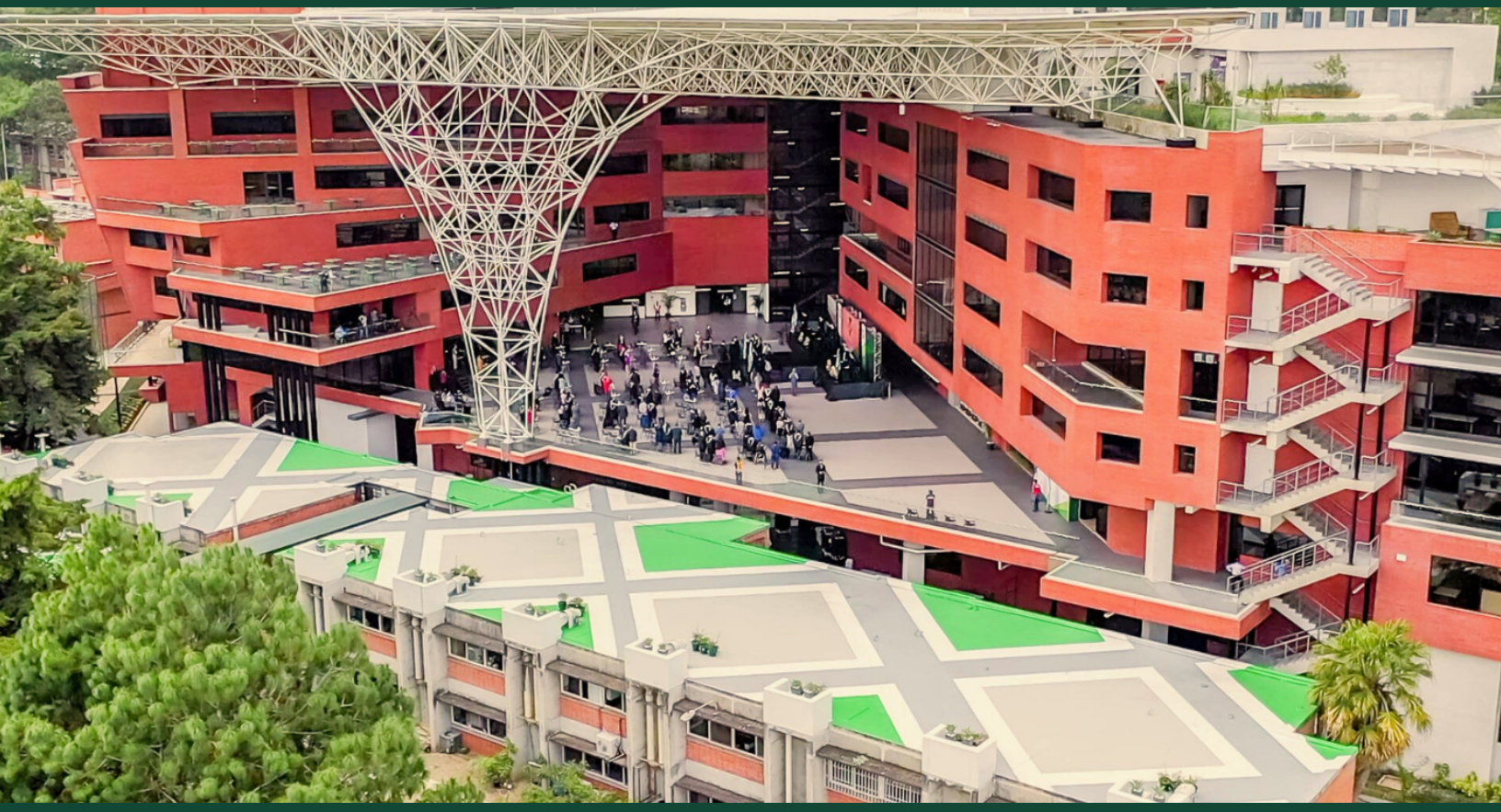
Automatización y mejora de la etapa de síntesis lógica, optimización del proceso de instalación de aplicaciones de Synopsys y simulación para el diseño del circuito integrado "El Gran Jaguar" en su adaptación a la tecnología de 65 nm

Carlos Julio Valdés Oajaca



UVG | UNIVERSIDAD DEL VALLE DE GUATEMALA

UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería

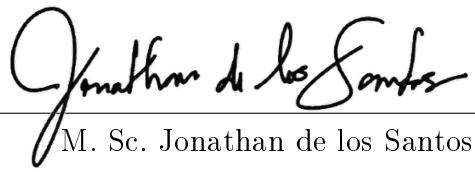
Automatización y mejora de la etapa de síntesis lógica, optimización del proceso de instalación de aplicaciones de Synopsys y simulación para el diseño del circuito integrado "El Gran Jaguar" en su adaptación a la tecnología de 65 nm

Trabajo de graduación presentado por Carlos Julio Valdés Oajaca para optar al grado académico de Licenciado en Ingeniería Electrónica

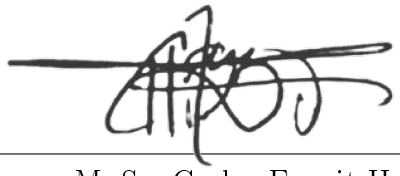
Guatemala,

2025

Vo.Bo.:

(f) 

M. Sc. Jonathan de los Santos

(f) 

M. Sc. Carlos Esquit Hernández

Fecha de aprobación: Guatemala, 20 de noviembre de 2025.

Prefacio

"La fe y la razón son como las dos alas con las que el espíritu humano se eleva hacia la verdad." — Santo Tomás de Aquino

Primeramente, este trabajo de tesis lo dedico a Dios, fuente de toda sabiduría y conocimiento. A Él gracias por concederme la vida y a través de ella, la oportunidad de desarrollarme no solo académicamente, sino también personalmente. A Jesucristo, porque siempre ha estado a mi lado cual gran amigo, y por enseñarme durante este tiempo que no hay valor más grande que el de amar a los demás sin medida. Además, de forma especial dedico este trabajo a la Santísima Virgen María, por su amor de madre incondicional que ha servido de esperanza durante todos los episodios de mi vida.

De la misma forma, quiero expresar mi más profundo agradecimiento a mis padres: Ing. Carlos Julio Valdés Marckwordt e Ing. Nydia Maribel Oajaca Portillo de Valdés, porque gracias a ellos soy el hombre que soy al día de hoy. Gracias por su enorme amor, su apoyo total y por enseñarme que todo es posible con Dios a mi lado; a mi querida hermana, a quien admiro mucho, la Ing. María Isabel Valdés Oajaca, quien siempre ha sido mi fuente de alegría, gracias por ser mi ejemplo de esfuerzo y dedicación constante; a mis abuelos: Telma, Carlos, Luisa y Conrado (†); a mis tíos: Ronald, Victoria y Erick; a mis primas: Luisa del Carmen y Ericka Isabel por su cariño, y a mi ahijado Ángel Gabriel.

También quiero expresar mi agradecimiento a la Universidad del Valle de Guatemala, especialmente al Departamento de Ingeniería Electrónica, Mecatrónica y Biomédica, por la excepcional educación recibida y la motivación para ser un profesional de impacto positivo para Guatemala y el mundo; al director del departamento, el Ing. Carlos Esquit, por toda su orientación y apoyo durante mi formación académica; a mi asesor, el Ing. Jonathan de los Santos, quien me dio la oportunidad de formar parte del proyecto del nanochip "El Gran Jaguar"; al profesor Ing. Gabriel Barrientos, por la oportunidad de crecer como auxiliar de curso y todos los consejos recibidos, y a la Ing. Magda Moscoso, quien desde un principio confió en mí para desarrollarme como auxiliar del laboratorio de Física.

Finalmente, quiero agradecer a mis amigos Marco, Paolo y Antonio, por su amistad incondicional durante toda mi vida; a mi buen amigo Giovanni, Johann, a mis colegas Estuardo, Miguel, Luis, Mauricio, Óscar, Byron y Aldo.

Índice general

Prefacio I

Índice de figuras IX

Índice de cuadros XII

Resumen XIII

Abstract XIV

1. Introducción 1

2. Antecedentes 3
2.1. El circuito integrado "El Gran Jaguar" 3

3. Justificación 13

4. Objetivos 15
4.1. Objetivo general 15
4.2. Objetivos específicos 15

5. Alcance 17

6. Marco teórico 18
6.1. Flujo de diseño para ASIC (*Application Specific Circuit*) 18
6.2. Herramientas de Synopsys 22

7. Archivos de TSMC de 65 nm 32
7.1. Librerías para las celdas de TSMC de 65 nm 32
7.2. Archivos PDF de las librerías de TSMC de 65 nm 35
7.3. Archivos Verilog de las librerías de TSMC de 65 nm 35

8. Estructura de directorios **37**
8.1. Estructura de directorios general 38
8.2. Estructura de directorios para la síntesis lógica 39

9. Flujo de diseño para la síntesis lógica **43**

10. Preparación para la síntesis lógica **48**
10.1. Diseño de circuitos digitales a nivel RTL 49
10.2. Desarrollo del *script* de síntesis lógica en TCL 56

11. Síntesis de circuitos digitales descritos en Verilog **63**
11.1. Síntesis lógica de una compuerta NOT 63
11.2. Síntesis lógica de un *Carry Look-Ahead Adder* de 4 bits 81
11.3. Síntesis lógica de un *Ripple Carry Counter* de 4 bits 99
11.4. Síntesis lógica de un *Ring Oscillator* 117
11.5. Síntesis lógica de una *Arithmetic Logic Unit* (ALU) de 4 bits con un *Ring Oscillator* 134
11.6. Síntesis lógica del nanochip "El Gran Jaguar" 159

12. ISE Design Suite **181**
12.1. Instalación 181
12.2. Celdas estándar (*Blackboxes*) 186
12.3. Simulación del nanochip "El Gran Jaguar" en ISE Design Suite 196

13. Verificación experimental del *gate-level netlist* del *core* del nanochip "El Gran Jaguar" (*FPGA prototyping*) **210**
13.1. Programación del FPGA Xilinx Virtex-5 Genesys 212
13.2. Verificación experimental junto a la Raspberry Pi 4B 217

14. Automatización para la instalación de las aplicaciones de Synopsys **222**
14.1. Explicación general del *script* Synopsys_Apps_Inst.sh 223
14.2. Ejecución del *script* Synopsys_Apps_Inst.sh 225

15. Conclusiones **234**

16. Recomendaciones **236**

17. Referencias **238**

18. Anexos **242**
18.1. Código de Python para la generación del Verilog del *core* del nanochip "El Gran Jaguar" 242
18.2. Nanochip "El Gran Jaguar" 248
18.3. Códigos para la simulación e implementación del *gate-level netlist* del *core* de "El Gran Jaguar" en el FPGA Xilinx Virtex-5 Genesys 592
18.4. Códigos para la automatización de la instalación de las aplicaciones de Synopsys 610

19. Glosario **647**

Índice de figuras

1. Flujo de diseño *front-end* (izquierda) y *back-end* (derecha) [2] 4
2. Síntesis lógica: (a) Esquemático sin el uso de restricciones y (b) Esquemático con el uso de restricciones [3] 5
3. Síntesis física de *full adder* en IC Compiler 6
4. Síntesis física de "El Gran Jaguar" en IC Compiler II 7
5. Errores de densidad en la capa de metal 5 y 6 8
6. Layout con 2 errores de densidad 8
7. Layout sin errores de densidad pero múltiples errores de diseño 9
8. Layout con red de alimentación correcta 10
9. Simulación de la síntesis lógica de "El Gran Jaguar" 11
10. Layout de "El Gran Jaguar" en 180 nm 12

11. Flujo de diseño para un ASIC digital 19
12. Flujo de diseño del *Front-End* propuesto por Synopsys [30] [33] 22
13. Interfaz gráfica de Design Compiler (Design Vision) 27
14. Interfaz gráfica de *Verdi* 28
15. FPGA Xilinx Virtex-5 Genesys 29
16. Conectores Pmod del FPGA Xilinx Virtex-5 Genesys 30
17. Entorno Xilinx ISE 31
18. Raspberry Pi 4B 31

19. Condición de operación de la librería tcbn65lpwc0d9 34
20. Condición de operación de la librería tpdn65lpnv2od3wc5 34

21. Estructura de directorios general 38
22. Estructura de directorios para la síntesis lógica 39

23. Diagrama de *inputs* y *outputs* para la síntesis lógica 43
24. Flujos de diseño para la síntesis lógica empleados en años anteriores 45
25. Flujo de diseño propuesto para la síntesis lógica 46

26. *Wire Bonding* de un circuito integrado 51
27. Esquemático del pad PDDW0204SCDG 52
28. Tabla de verdad del pad PDDW0204SCDG 53

29. Diagrama de flujo para la realización de un *script* de síntesis lógica en TCL . 57

30. Archivos necesarios para sintetizar la compuerta NOT 64
31. Interfaz gráfica de Verdi al simular el *core* de la NOT 65
32. Partes de la herramienta Verdi al simular 66
33. Simulación de la compuerta NOT 67
34. Modificación de parámetros para sintetizar el *core* de la NOT 68
35. Ejecución del *script* desde la terminal para realizar la síntesis de la NOT . . . 68
36. Celdas totales y verificación de la síntesis del *core* de la NOT 69
37. GUI de Design Compiler de la síntesis del *core* de la NOT 70
38. Esquemático del *core* de la NOT sintetizada 71
39. Archivos producidos por la síntesis del *core* de la NOT 72
40. Simulación del circuito sintetizado del *core* de la NOT 72
41. Explicación de la función de la sección de *Schematic* con la síntesis de la NOT 73
42. Modificación de la instancia del archivo *testbench* 74
43. Simulación del archivo de la NOT con pads 75
44. Modificación del *script* para realizar la síntesis del circuito con pads 76
45. Ejecución del *script* desde la terminal para realizar la síntesis de la NOT con pads 76
46. Ejecución del comando `report_cell` desde la terminal luego de sintetizar la NOT con pads 77
47. Ejecución del comando `check_design` desde la terminal luego de sintetizar la NOT con pads 78
48. Esquemático de la NOT con pads en Design Vision 79
49. Archivos producto de la síntesis de la NOT con pads 80
50. Simulación de la NOT sintetizada con pads 81
51. Archivos del CLA necesarios para la síntesis 82
52. Simulación del CLA de 4 bits 86
53. Modificación de los parámetros del *script* para sintetizar el CLA 87
54. Ejecución del *script* desde la terminal para realizar la síntesis del *core* del CLA 87
55. Celdas estándar y verificación de errores de la síntesis del *core* del CLA . . . 88
56. Esquemático del CLA en Design Vision 89
57. Simulación del CLA sintetizado 90
58. Modificación de la instancia dentro del *testbench* del CLA 91
59. Simulación del CLA con pads 92
60. Modificación del parámetro `USER_OPTION` 93
61. Ejecución del *script* desde la terminal para realizar la síntesis del CLA con pads 93
62. Celdas estándar totales del CLA sintetizado con pads 94
63. Ejecución del comando `check_design` luego de sintetizar el CLA con pads . . 95
64. Esquemático del CLA con pads 97
65. Archivos que resultan de la síntesis del CLA con pads 98
66. Simulación del CLA sintetizado con pads 98
67. Archivos para el RCC de 4 bits 101
68. Simulación del RCC de 4 bits 102
69. Cambio de parámetros para la síntesis del RCC 103
70. Ejecución del *script* desde la terminal para realizar la síntesis del *core* del RCC 103
71. Celdas de la síntesis del *core* del RCC 104

72. Esquemático del *core* del RCC . . . 106
73. Archivos de salida al sintetizar el *core* del RCC de 4 bits . . . 107
74. Simulación del *core* del RCC de 4 bits . . . 108
75. Cambio de instancia del *testbench* del RCC de 4 bits . . . 109
76. Simulación del RCC de 4 bits con pads . . . 110
77. Cambio del segundo parámetro para realizar la síntesis del RCC de 4 bits con pads . . . 110
78. Ejecución del *script* desde la terminal para realizar la síntesis del RCC de 4 bits con pads . . . 111
79. Celdas mapeadas del RCC de 4 bits con pads . . . 111
80. Probblemas asociados con la síntesis del RCC de 4 bits con pads . . . 112
81. Esquemático del RCC de 4 bits sintetizado con pads . . . 116
82. Archivos producto de la síntesis del RCC de 4 bits con pads . . . 116
83. Simulación de la síntesis del RCC de 4 bits con pads . . . 117
84. Archivos para llevar a cabo la síntesis del *ring oscillator* . . . 120
85. Adición del *delay* en el archivo `ring_osc.v` . . . 121
86. Simulación de *core* del *ring oscillator* . . . 122
87. Parámetros para la síntesis del *core* del *ring oscillator* . . . 122
88. Ejecución del *script* desde la terminal para realizar la síntesis del *core* del *ring oscillator* . . . 123
89. Celdas mapeadas y reporte de problemas para la síntesis del *core* del *ring oscillator* . . . 123
90. Esquemático de la síntesis del *core* del *ring oscillator* . . . 125
91. Archivos de salida como producto de la síntesis del *core* del *ring oscillator* . . . 125
92. Adición de *delay* a la celda OR del diseño sintetizado del *core* del *ring oscillator* 126
93. Simulación de la síntesis del *core* del *ring oscillator* . . . 127
94. Cambios en el *testbench* para la simulación del *ring oscillator* con pads . . . 128
95. Simulación del *ring oscillator* con pads . . . 129
96. Modificaión de parámetros para la síntesis del *ring oscillator* con pads . . . 129
97. Ejecución del *script* desde la terminal para realizar la síntesis del *ring oscillator* con pads . . . 130
98. Ejecución del comando `report_cell [get_cells -hier *]` para la síntesis del *ring oscillator* con pads . . . 130
99. Ejecución del comando `check_design` para la síntesis del *ring oscillator* con pads . . . 131
100. Esquemático del *ring oscillator* con pads . . . 132
101. Archivos de salida de la síntesis del *ring oscillator* con pads . . . 133
102. Simulación de la síntesis del *ring oscillator* con pads . . . 133
103. Archivos para la síntesis del *core* de la ALU con *ring oscillator* . . . 138
104. Simulación del *core* de la ALU con *ring oscillator* . . . 139
105. Parámetros para la síntesis del *core* de la ALU con *ring oscillator* . . . 139
106. Ejecución del *script* desde la terminal para realizar la síntesis del *core* de la ALU con *ring oscillator* . . . 140
107. Celdas y problemas de la síntesis del *core* de la ALU con *ring oscillator* . . . 140
108. Esquemático del *core* de ALU con *ring oscillator* . . . 145
109. Archivos de salida de la síntesis del *core* de la ALU con *ring oscillator* . . . 146
110. Adición de *delay* a la celda OR del diseño sintetizado del *core* de la ALU con *ring oscillator* . . . 146

111. Simulación de la síntesis del *core* de la ALU con *ring oscillator* 147
112. Cambio de instancia en el *testbench* de la ALU con *ring oscillator* 148
113. Simulación de la ALU con *ring oscillator* con pads antes de sintetizar 149
114. Parámetro modificado para la síntesis de la ALU con *ring oscillator* con pads 149
115. Ejecución del *script* desde la terminal para realizar la síntesis del *core* del *ring oscillator* 150
116. Celdas mapeadas de la síntesis de la ALU con *ring oscillator* con pads 150
117. Posibles problemas de la síntesis de la ALU con *ring oscillator* con pads . . . 151
118. Esquemático de la síntesis de la ALU con *ring oscillator* con pads 158
119. Archivos de salida de la síntesis de la ALU con *ring oscillator* con pads 158
120. Simulación de la síntesis de la ALU con *ring oscillator* con pads 159
121. Archivo de Python para la generación del *core* del nanochip 160
122. Ejecución del Python para la generación del *core* del nanochip 160
123. Archivos de entrada para el nanochip "El Gran Jaguar" 163
124. *Delay* para simular el *ring oscillator* del nanochip "El Gran Jaguar" 163
125. Simulación del *core* del nanochip 164
126. Opción para editar el *testbench* del nanochip desde Verdi 164
127. Edición de la señal `select` del *testbench* del nanochip 165
128. Opción para volver a simular el *testbench* del nanochip 165
129. Configuración para volver a simular el *testbench* del nanochip 165
130. Botón para volver a simular el *testbench* del nanochip 166
131. Simulación del *core* del nanochip previa a síntesis en inglés 166
132. Parámetros para la síntesis del *core* del nanochip 167
133. Ejecución del *script* desde la terminal para realizar la síntesis del *core* del nanochip 167
134. Celdas estándar mapeadas de la síntesis del *core* del nanochip 168
135. Advertencias y/o errores de la síntesis del *core* del nanochip 168
136. Esquemático del *gate-level netlist* del *core* del nanochip 169
137. Archivos de salida de la síntesis del *core* del nanochip 169
138. Adición de *delay* a la celda OR del diseño sintetizado del *core* del nanochip . 170
139. Simulación en español del *gate-level netlist* del *core* del nanochip 171
140. Simulación en inglés del *gate-level netlist* del *core* del nanochip 171
141. Modificación del *testbench* para simular el nanochip con pads 172
142. Simulación en español del nanochip con pads previo a síntesis 173
143. Simulación en inglés del nanochip con pads previo a síntesis 173
144. Cambio en el *script* para la síntesis del nanochip con pads 174
145. Ejecución del *script* desde la terminal para realizar la síntesis del nanochip con pads 174
146. Celdas estándar de la síntesis del nanochip con pads 175
147. Advertencias y/o errores de la síntesis del nanochip con pads 176
148. Esquemático del *gate-level netlist* del nanochip con pads 178
149. Archivos de salida por la síntesis del nanochip con pads 179
150. Adición de *delay* a la celda OR del diseño sintetizado del nanochip con pads . 179
151. Simulación del *gate-level netlist* en español del nanochip con pads 180
152. Simulación del *gate-level netlist* en inglés del nanochip con pads 180

153. Descarga de ISE Design Suite 14.7 182
154. Primera parte de la importación del archivo .ova en VMware Workstation . . 182

155. Segunda parte de la importación del archivo .ova en VMware Workstation . . 183
156. Última parte de la importación del archivo .ova en VMware Workstation . . . 183
157. Primera parte de la instalación de los VMware Tools 184
158. Ventana donde se solicita la descarga de la imagen de disco linux.iso 184
159. Selección de la imagen de disco linux.iso en la configuración de la máquina virtual 185
160. Instalación de los VMware Tools desde la terminal de la máquina virtual . . . 186
161. Esquemático de la celda OA211Dx 187
162. Esquemático de la celda OA21Dx 187
163. Esquemático de la celda OA221Dx 188
164. Esquemático de la celda OA22Dx 188
165. Esquemático de la celda OA31Dx 189
166. Esquemático de la celda OA32Dx 189
167. Esquemático de la celda OAI211Dx 190
168. Esquemático de la celda OAI21Dx 191
169. Esquemático de la celda OAI221Dx 191
170. Esquemático de la celda OAI222Dx 192
171. Esquemático de la celda OAI22Dx 192
172. Esquemático de la celda OAI31Dx 193
173. Esquemático de la celda OAI32Dx 193
174. Esquemático de la celda OR2Dx 194
175. Esquemático de la celda OR3Dx 194
176. Esquemático de la celda OR4Dx 195
177. Esquemático de la celda TIEL 195
178. Esquemático de la celda XNR2Dx 196
179. Opción para administrar licencias en ISE Design Suite usando Project Navigator 197
180. Carga de la licencia en ISE Design Suite usando Project Navigator 197
181. Verificación de la conexión del FPGA a la máquina virtual 198
182. Creación de un nuevo proyecto en ISE Design Suite 198
184. Configuración del proyecto en ISE Design Suite 199
183. Selección del proyecto en ISE Design Suite 199
185. Creación del proyecto en ISE Design Suite 200
186. Vinculación del archivo del *gate-level netlist* del *core* del nanochip con el de *blackboxes* 200
187. Paso 1 para agregar los archivos de Verilog al proyecto 201
188. Paso 2 para agregar los archivos de Verilog al proyecto 201
189. Archivos de Verilog agregados al proyecto 202
190. Primer paso para agregar el archivo UCF al proyecto 202
191. Archivo UCF agregado al proyecto 203
192. Adición de *delay* a la celda OR en el archivo `Blackboxes.v` 203
193. Adición de *delay* a la celda INV en el archivo `Blackboxes.v` 204
194. Opciones para sintetizar y mapear el diseño en ISE Design Suite 204
195. Opción para simular el diseño en ISE Design Suite usando ISim 205
196. Primer paso para asignar una constante o crear un reloj en ISim 205
197. Asignación de una constante a una señal en ISim 206
198. Creación de un reloj en ISim 206
199. Ejecución de la simulación en ISim 207
200. Desactivación de los *resets* en ISim 207

201. Simulación del diseño sintetizado del *core* del nanochip "El Gran Jaguar" en ISim para el texto en español . . . 208
202. Simulación del diseño sintetizado del *core* del nanochip "El Gran Jaguar" en ISim para el texto en inglés . . . 208

203. Diagrama de conexión entre el FPGA y la Raspberry Pi 4B . . . 211
204. Distribución de pines del FPGA Xilinx Virtex-5 Genesys . . . 212
205. Primer paso para implementar el *core* de "El Gran Jaguar" en el FPGA . . . 213
206. Generación del *bitstream* del diseño sintetizado del *core* de "El Gran Jaguar" . 213
207. Creación del archivo de iMPACT para programar el FPGA . . . 214
208. Archivo de iMPACT para programar el FPGA . . . 214
209. Opción para añadir el archivo *bitstream* . . . 215
210. Selección del archivo *bitstream* para programar el FPGA . . . 215
211. Opción para programar el FPGA . . . 216
212. Programación del FPGA con el diseño sintetizado del *core* de "El Gran Jaguar" 216
213. Señal del *ring oscillator* antes de ejecutar el código en la Raspberry Pi 4B . . 218
214. Ejecución del programa de Python en la RPI 4B para recibir el ASCII del nanochip emulado . . . 218
215. Terminal que muestra la cadena ASCII recibida en español . . . 219
216. Terminal que muestra la cadena ASCII recibida en inglés . . . 219
217. Señal del *ring oscillator* después de ejecutar el código en la Raspberry Pi 4B . 220
218. Señal del *ring oscillator* observada desde el osciloscopio . . . 220
219. Verificación experimental de la síntesis lógica del nanochip "El Gran Jaguar" . 221

220. Ejecución del *script* `Synopsys_Apps_Inst.sh` desde la terminal . . . 225
221. Selección de la opción 2 del *script* . . . 226
222. Verificación de la instalación previa del Synopsys Installer . . . 226
223. Versiones disponibles del Synopsys Installer . . . 227
224. Selección de la versión del Synopsys Installer . . . 227
225. Descarga del Synopsys Installer . . . 228
226. Descarga del Synopsys Installer . . . 228
227. Descarga del Synopsys Installer . . . 229
228. Ejecución de la nueva versión del Synopsys Installer . . . 229
229. Selección de la opción 3 del *script* . . . 230
230. Opciones para descargar con la opción 3 del *script* . . . 230
231. Selección de la opción 2 para descargar aplicaciones . . . 231
232. Versiones disponibles de StarRC . . . 231
233. Descarga de StarRC . . . 232
234. Instalación de StarRC . . . 232
235. Finalización de la instalación de StarRC . . . 233

Índice de cuadros

1. Generación de simulación VCS . . . 23
2. Comando para compilación y simulación en VCS . . . 23
3. Ejecución de la simulación con nombre propio . . . 23
4. Ejecución de la simulación con nombre genérico . . . 23
5. Comando para iniciar Library Compiler . . . 24
6. Comandos para leer información de una librería . . . 25
7. Comando para desplegar la información de una librería . . . 25
8. Comando para desplegar información de *timing* de una celda dentro de la librería . . . 25
9. Comando para desplegar el atributo de unidad de tiempo de la librería . . . 25
10. Comando para iniciar Design Compiler . . . 26
11. Comando para ejecutar un *script* en Design Compiler . . . 26
12. Comando para iniciar la interfaz gráfica de Design Compiler (Design Vision) . 26
13. Inicialización de Verdi . . . 28
14. Carga de un archivo .fsdb a Verdi . . . 28
15. Comandos para observar condiciones de operación de la librería . . . 33
16. Archivo Not.v: Diseño de una compuerta NOT en Verilog . . . 49
17. Archivo Not_tb.v: Diseño del *testbench* de una compuerta NOT en Verilog . 50
18. Parte del módulo en Verilog del pad PDDW0204SCDG . . . 54
19. Configuración del pad PDDW0204SCDG como entrada . . . 54
20. Configuración del pad PDDW0204SCDG como salida . . . 55
21. Archivo IO_Not.v: Diseño del módulo de una compuerta NOT con pads en Verilog . . . 56
22. Archivo script_dc.tcl: *Script* de síntesis lógica en TCL . . . 57
23. Comandos para la compilación y simulación del *core* de la NOT . . . 65
24. Comando para la ejecución de la síntesis de la NOT . . . 68
25. Comandos posteriores a la ejecución del *script* . . . 69
26. *Gate-level netlist* del *core* de la NOT . . . 69
27. Comando iniciar la GUI de Design Compiler . . . 70
28. Comando para salir de Design Compiler . . . 71
29. Comandos para simular el *gate-level netlist* del *core* de la NOT . . . 72
30. Comandos para la compilación y simulación de la NOT con pads . . . 74
31. Comandos posteriores a la ejecución del *script* . . . 77

32. *Gate-level netlist* de la NOT con pads . . . 78
33. Comandos para simular el *gate-level netlist* del *core* de la NOT . . . 80
34. Archivo ADDER4b.v . . . 82
35. Archivo ADDER4b_tb.v . . . 83
36. Archivo IO_ADDER4b.v . . . 84
37. Comandos para la compilación y simulación del *core* del ADDER4b . . . 85
38. Archivo ADDER4b_mapped.v en la carpeta `outputs_no_pads/` . . . 88
39. Comandos para la compilación y simulación del CLA con pads . . . 91
40. Archivo ADDER4b_mapped.v en la carpeta `outputs_with_pads/` . . . 95
41. Comandos para la compilación y simulación del CLA sintetizado con pads . . . 98
42. Archivo rc_counter.v . . . 99
43. Archivo rc_counter_tb.v . . . 100
44. Archivo IO_rc_counter.v . . . 100
45. Comandos para la compilación y simulación de la NOT con pads . . . 101
46. Archivo rc_counter_mapped.v en la carpeta `outputs_no_pads/` . . . 104
47. Comandos para la compilación y simulación de la NOT con pads . . . 107
48. Comandos para la compilación y simulación de la NOT con pads . . . 109
49. Archivo rc_counter_mapped.v en el directorio `outputs_with_pads/` . . . 112
50. Comandos para la compilación y simulación de la NOT con pads . . . 117
51. Archivo ring_osc.v . . . 118
52. Archivo ring_osc_tb.v . . . 118
53. Archivo IO_ring_osc.v . . . 119
54. Comandos para la compilación y simulación de la NOT con pads . . . 121
55. Archivo ring_osc_mapped.v en el directorio `outputs_no_pads/` . . . 124
56. Comandos para la compilación y simulación del *ring oscillator* . . . 126
57. Comandos para la compilación y simulación de la NOT con pads . . . 128
58. Archivo ring_osc_mapped.v en el directorio `outputs_with_pads` . . . 131
59. Comandos para la compilación y simulación del *ring oscillator* con pads . . . 133
60. Archivo ALU_with_ring_osc.v . . . 134
61. Archivo ALU_with_ring_osc_tb.v . . . 135
62. Archivo IO_ALU_with_ring_osc.v . . . 136
63. Comandos para la compilación y simulación de la ALU con *ring oscillator* . . . 138
64. Archivo ALU_with_ring_osc.v en el directorio `outputs_no_pads/` . . . 141
65. Comandos para la compilación y simulación de la ALU con *ring oscillator* . . . 146
66. Comandos para la compilación y simulación de la ALU con *ring oscillator* con pads . . . 148
67. Archivo ALU_with_ring_osc_mapped.v en el directorio `otuputs_with_pads` 151
68. Comandos para la compilación y simulación de la ALU con *ring oscillator* con pads . . . 158
69. Archivo nanochip_tb.v . . . 160
70. Archivo IO_nanochip.v . . . 161
71. Comandos para la compilación y simulación del *core* del nanochip . . . 164
72. Comandos para la compilación y simulación de la NOT con pads . . . 170
73. Comandos para la compilación y simulación del nanochip con pads . . . 172
74. Área y cantidad de celdas del nanochip con pads sintetizado . . . 176
75. Consumo de potencia del nanochip con pads sintetizado . . . 177
76. Comandos para la compilación y simulación del nanochip con pads . . . 179
77. Comandos para instalar VMware Tools . . . 185

78. Tabla de verdad de la celda OA211Dx . . . 187
79. Tabla de verdad de la celda OA21Dx . . . 187
80. Tabla de verdad de la celda OA221Dx . . . 188
81. Tabla de verdad de la celda OA22Dx . . . 189
82. Tabla de verdad de la celda OA31Dx . . . 189
83. Tabla de verdad de la celda OA32Dx . . . 190
84. Tabla de verdad de la celda OAI211Dx . . . 190
85. Tabla de verdad de la celda OAI21Dx . . . 191
86. Tabla de verdad de la celda OAI221Dx . . . 191
87. Tabla de verdad de la celda OAI222Dx . . . 192
88. Tabla de verdad de la celda OAI22Dx . . . 193
89. Tabla de verdad de la celda OAI31Dx . . . 193
90. Tabla de verdad de la celda OAI32Dx . . . 194
91. Tabla de verdad de la celda OR2Dx . . . 194
92. Tabla de verdad de la celda OR3Dx . . . 195
93. Tabla de verdad de la celda OR4Dx . . . 195
94. Tabla de verdad de la celda TIEL . . . 196
95. Tabla de verdad de la celda XNR2Dx . . . 196
96. Comando para ejecutar el código de Python en la Raspberry Pi 4B . . . 218
97. Comando para dar permisos de ejecución a los *scripts* . . . 225
98. Archivo escriturafinal_vp3.py ubicado en el directorio `verilog_generation/` 242
99. Archivo nanochip_mapped.v en el directorio `outputs_no_pads/` . . . 248
100. Archivo nanochip_mapped.v en el directorio `outputs_with_pads/` . . . 420
101. Archivo Blackboxes.v . . . 592
102. Código UCF para la asignación de pines del FPGA . . . 603
103. Archivo nano_RPI_modified.py . . . 604
104. Archivo APPS.sh . . . 610
105. Archivo Synopsys_Apps_Inst.sh . . . 632

Resumen

En este trabajo de tesis se presenta una metodología mejorada para la síntesis lógica y la verificación experimental del nanochip "El Gran Jaguar", alineada con el flujo de diseño recomendado por Synopsys. Como parte del proceso metodológico, se sintetizaron cinco circuitos preliminares con el fin de validar el flujo propuesto antes de abordar el diseño completo del nanochip. Todas las síntesis se realizaron utilizando la tecnología de 65 nm de TSMC, obteniéndose resultados correctos en las simulaciones. En particular, el *ring oscillator* mostró un comportamiento correcto, superando inconsistencias observadas en trabajos previos.

Posteriormente, el *core* del nanochip fue implementado en un FPGA Xilinx Virtex-5 Genesys para realizar su verificación mediante el enfoque de *FPGA prototyping*. En este esquema, el FPGA actúa como reemplazo temporal del ASIC mientras que la Raspberry Pi representa el sistema con el cual el chip interactuará una vez fabricado. Las pruebas demostraron la transmisión y recepción correctas de caracteres a través de dicha interfaz, así como el funcionamiento esperado del *ring oscillator*, verificado mediante mediciones con osciloscopio.

Adicionalmente, se desarrolló un *script* para automatizar la instalación de aplicaciones de Synopsys. Este procedimiento se validó mediante la instalación exitosa del Synopsys Installer y StarRC, reduciendo significativamente el tiempo y la complejidad en la preparación del entorno de trabajo.

Los resultados obtenidos confirman la validez de la metodología propuesta, la correcta operación del nanochip y la utilidad de la herramienta de automatización desarrollada.

Palabras clave: Síntesis lógica, nanochip, TSMC 65 nm, *Synopsys Design Flow*, *FPGA prototyping*.

Abstract

In this thesis, an improved methodology for the logic synthesis and experimental verification of the nanochip "El Gran Jaguar" is presented, aligned with the design flow recommended by Synopsys. As part of the methodological process, five preliminary circuits were synthesized in order to validate the proposed flow before addressing the complete design of the nanochip. All syntheses were carried out using TSMC's 65 nm technology, yielding correct results in the simulations. In particular, the ring oscillator exhibited proper behavior, overcoming inconsistencies reported in previous work.

Subsequently, the nanochip's core was implemented on a Xilinx Virtex-5 Genesys FPGA to perform its verification through a FPGA prototyping approach. In this scheme, the FPGA serves as a temporary replacement for the ASIC, while the Raspberry Pi represents the system with which the chip will interact once fabricated. The tests demonstrated correct transmission and reception of characters through this interface, as well as the expected operation of the ring oscillator, verified through oscilloscope measurements.

Additionally, a script was developed to automate the installation of Synopsys applications. This procedure was validated through the successful installation of Synopsys Installer and StarRC, significantly reducing the time and complexity required to prepare the working environment.

The results obtained confirm the validity of the proposed methodology, the correct operation of the nanochip, and the usefulness of the automation tool developed.

Keywords: Logic synthesis, nanochip, TSMC 65 nm, Synopsys Design Flow, FPGA prototyping.

CAPÍTULO 1

Introducción

La síntesis lógica de circuitos integrados digitales constituye una etapa fundamental dentro del flujo de diseño de un ASIC, ya que establece la estructura funcional que más adelante será utilizada en las fases de implementación física. En este contexto, el presente trabajo se enmarca en el campo del diseño digital y la validación de circuitos integrados digitales, con énfasis en la mejora de metodologías para la síntesis lógica y la verificación experimental mediante técnicas de *FPGA prototyping*. El estudio se desarrolla en torno al nanochip "El Gran Jaguar", un diseño cuyo *core* se describe en Verilog y cuya correcta operación debe garantizarse antes de su futura fabricación.

El propósito principal de esta investigación es mejorar la metodología utilizada en años anteriores para la síntesis lógica de circuitos digitales, alineándola con el flujo de diseño recomendado por Synopsys. Para ello, se implementa un proceso estructurado que incluye la definición de un solo *script* en TCL para la síntesis, la creación de una nueva jerarquía de directorios, la integración de herramientas EDA de Synopsys y la ejecución de verificaciones mediante simulación. Como parte del procedimiento metodológico, se sintetizan cinco circuitos antes de abordar el diseño completo del nanochip, con el fin de validar la robustez del flujo propuesto. Todas las síntesis se realizan con librerías de TSMC para la tecnología de 65 nm y se verifican mediante simulaciones a nivel RTL (presíntesis) y a nivel de *gate-level netlist* (postsíntesis).

La validación experimental del *core* del nanochip constituye un elemento esencial de este trabajo. Para ello, se implementa su *gate-level netlist* en un FPGA Xilinx Virtex-5 Genesys, lo que permite evaluar su comportamiento mediante el intercambio de datos con una Raspberry Pi 4B. Este proceso de emulación confirma el funcionamiento correcto del sistema bajo condiciones reales de operación y demuestra la viabilidad del flujo de síntesis adoptado. Entre los resultados más relevantes se encuentra la correcta operación del *ring oscillator*, elemento que en investigaciones anteriores no funcionaba.

Además del diseño y verificación del nanochip, el presente trabajo aborda la automatización del proceso de instalación de herramientas EDA de Synopsys. Se desarrolla un *script* en Bash que permite la instalación automática del Synopsys Installer y de diversas aplicaciones

complementarias, superando las limitaciones detectadas en herramientas generadas en años previos y optimizando significativamente la preparación del entorno de trabajo.

En conjunto, las aportaciones de esta tesis establecen un flujo metodológico más confiable, reproducible y alineado con las mejores prácticas en el diseño de ASIC. La correcta síntesis de los circuitos estudiados, la verificación experimental del *core* del nanochip y la automatización de la instalación del entorno EDA confirman la efectividad de la metodología propuesta y proporcionan una base sólida para investigaciones futuras en el área de diseño de circuitos integrados digitales.

CAPÍTULO 2

Antecedentes

Desde el año 2013 se realizó un cambio clave para los estudiantes de Ingeniería Electrónica de la Universidad del Valle de Guatemala, donde se introdujeron conocimientos básicos de diseño en VLSI *(Very Large Scale Integration)*. Dicho cambio fue novedoso para abrir las puertas a una nueva área de investigación donde participarían tanto estudiantes como profesores.

Al año siguiente, el Departamento de Ingeniería Electrónica, Mecatrónica y Biomédica de la Universidad del Valle de Guatemala estableció un convenio con la empresa Synopsys, líderes globales en el diseño de semiconductores. Este acuerdo permite el uso de herramientas de *software* especializado para la elaboración y simulación de circuitos nanoelectrónicos.

El fruto de estos esfuerzos permitieron que en el año del 2014 se llevara a cabo el primer trabajo de graduación relacionado a VLSI, donde se realizó un sumador/restador completo de 32 bits utilizando tecnología CMOS de 28 nm (nanómetros) aplicando herramientas de la empresa Synopsys, realizado por MSc. Jonathan de los Santos [1].

Posteriormente, en el 2015 se realizaron cambios en el mapa curricular de Ingeniería Electrónica, sustituyendo el curso **Introducción a sistemas de diseño VLSI** por los cursos de **Nanoelectrónica 1 y 2**. El fin es profundizar en el conocimiento de teoría y diseño VLSI, donde los estudiantes participen de forma activa utilizando herramientas de la empresa Synopsys. Gracias a ello surgió la iniciativa de creación del primer nanochip en Centroamérica diseñado por una universidad, con el apoyo de la empresa global número uno en fabricación de semiconductores **TSMC** (*Taiwan Semiconductor Manufacturing Company*) a través de **IMEC** (*Interuniversity Microelectronics Centre*).

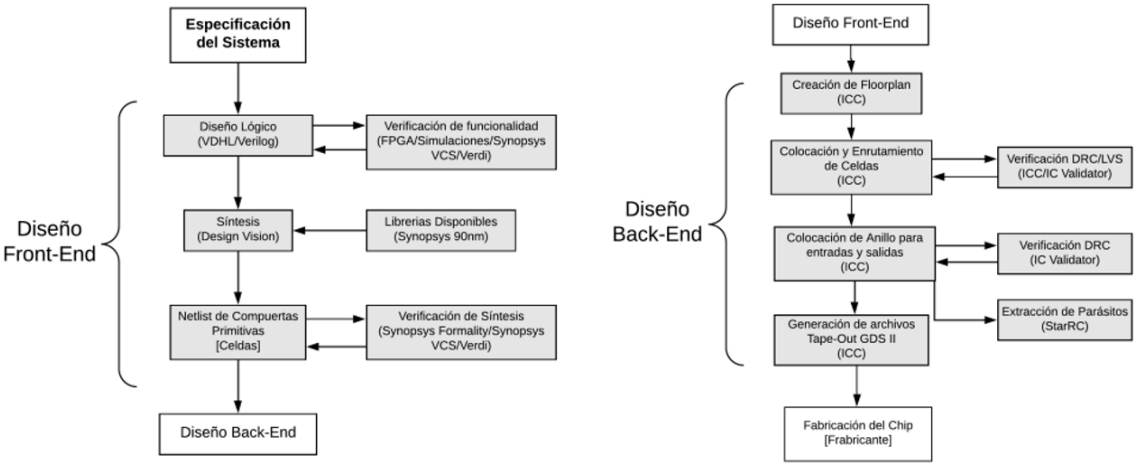
2.1. El circuito integrado "El Gran Jaguar"

Desde el año 2019 comenzaron los trabajos relacionados a la creación del nanochip titulado "El Gran Jaguar". En aquel momento, los esfuerzos de los Ingenieros Steven Rubio y Luis

Nájera se concentraron principalmente en establecer un flujo de diseño para la creación del chip usando tecnología CMOS y la implementación de circuitos sintetizados a nivel *netlist* a partir de un lenguaje descriptor de *hardware*.

El Ingeniero Rubio presentó el proceso completo de un flujo de diseño desde su funcionalidad hasta su fabricación. Según este trabajo de graduación, se plantean dos esquemas generales para lograr obtener los resultados esperados utilizando las herramientas a disposición el proceso se divide en dos categorías: **síntesis lógica (front-end)** y **síntesis física (back-end)** [2].

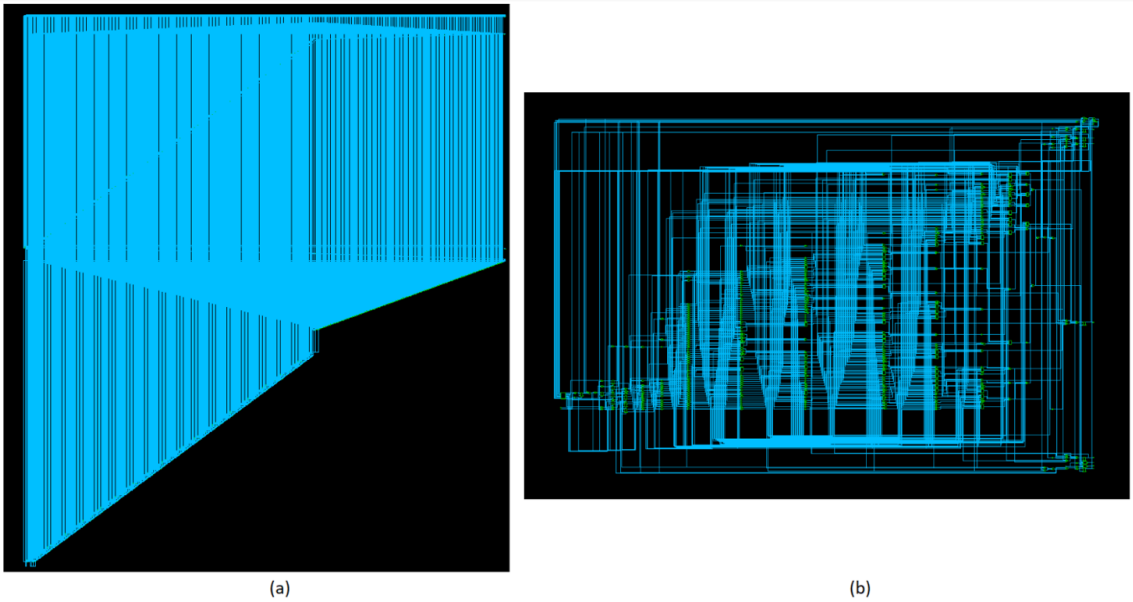
Figura 1. Flujo de diseño *front-end* (izquierda) y *back-end* (derecha) [2]



Nota. La imagen muestra los flujos de diseño para el diseño de un circuito integrado.

Por otro lado, el Ingeniero Luis Nájera se enfocó en la implementación de la arquitectura del utilizando Verilog como lenguaje descriptor de *hardware*. En su trabajo de graduación se logra la traducción del circuito elaborado en Verilog a un *netlist* de compuertas lógicas en tecnología CMOS. Para ello, se utilizó la herramienta de Design Vision y librerías de la empresa Synopsys. Aunado a ello, se utilizó VCS y Formality para comprobar el funcionamiento del circuito tanto antes como después de la síntesis lógica. A su vez, se demostró la utilidad de agregar restricciones de diseño para la optimización de un circuito conformado por 16 pines: 2 de entrada tanto para VDD como VSS, 2 de entrada para un selector, 1 de entrada para el reloj, 1 de salida para un *ring oscillator* y 8 de salida para los caracteres [3].

Figura 2. Síntesis lógica: (a) Esquemático sin el uso de restricciones y (b) Esquemático con el uso de restricciones [3]



Nota. La imagen muestra esquemáticos resultantes de la síntesis lógica en Design Compiler.

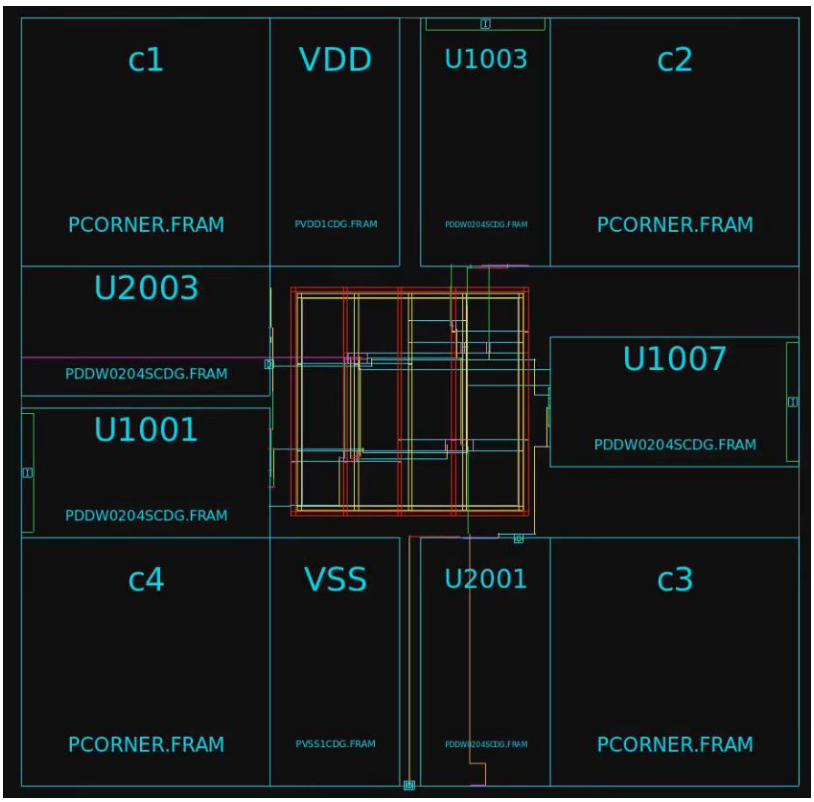
2.1.1. Progreso del año 2020

El año del 2020 fue un período clave para el proyecto, donde se obtuvo el apoyo de IMEC para fabricar el nanochip con la compañía TSMC en su tecnología CMOS de 180 nm. Los avances se realizaron principalmente para la parte del *Back-End*, donde se realizó la extracción de parásitos del diseño para crear un modelo real del circuito y poder realizar simulaciones más cercanas a la realidad [4].

Sin embargo, en cuanto al *Front-End* se incorporaron nuevos comandos en un *script* para llevar a cabo la síntesis lógica. A pesar de obtener el *netlist* a nivel transistor utilizando las librerías académicas de Synopsys, no se pudo completar la traducción utilizando las librerías académicas que proporcionó TSMC en su momento [5].

Por otro lado, se implementó el anillo de Entradas/Salidas donde se escogen los pines y pads adecuados para el funcionamiento del chip [6]. Se realizó la verificación DRC (*Design Rule Check*) modificando el *script* pero sin obtener los resultados esperados [7]. Además, se hicieron las pruebas ARC (*Antena Rule Check*) y ERC (*Electrical Rule Check*) para comprobar que el circuito funcionase físicamente de la forma adecuada [6].

Figura 3. Síntesis física de *full adder* en IC Compiler



Nota. La imagen muestra el layout de un *full adder* en el entorno de IC Compiler.

Finalmente, basado en los flujos de diseño anteriores se utilizó por primera vez IC Compiler I para la validación del diseño en silicio contra el esquemático (LVS). Sin embargo, la comprobación del funcionamiento no fue exitosa debido a la falta de información acerca de las celdas (*black boxes*) que provee TSMC para la tecnología CMOS de 180 nm [8].

2.1.2. Progreso del año 2021

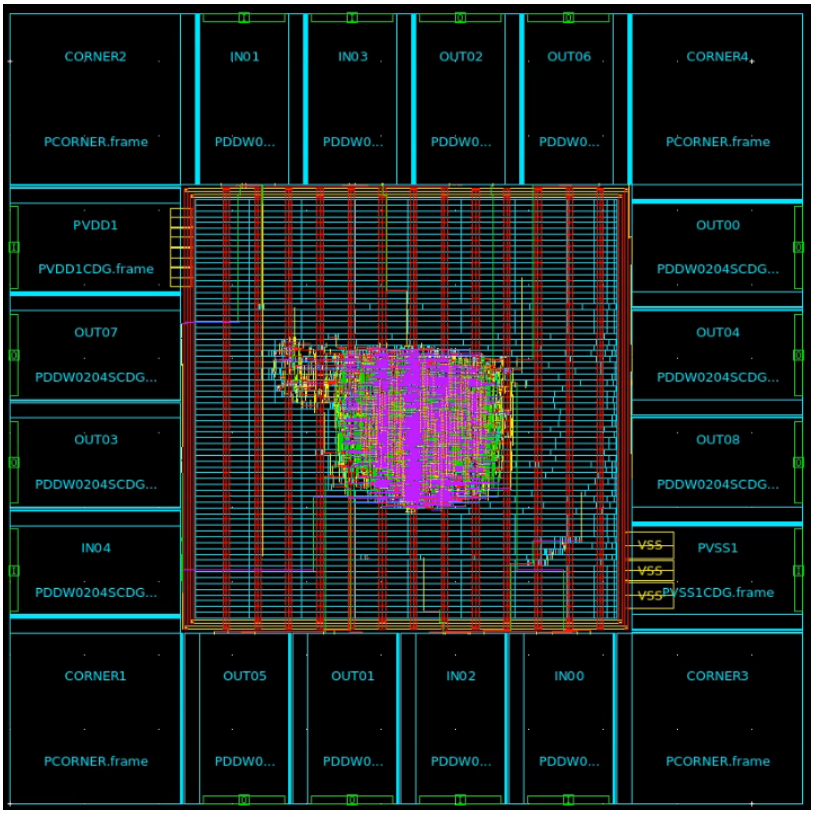
Durante el año 2021, las tareas entre los estudiantes fueron divididas. Hubo quienes se enfocaron en el apartado de síntesis lógica y otros en el apartado de síntesis física.

De esta manera, se propuso un flujo de diseño para la síntesis lógica donde se automatizó la generación del circuito en Verilog a través de un código de Python. Además, se logró obtener el *netlist* a nivel transistor por primera vez con la librería de TSMC de 180 nm. No obstante, se debe ejecutar el script dos veces para incluir los pines de Entradas/Salidas [9].

Por otra parte, como parte de los esfuerzos de este año, se propuso una metodología para la colocación (*Placement*) e interconexión de los componentes (*Interconnection/Routing*) [10]. Se logró manejar los errores heredados de años anteriores para la verificación LVS, con la creación de instancias dentro del *Runset* para cada celda no definida, agregando formato

al nombre del archivo GDS y la migración al uso de IC Compiler II [11].

Figura 4. Síntesis física de "El Gran Jaguar" en IC Compiler II



Nota. La imagen muestra el layout del circuito integrado "El Gran Jaguar" con tecnología de 180 nm de TSMC.

Cabe mencionar que es necesario replicar el trabajo de años anteriores para continuar con la línea de diseño, incluyendo las verificaciones. A pesar de que en este año se obtuvo resultados satisfactorios para LVS, ARC y ERC, durante la comprobación de reglas de diseño (DRC) se obtuvo errores de densidad sin solventar, acerca del no cumplimiento de densidad mínima de metal requerida en las capas del chip [12].

2.1.3. Progreso del año 2022

Durante este período se replicó el trabajo realizado en años anteriores, se continuó con la corrección de errores de diseño donde estaba pendiente solucionar los errores de densidad.

En el apartado de síntesis lógica se exploró un flujo de diseño alternativo donde se logró obtener el netlist con las librerías de TSMC y las librerías educativas de Synopsys. De igual forma se realiza la síntesis lógica en dos pasos, incluyendo el uso de la herramienta Formality para la validación del resultado con respecto al diseño original [13]. Además, se implementó PrimeTime para el análisis de tiempo estático y TetraMAX para la generación de patrones de prueba [14].

Para la corrección de los errores de densidad se mencionaban dos escenarios principales [15]:

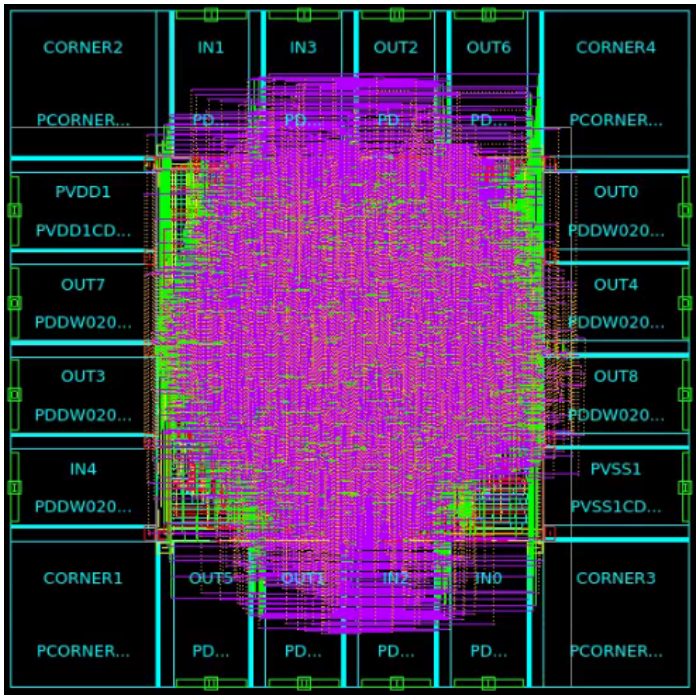
- El primero donde solo hay 2 errores de densidad en los metales 5 y 6. No hay errores de reglas de diseño.
- En el segundo no hay errores de densidad pero hay incontables errores de diseño.

Figura 5. Errores de densidad en la capa de metal 5 y 6

```
--------------------------------------------------------------------------
Density Report Summary:
--------------------------------------------------------------------------
Window = {{0 0 534.48 532.24}}
--------------------
Layer     Density(%)
--------------------
METAL1    0
METAL2    0
METAL3    0
METAL4    0
METAL5    21.1881
METAL6    22.9467
```

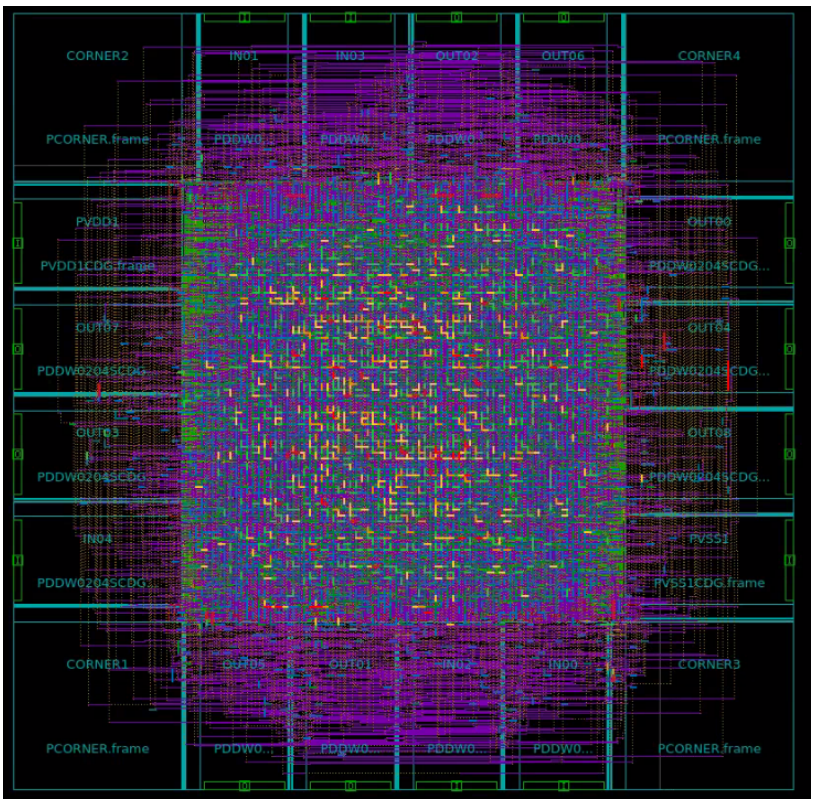
Nota. La imagen muestra el reporte de densidad para el layout de "El Gran Jaguar".

Figura 6. Layout con 2 errores de densidad



Nota. La imagen muestra el layout del circuito integrado "El Gran Jaguar" con errores de densidad en las capas de metal 5 y 6.

Figura 7. Layout sin errores de densidad pero múltiples errores de diseño



Nota. La imagen muestra el layout del circuito integrado "El Gran Jaguar" sin errores de densidad pero con múltiples errores de diseño.

Asimismo, se trabajó en una interfaz gráfica y script para automatizar la realización de la síntesis lógica y física. Además, se incluyen las verificaciones correspondientes a DRC, LVS, ERC, ARC y extracción de parásitos [15].

Aunado a todo ello, se trabajó en la automatización en la etapa de síntesis lógica, se exploró el uso de herramientas como StarRC para la extracción de parásitos y PrimSim para la simulación del circuito a nivel SPICE [16] [17] [18]. También se automatizó el proceso de las verificaciones físicas (DRC, ERC, LVS y ARC) y el proceso para la instalación de las aplicaciones de Synopsys [19] [20].

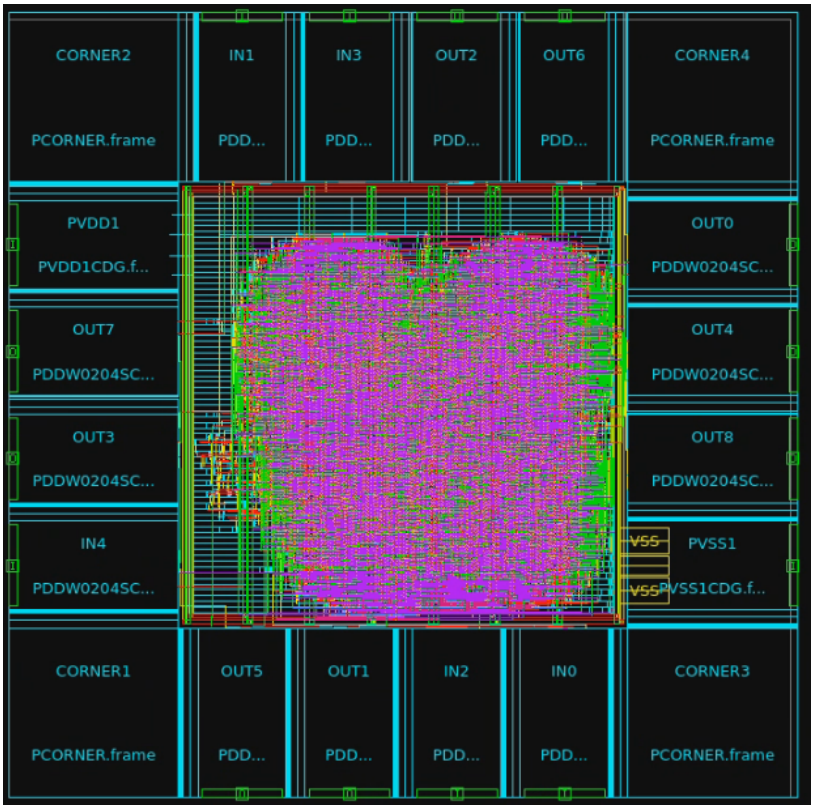
Por otra parte, a partir del archivo obtenido de síntesis lógica se crearon *black boxes* para su implementación en el FPGA Xilinx Virtex-5 Genesys. El objetivo era comprobar el funcionamiento del código en una plataforma física. Como complemento, a través del microcontrolador Tiva C se obtuvo la salida (bus de datos de 8 bits) del FPGA, para así procesar los caracteres que se envían y mediante comunicación serial (UART) enviarlos a una computadora. Luego mediante una aplicación realizada en Python, se realizó la conversión de una cadena de texto a audio para así lograr el sintetizador digital [21] [22].

2.1.4. Progreso del año 2023

En 2023, se implementaron cambios y mejoras en el flujo de diseño propuesto en años previos, optimizando la jerarquía de directorios utilizada tanto para la síntesis lógica como física. Se siguió trabajando en la solución a errores de densidad, un aspecto crítico, ya que si no son resueltos, no se cumplirían los requisitos mínimos para la fabricación del nanochip.

Para abordar los errores de densidad, se empleó el *runset* compatible con IC Validator proporcionado por TSMC para realizar el relleno de metal. Una vez solucionados estos errores, se realizó la verifación LVS (*Layout vs Schematic*) donde se detectaron errores con el *runset* ya mencionado. Estos errores se originaron por la falta de conexión entre los pines de alimentación y el anillo de alimentación, así como entre el anillo y la red de alimentación interna del circuito [23].

Figura 8. Layout con red de alimentación correcta



Nota. La imagen muestra el layout del circuito integrado "El Gran Jaguar" con la red de alimentación conectada de forma correcta.

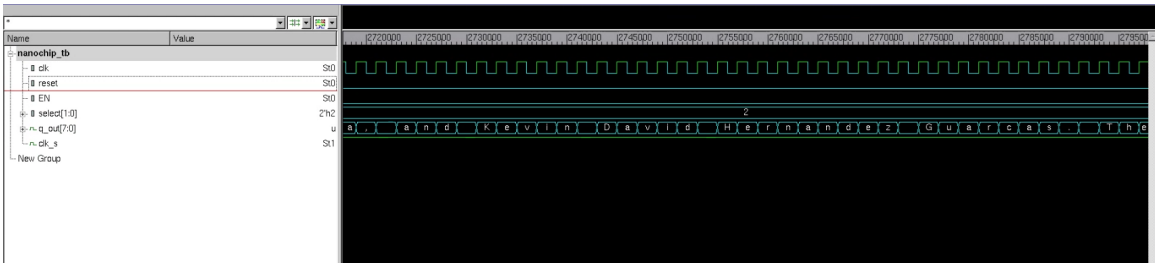
2.1.5. Progreso del año 2024

Para el año 2024, a finales del mismo, TSMC notifica al Departamento de Ingeniería Electrónica, Mecatrónica y Biomédica que el nanochip debe diseñarse en tecnología de 65 nm, debido a la descontinuación de la tecnología de 180 nm. Al ser un momento donde ya

se había trabajado para la tecnología de 180 nm, los logros de los trabajos de graduación de este año 2024 están enfocados a esta tecnología.

Se consiguió la realización de la síntesis lógica con un flujo de diseño distinto al de años anteriores, el cual permite en un nuevo *script* conseguir el *netlist* a nivel transistor con los pads de Entradas/Salidas en solo un paso, requeriendo que el *core* se sintetice antes del diseño con pads. De esta manera, se modificó la jerarquía de directorios heredada para lograr de forma más intuitiva el proceso de síntesis lógica [24].

Figura 9. Simulación de la síntesis lógica de "El Gran Jaguar"



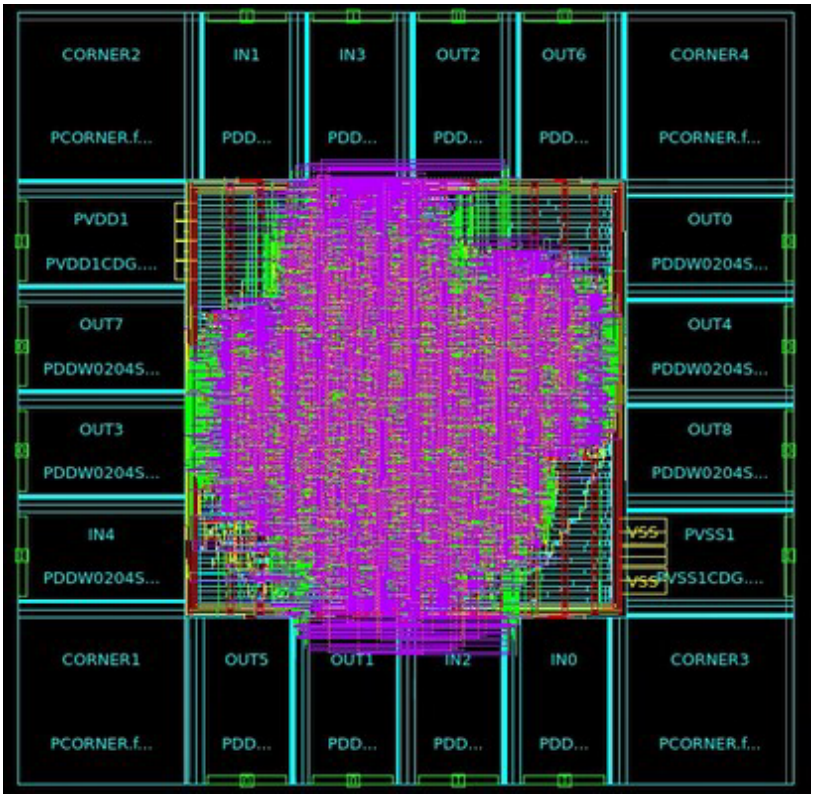
Nota. La imagen muestra parte de la simulación del circuito integrado "El Gran Jaguar" en DVE.

Por otro lado, en cuanto al *Back-End* se realizaron mejoras en el flujo de diseño donde se incorporaron funcionalidades como la generación de un archivo Verilog con componentes lógicos y físicos. Asimismo, se realizron pruebas DRC de varios circuitos: combinacionales, secuenciales y "El Gran Jaguar" [25].

Sin embargo, para la prueba DRC todos los circuitos la cumplieron satisfactoriamente menos "El Gran Jaguar". El error se encontró en la capa de metal 2. Se mencionó que el inconveniente espera resolverse a futuro con la implementación de la tecnología de 65 nm [25].

También se utilizó IC Compiler II y IC Validator para comprobar que los resultados obtenidos fueran fabricables. Las pruebas de LVS y ERC se cumplieron para todos los circuitos incluido "El Gran Jaguar". No obstante, luego de la extracción de parásitos el *deck* no es simulable, ya que no contiene pines de Entradas/Salidas y no hay mapeo existente para ellos. Por tanto, únicamente se realizó el mapeo de forma manual para una NOT, XOR y un sumador de 4 bits [26].

Figura 10. Layout de "El Gran Jaguar" en 180 nm



Nota. La imagen muestra el layout del circuito integrado "El Gran Jaguar" con tecnología de 180 nm de TSMC.

Como siguiente punto, se incorporó de nuevo la comprobación del funcionamiento del nanochip a través del FPGA Xilinx Virtex-5 Genesys pero esta vez utilizando la Raspberry Pi 3B+. Se simplificó el *hardware* al eliminar el uso de la Tiva C para enviar los caracteres por medio de comunicación serial (UART), implementando *multithreading* y *multiprocessing* para la lectura de los caracteres [27].

Finalmente, se realizaron mejoras en la instalación de aplicaciones de Synopsys y se incorporó el uso de de *pipeline* en Jenkins para automatizar a futuro la síntesis lógica y física de circuitos [24] [28].

CAPÍTULO 3

Justificación

El diseño de circuitos integrados digitales constituye un pilar fundamental dentro del área de la microelectrónica, particularmente en el entorno académico, donde la formación en metodologías profesionales de diseño VLSI resulta esencial para preparar ingenieros capaces de enfrentar los desafíos actuales de la industria de semiconductores. Esta industria, dominada por un reducido conjunto de empresas y países con acceso privilegiado a tecnología, conocimiento y herramientas especializadas, presenta barreras significativas para instituciones educativas, especialmente en Hispanoamérica. En este contexto, el acceso del Departamento de Ingeniería Electrónica, Mecatrónica y Biomédica de la Universidad del Valle de Guatemala a herramientas EDA profesionales de Synopsys representa una oportunidad estratégica para desarrollar capacidades locales en el diseño de ASIC.

El presente trabajo se justifica por la necesidad de fortalecer y modernizar el flujo de diseño aplicado al nanochip "El Gran Jaguar", integrando prácticas alineadas con las recomendaciones de Synopsys y corrigiendo limitaciones observadas en investigaciones previas. En primer lugar, se mejora la metodología de síntesis lógica mediante la creación de un *script* en TCL que automatiza y estandariza el proceso, optimiza el proceso de síntesis y soluciona fallas presentes en implementaciones anteriores. Asimismo, se desarrolla un *script* adicional en Bash que permite la descarga e instalación automática y funcional del Synopsys Installer y aplicaciones adicionales, resolviendo los problemas que habían impedido la automatización en trabajos previos. Este aporte incrementa la eficiencia del flujo de trabajo y facilita la preparación de futuros entornos de diseño.

En segundo lugar, el trabajo contribuye a la validación experimental del *core* del nanochip en su adaptación a la tecnología de 65 nm de TSMC mediante su implementación en un FPGA Xilinx Virtex-5 Genesys. Esta etapa permite demostrar con evidencia empírica que el *gate-level netlist* sintetizado opera correctamente en hardware real, confirmando la validez del flujo adoptado. Un resultado de particular relevancia es el funcionamiento adecuado del *ring oscillator*, un componente que había presentado inconsistencias en tesis anteriores y que en esta ocasión muestra un comportamiento coherente con su diseño teórico.

En conjunto, esta tesis responde a la necesidad de establecer un flujo de diseño más confia-

ble, reproducible y fundamentado para el desarrollo de ASIC dentro del entorno académico. El trabajo no solo garantiza el correcto funcionamiento del nanochip "El Gran Jaguar", sino que también aporta herramientas y documentación que fortalecen la capacidad del laboratorio para futuros proyectos. De esta manera, contribuye al desarrollo del conocimiento local en diseño de circuitos integrados digitales y promueve el uso eficiente de herramientas industriales en instituciones de educación superior.

CAPÍTULO 4

Objetivos

4.1. Objetivo general

Mejora del proceso de síntesis lógica para cualquier circuito integrado descrito en Verilog, alineándolo con el flujo de diseño recomendado por Synopsys, documentando cada etapa del flujo del *Front-End* de forma detallada. Además, se busca facilitar la instalación de las aplicaciones de Synopsys para eficientizar el tiempo para configurar el entorno de trabajo. También se propone validar el funcionamiento de los circuitos mediante herramientas de Synopsys, para finalmente en un FPGA, realizar pruebas experimentales implementando el nanochip "El Gran Jaguar".

4.2. Objetivos específicos

- Replicar el avance del proyecto del nanochip realizado en años anteriores.
- Mejora en la automatización de la fase de síntesis lógica de un circuito integrado con tecnología de 65 nm, alineándola con las recomendaciones de Synopsys para esta parte del flujo de diseño.
- Colaborar con el equipo de trabajo encargado de realizar las pruebas físicas ERC, LVS, antena y pruebas finales, proporcionándoles los circuitos sintetizados lógicamente y todos los archivos necesarios de entrada que necesiten para dichos procesos.
- Realizar mejoras en el proceso de instalación de las aplicaciones de Synopsys para su automatización.
- Comprobar la funcionalidad del RTL de los circuitos a través de Synopsys Verdi y la validez del nanochip "El Gran Jaguar" implementándolo en un FPGA Xilinx Virtex-5 Genesys.

- Describir en Verilog nuevas celdas en su adaptación a la tecnología de 65 nm en un FPGA Xilinx Virtex-5 Genesys para la validación del funcionamiento del nanochip.
- Generar una documentación detallada del proceso de diseño del *Front-End* de un circuito integrado y su implementación en el FPGA Xilinx Virtex-5 Genesys para la realización de pruebas experimentales.

CAPÍTULO 5

Alcance

El alcance de este trabajo de graduación se centra en la mejora de la metodología para la síntesis lógica de los circuitos integrados digitales descritos en Verilog, basándose en el flujo de diseño recomendado por Synopsys. Se utilizan herramientas EDA (*Electronic Design Automation*) de Synopsys y librerías de TSMC para la tecnología de 65 nm. Durante este proceso se utiliza Python, VCS, Verdi, Design Compiler y Library Compiler de la empresa Synopsys. Aunado a ello, se emplean herramientas de verificación, para la validación de los diseños mediante simulaciones. Adicionalmente, el diseño del nanochip "El Gran Jaguar" se verifica a través de la emulación del *core* en el FPGA de Xilinx Virtex-5 Genesys, con el fin de probar el funcionamiento lógico del circuito integrado antes de su fabricación. El trabajo se enfoca en la documentación detallada de cada etapa del flujo de diseño del *Front-End*, desde la generación del RTL (*Register Transfer Level*) hasta la verificación final del nanochip implementado en el FPGA. Además, se busca mejorar la automatización del proceso de instalación de las aplicaciones de Synopsys para optimizar el tiempo de trabajo. En esta ocasión, se mejora el *script* debido a que el desarrollado en trabajos previos no funciona correctamente al momento de instalar las aplicaciones de Synopsys.

Dicho trabajo se desarrollará desde enero de 2025 hasta noviembre de 2025. El alcance incluye el dominio y aplicación correcta de las herramientas Synopsys para el flujo de diseño de un ASIC, aspecto que representa un desafío significativo dado que trabajos previos han presentado limitaciones en su uso, lo cual afectó la calidad de los resultados. Para superar estas deficiencias, el presente trabajo contempla:

- Una comprensión profunda del flujo de diseño para ASIC recomendado por Synopsys.
- El correcto uso de las herramientas EDA de Synopsys.
- Una metodología rigurosa de verificación para cada diseño.
- Documentación detallada de las mejores prácticas identificadas.

De esta manera, se busca establecer una base sólida para futuras investigaciones en el área de diseño de circuitos integrados digitales.

CAPÍTULO 6

Marco teórico

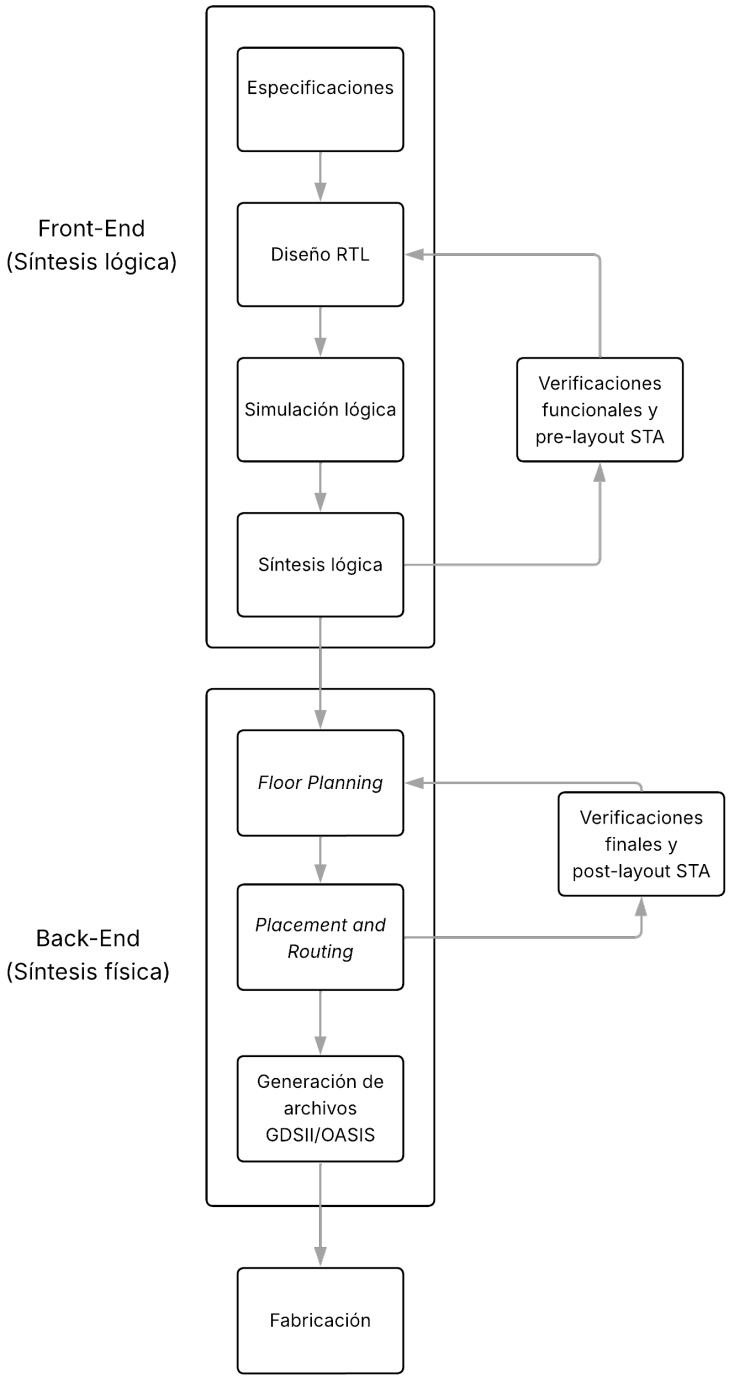
6.1. Flujo de diseño para ASIC (*Application Specific Circuit*)

La concepción de un Circuito Integrado de Aplicación Específica (ASIC, por sus siglas en inglés) demanda la ejecución de una secuencia de procedimientos rigurosamente estructurados, conocidos como flujos de diseño. Dichos flujos de diseño pueden ser completamente personalizados para satisfacer las necesidades específicas de un proyecto, o bien, pueden seguir metodologías semipersonalizadas ampliamente aceptadas en la industria [29].

Los flujos semipersonalizados pueden ser realizados en base a celdas o arreglos. Donde el flujo basado en celdas permite mayor optimización en cuanto a área, rendimiento y potencia. Mientras que el basado en arreglos se crea en base a bloques lógicos prediseñados, lo que reduce el tiempo de diseño y costos.

A continuación, se presenta el flujo de diseño para un ASIC digital propuesto por la empresa Synopsys [29]:

Figura 11. Flujo de diseño para un ASIC digital



Nota. En la presente imagen se muestra el flujo de diseño para un ASIC digital propuesto por Synopsys [29] [30].

Este marco metodológico resulta esencial para los arquitectos de circuitos integrados,

ya que dirige con precisión el complejo proceso de creación de un ASIC, garantizando que las etapas de diseño y fabricación cumplan con especificaciones estrictas que aseguran un funcionamiento impecable. Dicho Flujo de Diseño se segmenta en dos fases fundamentales: el *Front-End*, dedicado a la síntesis lógica, y el *Back-End*, enfocado en la síntesis física [29] [31] [32] [30].

6.1.1. Flujo de *Front-End*

La primera parte del flujo de diseño general presentado en la figura 11 contiene los y pasos necesarios para diseñar el funcionamiento del chip en lenguaje descriptivo de *hardware* [29] [31] [32] [30].

1. **Especificaciones:**
   Esta fase inicial exige la definición exhaustiva de las propiedades del chip, abarcando su estructura microarquitectónica, los requisitos funcionales y los parámetros de rendimiento, tales como consumo energético, área de silicio y restricciones temporales.

2. **Diseño RTL:**
   Posteriormente, el proceso avanza hacia la elaboración del modelo RTL (*Register Transfer Level*), una representación abstracta codificada en un HDL (*Hardware Description Language*), predominantemente Verilog, utilizando enfoques estructurales o de comportamiento. Este paso es seguido por una verificación mediante simulaciones destinadas a confirmar la integridad del diseño.

3. **Síntesis lógica:**
   Una vez validado el modelo RTL, se procede a la síntesis lógica, un proceso transformador mediante el cual el código RTL se convierte en una netlist a nivel de compuertas fabricables, empleando en este caso herramientas avanzadas de síntesis desarrolladas por Synopsys.

4. **Verificaciones funcionales:**
   Para verificar que el diseño sintetizado cumple con las especificaciones de diseño y *timing*, se llevan a cabo diversas pruebas, entre las que se incluyen:

   - **Simulación funcional:**
     Se realizan simulaciones para validar el comportamiento funcional del diseño sintetizado, asegurando que las salidas respondan correctamente a las entradas según lo especificado.
   - **Análisis de *timing*:**
     Se efectúa un análisis exhaustivo de temporización para garantizar que todas las rutas críticas del diseño cumplan con las restricciones de tiempo establecidas.
   - **Verificación formal:**
     Se emplean técnicas formales para comparar el diseño RTL original con la netlist sintetizada, asegurando que ambos sean funcionalmente equivalentes. También puede realizarse una verificación rigurosa por medio de una emulación en un FPGA.

6.1.2. Flujo de *Back-End*

Por otra parte, la segunda parte en la figura 11 hace referencia a la síntesis física, en otras palabras, la implementación física del circuito donde el *Netlist* de la síntesis lógica se transforma en un *layout* compuesto por varias celdas [29] [31] [32] [30].

1. ***Floor Planning*:**
   Durante este proceso el *netlist* a nivel de compuertas se transfiere a un diseño físico sobre el silicio. Este procedimiento implica la disposición estratégica del espacio de trabajo (*layout*) para optimizar el uso del área y minimizar los retrasos inducidos por las interconexiones.

2. ***Placement and Routing*:**
   Posteriormente, se lleva a cabo la fase de colocación y enrutamiento, que consiste en la ubicación precisa de celdas estándar y la configuración de interconexiones, considerando minuciosamente las restricciones de temporización, la optimización de la longitud de los cables y la disipación de potencia.

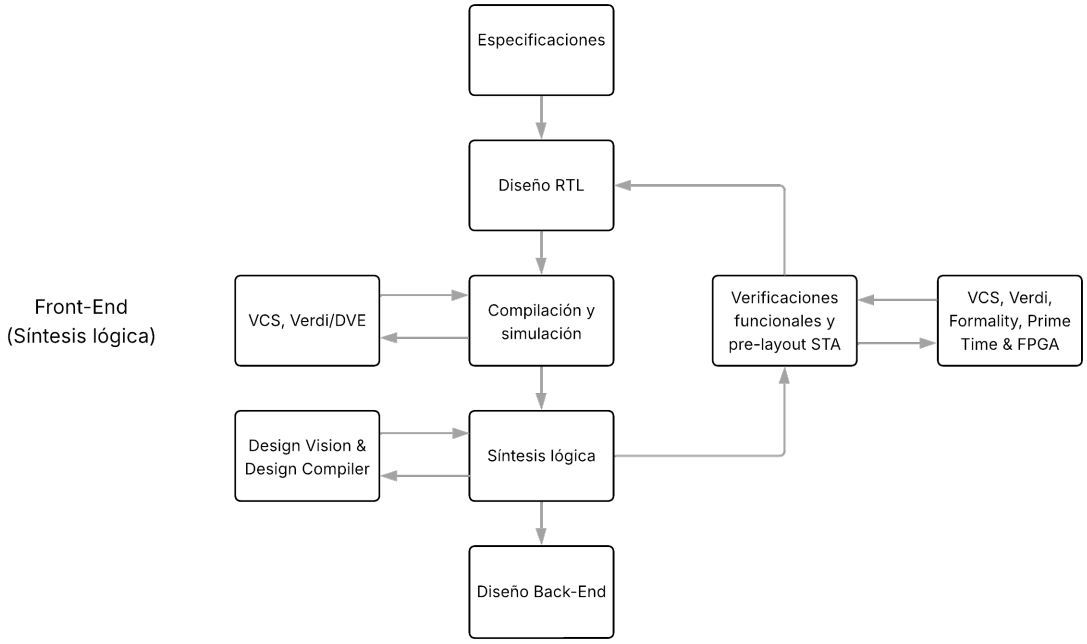
3. **Verificaciones finales:**
   Debe realizarse la ejecución de diversas pruebas destinadas a validar la conformidad del diseño físico con la funcionalidad. Además, se verifica si el circuito es fabricable:

   - **DRC (*Design Rule Check*):**
     Garantiza el cumplimiento de las restricciones de diseño para evitar anomalías en la fabricación.
   - **LVS (*Layout vs Schematic*):**
     Confirma la correspondencia entre el diseño físico y la *netlist* lógica.
   - **ERC (*Electrical Rule Check*)**
     Valida la corrección de las conexiones eléctricas.
   - **ARC (*Antena Rule Check*):**
     Prueba utilizada para prevenir posibles daños a los transistores durante la fabricación al abordar la acumulación de carga en las interconexiones metálicas.
   - ***Parasitic Extraction*:**
     Modela y analiza los efectos electromagnéticos derivados de elementos parásitos para verificar las características eléctricas. Se obtienen los parámetros eléctricos de todos los elementos del *layout*.

4. ***Sign-off* y *Tape-out*:**
   Esta es la fase final, donde el diseño del circuito integrado se exporta en formatos estandarizados, como GDSII (*Graphics Data System II*) u OASIS (*Open Artwork System Interchange Standard*).

6.2. Herramientas de Synopsys

Figura 12. Flujo de diseño del *Front-End* propuesto por Synopsys [30] [33]



Nota. En la imagen se muestra el flujo de diseño del *Front-End* con las herramientas de Synopsys que se utilizan en cada etapa del flujo.

En la figura 12 se presenta un flujo de diseño de síntesis lógica propuesto por la empresa Synopsys para el diseño de un ASIC. A la par de los pasos se encuentran el nombre de las herramientas de Synopsys que se utilizan en cada etapa del flujo.

Por otra parte, existe la familia de herramientas TestMAX que se utilizan para la generación automática de patrones de prueba (ATPG, por sus siglas en inglés) y el diseño para prueba (DFT, por sus siglas en inglés). En caso de ser necesario, dicho conjunto de herramientas se utilizaría en la etapa de síntesis lógica.

6.2.1. Verilog Compiler Simulator (VCS)

Una herramienta de vanguardia para el análisis, compilación y simulación de diseños basados en HDL, VCS agiliza la verificación del sistema al permitir una validación funcional integral antes y después de la síntesis Además, permite comprobar su funcionamiento antes y después de la elaboración de la síntesis lógica. Por medio de una interfaz gráfica permite observar el comportamiento del circuito para determinar si funciona correctamente.[34].

Por otra parte, para poder realizar simulaciones en VCS es necesario contar con un

archivo de *testbench* en formato *Verilog* que contenga las señales de entrada y salida del circuito. Además, es necesario que contenga las siguientes líneas de código:

Cuadro 1. Generación de simulación VCS

```
$fsdbDumpfile("file_name.fsdb");
$fsdbDumpvars(0, file_name_tb);
$dumpfile("file_name.vcd");
$dumpvars(0, file_name_tb);
```

La primera línea de código del Cuadro 1 genera aun archivo .fsdb con el nombre file_name.fsdb, donde se almacenará información sobre la simulación. La línea posterior indica al simulador que incluya las señales de la instancia 'file_name_tb' en el archivo .fsdb. Luego, se genera un archivo file_name.vcd que contiene el waveform de la simulación, por último se le indica al simulador que incluya las señales de la instancia 'file_name_tb' en el archivo .vcd.

Para compilar y simular un diseño realizado en lenguaje descriptor de *hardware*, se debe utilizar el siguiente comando:

Cuadro 2. Comando para compilación y simulación en VCS

```
vcs -full64 -debug_access+all -kdb -o name_simulacion
   file_name_tb.v file_name.v -v lib_name.v
```

Al ejecutar este comando se genera un archivo ejecutable con el nombre `name_simulacion`. En el caso de no indicar el nombre del archivo ejecutable con el comando `-o name_simulacion`, por defecto se llamará `./simv`.

El siguiente paso es escribir el siguiente nombre del archivo ejecutable en la terminal:

Cuadro 3. Ejecución de la simulación con nombre propio

```
./name_simulacion
```

En caso de no haber indicado el nombre del archivo ejecutable, se debe escribir:

Cuadro 4. Ejecución de la simulación con nombre genérico

```
./simv
```

A continuación se explica los detalles del comando de compilación y simulación del Cuadro 2:

- **vcs:** comando que hace referencia a la herramienta de Synopsys para compilar y simular lenguajes descriptores de *hardware*.
- **-full64:** indica que se utilizará la versión de 64 bits del compilador.
- **-debug_access+all:** permite el acceso completo a las variables de depuración durante la simulación.
- **-kdb:** genera la Verdi Knowledge Database (KDB) para la depuración avanzada en Verdi.
- **-o name_simulacion:** especifica el nombre del archivo ejecutable que se generará.
- **file_name_tb.v:** es el archivo que contiene el *testbench* del diseño.
- **file_name.v:** es el archivo que contiene el diseño en Verilog para su compilación y simulación.
- **-v:** especifica que el archivo que se coloca a continuación es una librería en formato Verilog.
- **-lib_name.v:** es el archivo de la librería en formato Verilog.

6.2.2. Library Compiler

Herramienta utilizada para la lectura, gestión y conversión de bibliotecas de celdas estándar, esencial para la síntesis lógica y el diseño físico. La información de caracterización de las celdas se encuentra en archivos en formato *Liberty* (.lib), la herramienta lee la información descrita en el archivo .lib y compila la descripción en una base de datos interna (formato de archivo .db). El archivo .db se construye en binario, sin embargo, para las herramientas de Synopsys es más eficiente trabajar con este formato que con el formato .lib. La diferencia entre ambos formatos es que el .lib es un archivo de texto que puede leer un humano, mientras que el .db está en lenguaje de máquina [35].

La información de la base de datos compilada (.db) se utiliza para *timing*, potencia, análisis de ruido con una síntesis compatible, *place and route*, *static timing analysis* y herramientas de verificación [35].

Con esta herramienta es posible realizar distintas acciones para obtener información de la librería a utilizar. A continuación, se muestran algunos comandos de ejemplo:

Cuadro 5. Comando para iniciar Library Compiler

```
lc_shell
```

Cuadro 6. Comandos para leer información de una librería

```
lc_shell> read_lib <lib_name.lib>
o bien:
lc_shell> read_db <lib_name.db>
```

Cuadro 7. Comando para desplegar la información de una librería

```
lc_shell> report_lib <lib_name>
```

Cuadro 8. Comando para desplegar información de *timing* de una celda dentro de la librería

```
lc_shell> report_lib <lib_name> <lib_cell> -timing
```

Cuadro 9. Comando para desplegar el atributo de unidad de tiempo de la librería

```
lc_shell> get_attribute [get_libs <lib_name>] time_unit
```

Tomar en cuenta que los nombres de los atributos pueden variar dependiendo de la librería.

Por otra parte, es posible utilizar los comandos de esta herramienta junto con Design Compiler durante el proceso de síntesis. Para ello, deben configurarse ambas herramientas para que trabajen en conjunto.

6.2.3. Design Compiler

Esencial para la síntesis de descripciones RTL en *netlists* optimizadas a nivel de compuertas, debido a ello, es una herramienta utilizada principalmente en la etapa de *Front-End*. No obstante las *netlists* a nivel compuerta son necesarias para la etapa de *Back-End* [36].

La capacidad del Design Compiler para ejecutar una optimización concurrente de parámetros críticos, tales como temporización, área, consumo de potencia y testabilidad, lo posiciona como una herramienta indispensable para cumplir con las estrictas especificaciones de diseño impuestas por los circuitos integrados de última generación. Entre sus innovaciones más destacadas se encuentra la integración de la tecnología topográfica, la cual permite modelar efectos físicos —como retardos de interconexión y congestión de enrutamiento— en una etapa temprana del flujo de diseño, específicamente durante la fase de síntesis [36].

Esta incorporación proactiva representa un cambio paradigmático en la metodología de síntesis, al abordar los desafíos del diseño físico de manera anticipada, lo que resulta en una reducción sustancial de las iteraciones costosas en etapas posteriores del proceso de diseño [36].

Cuadro 10. Comando para iniciar Design Compiler

```
dc_shell
```

En la vida real se construyen *scripts* en TCL para la automatizar la realización de la síntesis lógica. Para ello, se ejecuta el siguiente comando desde la terminal:

Cuadro 11. Comando para ejecutar un *script* en Design Compiler

```
dc_shell -f <script_name.tcl>
```

6.2.4. Design Vision

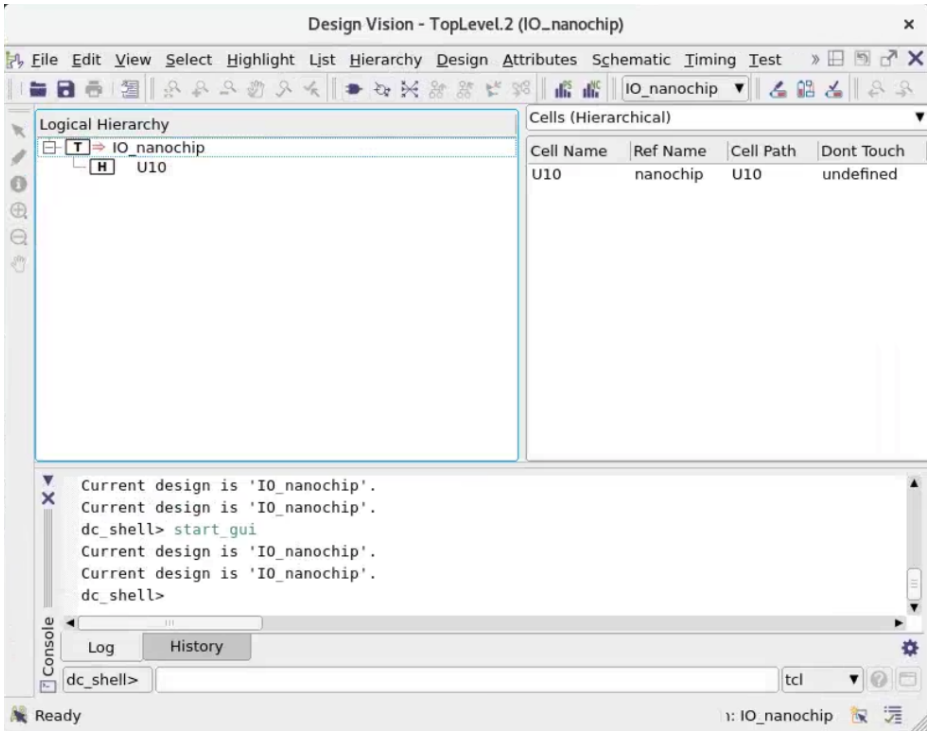
Esta plataforma de visualización y análisis de *netlists* permite un escrutinio detallado de la arquitectura a nivel de compuertas, con soporte para interfaces gráfica y de línea de comandos que ofrecen flexibilidad en la verificación [37].

Design Vision ofrece un entorno visual integral diseñado para la inspección detallada de diseños RTL, la definición y revisión precisa de restricciones, y el análisis exhaustivo de los resultados de síntesis. Esta herramienta de visualización eleva significativamente la interacción del usuario y fomenta una comprensión más profunda del proceso de síntesis multifacético. A través de su interfaz intuitiva, Design Vision permite la navegación y manipulación fluidas de diversas representaciones del diseño, tales como esquemáticos y *Netlists*, al tiempo que proporciona acceso directo a funcionalidades esenciales como el análisis de temporización y las características de optimización de potencia [37].

Cuadro 12. Comando para iniciar la interfaz gráfica de Design Compiler (Design Vision)

```
dc_shell> start_gui
```

Figura 13. Interfaz gráfica de Design Compiler (Design Vision)



Nota. En la imagen se muestra Design Vision, donde se puede observar la jerarquía del diseño, las señales de entrada y salida, el esquema del circuito y las opciones de análisis.

6.2.5. Verdi

Synopsys Verdi destaca como una herramienta preeminente en el ámbito de la automatización de diseño electrónico (EDA), distinguida específicamente como una plataforma de gestión de la identificación de errores y verificación. Constituye un elemento fundamental dentro del extenso portafolio de herramientas de Synopsys dedicadas al diseño de circuitos integrados. Esta herramienta asume un rol pivotal en el proceso de verificación funcional dentro de la industria de los semiconductores. Su función primordial radica en asegurar que los diseños digitales de alta complejidad operen con precisión conforme a lo estipulado y satisfagan sus especificaciones funcionales antes de avanzar hacia la etapa de fabricación, la cual es tanto costosa como irreversible [38].

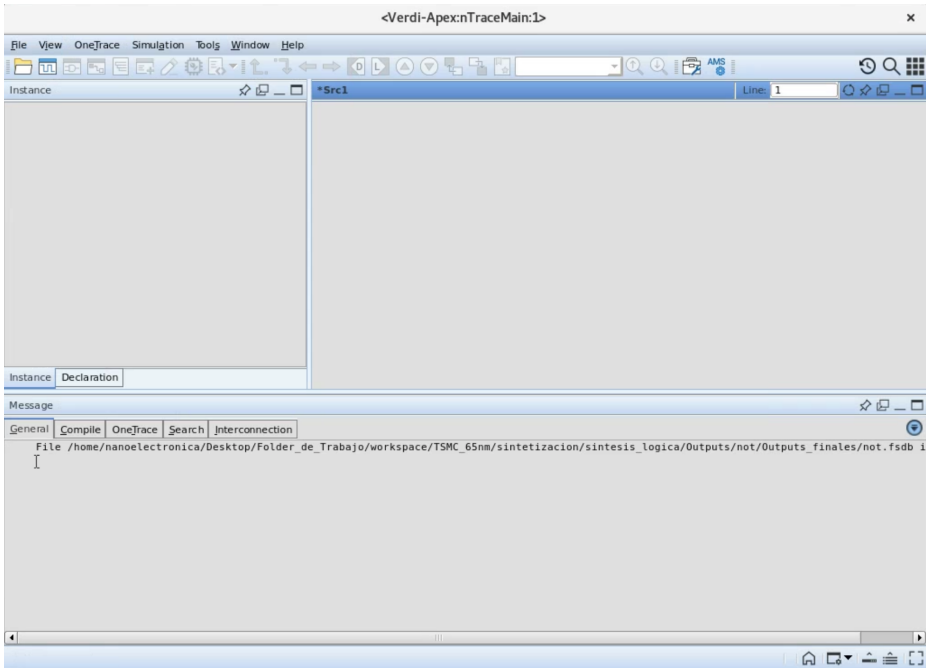
La herramienta posee compatibilidad con diversos lenguajes descriptores de *hardware* incluyendo Verilog, VHDL y SystemVerilog, lo que la convierte en una opción versátil para diversos ambientes de diseño. Las capacidades de primera clase mundial para la identificación de errores ofrecen características avanzadas de análisis y visualización. Su papel principal es ayudar a los ingenieros a identificar y resolver eficazmente las discrepancias entre el comportamiento esperado y el real del diseño lógico de un chip [38].

Cuadro 13. Inicialización de Verdi

```
verdi
```

En el Cuadro 14 se muestra el comando para iniciar la herramienta, una vez ejecutado se mostrará la interfaz gráfica de Verdi.

Figura 14. Interfaz gráfica de *Verdi*



Nota. En la imagen se muestra la interfaz gráfica de la herramienta Synopsys Verdi.

Por otro lado, Verdi puede integrarse con simuladores como VCS para proporcionar una experiencia de depuración más completa. Al combinar las capacidades de simulación de VCS con las herramientas de análisis y visualización de Verdi, los ingenieros pueden identificar y resolver problemas de diseño de manera más eficiente [38]. Para ello, es necesario cargar un archivo .fsdb generado por VCS en Verdi.

A continuación, se muestra el comando utilizado para cargar un archivo .fsdb en Verdi:

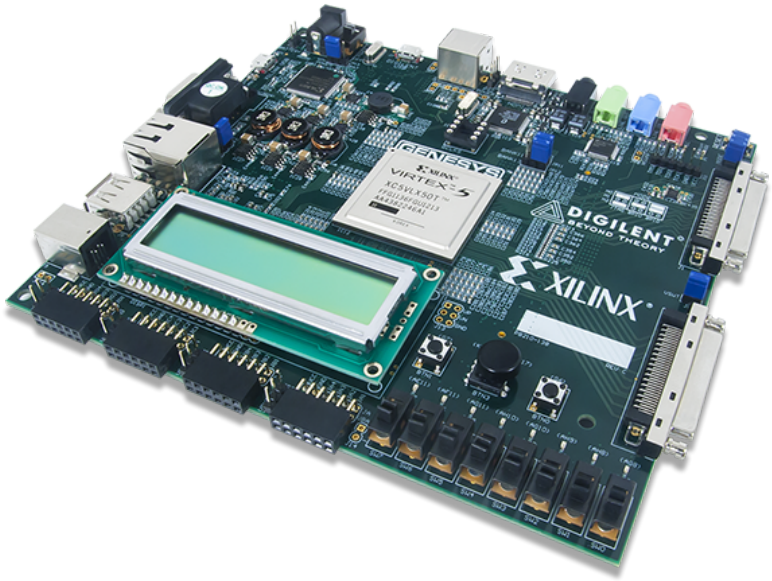
Cuadro 14. Carga de un archivo .fsdb a Verdi

```
verdi -ssf <file_name.fsdb>
```

Verificación experimental y uso del FPGA Xilinx Virtex-5 Genesys

Este FPGA constituye una herramienta de vanguardia en el ámbito de los circuitos digitales de alto rendimiento. Su diseño sobresale por una versatilidad excepcional, permitiendo la implementación de proyectos que transitan desde aplicaciones fundamentales hasta desarrollos de extrema complejidad. Este dispositivo está equipado con un FPGA de alta velocidad, memorias externas, puertos de video digital y una diversidad de opciones de expansión, todas caracterizadas por su elevado desempeño. Entre sus componentes se incluyen una pantalla LCD, memoria integrada y conectores multifuncionales como USB, audio y video, los cuales facilitan la ejecución de pruebas rigurosas y la validación funcional del nanochip "El Gran Jaguar" [39].

Figura 15. FPGA Xilinx Virtex-5 Genesys

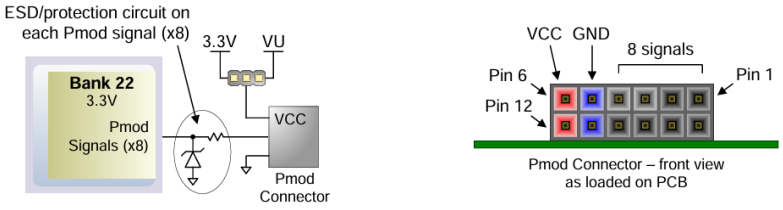


Nota. En la imagen se muestra la placa del FPGA Xilinx Virtex-5 Genesys.

Conectores Pmod

Los conectores Pmod representan una interfaz estandarizada para la integración de periféricos, habilitando pruebas físicas mediante su diseño adaptable. La placa dispone de 68 pines de Entradas/Salidas y cuatro puertos Pmod, cada uno equipado con ocho pines. De estos, dos están destinados a la alimentación (VCC de 3.3V) y dos a tierra (GND), como se muestra en la figura 3. Esta configuración optimiza la incorporación de módulos adicionales, expandiendo las capacidades funcionales del sistema [39].

Figura 16. Conectores Pmod del FPGA Xilinx Virtex-5 Genesys



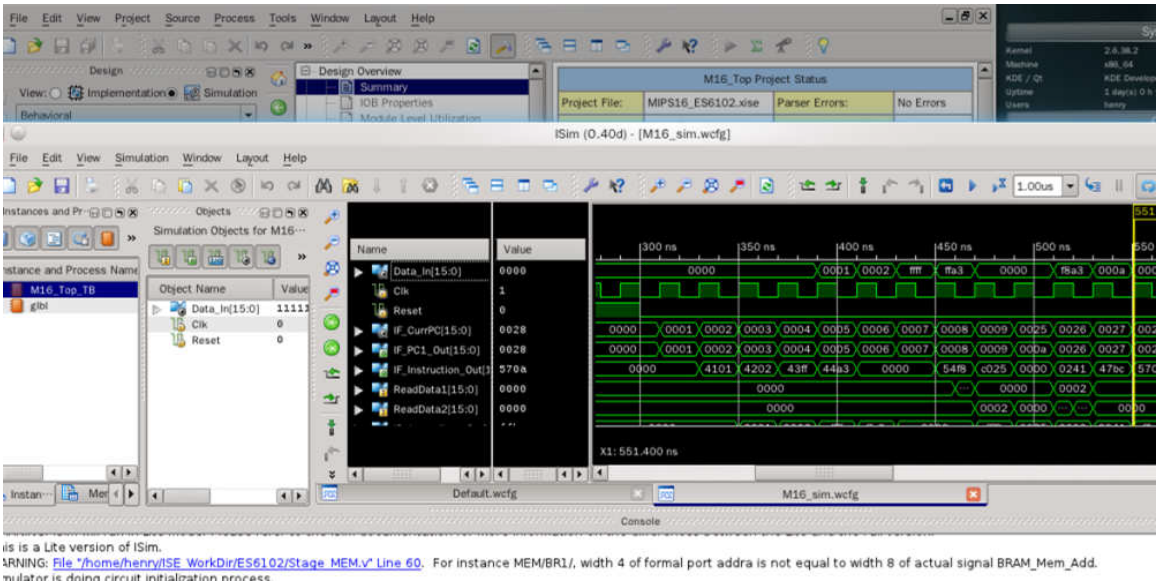
Nota. En la imagen anterior se muestran los conectores Pmod del FPGA Xilinx Virtex-5 Genesys.

Entorno Xilinx ISE

El entorno Xilinx ISE (*Integrated Synthesis Environment*) se erige como una herramienta referente en el diseño de lógica programable, permitiendo la síntesis y el análisis de descripciones en lenguajes HDL. Este sistema posibilita simulaciones que estiman retardos temporales y evalúan el comportamiento integral del diseño, consolidándose como un recurso esencial para la verificación previa a la implementación física. Las etapas fundamentales del flujo de trabajo —*Translate*, *Map*, *Place* y *Route*— garantizan una transición eficiente desde el código fuente hasta la configuración del FPGA. Adicionalmente, el simulador ISim, integrado en ISE, proporciona una interfaz gráfica para la visualización de formas de onda, optimizando la depuración y el análisis detallado del diseño [40] [41] [42].

Por otro lado, *ISim Simulator* es el simulador lógico dentro del entorno Xilinx ISE. La herramienta se utiliza para simular el comportamiento de un diseño y su reacción a diferentes estímulos. Los programas de banco de pruebas, escritos en lenguajes descripores de *hardware* como VHDL o Verilog, se emplean para proporcionar formas de onda de señales de entrada simuladas para monitorear y verificar las salidas del dispositivo bajo prueba. Con esta herramienta es posible identificar problemas de la síntesis del chip [43].

Figura 17. Entorno Xilinx ISE

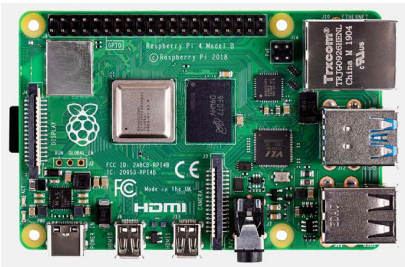


Nota. En la imagen se muestra la interfaz gráfica del entorno Xilinx ISE, donde se puede observar el esquema del circuito, las señales de entrada y salida, y las opciones de simulación.

Rapberry Pi 4B

La Raspberry Pi 4B es una plataforma de computación embebida de código abierto, basada en el sistema operativo Linux, con capacidades comparables a las de un ordenador convencional. Su arquitectura incluye el procesador de Broadcom BCM2711 basado en ARM Cortex-A72, de 1 GB a 8 GB de RAM dependiendo del modelo, conectividad Wi-Fi y Bluetooth y un puerto Ethernet. Esto la posiciona como una solución idónea para aplicaciones de procesamiento intensivo y flexibilidad periférica. Entre sus características se encuentran un puerto para tarjeta microSD, dos salidas HDMI para video y audio, conector JACK para audio, puertos USB y conectores MPI de *display*. Esta configuración la convierte en una herramienta óptima para el desarrollo de sistemas complejos y la interacción con el nanochip "El Gran Jaguar" en el FPGA Digilent Xilinx Virtex-5 Genesys [44] [45].

Figura 18. Raspberry Pi 4B



Nota. En la imagen se muestra la placa de la Raspberry Pi 4B.

CAPÍTULO 7

Archivos de TSMC de 65 nm

7.1. Librerías para las celdas de TSMC de 65 nm

Para comenzar con el trabajo de diseño de un ASIC es importante identificar las librerías necesarias para realizar la síntesis del circuito digital descrito en Verilog. En este caso específico se trabajará con celdas específicas de TSMC para la tecnología de 65 nm.

Para seleccionar las librerías es importante primero tomar en cuenta las celdas para el circuito integrado y las celdas para los pads de entrada y salida. Aunado a ello, tomar en cuenta el voltaje de operación que se desea utilizar, en este caso, 1.3 V para las celdas del circuito y 2.5 V para los pads. Asimismo, es fundamental que el *foundry* (TSMC en este caso) provea los archivos de descripción correspondientes a las celdas.

Es importante mencionar que los archivos de caracterización de las celdas son en formato `.lib` o `.db`. La diferencia es que el archivo `.lib` es un archivo de texto que contiene la información de las celdas en lenguaje humano legible, mientras que el archivo `.db` es un archivo binario que contiene la misma información pero en un formato más compacto y eficiente para la herramienta de síntesis.

A continuación se explica la identificación de las librerías de TSMC de 65 nm:

7.1.1. *Target Library*

Celdas para el circuito integrado

- `TSMC > 65 > CMOS > LP > stclib > 9-track > tcbn65lp-set > tcbn65lp_220a_FE > tcbn65lp_220a_nldm > TSMCHOME > digital >`

`Front_End > timing_power_noise > NLDM > tcbn65lp_220a > tcbn65lpwc0d9.db`

Celdas para los pads

- `TSMC > 65 > CMOS > LP > IO2.5V > iolib > LINEAR > tpdn65lpnv2od3_200a_FE > tpdn65lpnv2od3_200a_nldm > TSMCHOME > digital > Front_End > timing_power_noise > NLDM > tpdn65lpnv2od3_200a > tpdn65lpnv2od3wc5.db`

La *target library* es la librería principal, cuyo contenido es utilizado durante la compilación del circuito integrado para crear el *gate-level netlist* en base a una tecnología específica.

7.1.2. *Link Library*

La *link library* es utilizada para resolver instancias dentro del RTL (*Register Transfer Level*) del circuito integrado (el Verilog en este caso) o el *gate-level netlist*. Por convención, se utiliza la misma librería que la *target library*.

Es importante destacar el motivo por el cual se escogieron estas librerías. Toda librería es caractericada por un *process/voltage/temperature* (PVT) *corner*, también conocido como condición de operación. En este caso, se utilizaron librerías caracterizadas por el peor escenario de operación, ya que, si el circuito funciona correctamente bajo estas condiciones, se asegura que funcione en cualquier otro escenario de operación. A continuación se explica cómo identificar las condiciones de operación de las librerías:

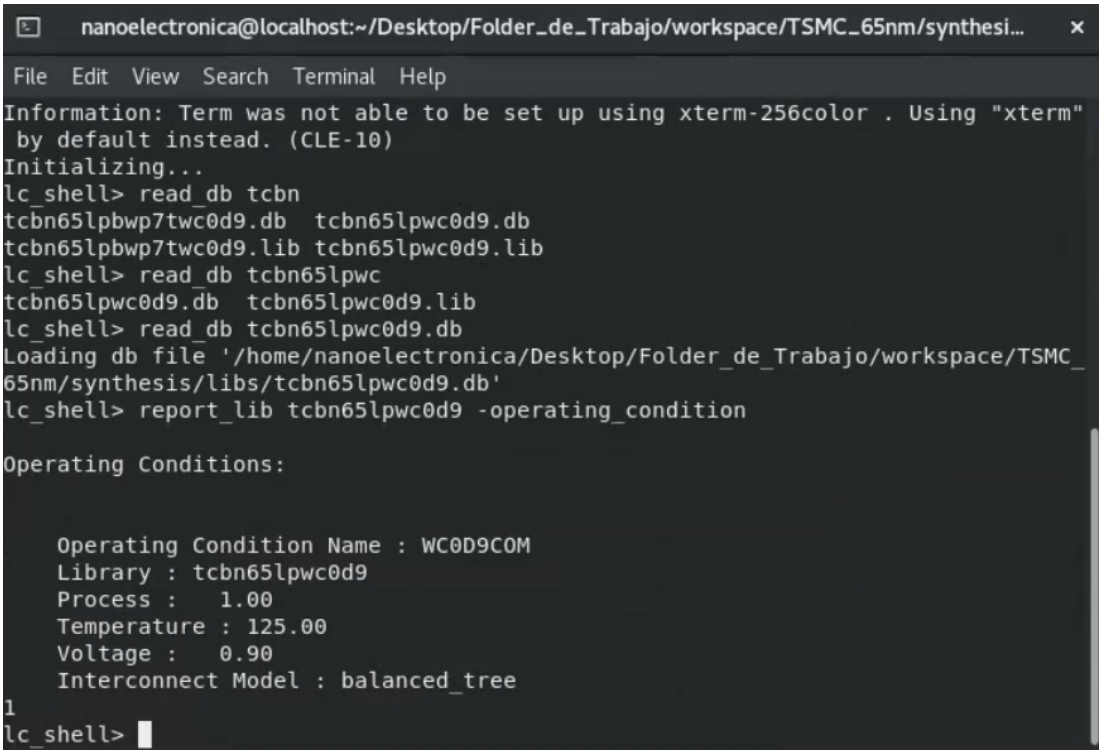
- tcbn65lpwc0d9: `wc` indica el *worst case*. `0d9` indica que el voltaje que se utilizó para la caracterización es de 0.9 V.
- tpdn65lpnv2od3wc5: `wc` indica el *worst case*. `5` indica que es la librería número 5 de los peores escenarios de operación.

Para comprobar la información de las condiciones de operación de las librerías, utilizar la herramienta de *Library Compiler* de Synopsys. A continuación se muestra los comandos a emplear en conjunto con capturas del resultado:

Cuadro 15. Comandos para observar condiciones de operación de la librería

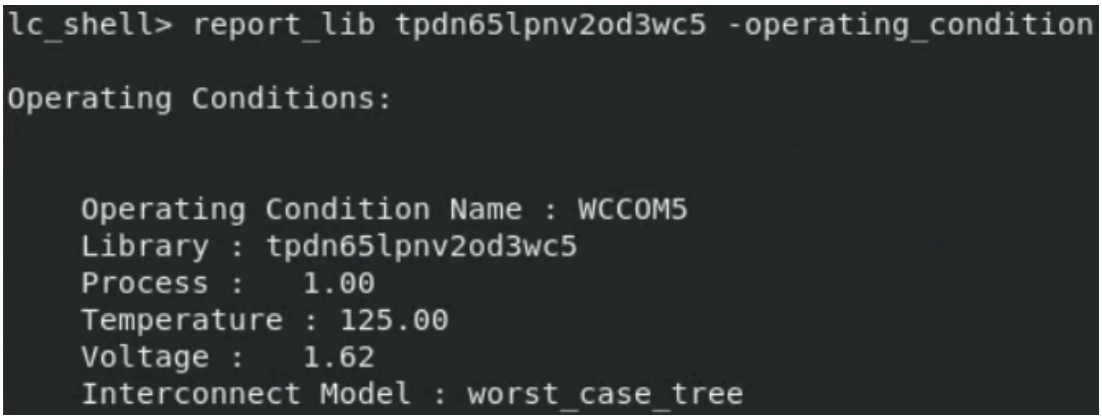
```
lc_shell
lc_shell> read_db file_name.db
lc_shell> report_lib file_name -operating_condition
```

Figura 19. Condición de operación de la librería tcbn65lpwc0d9



Nota. La imagen muestra las condiciones PVT de la librería tcbn65lpwc0d9.

Figura 20. Condición de operación de la librería tpdn65lpnv2od3wc5



Nota. La imagen muestra las condiciones PVT de la librería tpdn65lpnv2od3wc5.

Tanto para el caso de las celdas del circuito integrado como para las celdas de los pads, estas son las librerías de 65 nm con las peores condiciones de operación.

Importante tomar en cuenta que las librerías deben estar en un mismo directorio para que durante el proceso de síntesis la herramienta realice el mapeo de forma correcta. Más

adelante se explicará la estructura de directorios a emplear.

7.2. Archivos PDF de las librerías de TSMC de 65 nm

Es importante tener a disposición estos archivos que contienen información acerca de las celdas que se utilizarán para obtener el *gate-level netlist* del circuito integrado. Estos documentos proporcionan detalles clave sobre el funcionamiento, características eléctricas, características de *timing* y parámetros de las celdas, lo cual es esencial para la selección de la librería y comprensión del funcionamiento de sus celdas.

A continuación se muestra la ubicación de dichos archivos PDF para las celdas del circuito y pads:

PDF para las celdas del circuito integrado

- `TSMC > 65 > CMOS > LP > stclib > 9-track > tcbn65lp-set > tcbn65lp_220a_FE > tcbn65lp_200c_doc > TSMCHOME > digital > Documentation > documents > tcbn65lp_200c > DB_TCBN65LP_WC0D9.pdf`

PDF para las celdas de los pads

- `TSMC > 65 > CMOS > LP > IO2.5V > iolib > LINEAR > tpdn65lpnv2od3_200a_FE > tpdn65lpnv2od3_200a_doc > TSMCHOME > digital > Documentation > documents > tpdn65lpnv2od3_200a > tpdn65lpnv2od3wc5.pdf`

7.3. Archivos Verilog de las librerías de TSMC de 65 nm

Aunado a lo anterior, es importante contar con los archivos Verilog, tanto de las celdas del circuito integrado como de los pads. Estos archivos son necesarios para realizar la simulación del circuito integrado sintetizado, obtenido posteriormente a ejecutar los correspondientes comandos de la herramienta Design Compiler. Esto se debe a que el *gate-level netlist* generado por la herramienta hace referencia a las celdas de las librerías, y para simular el comportamiento del circuito es necesario tener la descripción de dichas celdas en Verilog.

A continuación se muestra la ubicación de los archivos Verilog para las celdas del circuito y pads:

Verilog para las celdas del circuito integrado

- `TSMC > 65 > CMOS > LP > stclib > 9-track > tcbn65lp-set > tcbn65lp_220a_FE > tcbn65lp_200a_vlg > TSMCHOME > digital > Front_End > verilog > tcbn65lp_200a > tcbn65lp.v`

Verilog para las celdas de los pads

- TSMC > 65 > CMOS > LP > IO2.5V > iolib > LINEAR > tpdn65lpnv2od3_200a_FE > tpdn65lpnv2od3_140b_vlg > TSMCHOME > digital > Front_End > verilog > tpdn65lpnv2od3_140b > tpdn65lpnv2od3.v

CAPÍTULO 8

Estructura de directorios

Para organizar los archivos necesarios para la síntesis del circuito integrado, es fundamental establecer una estructura de directorios clara y coherente. En este capítulo se explicará la estructura utilizada y recomendada, para que todos los *scripts*, RTLs y archivos de las librerías estén ubicados de manera intuitiva y accesible.

Anteriormente, se ha realizado la síntesis lógica y física de distintos circuitos integrados, principalmente "El Gran Jaguar". En el año 2024 el estudiante Kevin Hernández modificó la estructura de directorios para poder realizar la síntesis lógica de varios circuitos integrados, evitando la sobrescritura de los archivos generados y mejorando el nombramiento de las carpetas [24].

No obstante, a través del *LATAM Digital IC Design Bootcamp* realizado en la Ciudad de Guatemala en el año 2025, se tuvo la oportunidad de profundizar sobre las herramientas de Synopsys y las mejores prácticas para el diseño de circuitos integrados. Tomando en cuenta tanto el conocimiento disponible a través de SolvNetPlus y lo de este evento, se realizó una nueva estructura de directorios que mejora la organización y claridad de los archivos para llevar a cabo la síntesis lógica de cualquier circuito integrado descrito en Verilog (referencia jerarquia directorios). Con esta nueva estructura, los nombres de los directorios son más intuitivos y la navegación a través de los archivos es más comprensiva.

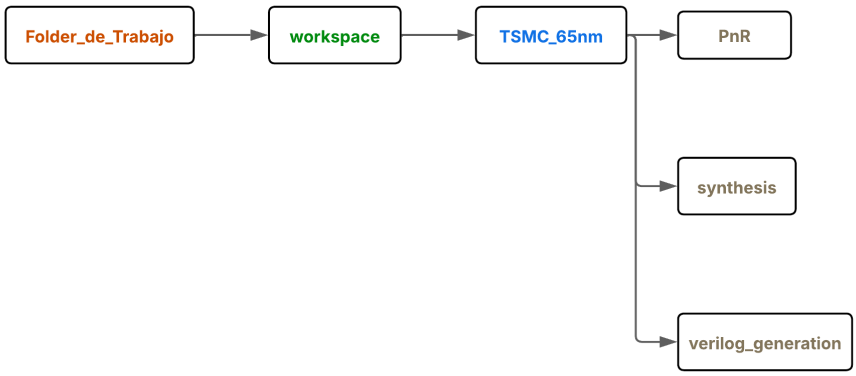
Esta nueva estructura de directorios general permite que se organicen las librerías y demás archivos relacionados con las etapas de *Front-End* (*synthesis*) y *Back-End* (*Placement and Routing*) por separado para evitar confusiones. De esta manera se favorece la claridad y organización del trabajo.

A continuación se muestra cómo se conforma la nueva estructura de directorios general, explicando la función de cada uno. Además, se realizará una mayor profundización en la estructura de directorios de la síntesis lógica.

8.1. Estructura de directorios general

A continuación se muestra la jerarquía general de directorios que se utilizó. Importante mencionar que únicamente se muestra el nombre de las carpetas:

Figura 21. Estructura de directorios general



Nota. La imagen muestra la jerarquía de directorios general utilizada para el diseño de circuitos integrados.

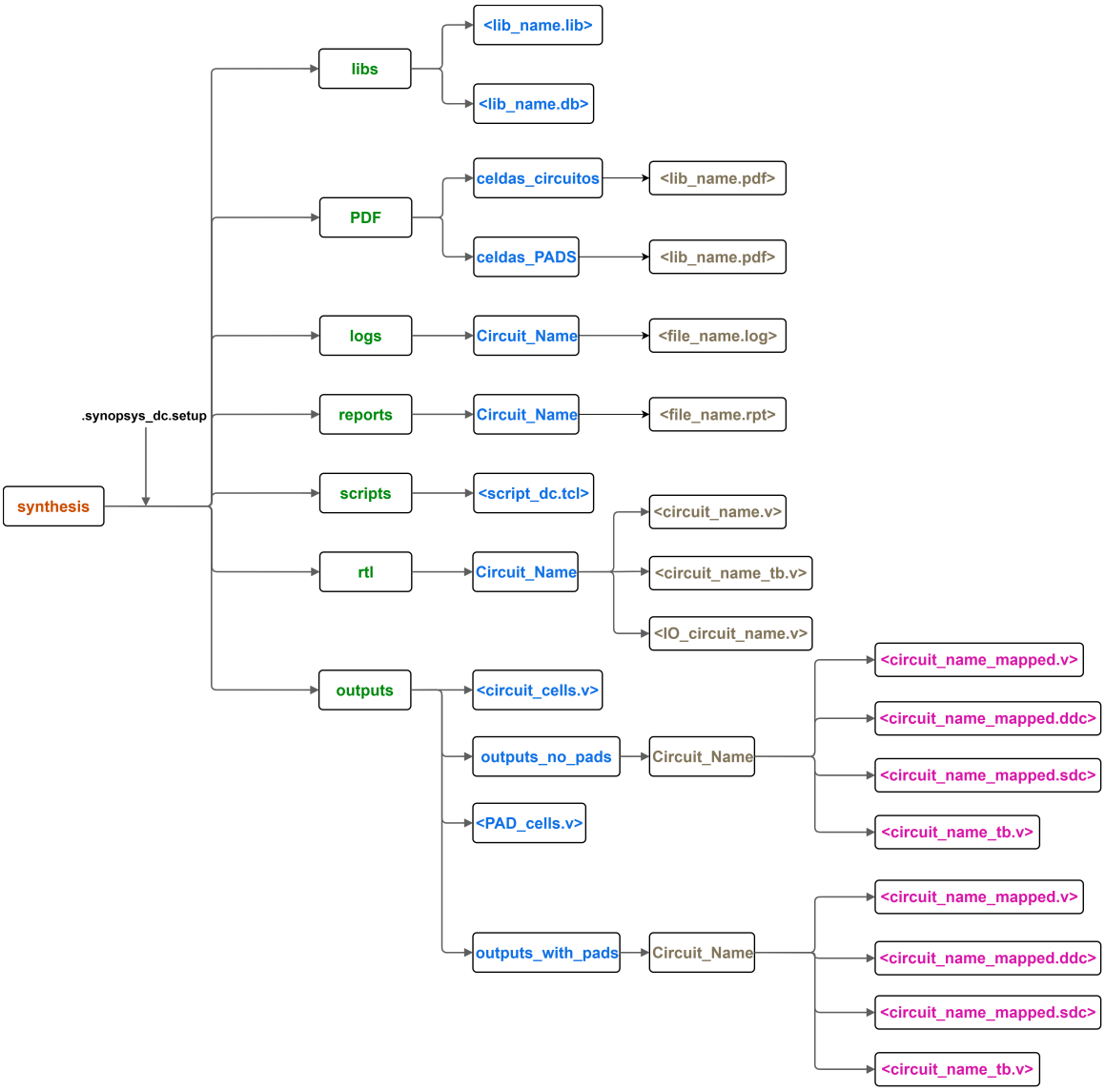
- **Folder_de_Trabajo:** espacio principal donde se almacenan todos los archivos relacionados con el proyecto de diseño del circuito integrado. Si se desean realizar otro tipos de pruebas, pueden crearse otro directorios dentro de esta carpeta. El objetivo es mantener organizado todo lo relacionado con el proyecto.
- ***workspace*:** el objetivo de dicho directorio es organizar por tecnología de nodo los archivos relacionados con la síntesis lógica y física del circuito integrado. en este caso, se trabajará con la tecnología de 65 nm de TSMC. De igual manera, pueden crearse otros directorios como "TSMC_180nm", "GF_22nm", etc.
- **TSMC_65nm:** dentro del directorio de la tecnología específica se encuentran todos los archivos necesarios para llevar a cabo la síntesis lógica y física del circuito integrado: desde las librerías hasta los archivos sintetizados.
  - **PnR:** este directorio contiene todos los archivos necesarios para realizar la síntesis física (también conocida como *Placement and Routing* en inglés) de cualquier circuito integrado. Esto incluye desde los subdirectorios, los *scripts* en TCL, *Runsets* hasta los GDSII generados. No obstante, en este trabajo no se realizará una profundización de esta parte debido a que el enfoque es la síntesis lógica.
  - **synthesis:** este directorio contiene todos los archivos necesarios para llevar a cabo la síntesis lógica de cualquier CI. En esta carpeta se encuentran subdirectorios para las librerías, los PDF de las librerías, *logs*, *scripts* en TCL, archivos RTL y los archivos de salida generados posteriormente a la síntesis. Importante mencionar que desde esta ubicación se ejecutará el *script* para realizar la síntesis. Más adelante se explicará con mayor detalle esta parte.

- **verilog_generation:** en esta ubicación se encuentra el archivo en Python 3 utilizado en la generación del archivo Verilog de "El Gran Jaguar".

A continuación se detalla la jerarquía de directorios para la síntesis lógica:

8.2. Estructura de directorios para la síntesis lógica

Figura 22. Estructura de directorios para la síntesis lógica



Nota. La imagen muestra la jerarquía de directorios utilizada para la síntesis lógica de diversos circuitos integrados.

Es de destacar que dentro del directorio *synthesis* se ejecutará el *script* para llevar a cabo la síntesis (misma ubicación del archivo .setup).

1. **.synopsys_dc.setup:** archivo de inicialización y configuración para la herramienta Design Compiler de Synopsys. Este archivo contiene parámetros, palabras reservedas y nomenclatura de TSMC. El archivo debe ubicarse en la carpeta principal de *synthesis* junto con otro archivo llamado TSMC_naming_rule.script. Tanto el archivo .setup como el .script son proporcionados por TSMC junto con sus librerías. Al ubicar el archivo de las librerías de pads .db, en el camino será notoria una carpeta adicional cuyo nombre es: *script*. Dentro de esta carpeta se encuentran ambos archivos.

2. **libs:** dentro de esta carpeta se ubicarán los archivos .db y .lib correspondientes a las librerías que se utilizarán para el proceso de síntesis lógica.

   - **<lib_name.lib>:** archivo de texto legible que contiene la información de caracterización de las celdas.
   - **<lib_name.db>:** archivo binario que resulta de la compilación del archivo .lib. Este archivo es al que se hace referencia al realizar la síntesis lógica.

3. **PDF:** carpeta donde se almacenan los archivos PDF de las librerías que contienen información relevante sobre las celdas que se utilizarán en la síntesis lógica. Se recomienda tener los archivos separados en celdas para el circuito y celdas para los pads para tener los archivos organizados de mejor manera.

   - **celdas_circuitos:** en este directorio se encuentran los archivos PDF de las celdas que se utilizarán para el circuito integrado.
     - **<lib_name.pdf>:** archivo PDF que contiene información relevante sobre las celdas que se utilizarán para el circuito integrado.
   - **celdas_PADS:** en este directorio se encuentran los archivos PDF de las celdas que se utilizarán para los pads de entrada y salida.
     - **<lib_name.pdf>:** archivo PDF que contiene información relevante sobre las celdas que se utilizarán para los pads de entrada y salida.

4. **logs:** directorio donde se van a almacenar los archivos *log* por nombre de circuito, esto para saber qué comandos se ejecutaron dependiendo del circuito sintetizado.

   - **Circuit_Name:** carpeta por nombre de circuito sintetizado donde se almacenan los archivos *log* generados.
     - **<file_name.log>:** archivo *log* generado por la herramienta Design Compiler que contiene un registro detallado de los comandos ejecutados durante el proceso de síntesis lógica del circuito integrado.

5. **reports:** directorio donde se almacenan los reportes generados por la herramienta Design Compiler, organizados por nombre de circuito.

   - **Circuit_Name:** carpeta por nombre de circuito sintetizado donde se almacenan los reportes generados.

        - **<file_name.log>:** archivo de reporte generado por la herramienta Design Compiler que contiene información detallada sobre el proceso de síntesis lógica del circuito integrado. Existen reportes previa a la síntesis sobre el diseño, puertos y *timing*, así como reportes posteriores a la síntesis sobre potencia, restricciones de diseño, *timing*, entre otros.

6. **scripts:** directorio donde se almacenan los *scripts* en TCL utilizados para llevar a cabo la síntesis lógica.

    - **<script_dc.tcl>:** archivo en TCL que contiene los comandos necesarios para ejecutar la síntesis lógica del circuito integrado. Más adelante se explicará de forma detallada el contenido de este archivo.

7. **rtl:** directorio donde se almacenan los archivos RTL en Verilog que describen el comportamiento de cada circuito integrado según su nombre.

    - **Circuit_Name:** carpeta por nombre de circuito integrado donde se almacenan los archivos correspondientes RTL en Verilog.
        - **<circuit_name.v>:** archivo en Verilog que describe el comportamiento del circuito integrado.
        - **<circuit_name_tb.v>:** archivo en Verilog que contiene el banco de pruebas (*testbench*) utilizado para simular y verificar el funcionamiento del circuito integrado.
        - **<IO_circuit_name.v>:** archivo en Verilog que une la instancia del respectivo circuito integrado con las instancias de los pads de entrada y salida.

8. **outputs:** directorio donde se almacenan los archivos generados posteriormente a la síntesis lógica, organizados por síntesis sin pads o con pads. Dentro de cada uno de estos subdirectorios, se crean carpetas por nombre de circuito.

    - **<circuit_cells.v>:** archivo en Verilog que contiene la descripción de las celdas estándar de TSMC utilizadas en el circuito integrado. Se ubica en este directorio ya que se hará referencia a este archivo al momento de realizar la verificación funcional y simulación del circuito integrado sintetizado.
    - **outputs_no_pads:** directorio que organiza los archivos de salida generados debido a la síntesis lógica de cada circuito integrado sin pads. Estos archivos son organizados por circuito integrado.
        - **Circuit_Name:** carpeta por nombre de circuito sintetizado donde se almacenan los archivos generados por la síntesis lógica del circuito integrado sin pads.
        - **<circuit_name_mapped.v>:** archivo en Verilog que contiene el *gate-level netlist* del circuito integrado sintetizado sin pads. Este archivo es generado por la herramienta Design Compiler después de mapear el diseño RTL a las celdas estándar de la librería.
        - **<circuit_name_mapped.ddc>:** archivo encriptado en formato binario de Synopsys que contiene el *design database* del circuito integrado sintetizado sin pads. Este archivo es generado por la herramienta Design Compiler y contiene el *gate-level netlist* junto con información adicional sobre el diseño, como las restricciones de diseño.

- **<circuit_name_mapped.sdc>:** archivo en formato de texto que contiene las restricciones de diseño para el circuito integrado sintetizado sin pads. Este archivo es generado por la herramienta Design Compiler y se utiliza para definir las restricciones de *timing* y otras restricciones relevantes para el diseño.
- **<circuit_name_tb.v>:** este archivo es el mismo que el archivo de banco de pruebas (*testbench*) ubicado en el directorio *rtl*. Se incluye aquí para facilitar la simulación y verificación del circuito integrado sintetizado sin pads.

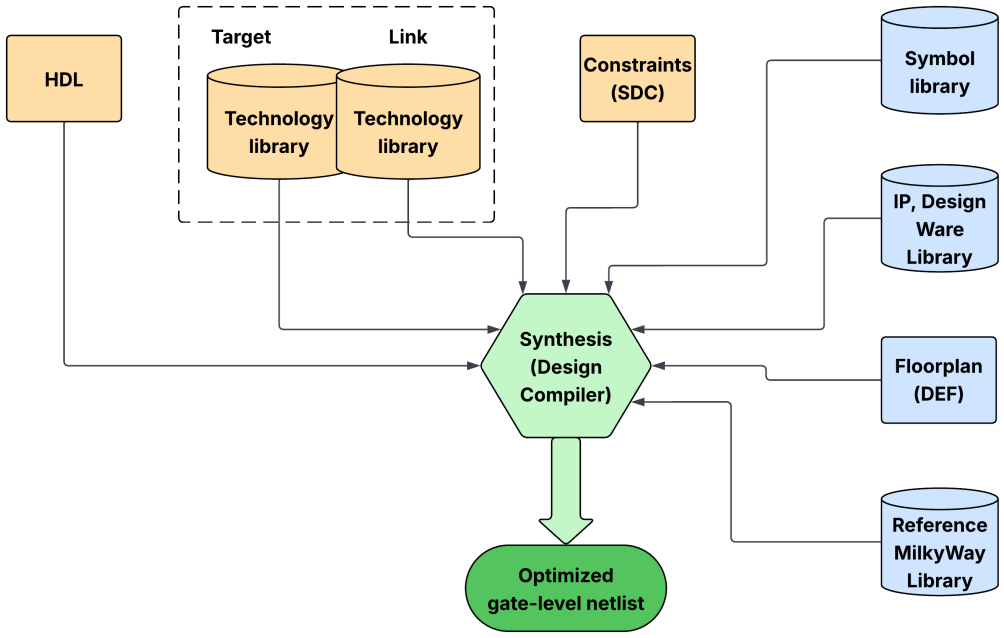
- **<PAD_cells.v>:** archivo en Verilog que contiene la descripción de las celdas de pads de TSMC utilizadas en el circuito integrado. Se ubica en este directorio ya que se hará referencia a este archivo al momento de realizar la verificación funcional y simulación del circuito integrado con pads sintetizado.
- **outputs_with_pads:** directorio que organiza los archivos de salida generados debido a la síntesis lógica de cada circuito integrado con pads. Estos archivos son organizados por circuito integrado.
  - **Circuit_Name:** carpeta por nombre de circuito sintetizado donde se almacenan los archivos generados por la síntesis lógica del circuito integrado con pads.
  - **<circuit_name_mapped.v>:** archivo en Verilog que contiene el *gate-level netlist* del circuito integrado sintetizado con pads. Este archivo es generado por la herramienta Design Compiler después de mapear el diseño RTL a las celdas estándar de la librería, incluyendo las celdas de pads.
  - **<circuit_name_mapped.ddc>:** archivo encriptado en formato binario de Synopsys que contiene el *design database* del circuito integrado sintetizado con pads. Este archivo es generado por la herramienta Design Compiler y contiene el *gate-level netlist* junto con información adicional sobre el diseño, como las restricciones de diseño.
  - **<circuit_name_mapped.sdc>:** archivo en formato de texto que contiene las restricciones de diseño para el circuito integrado sintetizado con pads. Este archivo es generado por la herramienta Design Compiler y se utiliza para definir las restricciones de *timing* y otras restricciones relevantes para el diseño.
  - **<circuit_name_tb.v>:** este archivo es el mismo que el archivo de banco de pruebas (*testbench*) ubicado en el directorio *rtl*. Se incluye aquí para facilitar la simulación y verificación del circuito integrado sintetizado con pads.

CAPÍTULO 9

Flujo de diseño para la síntesis lógica

Para realizar la síntesis lógica de cualquier circuito descrito en Verilog es fundamental entender qué se necesita como insumo y cuál es el resultado del proceso. A continuación se presenta un diagrama que ilustra los *inputs* y *outputs* necesarios para Design Compiler [36]:

Figura 23. Diagrama de *inputs* y *outputs* para la síntesis lógica



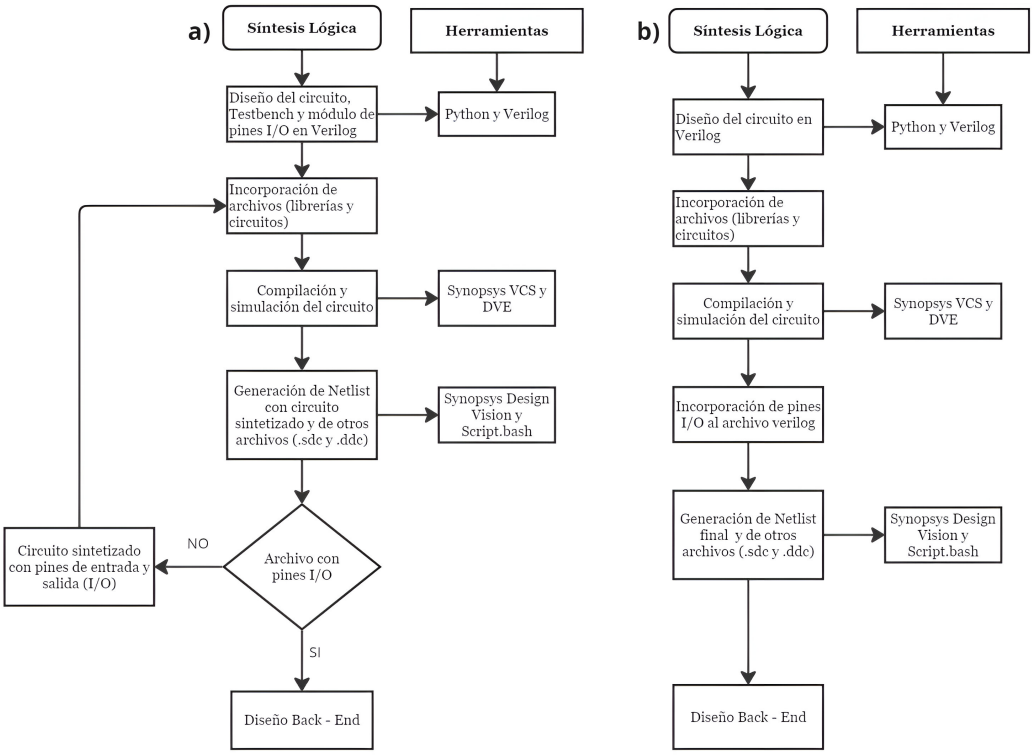
Nota. La imagen muestra los insumos que necesita Design Compiler y el *output* del *gate-level netlist*.

1. **HDL:** circuito descrito en VHDL, Verilog o SystemVerilog.
2. ***Target Library:*** librería principal utilizada para mapear el diseño RTL a las celdas estándar.
3. ***Link Library:*** librería utilizada para resolver instancias dentro del RTL o el *gate-level netlist*.
4. ***Constraints (SDC):*** archivo que contiene las restricciones de diseño, como las restricciones de *timing*.
5. ***Symbol Library:*** archivo para la representación gráfica de celdas y componentes.
6. ***IP, DesignWare Library:*** librería de Synopsys que contiene distintos componentes funcionales listos para incluir en diseños de mayor complejidad (memorias, componentes USB, etc.).
7. ***Floorplan (DEF):*** archivo que define la distribución física del *layout*.
8. ***Reference MilkyWay Library:*** librería cuyo formato de almacenamiento contiene información lógica y física acerca de componentes. Incluye información de funcionalidad, *routing*, dimensiones de celdas, etc.
9. ***Optimized gate-level netlist:*** archivo producto de la síntesis lógica que contiene la descripción del circuito integrado sintetizado en términos de celdas estándar.

Importante mencionar que no todos los *inputs* son obligatorios para realizar la síntesis lógica. En este caso, los primeros cuatro elementos de la lista anterior son los esenciales en esta etapa.

Una vez comprendidos los *inputs* y *outputs* necesarios para la síntesis lógica, es importante entender el flujo de diseño que se debe seguir para llevar a cabo este proceso. En años anteriores, el estudiante Kevin Hernández propuso un flujo de diseño para la síntesis lógica que optimiza el proceso de síntesis comparado al propuesto por el estudiante Noel Prado. La ventaja de este nuevo flujo es que la síntesis con pads se hace de forma automática con el script llamado: `ejecutar_sintesis_logica_2.bash`. Este *script* llama a otro archivo llamado: `comandos_sintesis_logica_2.script`, que contiene los comandos de Design Compiler para realizar la síntesis. Ya no se necesita sintetizar primero el *core* sin pads y luego incorporar los pads manualmente para volver a sintetizar como se hacía anteriormente con el flujo propuesto por el estudiante Noel Prado [24] [23].

Figura 24. Flujos de diseño para la síntesis lógica empleados en años anteriores



Nota. La imagen muestra: a)Flujo de diseño empleado por el estudiante Kevin Hernández [24]. b)Flujo de diseño empleado por el estudiante Noel Prado [23].

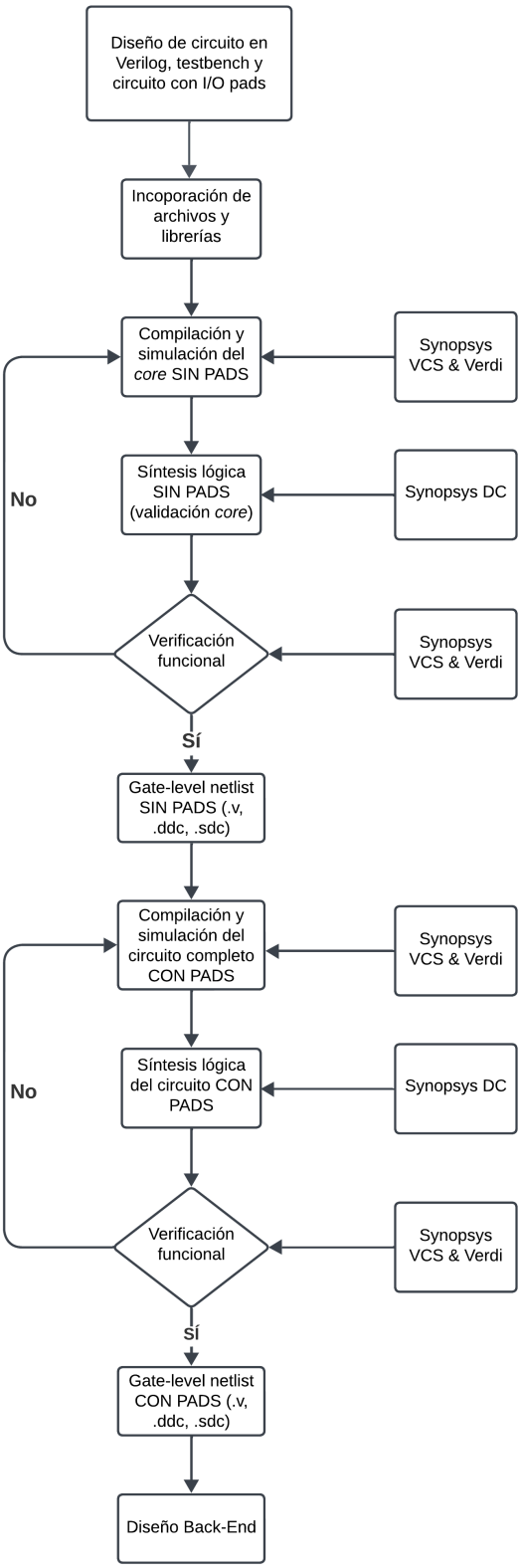
Si bien esta manera de sintetizar permite la identificación de problemas antes de incluir los pads, existen diversas desventajas en el flujo de diseño a):

- La utilización de dos archivos *bash* y *script* puede generar confusión al momento de ejecutar la síntesis lógica.
- La síntesis lógica no se está realizando en lenguaje TCL, que es el lenguaje nativo de las herramientas de Synopsys.
- El tomar el archivo sintetizado del *core* para incorporar los pads y luego volver a sintetizar no es lo más óptimo, ya que se debe sintetizar dos veces el mismo circuito.

Es posible conservar las ventajas del flujo de diseño a) pero optimizando el proceso de síntesis lógica. Para ello, se propone un nuevo flujo de diseño que utiliza únicamente un archivo en TCL para realizar la síntesis lógica.

Es importante mencionar que con este nuevo flujo de diseño no es necesario realizar la síntesis sin pads antes de la síntesis con pads. El usuario puede elegir si hacer la síntesis con o sin pads independientemente. Esto será evidente más adelante al explicar el *script* utilizado.

Figura 25. Flujo de diseño propuesto para la síntesis lógica



Nota. La imagen muestra el flujo de diseño propuesto en este trabajo.

Como se puede observar en la Figura 25, se coloca primero la síntesis sin pads para validar primero el funcionamiento del *core* del circuito integrado. Posteriormente, se realiza la síntesis con pads para obtener el *gate-level netlist* final que será utilizado en la etapa de *Back-End* con la herramienta de ICC2.

Es una buena práctica sintetizar primero solo el *core* para identificar posibles errores en el diseño del *core* antes de incorporar los pads. Esto permite conservar las ventajas del flujo de diseño anterior, pero sin la necesidad de sintetizar dos veces el mismo circuito para poder incorporar los pads.

CAPÍTULO 10

Preparación para la síntesis lógica

Antes de realizar la síntesis lógica de cualquier circuito integrado, es fundamental preparar todos los archivos necesarios y asegurarse de que estén correctamente organizados. Esto incluye la verificación de las librerías y la configuración del entorno de trabajo. Aunado a ello, es importante tener los archivos RTL del circuito integrado listos y el correspondiente *script* en TCL para ejecutar la síntesis lógica. Una adecuada organización y verificación de estos elementos garantiza que el proceso de síntesis se lleve a cabo sin contratiempos y con resultados reproducibles, minimizando errores.

Además, es esencial comprender que la fase de diseño a nivel RTL representa el punto de partida para la implementación física del circuito, por lo que las decisiones tomadas en esta etapa influyen directamente en la calidad del diseño final, su área, consumo de potencia y desempeño. Por ello, es importante dedicar tiempo a realizar diseños bien estructurados desde el inicio.

Como primer punto, en este capítulo se explicará el proceso de diseño de circuitos digitales a nivel RTL utilizando Verilog. Principalmente se realizará énfasis en el diseño del módulo del circuito como tal, su *testbench* y el módulo del circuito con los pads de entrada y salida. Estos son los tres archivos cruciales de diseño que se alojan en el directorio *rtl*, como se explicó en el Capítulo 8.

Por otra parte, se explicará el desarrollo del nuevo *script* en TCL utilizado para llevar a cabo la síntesis lógica de cualquier circuito digital descrito en Verilog. Se esclarecerá cómo se construye dicho *script* y se detallará el propósito de cada sección del mismo. Finalmente, se mostrará cómo se puede adaptar el *script* para realizar la síntesis de distintos circuitos digitales.

10.1. Diseño de circuitos digitales a nivel RTL

Como se mencionó anteriormente se explicará el proceso de diseño de tres módulos en concreto: el módulo del circuito digital, el *testbench* y el módulo del circuito con pads de entrada y salida. Se utilizará Verilog como lenguaje de descripción de hardware (HDL) al ser de los más empleados en la industria.

Es fundamental tener estos tres módulos antes de ejecutar la síntesis lógica, ya que son los archivos de entrada necesarios para que la herramienta Design Compiler pueda mapear el diseño RTL a las celdas estándar de la librería.

En este caso, se tomará como ejemplo el diseño de una compuerta NOT. Esto permitirá mostrar de forma clara cómo se construyen los módulos de los demás circuitos sintetizados posteriormente.

10.1.1. Diseño del módulo principal de un circuito digital en Verilog

Como se mencionó anteriormente, a lo largo de este trabajo se estará trabajando con Verilog. A continuación se presenta el diseño de una compuerta NOT:

Cuadro 16. Archivo Not.v: Diseño de una compuerta NOT en Verilog

```
module Not(A,Y);
        input A;
        output Y;
        assign Y = ~A;
endmodule
```

Es importante revisar el diseño para asegurarse que cumple con las especificaciones deseadas además de verifcar que se cumple con la sintaxis.

El diseño de un circuito en Verilog puede realizar de forma *behavioral* o *structural*. En este caso, se utilizó la descripción *behavioral* al emplear la sentencia `assign` para definir la salida en función de la entrada. Dependiendo de la complejidad del circuito y las preferencias del diseñador, se puede optar por una, la otra o incluso una combinación de ambas. Tomar en cuenta que para bloques combinacionales se hace uso del *blocking assignment* (símbolo =), mientras que para bloques secuenciales se utiliza el *non-blocking assignment* (símbolo <=).

Durante el proceso de síntesis la herramienta Design Compiler es capaz de reconocer ambos tipos de descripciones y mapearlas a las celdas estándar de la librería de TSMC de 65 nm. Ahora bien, si el programa no encuentra una celda que directamente represente la función lógica descrita en el RTL, la herramienta puede implementar la función utilizando una combinación de celdas disponibles en la librería.

10.1.2. Diseño del módulo *testbench* de un circuito digital en Verilog

Como parte del proceso de diseño de un circuito digital descrito en Verilog, es fundamental crear un módulo de *testbench* para verificar su funcionamiento. El *testbench* es un entorno de simulación que permite probar y validar el comportamiento del circuito antes de proceder a la síntesis lógica. A continuación se presenta el diseño del *testbench* para la compuerta NOT:

Cuadro 17. Archivo Not_tb.v: Diseño del *testbench* de una compuerta NOT en Verilog

```
`timescale  1ns/10ps
module Not_tb;
wire Y;
reg A;
Not compuerta(.A(A), .Y(Y));

initial
begin
        $fsdbDumpfile("Not.fsdb");
        $fsdbDumpvars(0, Not_tb);
        $dumpfile("Not.vcd");
        $dumpvars(0, Not_tb);
        A = 1'b0;
        #1;
        A = 1'b1;
        #1;
        A = 1'b0;
        #1;
        A = 1'b1;

end
endmodule
```

Importante destacar que se incluye al inicio del archivo la directiva `` `timescale `` para definir la unidad de tiempo y la precisión de la simulación. En este caso, se establece una unidad de tiempo de 1 nanosegundo y precisión de 1 picosegundo. El motivo por el cual se incluye esta directiva es debido a que este mismo archivo será utilizado para la simulación del *netlist* generado posterior a la síntesis lógica. Lo que sucede es que el archivo Verilog de las celdas estándar que proporciona TSMC contiene la directiva `` `timescale ``, por lo que es necesario que el *testbench* también la incluya para evitar errores al tratar de simular.

Verificar que las unidades de tiempo y precisión del *testbench* coincidan con las del archivo de las celdas estándar.

Por otro lado, al momento de diseñar un *testbench* se deben tomar en cuenta las siguientes consideraciones:

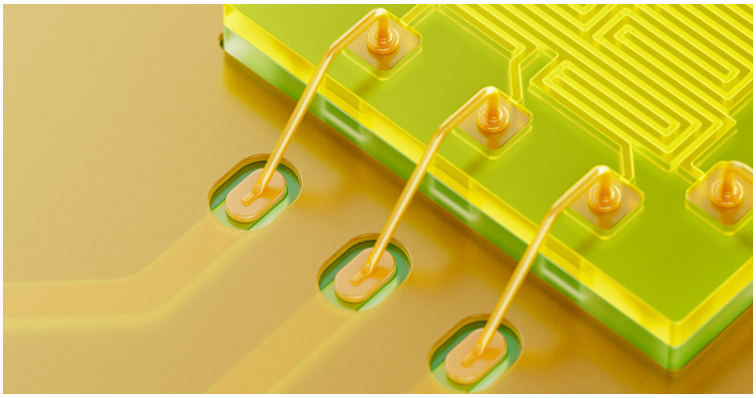
- **Definición de las señales:**
  - **Inputs:** para simular las señales de entrada se utilizan registros definidos con la palabra reservada `reg`. Esto debido a que los registros permiten almacenar valores y cambiar su estado durante la simulación.
  - **Outputs:** para simular las señales de salida se utilizan cables definidos con la palabra reservada `wire`. Esto debido a que los cables representan conexiones físicas y no almacenan valores.
- **Inicialización:** es importante inicializar todas las señales relevantes al inicio antes de comenzar la simulación. Esto asegura que el circuito comience en un estado conocido.
- **Simulación del tiempo transcurrido:** se utilizan retardos de tiempo mediante la sentencia `#` para simular el paso del tiempo y observar cómo responde el circuito a diferentes entradas.
- **Generación de archivos de simulación:** se utilizan sentencias como `$fsdbDumpfile`, `$fsdbDumpvars`, `$dumpfile` y `$dumpvars` para generar archivos VCD o FSDB respectivamente. Estos archivos de simulación permiten visualizar las señales en la simulación y así analizar el comportamiento del circuito.

10.1.3. Diseño del módulo de un circuito digital con pads en Verilog

Como se mencionó anteriormente, el *netlist* que se utilizará en la etapa de *Back-End* debe incluir las celdas de pads de entrada y salida. Por ello, es necesario diseñar un módulo en Verilog que integre el circuito digital con las instancias de los pads correspondientes para sintetizarlo.

Al integrarse los pads con el *core* del circuito en esta etapa facilita el diseño físico posterior. Los pads sirven como puntos de conexión entre el circuito integrado y el mundo exterior, permitiendo la comunicación con otros dispositivos y sistemas. En muchos circuitos integrados digitales se utiliza una técnica llamada *wire bonding* para conectar los pads a los pines de encapsulado del chip que permitirán la interacción con el entorno externo.

Figura 26. *Wire Bonding* de un circuito integrado

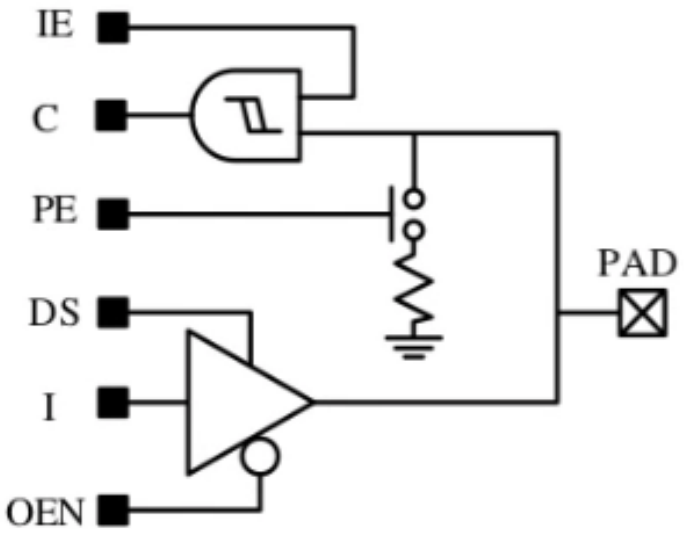


Nota. La imagen muestra el *wire bonding* entre los pads de un circuito y los *tracks* de una PCB [46].

Para el diseño de este módulo es necesario definir cuáles serán las celdas de pads que se utilizarán para las entradas y salidas del circuito digital. En este caso, los pads se deben escoger entre las librerías de pads que proporciona TSMC para la tecnología de 65 nm. La librería de pads que se utilizará se indicó en el Capítulo 7. A continuación se detalla qué celda se eligió y por qué:

La celda seleccionada es la PDDW0204SCDG. El módulo de Verilog correspondiente a esta celda se encuentra en el archivo `tpdn65lpnv2od3.v` y la información acerca de su funcionamiento y *timing* se encuentra en el archivo PDF `tpdn65lpnv2od3wc5.pdf`. Esto se detalla en el Capítulo 7.

Figura 27. Esquemático del pad PDDW0204SCDG



Nota. La imagen muestra el esquemático del pad a utilizar.

El motivo por el cual se elegió esta celda es debido a las siguientes razones que se encuentran en el PDF:

- Esta celda posee un Schmitt Trigger entre C y PAD. Esto ayuda a prevenir problemas de ruido y evitar oscilaciones.
- Comparada a las demás celdas, es la que menos capacitancia de pines posee, lo que ayuda a mejorar el desempeño del circuito.

Al hacer la comparación con las demás celdas de pads disponibles, esta era la que mejor cumplía con las observaciones ya mencionadas.

Ahora bien, estas celdas pueden funcionar como entradas o salidas dependiendo de la configuración de sus pines. A continuación se presenta la tabla de verdad de la celda:

Figura 28. Tabla de verdad del pad PDDW0204SCDG

Truth Table

INPUT						OUTPUT	
DS	OEN	I	PAD	PE	IE	PAD	C
0/1	0	0	-	0/1	0	0	0
0/1	0	0	-	0/1	1	0	0
0/1	0	1	-	0/1	0	1	0
0/1	0	1	-	0/1	1	1	1
0/1	1	0/1	0	0/1	0	-	0
0/1	1	0/1	0	0/1	1	-	0
0/1	1	0/1	1	0/1	0	-	0
0/1	1	0/1	1	0/1	1	-	1
0/1	1	0/1	Z	0	0	-	0
0/1	1	0/1	Z	0	1	-	X
0/1	1	0/1	Z	1	0	L	0
0/1	1	0/1	Z	1	1	L	L

Nota. La imagen muestra la tabla de verdad para configurar el pad.

Encerrado en el recuadro verde se observa la configuración para que el pad actúe como salida. En este caso I es el pin que actúa como entrada.

Mientras que encerrado en el recuadro azul se observa la configuración para que el pad actúe como entrada. En este caso el pin C es el que actúa como salida.

En el Cuadro 18 se observa parte del módulo en Verilog del pad. Importante notar el orden de las señales de entrada y salida para poder instanciar y configurar de forma adecuada el pad.

Cuadro 18. Parte del módulo en Verilog del pad PDDW0204SCDG

```
`celldefine
module PDDW0204SCDG (I,DS,OEN,PAD,C,PE,IE);
   input I,DS,OEN,PE,IE;
   inout PAD;
   output C;
   wire  MG;
   parameter PullTime = 100000;
   reg lastPAD, pull_uen, pull_den,PS;
initial begin
  pull_uen = 0;
  pull_den = 0;
  PS = 0;
end
...
```

Al configurar el pad como entrada, se debe conectar el pin C a la señal de salida del circuito digital. El objetivo es que sin importar sea cual sea la señal que entre por el pad, esta se refleje a través de C.

La manera más sencilla para ubicar la configuración adecuada del pad es observar la tabla de verdad en la Figura 28 y poner atención a las filas donde el pin pad no posee un estado definido y solamente el pin C actúa como salida (desde la fila 5 a la 10). A partir de ahí, se observa que tanto la configuración de la fila 6 como la de la fila 8 alcanzan el funcionamiento deseado, ya que el valor del pad como entrada se refleja en C como salida.

La configuración del pad como entrada se debe realizar de la siguiente manera:

Cuadro 19. Configuración del pad PDDW0204SCDG como entrada

```
PDDW0204SCDG U10(.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(A), .C(
   A_w), .PE(1'b0), .IE(1'b1));
```

- **I:** puede adquirir el valor de 0 ó 1, en este caso se le asigna 0.
- **DS:** puede adquirir el valor de 0 ó 1, en este caso se le asigna 0.
- **OEN:** según la tabla de verdad se le debe asignar el valor de 1.
- **PAD:** en este caso que el pad funciona como entrada, se conecta a la señal definida como la entrada del módulo.
- **C:** en este caso se conecta a un cable interno llamado que llevará la señal de entrada hasta el *core* del circuito digital.
- **PE:** puede adquirir el valor de 0 ó 1, en este caso se le asigna 0.

- **IE:** según la tabla de verdad se le debe asignar el valor de 1.

Por otro lado, al configurar el pad como salida, el objetivo es que la señal que ingrese por I se refleje a través del pad.

Para poder ubicar fácilmente la configuración adecuada para que el pad actúe como salida, se debe observar nuevamente la tabla de verdad en la Figura 28 y poner atención a las filas donde el estado del pad como entrada no está definido, ya que se desea que funcione como salida. En este caso, de la fila 1 a la 4 se extrae la configuración adecuada.

La configuración del pad como salida se debe realizar de la siguiente manera:

Cuadro 20. Configuración del pad PDDW0204SCDG como salida

```
PDDW0204SCDG U11(.I(Y_w), .DS(1'b0), .OEN(1'b0), .PAD(Y), .C(1'
   bx), .PE(1'b0), .IE(1'b1));
```

- **I:** en este caso se conecta a un cable interno que transporta la señal de salida del *core* hacia el pad.
- **DS:** puede adquirir el valor de 0 ó 1, en este caso se le asigna 0.
- **OEN:** según la tabla de verdad se le debe asignar el valor de 0.
- **PAD:** en este caso que el pad funciona como salida, se conecta a la señal definida como la salida del módulo.
- **C:** puede adquirir cualquier valor, ya que no está involucrado en el funcionamiento.
- **PE:** puede adquirir el valor de 0 ó 1, en este caso se le asigna 0.
- **IE:** puede adquirir el valor de 0 ó 1, en este caso se le asigna 1.

Si existen múltiples señales de entrada y salida, se deben instanciar tantos pads como sea necesario para cada una de ellas.

Luego de haber comprendido el funcionamiento y la configuración de los pads de entrada y salida, se procede a diseñar el módulo en Verilog que integra el *core* del circuito digital con los pads correspondientes. A continuación se presenta el módulo correspondiente para la compuerta NOT con pads:

Cuadro 21. Archivo IO_Not.v: Diseño del módulo de una compuerta NOT con pads en Verilog

```
module IO_Not(A, Y);
        input A;
        output Y;
        wire A_w, Y_w;
        Not U20(A_w,Y_w);
        //Configuracion de PIN 1/0
        //Entradas (C):
        //PDDW0204SCDG U10(1'b0,1'b0, 1'b1,A,A_w,1'b0,1'b1);
        //Salidas (I):
        //PDDW0204SCDG U11(Y_w,1'b0, 1'b0, Y, 1'bx, 1'b0,1'b1);

        //Entradas:
        PDDW0204SCDG U10(.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(
           A), .C(A_w), .PE(1'b0), .IE(1'b1));
        //Salidas:
        PDDW0204SCDG U11(.I(Y_w), .DS(1'b0), .OEN(1'b0), .PAD(Y
           ), .C(1'bx), .PE(1'b0), .IE(1'b1));
endmodule
```

La versión final del módulo se observa en el Cuadro 21. En este módulo se instanció el *core* del circuito digital (componente NOT) y se conectaron las señales de entrada y salida a los pads correspondientes. Hay que mencionar que no se incluyen pines para VDD y VSS en este módulo ya que dichas señales no existen dentro del RTL.

Como se verá más adelante, la herramienta Design Compiler instancia un par de celdas llamadas TIEH y TIEL, las cuales son *buffers* que conectan VDD a las señales que reciben el 1 lógico y VSS a las señales que reciben el 0 lógico. La herramienta crea *wires* para definir "VDD" y "VSS", y así conectar las señales correspondientes a estas referencias de voltaje.

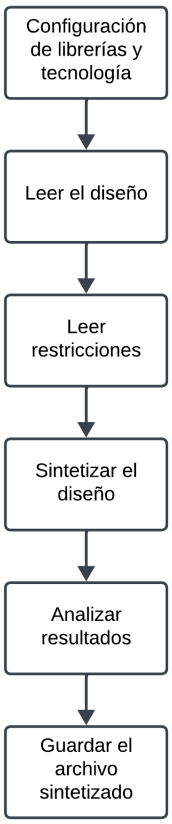
Por tanto, aunque se incluyan pads de alimentación en el Verilog como se realizó en años anteriores, al momento de sintetizar la herramienta los removerá automáticamente y colocará los *buffers* ya mencionados.

10.2. Desarrollo del *script* de síntesis lógica en TCL

Antes de realizar la síntesis lógica de cualquier circuito es importante contar con un *script* en TCL que contenga los comandos necesarios para ejecutar el proceso de síntesis. Este *script* debe estar bien estructurado y organizado para facilitar su comprensión y modificación en el futuro. La idea de un *script* es automatizar el proceso para optimizar el tiempo y reducir errores.

A continuación se presenta un diagrama de flujo para la construcción de un *script* de síntesis lógica en TCL:

Figura 29. Diagrama de flujo para la realización de un *script* de síntesis lógica en TCL



Nota. La imagen muestra los pasos a seguir para crear un *script* de síntesis lógica en TCL.

A partir de dichos pasos, se desarrolló el *script* llamado `script_dc.tcl`, el cual debe ser ejecutado en el directorio *synthesis* para realizar la síntesis como se explicó en el Capítulo 8. A continuación se presenta el *script* completo y se explica cada sección del mismo:

Cuadro 22. Archivo script_dc.tcl: *Script* de síntesis lógica en TCL

```
# Author: Carlos Julio Valdes Oajaca
# Date: July 2025

# Parameters the user should modify
# PARAMETRO -> name of the circuit and instance (verilog file
    and instance must have the same name)
# USER_OPTION -> select between 1 and 2 only
# 1 -> synthesis without PADS
# 2 -> synthesis with PADS

set PARAMETRO "Not"
set USER_OPTION 1
```

```
# Depending on user's selection do the synthesis without or
   with pads
if {$USER_OPTION == 1} { # Option without pads
        set PARAMETRO_2 "no"
        set rtl_files "${PARAMETRO}.v"
        set top_module "${PARAMETRO}"
} elseif {$USER_OPTION == 2} { # Option with pads
        set PARAMETRO_2 "with"
        set rtl_files [list "IO_${PARAMETRO}.v" "${PARAMETRO}.v
           "]
        set top_module "IO_${PARAMETRO}"
} else {
        puts "Error: please select 1 or 2"
        exit
}

#if {![file exists reports/$PARAMETRO]} {
    file mkdir reports/$PARAMETRO
#}

#if {![file exists outputs/outputs_no_pads/$PARAMETRO]} {
    file mkdir outputs/outputs_no_pads/$PARAMETRO
#}

#if {![file exists outputs/outputs_with_pads/$PARAMETRO]} {
    file mkdir outputs/outputs_with_pads/$PARAMETRO
#}

#if {![file exists logs/$PARAMETRO]} {
    file mkdir logs/$PARAMETRO
#}

# Search Path and Logic Library Setup
# tcbn65lpbwp7twc0d9.db ->lib 7t
# tcbn65lpwc0d9.db ->lib 9t
set_app_var search_path "$search_path ./rtl/$PARAMETRO ./libs"
set_app_var target_library "tcbn65lpwc0d9.db tpdn65lpnv2od3wc5.
   db"
set_app_var link_library "* $target_library"

# RTL Reading and Link
analyze -format verilog $rtl_files
# Colocar el nombre del modulo top
elaborate $top_module
#
link

# Constraints Setup
```

```
#set clk_val 10.0
#create_clock -period $clk_val [get_ports clk] -name clk
#set_clock_uncertainty -setup [expr $clk_val*0.1] [get_clocks
   clk]
#set_clock_transition -max [expr $clk_val*0.1] [get_clocks clk]
# source latency
#set_clock_latency -source [expr $clk_val*0.05] [get_clocks clk
   ]
# network latency
#set_clock_latency -max [expr $clk_val*0.03] [get_clocks clk]
# NO input delays to clk
#set_input_delay -max  [expr $clk_val*0.4] -clock clk [
   remove_from_collection [all_inputs] clk]
# Output delay
#set_output_delay -max [expr $clk_val*0.5] -clock clk [
   all_outputs]
# load
#set_load -max 0.04 [all_outputs]

#set_input_transition -min [expr $clk_val*0.01] [
   remove_from_collection [all_inputs] clk]
#set_input_transition -max [expr $clk_val*0.10] [
   remove_from_collection [all_inputs] clk]

# Pre-compile Reports
report_clock > ./reports/$PARAMETRO/pre_syn_report_clock.rpt
report_clock -skew > ./reports/$PARAMETRO/pre_syn_report_skew.
   rpt
report_port -verbose > ./reports/$PARAMETRO/
   pre_syn_report_port_constraints.rpt
check_timing > ./reports/$PARAMETRO/pre_syn_check_timing.rpt
check_design > ./reports/$PARAMETRO/pre_syn_check_design.rpt

# Compile/Synthesis
compile_ultra -no_auto_ungroup
#compile -map_effort high -area_effort high

# Post-compile Reports
check_timing > ./reports/$PARAMETRO/report_timing.rpt
report_qor > ./reports/$PARAMETRO/report_qor.rpt
report_constraints -all_violators > ./reports/$PARAMETRO/
   report_constraints.rpt
report_power > ./reports/$PARAMETRO/report_power.rpt
check_design > ./reports/$PARAMETRO/check_design.rpt
report_cell [get_cells -hier *] > ./reports/$PARAMETRO/
   report_cell.rpt

# Save Design
```

```
write_file -format ddc -hierarchy -out outputs/outputs_${
   PARAMETRO_2}_pads/$PARAMETRO/${PARAMETRO}_mapped.ddc
write_file -format verilog -hierarchy -out outputs/outputs_${
   PARAMETRO_2}_pads/$PARAMETRO/${PARAMETRO}_mapped.v
write_sdc outputs/outputs_${PARAMETRO_2}_pads/$PARAMETRO/${
   PARAMETRO}_mapped.sdc

write_icc2_files -force -out outputs/outputs_${PARAMETRO_2}
   _pads/$PARAMETRO/icc2_files


# Move Logs
if {[file exists command.log]} {
        file rename -force command.log logs/$PARAMETRO/command.
           log
}
if {[file exists filenames.log]} {
        file rename -force filenames.log logs/$PARAMETRO/
           filenames.log
}
if {[file exists default.svf]} {
        file rename -force default.svf logs/$PARAMETRO/default.
           svf
}

# Copy testbench to outputs
if {![file exists outputs/outputs_${PARAMETRO_2}_pads/
   $PARAMETRO/${PARAMETRO}_tb.v]} {
file copy -force rtl/$PARAMETRO/${PARAMETRO}_tb.v outputs/
   outputs_${PARAMETRO_2}_pads/$PARAMETRO/${PARAMETRO}_tb.v
}

# Delete unnecessary files
file delete -force {*}[glob -nocomplain *.mr]
file delete -force {*}[glob -nocomplain *-verilog.pvl]
file delete -force {*}[glob -nocomplain *-verilog.syn]

return
```

Como se observa en el Cuadro 22, el *script* está dividido en varias secciones que cumplen funciones espcíficas. Al inicio se mencionan los parámetros que el usuario debe modificar para realizar la síntesis lógica del circuito que desee. El parámetro `PARAMETRO` define el nombre del circuito y de la instancia. Es importante que el archivo Verilog del circuito y la instancia tengan el mismo nombre para evitar errores. El parámetro `USER_OPTION` permite seleccionar si se desea realizar la síntesis lógica sin pads (opción 1) o con pads (opción 2). Dependiendo de la opción seleccionada, se ajustan las variables `PARAMETRO_2`, `rtl_files` y `top_module` para adecuar el proceso de síntesis.

Posteriormente, se crean directorios de ser necesario para organizar todos los archivos de la síntesis como se mencionó en el Capítulo 8.

A continuación se explican las demás secciones del *script*:

1. ***Search Path and Logic Library Setup:*** se establecen las rutas donde se encuentran los archivos RTL y las librerías de celdas estándar necesarias para la síntesis lógica. Notar que como se mencionó en el Capítulo 7, la *link library* se define como la misma *target library*. La *target library* incluye de primero la librerías para las celdas del *core* y después la librería de pads. Puede colocarse otra librería en lugar de la que se utiliza, por ejemplo puede utilizarse la librería de 9t en lugar de la de 7t.

2. ***RTL reading and Link:***

   - **analyze:** este comando lee todos los archivos RTL especificados, verifica la sintaxis, reporta errores y crea archivos intermedios necesarios para la síntesis. Con este comando pueden analizarse múltiples archivos Verilog en orden jerárquico. Por ejemplo, el archivo con pads depende del archivo del *core*, por lo que se coloca primero el archivo con pads (IO_${PARAMETRO}.v) y después el archivo del *core* (${PARAMETRO}.v). El archivo donde se encuentran definidos los pads ya está incluido en la *target library*, por lo que no es necesario incluirlo en esta parte. Este es el comando que se aprovecha para realizar la síntesis lógica de cualquier circuito con pads sin necesidad de tener que sintetizar el circuito dos veces.
   - **elaborate:** este comando traduce el diseño a un *netlist* interno basado en compuertas genéricas de Synopsys llamadas GTECH a través de los archivos intermedios y establece el diseño actual. En este caso, se debe colocar el nombre del módulo top que se definió anteriormente en la variable `top_module`, en el caso del diseño con pads sería IO_${PARAMETRO} y en el caso del diseño sin pads sería ${PARAMETRO}.
   - **link:** a pesar de que el comando `elaborate` ya realice un `link`, es una buena práctica incluir este comando es una buena práctica, ya que conecta al "top" con sus subdiseños y resuelve todas las instancias de las celdas del diseño.

3. ***Constraints Setup:*** en esta sección se pueden definir las restricciones de temporización y carga para el diseño. En este caso, no se incluyen restricciones ya que con los circuitos que se sintetizan en este trabajo no se busca un desempeño específico. De hecho, para el circuito integrado "El Gran Jaguar" no es necesario definir restricciones de tiempo porque su función es devolver un texto que será reproducido en audio. Al pensar cuántas palabras por segundo puede decir un ser humano comparado con el tiempo de *switcheo* del circuito que para este caso es del orden de nanosegundos, se concluye que no es necesario imponer restricciones de tiempo. Por esta razón, se encuentra comentada esta sección, no obstante, se deja como referencia para futuros trabajos.

4. ***Pre-compile Reports:*** antes de ejecutar la síntesis lógica, es recomendable generar reportes preliminares para verificar el estado del diseño y las restricciones definidas. Estos reportes incluyen información sobre los relojes, puertos, verificación de temporización y verificación del diseño.

5. ***Compile/Synthesis:*** para lograr con los objetivos de diseño es necesario tener un código a nivel RTL de buena calidad óptimo para Design Compiler, restricciones de diseño adecuadas si aplica y la ejecución de la síntesis con las optimizaciones correctas. En este caso se utiliza el comando `compile_ultra -no_auto_ungroup` para realizar la síntesis lógica con optimizaciones avanzadas. Dicho comando primero realiza optimizaciones en cuanto a la arquitectura del diseño (código RTL), luego realiza optimizaciones a nivel lógico (GTECH) y finalmente realiza el mapeo más óptimo a celdas estándar. La opción `-no_auto_ungroup` evita que la herramienta desagrupe automáticamente las instancias jerárquicas, lo que puede ser beneficioso para mantener la estructura del diseño original. El comando prioriza por medio de costos dependiendo de las restricciones a cumplir:

    a) Restricciones de diseño como el tiempo de transición y capacitancia máxima tienen prioridad máxima de costo.

    b) Restricciones definidas por el usuario como *delay* máximo, área máxima y potencia máxima.

6. ***Post-compile Reports:*** después de completar la síntesis lógica, es fundamental generar reportes detallados para evaluar el rendimiento del diseño sintetizado. Estos reportes incluyen información sobre la temporización, calidad del diseño, violaciones de restricciones, consumo de potencia, verificación del diseño y detalles de las celdas utilizadas.

7. ***Save Design:*** es primordial guardar el diseño sintetizado junto con las restricciones de diseño utilizadas. El archivo .ddc es un formato propietario de Synopsys que almacena el *netlist* sintetizado y la información de diseño, es puede utilizar entre herramientas del mismo Synopsys como ICC2. El archivo .v generado contiene el *netlist* en formato Verilog, lo que facilita su integración con otras herramientas de diseño de otras empresas. El archivo .sdc guarda las restricciones de diseño utilizadas durante la síntesis lógica. También es posible generar los archivos necesarios para el *Back-End* utilizando el comando `write_icc2_files`, los cuales incluyen archivos de configuración y scripts específicos para la herramienta ICC2 de Synopsys. Lo más importante es poder brindarle al equipo de *Back-End* el archivo del *gate-level netlist*, ya sea en formato .ddc ó .v junto con el archivo de restricciones .sdc.

8. ***Move Logs:*** al finalizar la síntesis lógica se mueven de ubicación archivos .log generados durante la síntesis. Estos archivos se crean automáticamente en el directorio *synthesis* y se mueven a la carpeta *logs* correspondiente al circuito sintetizado para mantener una mejor organización en el espacio de trabajo.

9. ***Copy testbench to outputs:*** se copia el archivo del *testbench* del circuito sintetizado al directorio de *outputs* correspondiente (ya sea con pads o sin pads). Esto facilita la simulación posterior del *netlist* generado durante la síntesis lógica.

10. ***Delete unnecessary files:*** finalmente, se eliminan archivos temporales generados durante la síntesis lógica que no son necesarios para el análisis posterior. Esto ayuda a mantener el espacio de trabajo limpio y organizado.

CAPÍTULO 11

Síntesis de circuitos digitales descritos en Verilog

En este capítulo se describe el proceso de síntesis que se llevó a cabo con diversos circuitos utilizando el *script* `script_dc.tcl`.

No se utiliza la interfaz gráfica para la síntesis en este caso debido a que consume mucho más tiempo. En la industria se utlizan *scripts* para optimizar la mayor cantidad de tiempo.

Para cada circuito a continuación y como se mencionó anteriormente, se utilizan los módulos del *core*, *testbench* y el *core* con pads.

11.1. Síntesis lógica de una compuerta NOT

11.1.1. Módulo *core* del circuito, *testbench* y módulo del circuito con pads

Lo primero a considerar es la creación de la carpeta llamada `rtl` y la carpeta `scripts` dentro del directorio `synthesis`. Posteriormente, se crea una carpeta con el nombre del circuito que se planea sintetizar dentro del directorio `rtl`.

De esta manera se diseñan los módulos del circuito en Verilog ya sea utilizando la descripción *behavioral* o *structural*.

Si el archivo del circuito con pads no se encuentra listo, no será posible realizar la síntesis del mismo, por lo que el usuario logrará sintetizar con éxito el *core* únicamente.

Para esta parte, se diseñó el circuito de una compuerta NOT en Verilog con descripción *behavioral*.

Una vez lograda la síntesis de la compuerta NOT sin y con pads, se escala en complejidad hasta llegar a la síntesis del nanochip "El Gran Jaguar".

Es importante destacar que para nombrar los archivos y los módulos existe una forma establecida:

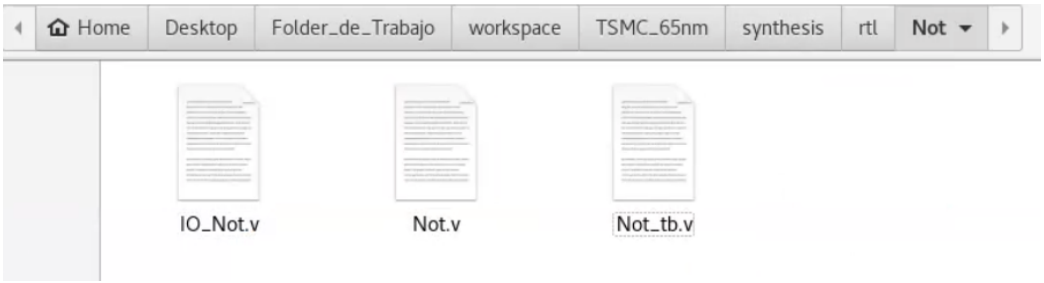
- **Archivo *core*:** ${PARAMETRO}.v
- **Archivo *testbench*:** ${PARAMETRO}_tb.v
- **Archivo *core* con pads:** IO_${PARAMETRO}.v

En el caso de la compuerta NOT:

- **Not.v**
- **Not_tb.v**
- **IO_Not.v**

Es importante tener estos archivos dentro de la carpeta correspondiente al circuito en el directorio `rtl` como se observa en la Figura 30.

Figura 30. Archivos necesarios para sintetizar la compuerta NOT



Nota. La imagen muestra los archivos necesarios para la síntesis de la NOT.

Es importante que la carpeta del diseño dentro del directorio `rtl` sea igual al del nombre del archivo y módulo, en este caso: `Not`.

Como en la Sección 10.1 ya se explicó y mostró cómo están diseñados los módulos de la compuerta NOT, se continua con la simulación del *core*.

11.1.2. Simulación del módulo *core* del circuito

Con el objetivo de verificar la codificación del *core*, se realiza una simulación del mismo utilizando el *testbench*. De no relizarse esta simulación se corre el riesgo de fallos en el diseño

en etapas posteriores, incluso un mal diseño a nivel RTL puede causar grandes problemas en la etapa de *Back-End*.

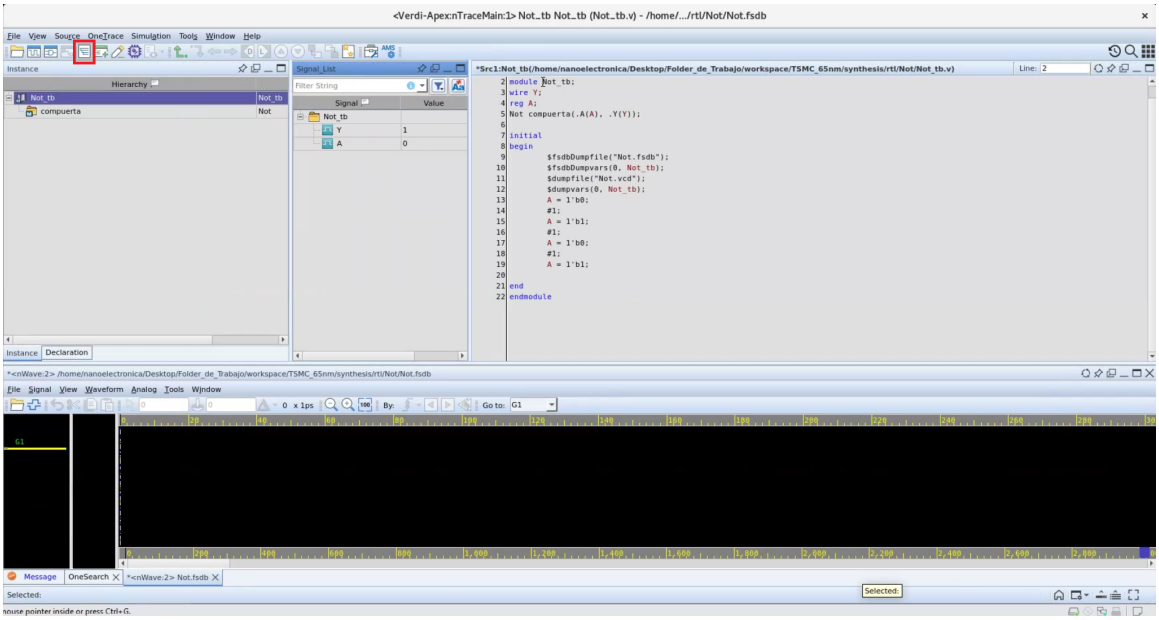
Para simular se utilizaron las herramientas de VCS para la verificación de sintaxis y Verdi para el análisis de la simulación. Los comandos para la simulación del circuito se presentan a continuación:

Cuadro 23. Comandos para la compilación y simulación del *core* de la NOT

```
vcs -full64 -debug_access+all -kdb Not_tb.v Not.v
./simv
verdi -ssf Not.fsdb
```

Luego ejecutar estos comandos, la herramienta de Verdi aparece con su interfaz gráfica. Una vez aparezca la ventana de la aplicación, hacer *click* en el ícono encerrado dentro del rectángulo rojo (*Signal List*) como se muestra a continuación:

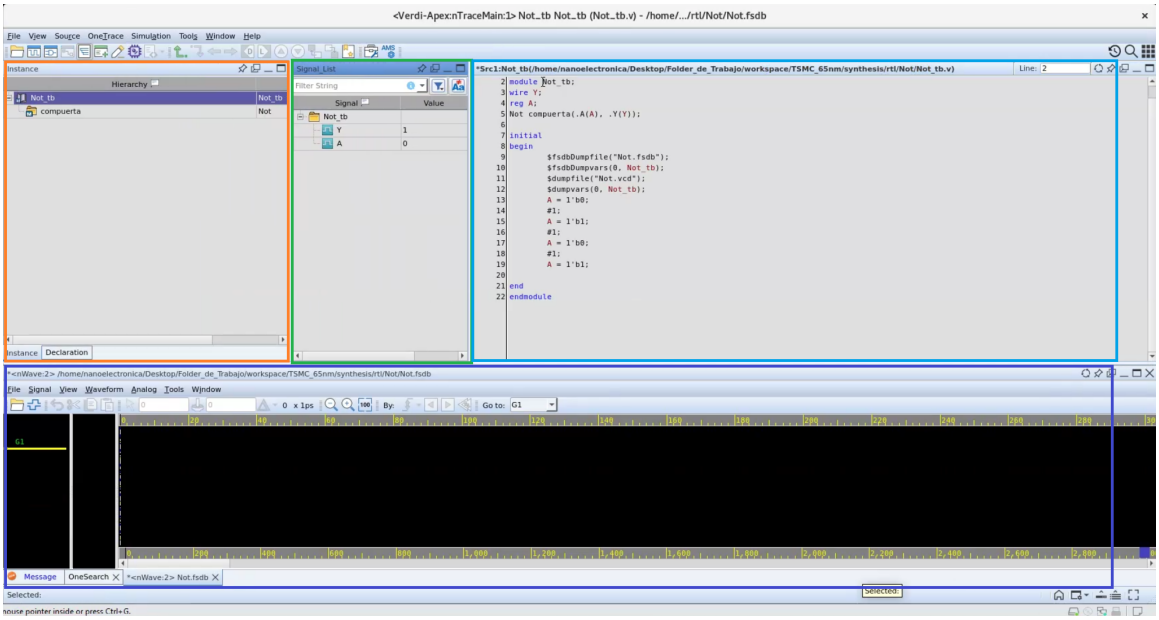
Figura 31. Interfaz gráfica de Verdi al simular el *core* de la NOT



Nota. La imagen muestra la interfaz de Verdi al simular.

Ahora se procede a explicar cada parte que se muestra en la ventana de Verdi, de igual manera más adelante se mostrará cómo se puede editar dentro de la misma herramienta y cómo obtener el esquemático del circuito simulado.

Figura 32. Partes de la herramienta Verdi al simular

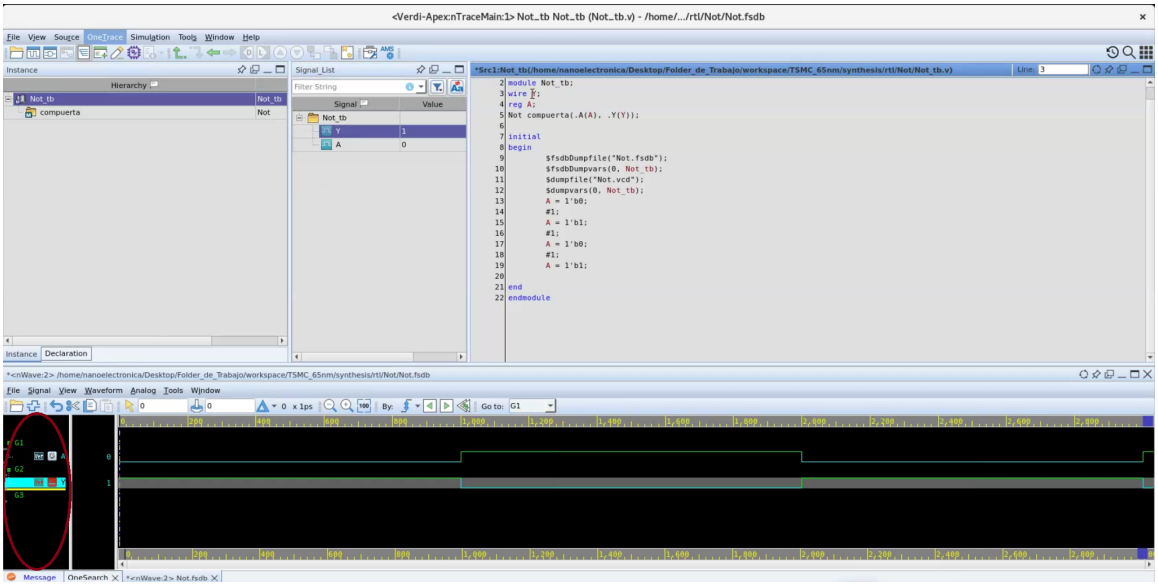


Nota. La imagen muestra las partes dentro de la GUI de Verdi.

- **Recuadro naranja:** en esta sección de la herramienta se muestra la jerarquía del diseño. Se lista del módulo top hasta las celdas o instancias más pequeñas que componen al circuito simulado.
- **Recuadro verde:** lista de señales del circuito seleccionado de la jerarquía. Es posible observar el valor de cada señal al arrancar la simulación.
- **Recuadro celeste:** código del circuito fuente que se encuentra seleccionado en la jerarquía. Es posible editarlo dentro de Verdi pero esto se explicará más adelante.
- **Recuadro morado:** sección de *Waveform* donde se muestran las señales de la simulación de forma gráfica.

Para poder visualizar las señales del circuito y analizarlo, arrastrar desde la sección de señales tanto `A` como `Y` hasta la parte izquierda de la sección *Waveform*:

Figura 33. Simulación de la compuerta NOT



Nota. La imagen muestra la simulación de la compuerta NOT en Verdi.

De esta manera como se observa en la Figura 33, la señal salida `Y` se encuentra invertida con respecto a la señal de entrada `A`. Por lo que el funcionamiento del *core* de la compuerta es correcto.

11.1.3. Síntesis del módulo *core* del circuito

Antes de realizar la síntesis del circuito, es importante abrir el *script* `script_dc.tcl` para colocar los parámetros adecuados:

Figura 34. Modificación de parámetros para sintetizar el *core* de la NOT

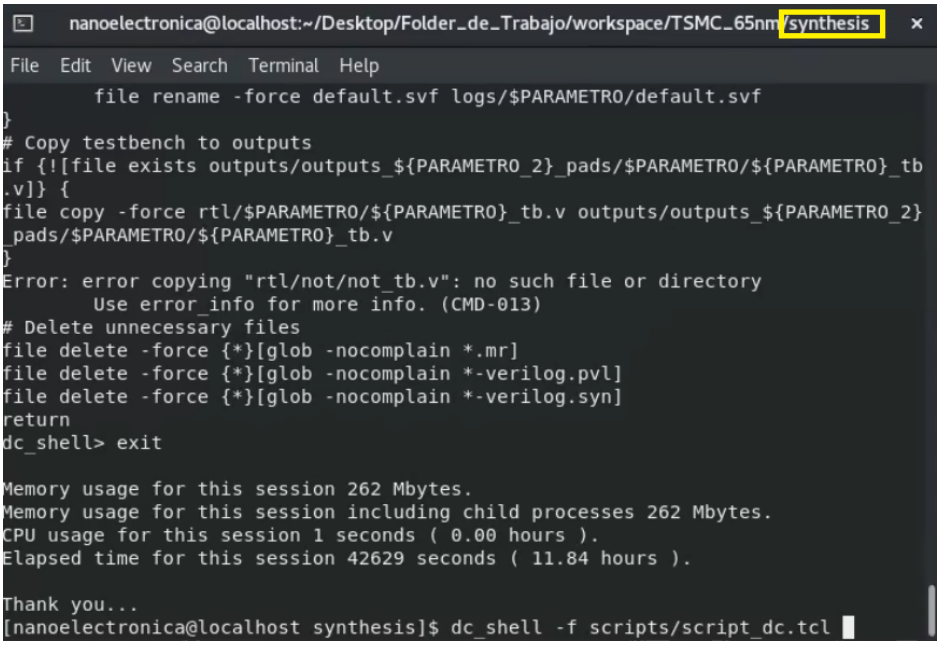
Nota. La imagen muestra los parámetros para sintetizar el *core* de la NOT.

Una vez modificado el *script* de forma acorde al diseño, se ejecuta el siguiente comando desde el directorio `synthesis`:

Cuadro 24. Comando para la ejecución de la síntesis de la NOT

```
dc_shell -f scripts/script_dc.tcl
```

Figura 35. Ejecución del *script* desde la terminal para realizar la síntesis de la NOT



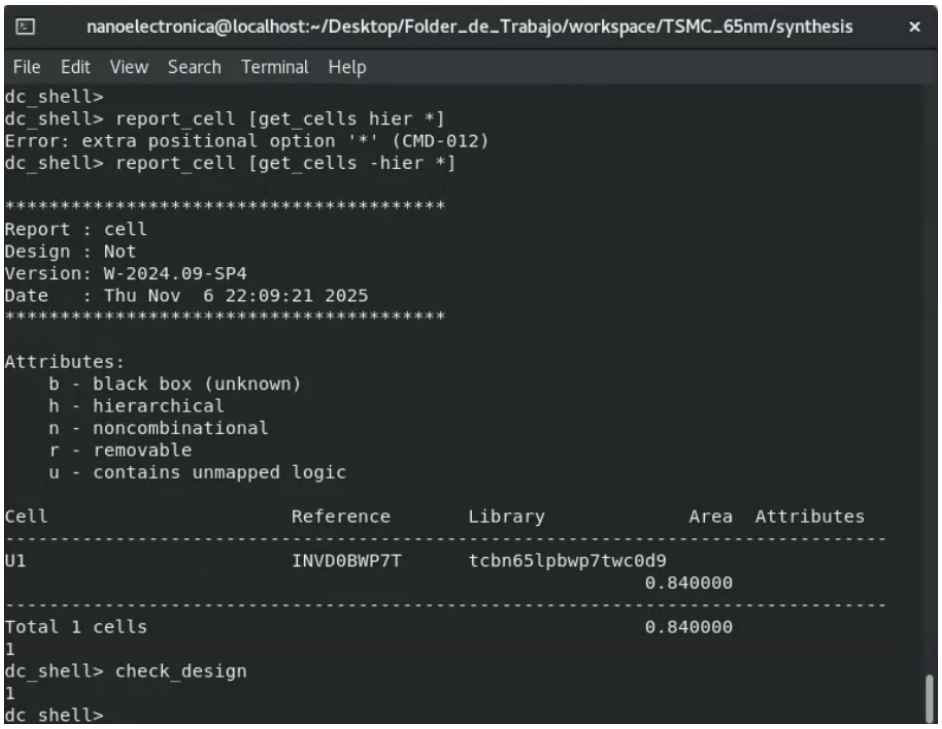
Nota. La imagen muestra la ejecución del *script* desde la terminal.

Una vez ejecutado el *script* se recomienda ejecutar los siguientes comandos para primero observar la cantidad de celdas totales del circuito sintetizado y luego, para revisar si el diseño sintetizado posee errores:

Cuadro 25. Comandos posteriores a la ejecución del *script*

```
report_cell [get_cells -hier *]
check_design
```

Figura 36. Celdas totales y verificación de la síntesis del *core* de la NOT



Nota. La imagen muestra la salida al ejecutar los comandos del Cuadro 31.

Como se puede notar en la figura anterior, la devolución al ejecutar el comando `check_design` fue 1, lo que indica que no hay errores ni advertencias para el diseño sintetizado.

A continuación se presenta el *gate-level netlist* como resultado de la síntesis lógica del *core* de la compuerta NOT:

Cuadro 26. *Gate-level netlist* del *core* de la NOT

```
////////////////////////////////////////////////////////////////
// Created by: Synopsys DC Ultra(TM) in wire load mode
// Version   : W-2024.09-SP4
// Date      : Thu Nov  6 22:02:26 2025
////////////////////////////////////////////////////////////////
```

```
module Not ( A, Y );
  input A;
  output Y;


  INVDO U1 ( .I(A), .ZN(Y) );
endmodule
```

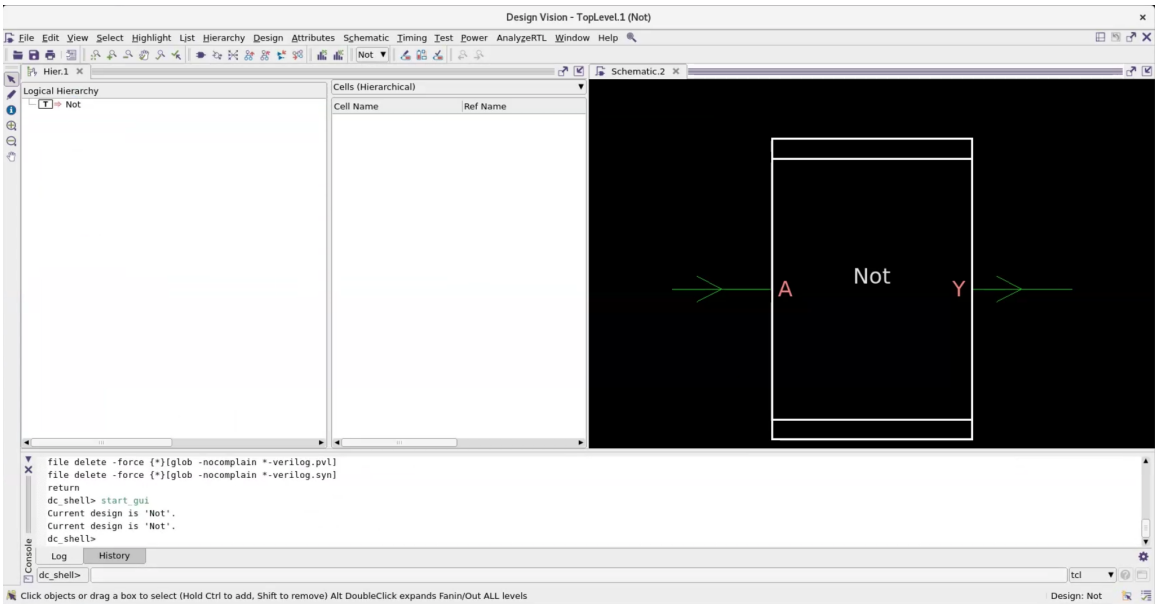
Cuando se termina de ejecutar el *script*, se puede observar el esquemático del circuito sintetizado mediante Design Vision (la interfaz gráfica de Design Compiler):

Cuadro 27. Comando iniciar la GUI de Design Compiler

```
dc_shell> start_gui
```

Una vez aparezca la interfaz hacer *click* en el botón del recuadro amarillo:

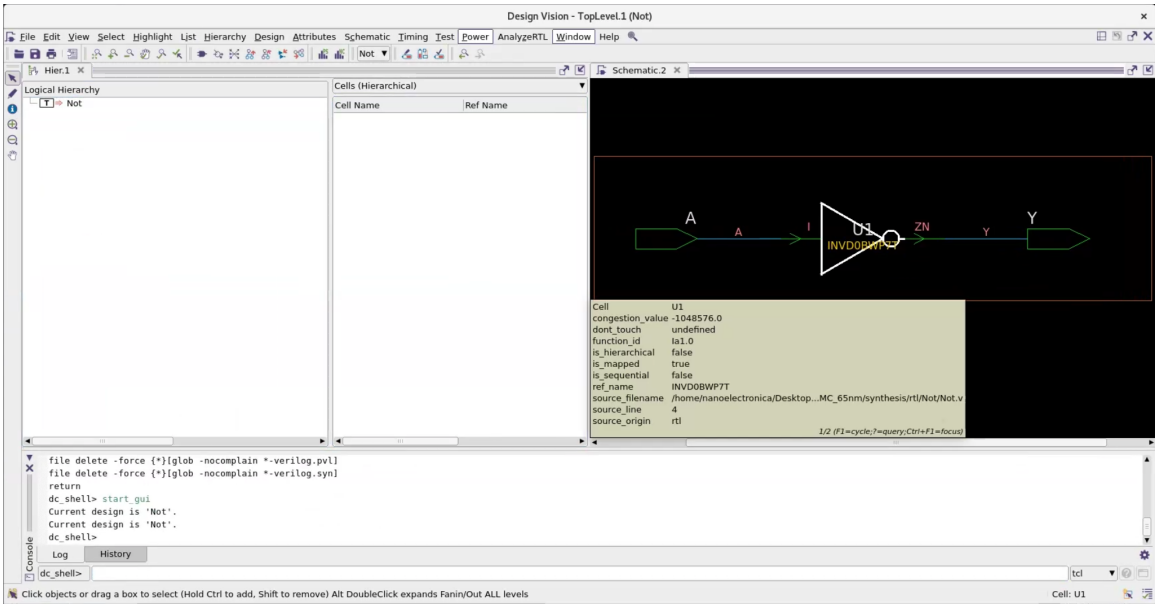
Figura 37. GUI de Design Compiler de la síntesis del *core* de la NOT



Nota. La imagen muestra Design Vision de la síntesis del *core* de la NOT.

Para observar las celdas, hacer doble *click* sobre la caja llamada `Not` para ver las instancias dentro de la misma:

Figura 38. Esquemático del *core* de la NOT sintetizada



Nota. La imagen muestra Design Vision con el esquemático del *core* de la NOT sintetizada.

En la imagen se observa el esquemático del circuito sintetizado, puede notarse el nombre de la instancia, el nombre de la celda, el nombre de las señales del circuito, etc.

Para salir, cerrar la ventana de la GUI y colocar el siguiente comando en la terminal:

Cuadro 28. Comando para salir de Design Compiler

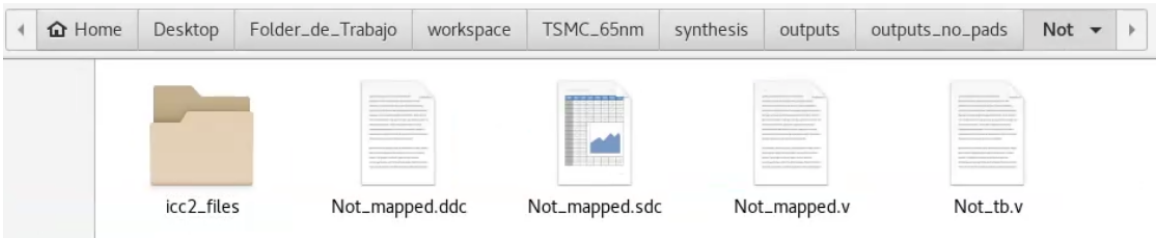
```
dc_shell> exit
```

11.1.4. Simulación del *core* del circuito sintetizado

Como siguiente paso, una vez se haya verificado que se realizó la síntesis con éxito, se simula el circuito sintetizado para verificar la funcionalidad del *gate-level netlist*. Aunque se logre sintetizar un circuito, esto no garantiza que la funcionalidad esté garantizada, por esto es fundamental simularlo.

Antes de realizar la simulación del circuito sintetizado, verificar que los archivos necesarios se encuentren en su ubicación correspondiente:

Figura 39. Archivos producidos por la síntesis del *core* de la NOT



Nota. La imagen muestra los archivos de salida producidos por la síntesis.

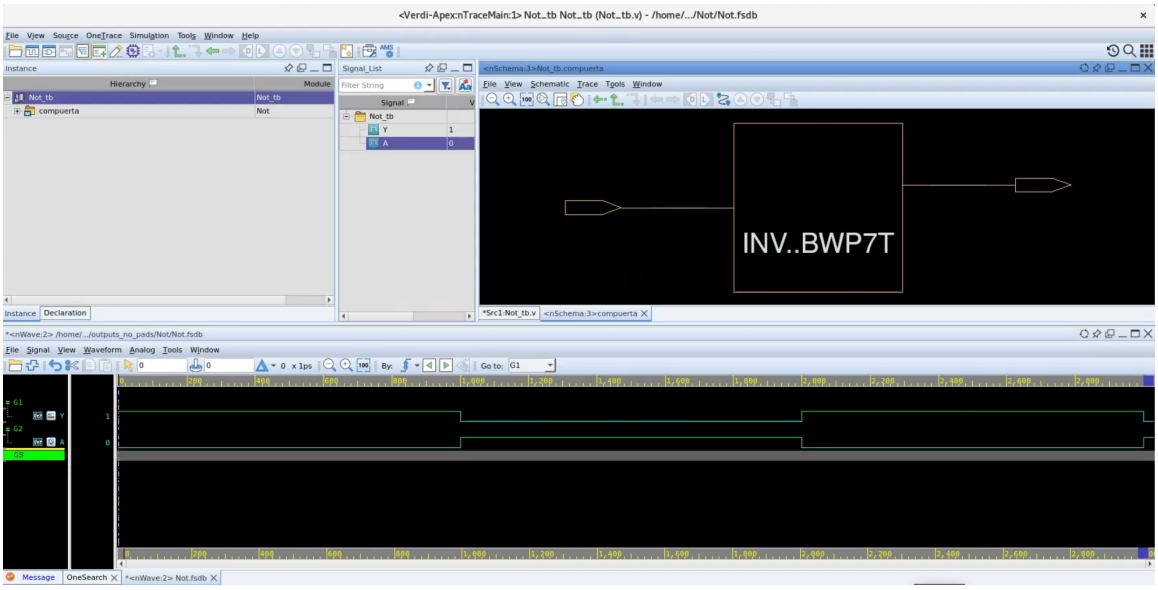
Para simular el circuito sintetizado se realizó la ejecución de los siguientes comandos:

Cuadro 29. Comandos para simular el *gate-level netlist* del *core* de la NOT

```
vcs -full64 -debug_access+all -kdb Not_tb.v Not_mapped.v -v
    ../../tcbn65lp.v
./simv
verdi -ssf Not.fsdb
```

Esto da como resultado la simulación del circuito sintetizado:

Figura 40. Simulación del circuito sintetizado del *core* de la NOT



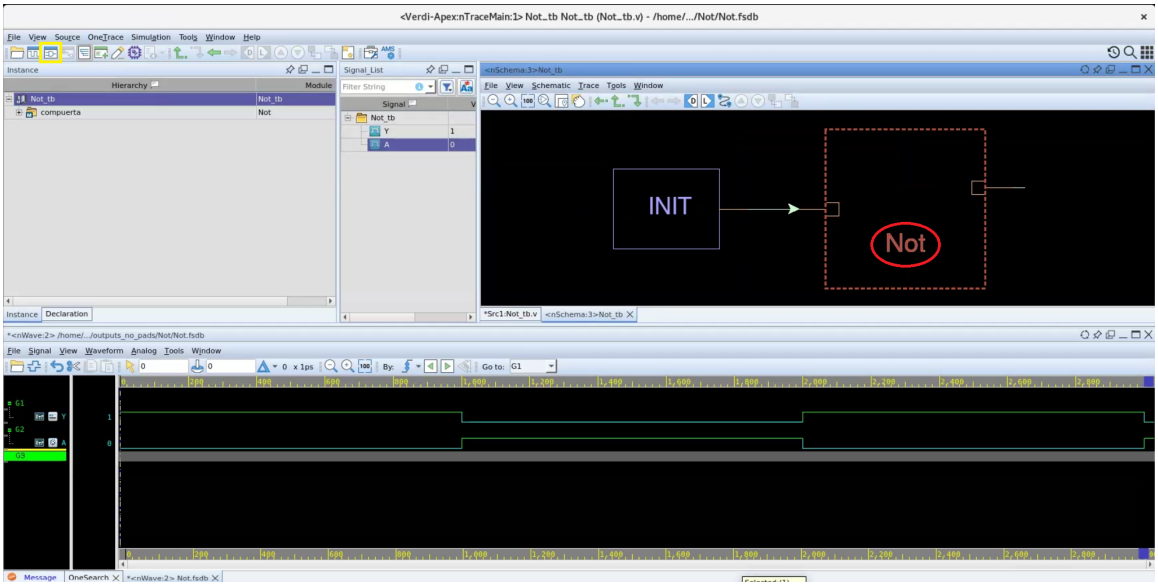
Nota. La imagen muestra la simulación del archivo sintetizado de la NOT.

En la parte superior derecha de la Figura 40 se puede observar el esquemático del circuito sintetizado, donde aparece el mismo nombre de celda que se muestra al ejecutar el comando

`report_cell [get_cells -hier *]` en el Cuadro 31.

Para observar el esquemático en Verdi hacer *click* en la opción *New Schematic* que se encuentra encerrada en el recuadro amarillo de la Figura 41. Luego, para ver el nombre de la celda que compone la NOT hacer doble *click* sobre la caja de nombre `Not` en el diagrama (encerrada por la elipse roja).

Figura 41. Explicación de la función de la sección de *Schematic* con la síntesis de la NOT



Nota. La imagen muestra la explicación de la sección de *Schematic*.

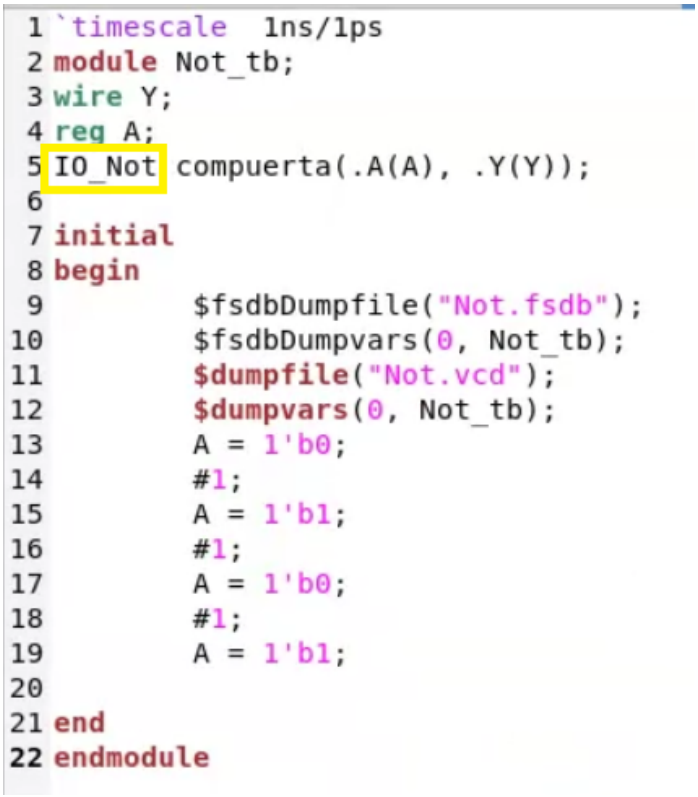
Al observar la Figura 40 se puede notar que la señal de salida `Y` se encuentra invertida respecto a la señal de entrada `A`, por tanto, el circuito sintetizado del *core* de la NOT es correcto.

11.1.5. Simulación del módulo del circuito con pads

Una vez se haya verificado la validez de la síntesis del *core*, se comienza el proceso de síntesis del diseño junto con los pads de entrada y salida.

Para ello, utilizar el mismo archivo de *testbench*, únicamente cambiar la instancia a la que se hace referencia dentro de este archivo para simular el archivo del circuito con pads:

Figura 42. Modificación de la instancia del archivo *testbench*



```
1  `timescale  1ns/1ps
2  module Not_tb;
3  wire Y;
4  reg A;
5  IO_Not compuerta(.A(A), .Y(Y));
6
7  initial
8  begin
9          $fsdbDumpfile("Not.fsdb");
10         $fsdbDumpvars(0, Not_tb);
11         $dumpfile("Not.vcd");
12         $dumpvars(0, Not_tb);
13         A = 1'b0;
14         #1;
15         A = 1'b1;
16         #1;
17         A = 1'b0;
18         #1;
19         A = 1'b1;
20
21 end
22 endmodule
```

Nota. La imagen muestra la modificación del archivo *testbench*.

Luego se ejecutan los siguientes comandos para simular el circuito con pads:

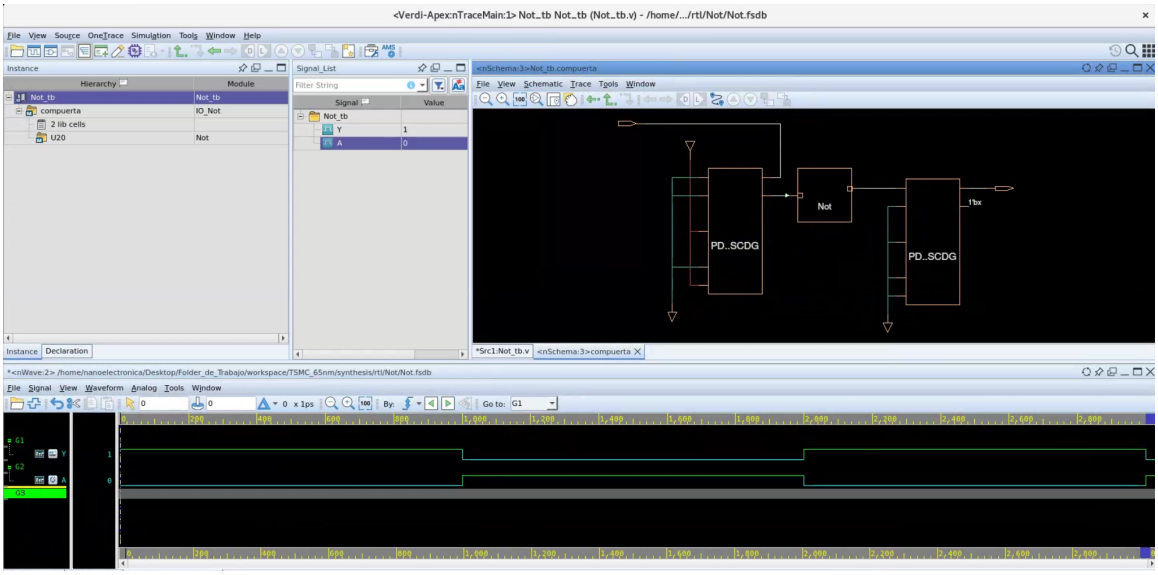
Cuadro 30. Comandos para la compilación y simulación de la NOT con pads

```
vcs -full64 -debug_access+all -kdb Not_tb.v IO_Not.v Not.v -v
   ../../outputs/tpdn65lpnv2od3.v
./simv
verdi -ssf Not.fsdb
```

Puede notarse en el Cuadro 30 que el comando de VCS se modifica de tal manera que los archivos se listan de manera jerárquica. Además, se incluye la librería de pads en Verilog para que el programa pueda compilar el diseño completo.

De esta forma, se obtiene lo siguiente en la simulación:

Figura 43. Simulación del archivo de la NOT con pads



Nota. La imagen muestra la simulación de la NOT con pads.

Puede notarse en la vista de esquemático que las celdas de los pads se encuentran tanto en la entrada como en la salida del *core* de la NOT.

Al analizar el comportamiento del circuito en la Figura 43 se concluye que la codificación del circuito con pads es correcta.

11.1.6. Síntesis del módulo del circuito con pads

Ahora se procede a realizar la síntesis de la NOT con pads, luego de haber verificado que la codificación del archivo sea correcta. Antes de ejecutar el *script*, modificar en el archivo el parámetro de nombre `USER_OPTION` para que adquiera el valor de 2:

Figura 44. Modificación del *script* para realizar la síntesis del circuito con pads

```
# Author: Carlos Julio Valdes Oajaca
# Date: July 2025

# Parameters the user should modify
# PARAMETRO -> name of the circuit and instance (verilog file and instance must have the same name)
# USER_OPTION -> select between 1 and 2 only
# 1 -> synthesis without PADS
# 2 -> synthesis with PADS

set PARAMETRO "Not"
set USER_OPTION 2

# Depending on user's selection do the synthesis without or with pads
if {$USER_OPTION == 1} { # Option without pads
        set PARAMETRO_2 "no"
        set rtl_files "${PARAMETRO}.v"
        set top_module "${PARAMETRO}"
} elseif {$USER_OPTION == 2} { # Option with pads
        set PARAMETRO_2 "with"
        set rtl_files [list "IO_${PARAMETRO}.v" "${PARAMETRO}.v"]
        set top_module "IO_${PARAMETRO}"
} else {
        puts "Error: please select 1 or 2"
        exit
}
```

Nota. La imagen muestra la modificación del *script* para realizar la síntesis del circuito con pads.

Luego, se ejecuta el *script*:

Figura 45. Ejecución del *script* desde la terminal para realizar la síntesis de la NOT con pads

```
nanoelectronica@localhost:~/Desktop/Folder_de_Trabajo/workspace/TSMC_65nm/synthesis
File  Edit  View  Search  Terminal  Help
if {[file exists filenames.log]} {
        file rename -force filenames.log logs/$PARAMETRO/filenames.log
}
if {[file exists default.svf]} {
        file rename -force default.svf logs/$PARAMETRO/default.svf
}
# Copy testbench to outputs
if {![file exists outputs/outputs_${PARAMETRO_2}_pads/$PARAMETRO/${PARAMETRO}_tb.v]
} {
file copy -force rtl/$PARAMETRO/${PARAMETRO}_tb.v outputs/outputs_${PARAMETRO_2}_pa
ds/$PARAMETRO/${PARAMETRO}_tb.v
}
# Delete unnecessary files
file delete -force {*}[glob -nocomplain *.mr]
file delete -force {*}[glob -nocomplain *-verilog.pvl]
file delete -force {*}[glob -nocomplain *-verilog.syn]
return
dc_shell> start_gui
dc_shell> Current design is 'Not'.
dc_shell> Current design is 'Not'.
dc_shell> exit

Memory usage for this session 372 Mbytes.
Memory usage for this session including child processes 372 Mbytes.
CPU usage for this session 21 seconds ( 0.01 hours ).
Elapsed time for this session 1411 seconds ( 0.39 hours ).

Thank you...
[nanoelectronica@localhost synthesis]$ dc_shell -f scripts/script_dc.tcl
```

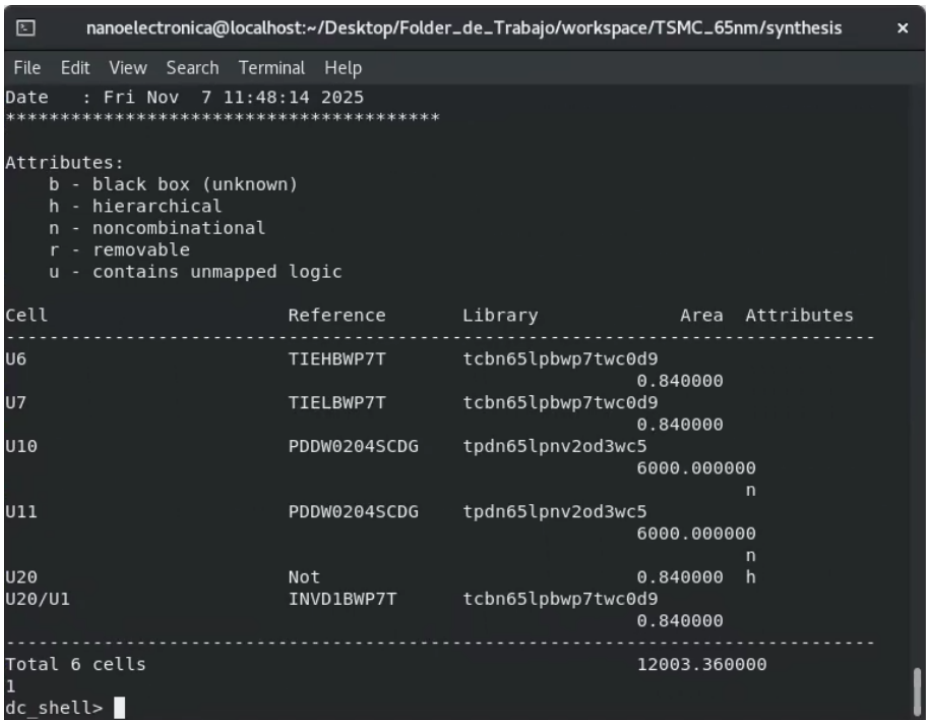
Nota. La imagen muestra la ejecución del *script* desde la terminal.

Una vez sintetizado el diseño, verifica que las celdas estén mapeadas y no haya errores en el diseño sintetizado:

Cuadro 31. Comandos posteriores a la ejecución del *script*

```
report_cell [get_cells -hier *]
check_design
```

Figura 46. Ejecución del comando `report_cell` desde la terminal luego de sintetizar la NOT con pads

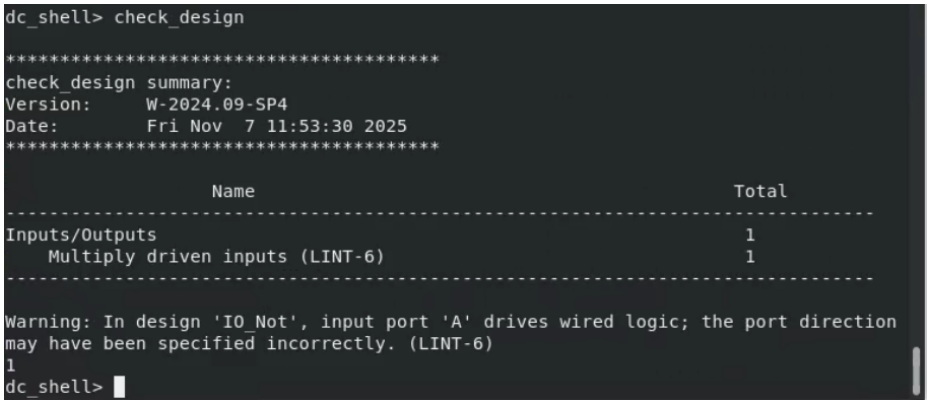


Nota. La imagen muestra la ejecución del comando `report_cell` luego de sintetizar la NOT con pads.

Puede notarse que en el diseño aparecen las celdas TIEH y TIEL, tal como se mencionó al final de la Subsección 10.1.3. Estas celdas son utilizadas para "definir" VDD y VSS dentro del diseño, de esta manera se pueden conectar los pines de los pads y cualquier otra señal a 1'b1 ó 1'b0 según sea el caso.

Además, también aparecen las celdas de los pads que se seleccionaron y la celda de la propia compuerta NOT.

Figura 47. Ejecución del comando `check_design` desde la terminal luego de sintetizar la NOT con pads



Nota. La imagen muestra la ejecución del comando `check_design` luego de sintetizar la NOT con pads.

La figura anterior muestra una advertencia, en la que la señal `A` dice estar manejando una lógica cableada, lo cual indica que la dirección del puerto se haya especificado incorrectamente.

La naturaleza de esta advertencia se debe a que el pin PAD de la celda puede actuar de forma bidireccional, sin embargo, herramienta Design Compiler identifica que dicho pin actúa como salida en todo momento, por lo que se genera esta advertencia. No hay que preocuparse por esto, ya que al no tratarse de un error, el diseño igualmente puede funcionar de manera correcta. Más adelante se verá, que dicha advertencia no posee importancia alguna, ya que la simulación del circuito sintetizado demostrará que todo está funcionando de la manera esperada.

Incluso, cuando se muestren los resultados de los otros circuitos esta advertencia seguirá apareciendo, pero no representará ningún problema para comprobar que los circuitos estén funcionando de forma correcta.

El resultado de la síntesis de la NOT con pads se muestra a continuación:

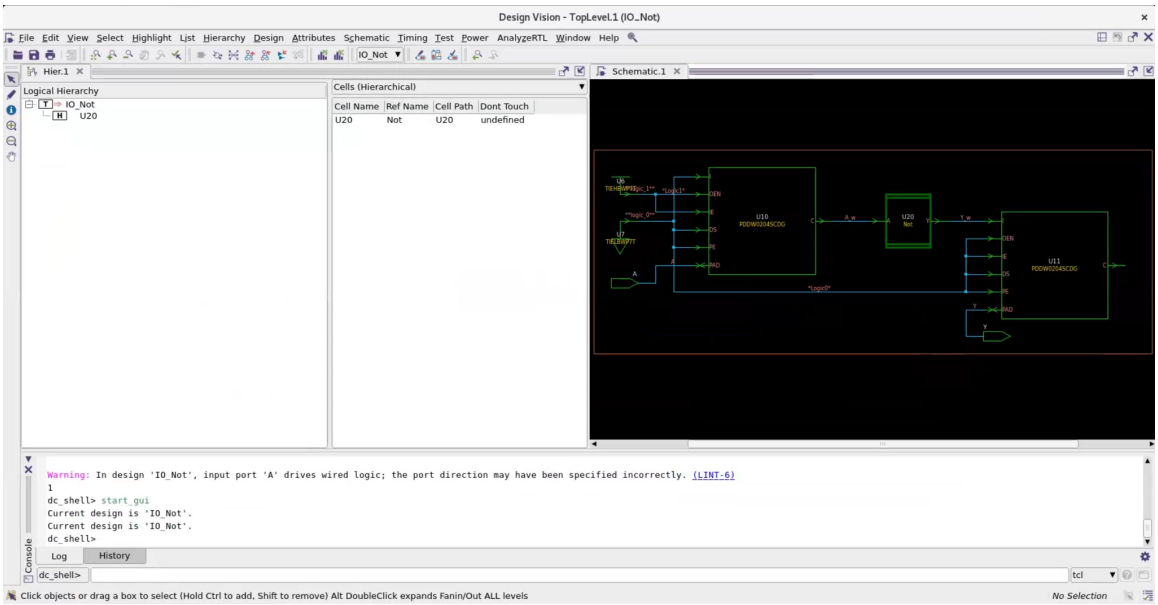
Cuadro 32. *Gate-level netlist* de la NOT con pads

```
////////////////////////////////////////////////////////////
// Created by: Synopsys DC Ultra(TM) in wire load mode
// Version   : W-2024.09-SP4
// Date      : Fri Nov  7 11:47:20 2025
////////////////////////////////////////////////////////////


module Not ( A, Y );
  input A;
  output Y;
```

```
  CKND1 U1 ( .I(A), .ZN(Y) );
endmodule


module IO_Not ( A, Y );
  input A;
  output Y;
  wire   \*Logic1* , \*Logic0* , A_w, Y_w;

  Not U20 ( .A(A_w), .Y(Y_w) );
  PDDW0204SCDG U10 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(A), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(
     A_w) );
  PDDW0204SCDG U11 ( .I(Y_w), .OEN(\*Logic0* ), .IE(\*Logic0* )
     , .PAD(Y), .DS(
        \*Logic0* ), .PE(\*Logic0* ) );
  TIEHBWP7T U6 ( .Z(\*Logic1* ) );
  TIELBWP7T U7 ( .ZN(\*Logic0* ) );
endmodule
```

Posteriormente, se inicia Design Vision para observar el esquemático de la NOT con pads sintetizada:

Figura 48. Esquemático de la NOT con pads en Design Vision



Nota. La imagen muestra el esquemático en Design Vision luego de sintetizar la NOT con pads.

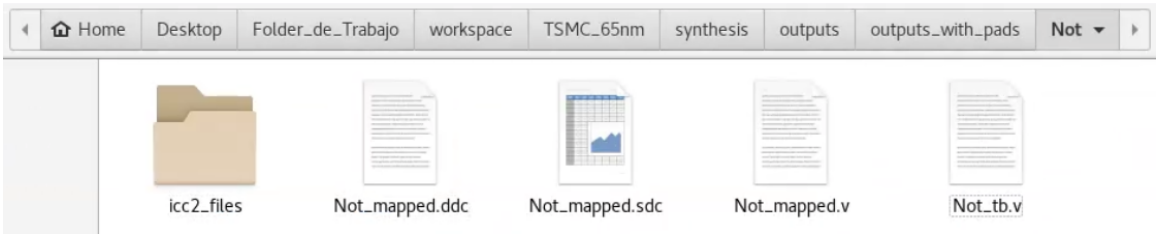
En la Figura 48 se puede observar que las celdas de los pads están presentes tanto en la

entrada como salida de la NOT. También se puede apreciar las distintas conexiones de cada instancia.

11.1.7. Simulación del módulo del circuito sintetizado con pads

Antes de simular la NOT con pads verificar que los archivos generados por la síntesis se encuentren en su respectivo directorio.

Figura 49. Archivos producto de la síntesis de la NOT con pads



Nota. La imagen muestra los archivos producto de la síntesis de la NOT con pads.

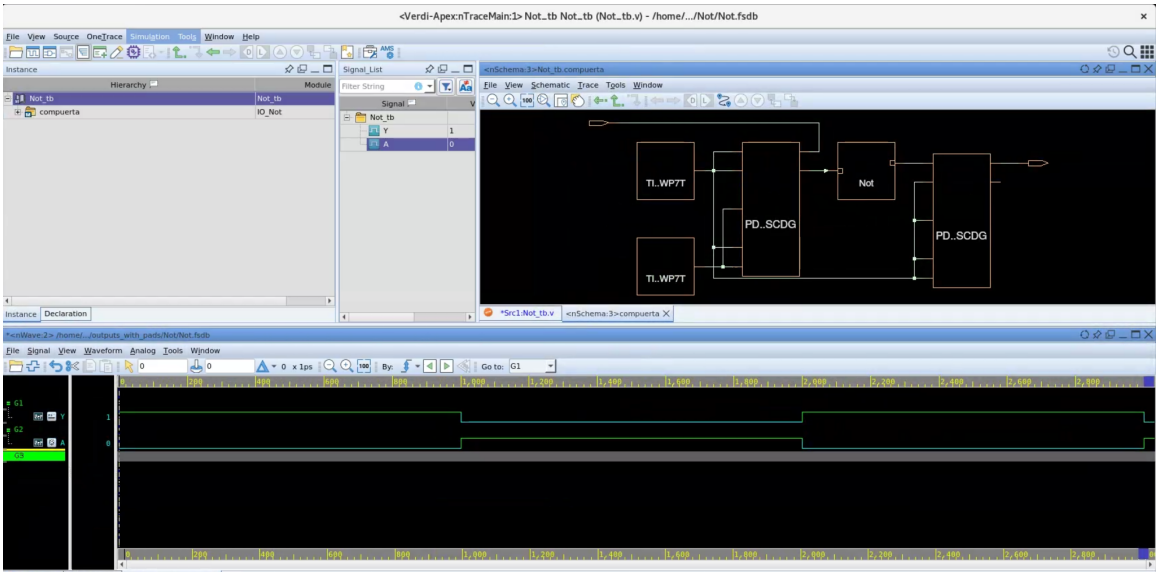
Importante destacar que el directorio donde se encuentran estos resultados con pads es distinto al directorio de los resultados con pads (ver Capítulo 8).

Para simular la NOT con pads ejecutar los siguientes comandos desde la ubicación de estos archivos:

Cuadro 33. Comandos para simular el *gate-level netlist* del *core* de la NOT

```
vcs -full64 -debug_access+all -kdb Not_tb.v Not_mapped.v -v
   ../../tcbn65lp.v -v ../../tpdn65lpnv2od3.v
./simv
verdi -ssf Not.fsdb
```

Figura 50. Simulación de la NOT sintetizada con pads



Nota. La imagen muestra la simulación de la NOT sintetizada con pads.

En la Figura anterior puede notarse las celdas que componen al circuito, aparecen las celdas TIEH, TIEL, los pads y el módulo *core* de la NOT. Al analizar el funcionamiento, la señal de salida `Y` es inversa a la señal entrada `A`, por lo que la síntesis del circuito con pads es correcta.

11.2. Síntesis lógica de un *Carry Look-Ahead Adder* de 4 bits

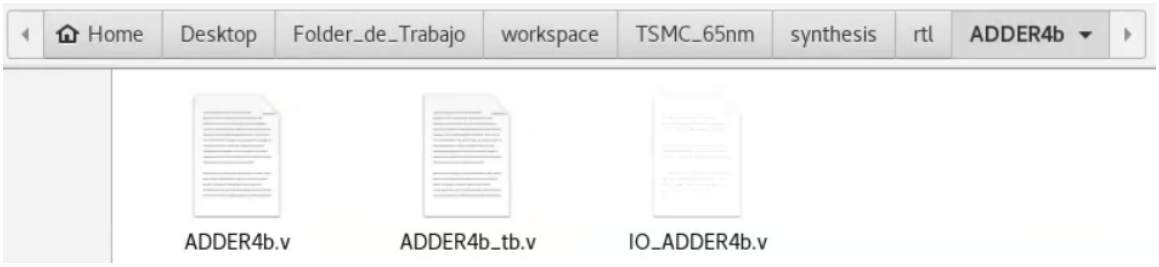
Posteriormente a sintetizar un circuito sencillo como lo es una compuerta NOT, se procedió con la síntesis de un sumador tipo CLA de 4 bits.

En este caso la nomenclatura para los archivos y carpetas cambia, ahora es: `ADDER4b` en lugar de `Not`.

11.2.1. Módulo *core* del circuito, *testbench* y módulo del circuito con pads

Para iniciar con el proceso de síntesis de dicho sumador, asegurarse de tener el archivo del módulo del *core*, *testbench* y el *core* con pads:

Figura 51. Archivos del CLA necesarios para la síntesis



Nota. La imagen muestra los archivos necesarios para la síntesis del CLA en su carpeta respectiva.

Importante destacar la ubicación de estos archivos dentro de la carpeta `rtl`.

A continuación se muestra en el Cuadro 34 el módulo del *core* del circuito, en el Cuadro 35 *testbench* y en el Cuadro 36 el circuito con pads.

Cuadro 34. Archivo ADDER4b.v

```
module ADDER4b(a,b, cin, S,cout);
input [3:0] a,b;
input cin;

output [3:0] S;
output cout;

wire [3:0] p,g,c;
wire w1, w2, w3, w4, w5, w6, w7;

    xor  G1(p[0], a[0], b[0]);
    xor  G17(p[1], a[1], b[1]);
    xor  G18(p[2], a[2], b[2]);
    xor  G19(p[3], a[3], b[3]);
    and  G2(g[0], a[0], b[0]);
    and  G20(g[1], a[1], b[1]);
    and  G21(g[2], a[2], b[2]);
    and  G22(g[3], a[3], b[3]);

    and  G3(w1, p[0], cin);
    or  G4(c[1], g[0], w1);
    and  G5(w2, p[1], g[0]);
    and  G6(w3, p[1], p[0], cin);
    or  G7(c[2], g[1], w2, w3);
    and  G8(w4, p[2], g[1]);
    and  G16(w5, p[2], p[1], g[0]);
    and  G9(w6, p[2], p[1], p[0], cin);
    or  G10(c[3], g[2], w4, w5, w6);
```

```
    and  G11(w7, p[3], g[2]);
    and  G12(w8, p[3], p[2], g[1]);
    and  G13(w9, p[3], p[2], p[1], g[0]);
    and  G14(w10, p[3], p[2], p[1], p[0], cin);
    or  G15(cout, g[3], w7, w8, w9, w10);

    xor  G23(S[0], p[0], cin);
    xor  G24(S[1], p[1], c[1]);
    xor  G25(S[2], p[2], c[2]);
    xor  G26(S[3], p[3], c[3]);

endmodule
```

Cuadro 35. Archivo ADDER4b_tb.v

```
`timescale  1ns/10ps
module ADDER4b_tb;
reg [3:0] A, B;
reg cin;
wire [3:0] S;
wire Co;

// Instanciar el modulo carry_look_ahead_4bit
ADDER4b U10(.a(A), .b(B), .cin(cin), .S(S), .cout(Co));

initial
begin
    // Inicializacion de archivos de volcado de datos
    $fsdbDumpfile("ADDER4b.fsdb");
    $fsdbDumpvars(0, ADDER4b_tb);
    $dumpfile("ADDER4b.vcd");
    $dumpvars(0, ADDER4b_tb);

    // Pruebas con diferentes valores de entrada
    cin = 0; A = 4'd5; B = 4'd3;
    #100;

    A = 4'd6;
    B = 4'd9;
    #100;

    A = 4'd12;
    B = 4'd4;
    #100;

    A = 4'd7;
    B = 4'd8;
    #100;
```

```
    A = 4'd15;
    B = 4'd15;
    #100;

    // Finalizar simulacion
    $finish;
end
endmodule
```

Cuadro 36. Archivo IO_ADDER4b.v

```
module IO_ADDER4b (
    input [3:0] a,
    input [3:0] b,
    input cin,
    output [3:0] S,
    output cout
);
    wire [3:0] a_w, b_w, S_w;
    wire cin_w, cout_w;
    wire [3:0] p, g, c;
    wire w1, w2, w3, w4, w5, w6, w7, w8, w9, w10;

    // Instancia del modulo carry_look_ahead_4bit
    ADDER4b U (
        .a(a_w),
        .b(b_w),
        .cin(cin_w),
        .S(S_w),
        .cout(cout_w)
    );

    // Interconexion de pines de entrada (a, b, cin)
    // Entradas a
    PDDW0204SCDG U_A0 (.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(a
       [0]), .C(a_w[0]), .PE(1'b0), .IE(1'b1));
    PDDW0204SCDG U_A1 (.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(a
       [1]), .C(a_w[1]), .PE(1'b0), .IE(1'b1));
    PDDW0204SCDG U_A2 (.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(a
       [2]), .C(a_w[2]), .PE(1'b0), .IE(1'b1));
    PDDW0204SCDG U_A3 (.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(a
       [3]), .C(a_w[3]), .PE(1'b0), .IE(1'b1));

    // Entradas b
    PDDW0204SCDG U_B0 (.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(b
       [0]), .C(b_w[0]), .PE(1'b0), .IE(1'b1));
    PDDW0204SCDG U_B1 (.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(b
```

```
        [1]), .C(b_w[1]), .PE(1'b0), .IE(1'b1));
    PDDW0204SCDG U_B2 (.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(b
        [2]), .C(b_w[2]), .PE(1'b0), .IE(1'b1));
    PDDW0204SCDG U_B3 (.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(b
        [3]), .C(b_w[3]), .PE(1'b0), .IE(1'b1));

    // Entrada cin
    PDDW0204SCDG U_Cin (.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(
        cin), .C(cin_w), .PE(1'b0), .IE(1'b1));

    // Interconexion de pines de salida (S, cout)
    // Salidas S
    PDDW0204SCDG U_S0 (.I(S_w[0]), .DS(1'b0), .OEN(1'b0), .PAD(
        S[0]), .C(1'bx), .PE(1'b0), .IE(1'b0));
    PDDW0204SCDG U_S1 (.I(S_w[1]), .DS(1'b0), .OEN(1'b0), .PAD(
        S[1]), .C(1'bx), .PE(1'b0), .IE(1'b0));
    PDDW0204SCDG U_S2 (.I(S_w[2]), .DS(1'b0), .OEN(1'b0), .PAD(
        S[2]), .C(1'bx), .PE(1'b0), .IE(1'b0));
    PDDW0204SCDG U_S3 (.I(S_w[3]), .DS(1'b0), .OEN(1'b0), .PAD(
        S[3]), .C(1'bx), .PE(1'b0), .IE(1'b0));

    // Salida cout
    PDDW0204SCDG U_Cout (.I(cout_w), .DS(1'b0), .OEN(1'b0), .
        PAD(cout), .C(1'bx), .PE(1'b0), .IE(1'b0));

    // Pines de VDD y VSS
    PVDD1CDG PVDD();
    PVSS1CDG PVSS();

endmodule
```

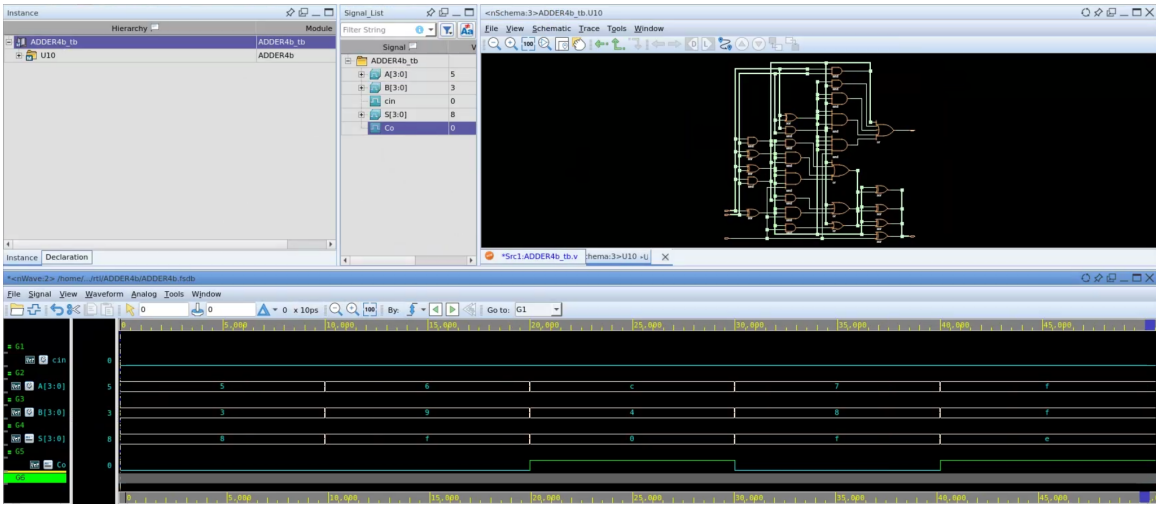
11.2.2. Simulación del módulo *core* del circuito

Posteriormente, se realiza la simulación del sumador como tal sin pads utilizando VCS y Verdi dentro del directorio donde estos archivos se encuentran.

Cuadro 37. Comandos para la compilación y simulación del *core* del ADDER4b

```
vcs -full64 -debug_access+all -kdb ADDER4b_tb.v ADDER4b.v
./simv
verdi -ssf ADDER4b.fsdb
```

Figura 52. Simulación del CLA de 4 bits



Nota. La imagen muestra la simulación del CLA de 4 bits en Verdi.

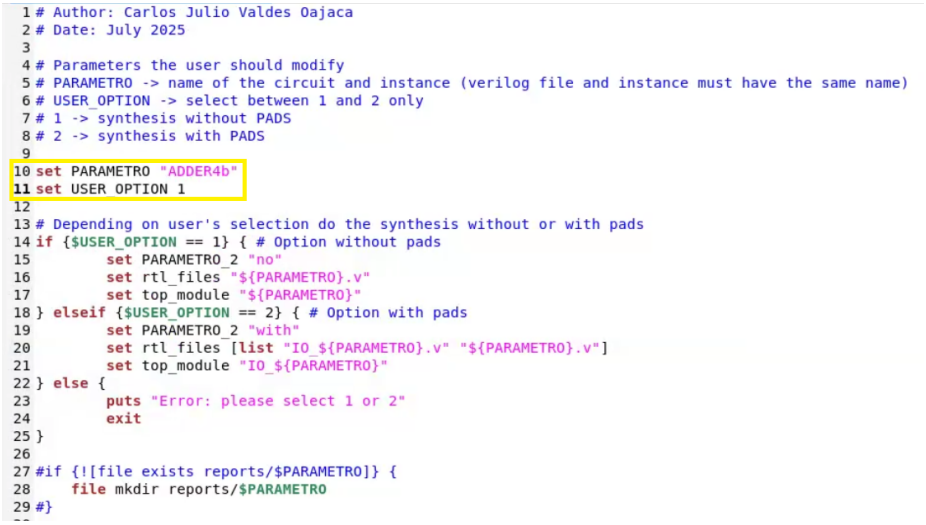
La figura anterior muestra la simulación del CLA. Puede comprobarse que el esquemático muestra diversas compuertas conectada entre sí, esto se debe a que la descripción que se utilizó para diseñar el CLA en Verilog fue *behavioral*.

Al analizar la señales puede observarse que la suma está en formato hexadecimal. La suma se hace correctamente, y notar que para los casos donde la suma es mayor a 15 el bit de *carry out* (`Co`) se enciende como es de esperarse.

11.2.3. Síntesis del módulo *core* del circuito

Antes de ejecutar el *script* de síntesis lógica es importante modificar los parámetros de usuario para definir correctamente el circuito a sintetizar y si la síntesis será con o sin pads.

Figura 53. Modificación de los parámetros del *script* para sintetizar el CLA

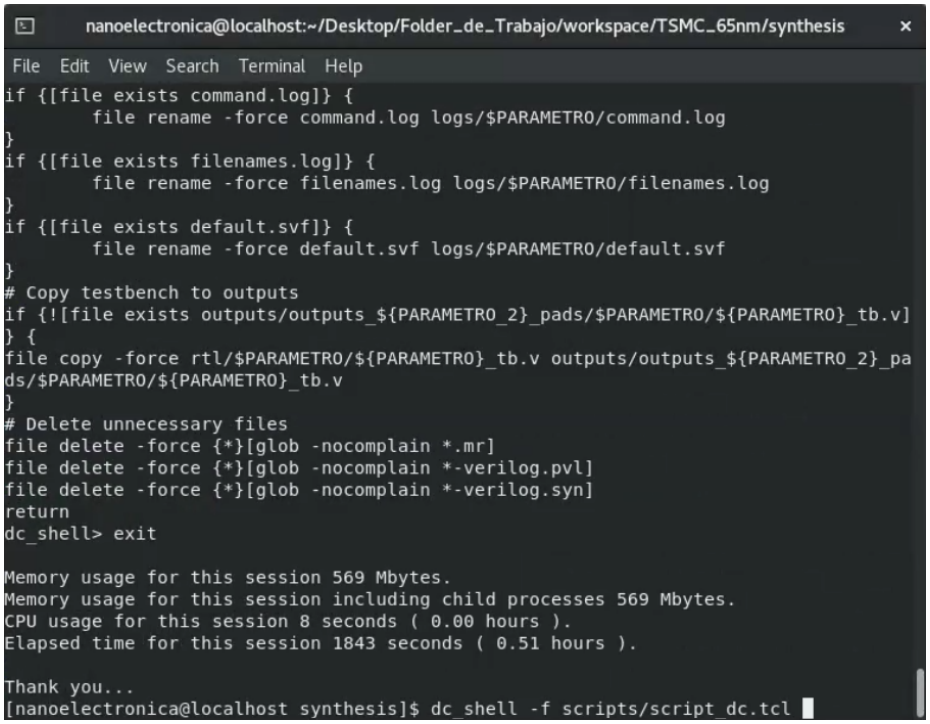


```
# Author: Carlos Julio Valdes Oajaca
# Date: July 2025

# Parameters the user should modify
# PARAMETRO -> name of the circuit and instance (verilog file and instance must have the same name)
# USER_OPTION -> select between 1 and 2 only
# 1 -> synthesis without PADS
# 2 -> synthesis with PADS

set PARAMETRO "ADDER4b"
set USER_OPTION 1

# Depending on user's selection do the synthesis without or with pads
if {$USER_OPTION == 1} { # Option without pads
        set PARAMETRO_2 "no"
        set rtl_files "${PARAMETRO}.v"
        set top_module "${PARAMETRO}"
} elseif {$USER_OPTION == 2} { # Option with pads
        set PARAMETRO_2 "with"
        set rtl_files [list "IO_${PARAMETRO}.v" "${PARAMETRO}.v"]
        set top_module "IO_${PARAMETRO}"
} else {
        puts "Error: please select 1 or 2"
        exit
}

#if {![file exists reports/$PARAMETRO]} {
    file mkdir reports/$PARAMETRO
#}
```

Nota. La imagen muestra la modificación de PARAMETRO y USER_OPTION para sintetizar el CLA.

Posteriormente ejecutar el *script*:

Figura 54. Ejecución del *script* desde la terminal para realizar la síntesis del *core* del CLA

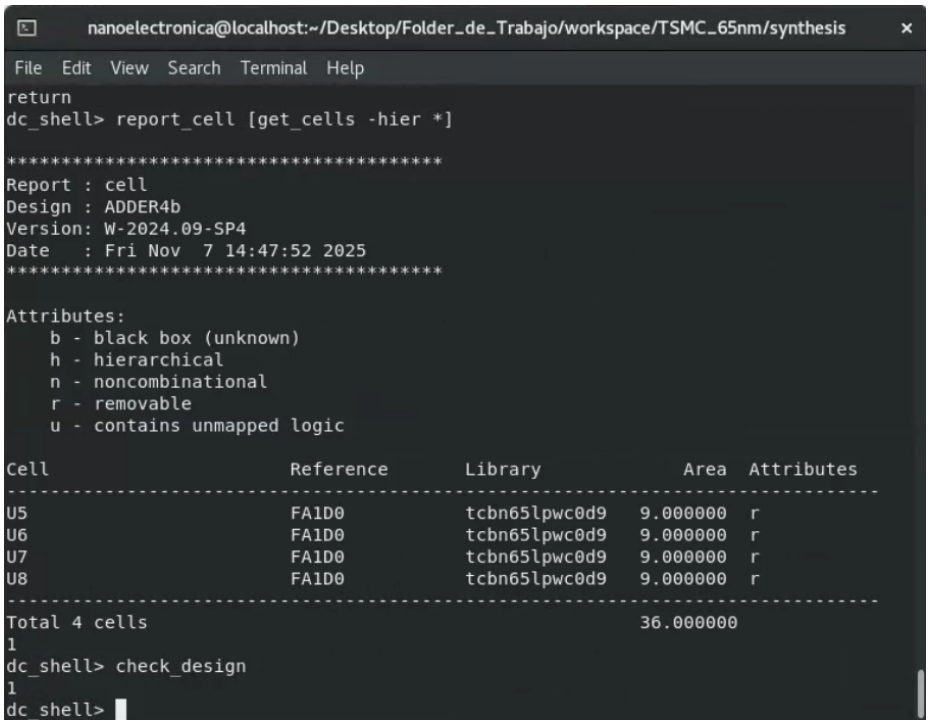


```
nanoelectronica@localhost:~/Desktop/Folder_de_Trabajo/workspace/TSMC_65nm/synthesis
File  Edit  View  Search  Terminal  Help
if {[file exists command.log]} {
        file rename -force command.log logs/$PARAMETRO/command.log
}
if {[file exists filenames.log]} {
        file rename -force filenames.log logs/$PARAMETRO/filenames.log
}
if {[file exists default.svf]} {
        file rename -force default.svf logs/$PARAMETRO/default.svf
}
# Copy testbench to outputs
if {![file exists outputs/outputs_${PARAMETRO_2}_pads/$PARAMETRO/${PARAMETRO}_tb.v]
} {
file copy -force rtl/$PARAMETRO/${PARAMETRO}_tb.v outputs/outputs_${PARAMETRO_2}_pa
ds/$PARAMETRO/${PARAMETRO}_tb.v
}
# Delete unnecessary files
file delete -force {*}[glob -nocomplain *.mr]
file delete -force {*}[glob -nocomplain *-verilog.pvl]
file delete -force {*}[glob -nocomplain *-verilog.syn]
return
dc_shell> exit

Memory usage for this session 569 Mbytes.
Memory usage for this session including child processes 569 Mbytes.
CPU usage for this session 8 seconds ( 0.00 hours ).
Elapsed time for this session 1843 seconds ( 0.51 hours ).

Thank you...
[nanoelectronica@localhost synthesis]$ dc_shell -f scripts/script_dc.tcl
```

Nota. La imagen muestra la ejecución del *script* desde la terminal.

Verificar las celdas del diseño y si hay errores:

Figura 55. Celdas estándar y verificación de errores de la síntesis del *core* del CLA



Nota. La imagen muestra las celdas y verificación de errores del CLA sintetizado.

Al observar la figura anterior puede verse que el diseño está conformado por cuatro celdas FA1D0, las cuales al revisar el PDF de las celdas se concluye que cada una es un sumador de 1 bit cada una.

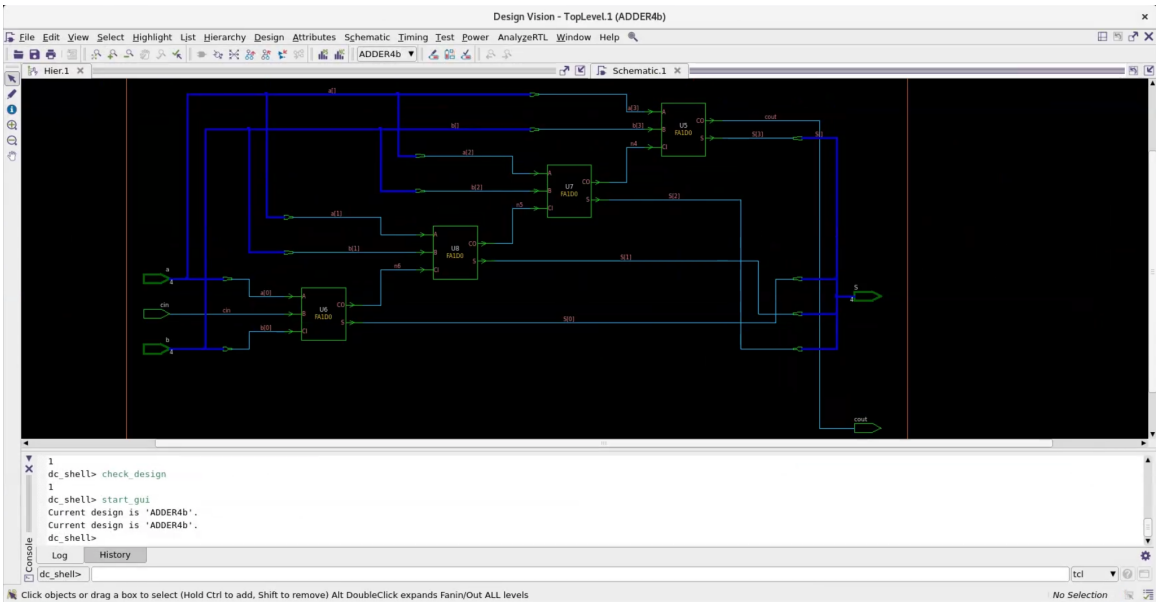
A continuación se muestra el archivo del CLA sintetizado junto con su esquemático en Design Vision:

Cuadro 38. Archivo ADDER4b_mapped.v en la carpeta `outputs_no_pads/`

```
////////////////////////////////////////////////////////////////
// Created by: Synopsys DC Ultra(TM) in wire load mode
// Version   : W-2024.09-SP4
// Date      : Fri Nov  7 14:46:52 2025
////////////////////////////////////////////////////////////////


module ADDER4b ( a, b, cin, S, cout );
  input [3:0] a;
  input [3:0] b;
  output [3:0] S;
  input cin;
  output cout;
```

```
  wire    n4, n5, n6;

  FA1D0 U5 ( .A(a[3]), .B(b[3]), .CI(n4), .CO(cout), .S(S[3]) )
      ;
  FA1D0 U6 ( .A(a[0]), .B(cin), .CI(b[0]), .CO(n6), .S(S[0]) );
  FA1D0 U7 ( .A(a[2]), .B(b[2]), .CI(n5), .CO(n4), .S(S[2]) );
  FA1D0 U8 ( .A(a[1]), .B(b[1]), .CI(n6), .CO(n5), .S(S[1]) );
endmodule
```

Figura 56. Esquemático del CLA en Design Vision

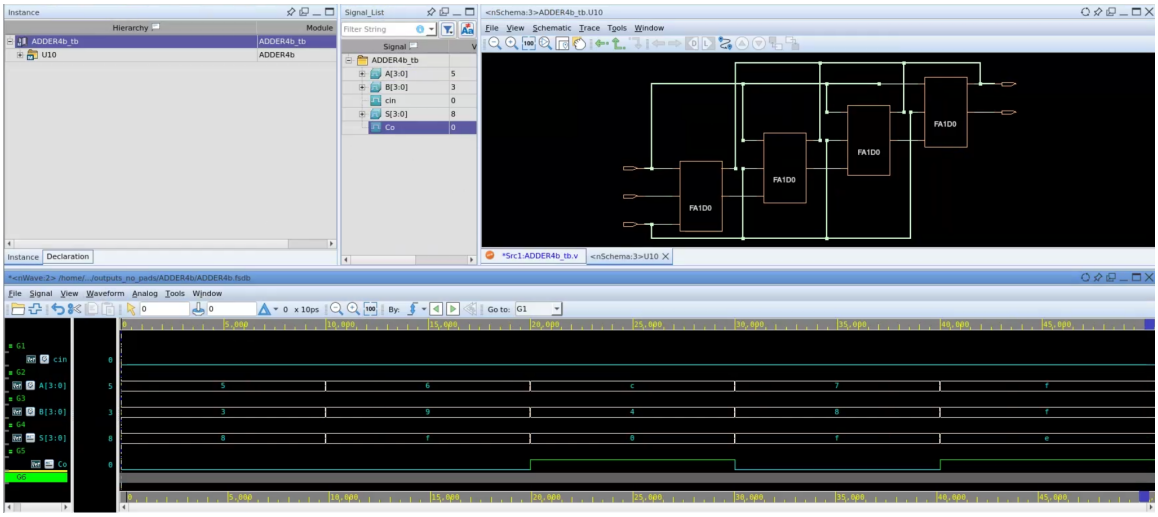


Nota. La imagen muestra el esquemático del CLA en Design Vision.

11.2.4. Simulación del *core* del circuito sintetizado

Luego de verificar con el comando `check_design` que no existen errores ni advertencias de importancia se procede con la simulación del circuito sintetizado.

Figura 57. Simulación del CLA sintetizado



Nota. La imagen muestra la simulación del CLA sintetizado en Verdi.

La Figura 57 demuestra que el CLA sintetizado es correcto.

11.2.5. Simulación del módulo del circuito con pads

Una vez realizado la verificación correspondiente al *core* del CLA se comienza con el proceso de síntesis del circuito con pads.

Para ello, se comienza simulando el archivo con pads antes de la síntesis, en este caso: `IO_ADDER4b.v` con el *testbench*. Importante recordar que la instancia dentro del archivo *testbench* ya no es `ADDER4b`, ahora es `IO_ADDER4b`.

Figura 58. Modificación de la instancia dentro del *testbench* del CLA

```
1 `timescale 1ns/10ps
2 module ADDER4b_tb;
3 reg [3:0] A, B;
4 reg cin;
5 wire [3:0] S;
6 wire Co;
7
8 // Instanciar el módulo carry_look_ahead_4bit
9 IO_ADDER4b U10(.a(A), .b(B), .cin(cin), .S(S), .cout(Co));
10
11 initial
12 begin
13     // Inicialización de archivos de volcado de datos
14     $fsdbDumpfile("ADDER4b.fsdb");
15     $fsdbDumpvars(0, ADDER4b_tb);
16     $dumpfile("ADDER4b.vcd");
17     $dumpvars(0, ADDER4b_tb);
18
19     // Pruebas con diferentes valores de entrada
20     cin = 0; A = 4'd5; B = 4'd3;
21     #100;
```

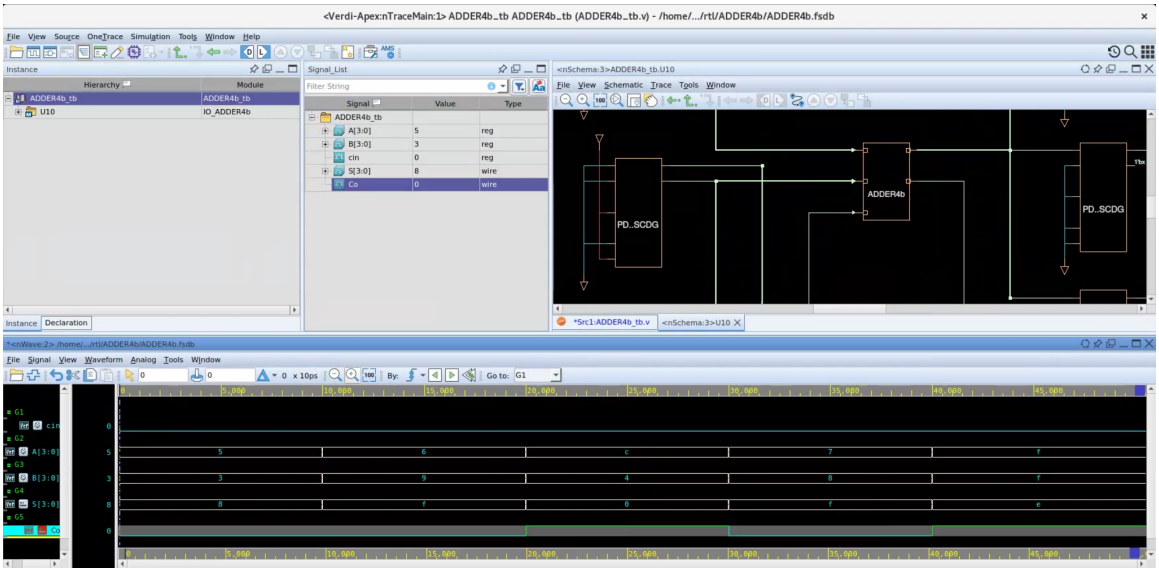
Nota. La imagen muestra la modificación de la instancia dentro del *testbench* del CLA.

Luego, ejecutar los comandos respectivos para simular el CLA con pads:

Cuadro 39. Comandos para la compilación y simulación del CLA con pads

```
vcs -full64 -debug_access+all -kdb ADDER4b_tb.v IO_ADDER4b.v
   ADDER4b.v -v ../../outputs/tpdn65lpnv2od3.v
./simv
verdi -ssf ADDER4b.fsdb
```

Figura 59. Simulación del CLA con pads



Nota. La imagen muestra la simulación del CLA con pads en Verdi.

Como la Figura 59 muestra, los pads de entradas y salida se encuentran a ambos extremos del *core* del CLA. Se realizó *zoom* en la parte del esquemático para que se pudiese apreciar el nombre de las instancias.

Con respecto al funcionamiento, el CLA suma correctamente tal como lo hace el circuito sin los pads.

11.2.6. Síntesis del módulo del circuito con pads

Para sintetizar el CLA con pads primero hay que modificar el parámetro `USER_OPTION` del *script*:

Figura 60. Modificación del parámetro `USER_OPTION`

```
# Author: Carlos Julio Valdes Oajaca
# Date: July 2025

# Parameters the user should modify
# PARAMETRO -> name of the circuit and instance (verilog file and instance must have the same name)
# USER_OPTION -> select between 1 and 2 only
# 1 -> synthesis without PADS
# 2 -> synthesis with PADS

set PARAMETRO "ADDER4b"
set USER_OPTION 2

# Depending on user's selection do the synthesis without or with pads
if {$USER_OPTION == 1} { # Option without pads
        set PARAMETRO_2 "no"
        set rtl_files "${PARAMETRO}.v"
        set top_module "${PARAMETRO}"
} elseif {$USER_OPTION == 2} { # Option with pads
        set PARAMETRO_2 "with"
        set rtl_files [list "IO_${PARAMETRO}.v" "${PARAMETRO}.v"]
        set top_module "IO_${PARAMETRO}"
} else {
        puts "Error: please select 1 or 2"
        exit
}
```

Nota. La imagen muestra la modificación del parámetro `USER_OPTION`.

Posteriormente, ejecutar el script:

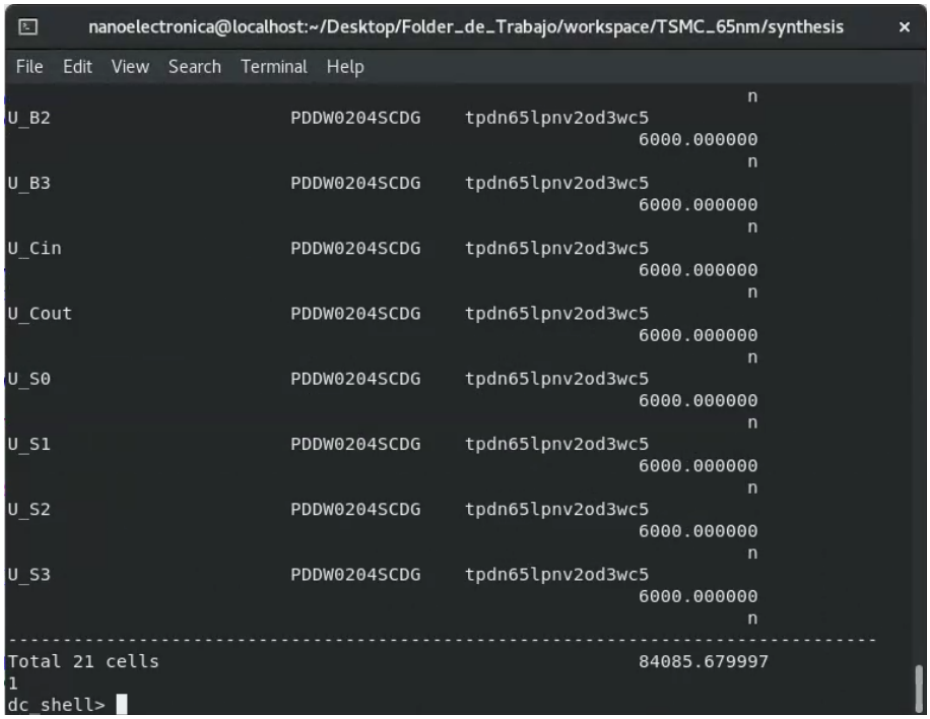
Figura 61. Ejecución del *script* desde la terminal para realizar la síntesis del CLA con pads

```
nanoelectronica@localhost:~/Desktop/Folder_de_Trabajo/workspace/TSMC_65nm/synthesis
File  Edit  View  Search  Terminal  Help
if {[file exists command.log]} {
        file rename -force command.log logs/$PARAMETRO/command.log
}
if {[file exists filenames.log]} {
        file rename -force filenames.log logs/$PARAMETRO/filenames.log
}
if {[file exists default.svf]} {
        file rename -force default.svf logs/$PARAMETRO/default.svf
}
# Copy testbench to outputs
if {![file exists outputs/outputs_${PARAMETRO_2}_pads/$PARAMETRO/${PARAMETRO}_tb.v]
} {
file copy -force rtl/$PARAMETRO/${PARAMETRO}_tb.v outputs/outputs_${PARAMETRO_2}_pa
ds/$PARAMETRO/${PARAMETRO}_tb.v
}
# Delete unnecessary files
file delete -force {*}[glob -nocomplain *.mr]
file delete -force {*}[glob -nocomplain *-verilog.pvl]
file delete -force {*}[glob -nocomplain *-verilog.syn]
return
dc_shell> exit

Memory usage for this session 569 Mbytes.
Memory usage for this session including child processes 569 Mbytes.
CPU usage for this session 8 seconds ( 0.00 hours ).
Elapsed time for this session 1843 seconds ( 0.51 hours ).

Thank you...
[nanoelectronica@localhost synthesis]$ dc_shell -f scripts/script_dc.tcl
```

Nota. La imagen muestra la ejecución del *script* desde la terminal.

Verificar las celdas del CLA sintetizado y si existen errores:

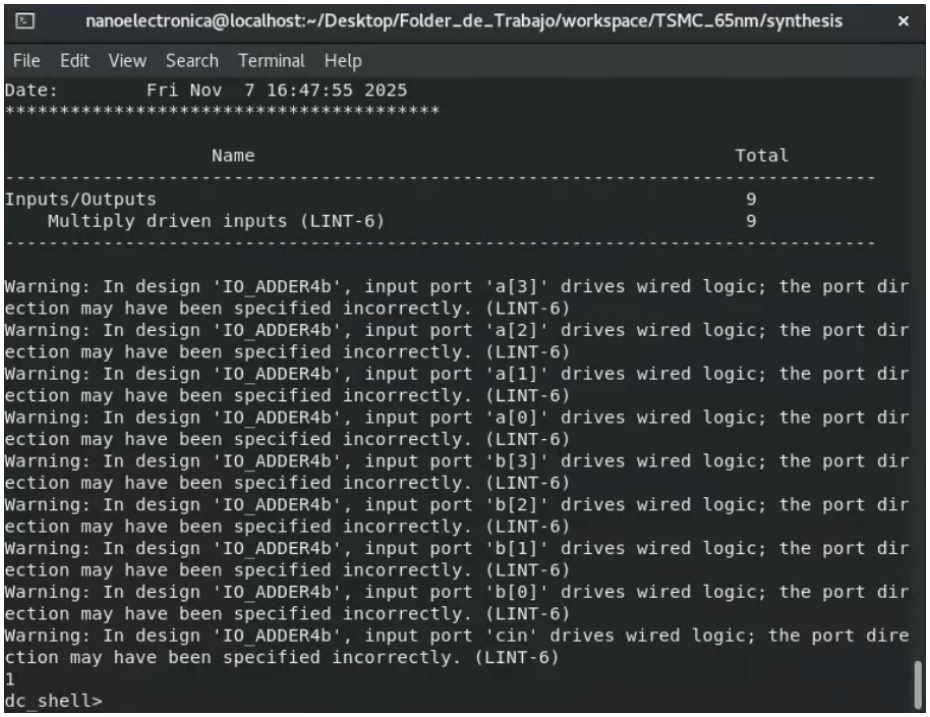
Figura 62. Celdas estándar totales del CLA sintetizado con pads



Nota. La imagen muestra el total de celdas del CLA sintetizado con pads.

Conforme la complejidad de los diseños aumenta, la cantidad de celdas también. Es por eso que la creación de reportes se vuelve valiosa. Como se observa en el *script* en la Sección 10.2, se genera un reporte con todas las celdas del archivo sintetizado en el directorio: `reports/PARAMETRO`. Se sugiere revisar los reportes a medida que el diseño se vuelve más complejo.

Figura 63. Ejecución del comando `check_design` luego de sintetizar el CLA con pads



Nota. La imagen muestra la ejecución del comando `check_design`.

De igual manera, existe un reporte en `reports/PARAMETRO` que muestra lo mismo que la Figura 63.

A continuación, se muestra el archivo del CLA con pads sintetizado y el esquemático en Design Vision:

Cuadro 40. Archivo ADDER4b_mapped.v en la carpeta `outputs_with_pads/`

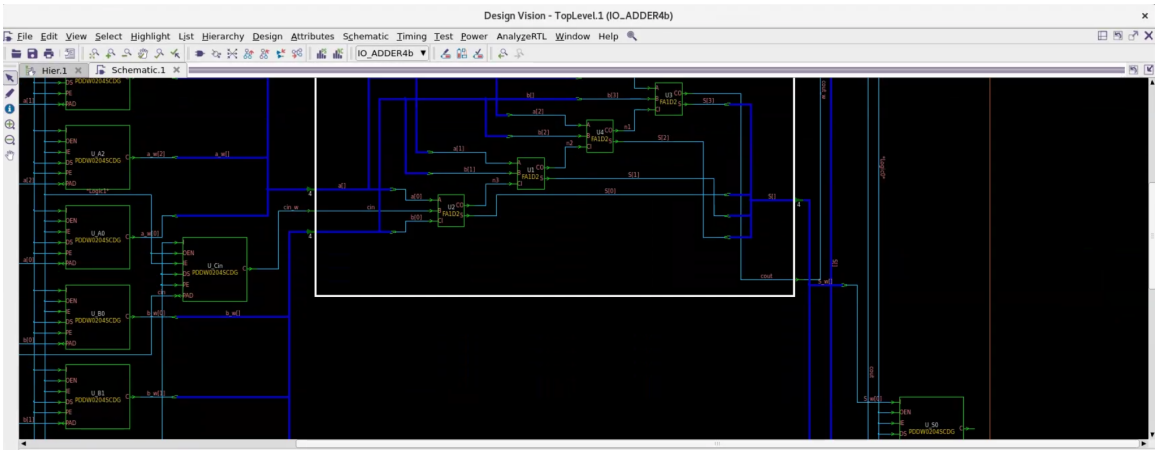
```
////////////////////////////////////////////////////////////////
// Created by: Synopsys DC Ultra(TM) in wire load mode
// Version   : W-2024.09-SP4
// Date      : Fri Nov  7 16:38:41 2025
////////////////////////////////////////////////////////////////


module ADDER4b ( a, b, cin, S, cout );
  input [3:0] a;
  input [3:0] b;
  output [3:0] S;
  input cin;
  output cout;
  wire   n1, n2, n3;

  FA1D2 U1 ( .A(a[1]), .B(b[1]), .CI(n3), .CO(n2), .S(S[1]) );
```

```
  FA1D2 U2 ( .A(a[0]), .B(cin), .CI(b[0]), .CO(n3), .S(S[0]) );
  FA1D2 U3 ( .A(a[3]), .B(b[3]), .CI(n1), .CO(cout), .S(S[3]) )
     ;
  FA1D2 U4 ( .A(a[2]), .B(b[2]), .CI(n2), .CO(n1), .S(S[2]) );
endmodule


module IO_ADDER4b ( a, b, cin, S, cout );
  input [3:0] a;
  input [3:0] b;
  output [3:0] S;
  input cin;
  output cout;
  wire   \*Logic1* , \*Logic0* , cin_w, cout_w;
  wire   [3:0] a_w;
  wire   [3:0] b_w;
  wire   [3:0] S_w;

  ADDER4b U ( .a(a_w), .b(b_w), .cin(cin_w), .S(S_w), .cout(
     cout_w) );
  PDDW0204SCDG U_A0 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(
        a[0]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(a_w[0]) );
  PDDW0204SCDG U_A1 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(
        a[1]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(a_w[1]) );
  PDDW0204SCDG U_A2 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(
        a[2]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(a_w[2]) );
  PDDW0204SCDG U_A3 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(
        a[3]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(a_w[3]) );
  PDDW0204SCDG U_B0 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(
        b[0]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(b_w[0]) );
  PDDW0204SCDG U_B1 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(
        b[1]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(b_w[1]) );
  PDDW0204SCDG U_B2 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(
        b[2]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(b_w[2]) );
  PDDW0204SCDG U_B3 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(
        b[3]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(b_w[3]) );
  PDDW0204SCDG U_Cin ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
     Logic1* ),
        .PAD(cin), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(cin_w)
           );
```

```
  PDDW0204SCDG U_S0 ( .I(S_w[0]), .OEN(\*Logic0* ), .IE(\*
     Logic0* ), .PAD(S[0]), .DS(\*Logic0* ), .PE(\*Logic0* ) );
  PDDW0204SCDG U_S1 ( .I(S_w[1]), .OEN(\*Logic0* ), .IE(\*
     Logic0* ), .PAD(S[1]), .DS(\*Logic0* ), .PE(\*Logic0* ) );
  PDDW0204SCDG U_S2 ( .I(S_w[2]), .OEN(\*Logic0* ), .IE(\*
     Logic0* ), .PAD(S[2]), .DS(\*Logic0* ), .PE(\*Logic0* ) );
  PDDW0204SCDG U_S3 ( .I(S_w[3]), .OEN(\*Logic0* ), .IE(\*
     Logic0* ), .PAD(S[3]), .DS(\*Logic0* ), .PE(\*Logic0* ) );
  PDDW0204SCDG U_Cout ( .I(cout_w), .OEN(\*Logic0* ), .IE(\*
     Logic0* ), .PAD(
        cout), .DS(\*Logic0* ), .PE(\*Logic0* ) );
  TIEH U6 ( .Z(\*Logic1* ) );
  TIEL U7 ( .ZN(\*Logic0* ) );
endmodule
```

De igual manera que con la NOT con pads, puede apreciarse la presencia de las celdas TIEH y TIEL.

Figura 64. Esquemático del CLA con pads



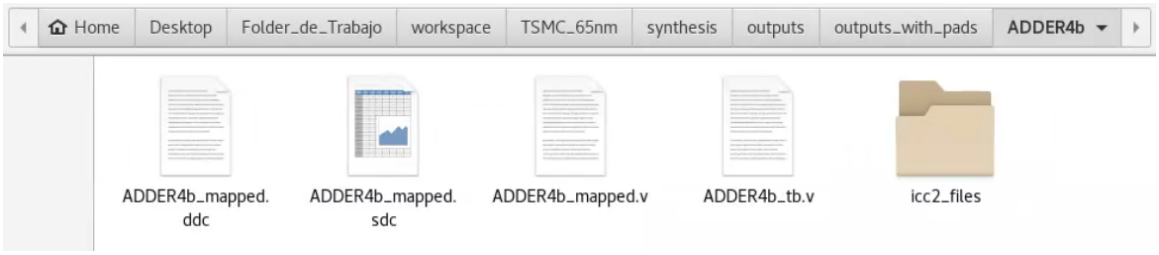
Nota. La imagen muestra el esquemático del CLA con pads en Design Vision.

En la figura anterior se pueden observar los cuatro sumadores de 1 bit conectados con pads a ambos extremos.

11.2.7. Simulación del módulo del circuito sintetizado con pads

Antes de simular el circuito del CLA con pads se deben tener los archivos correspondientes dentro del directorio apropiado:

Figura 65. Archivos que resultan de la síntesis del CLA con pads



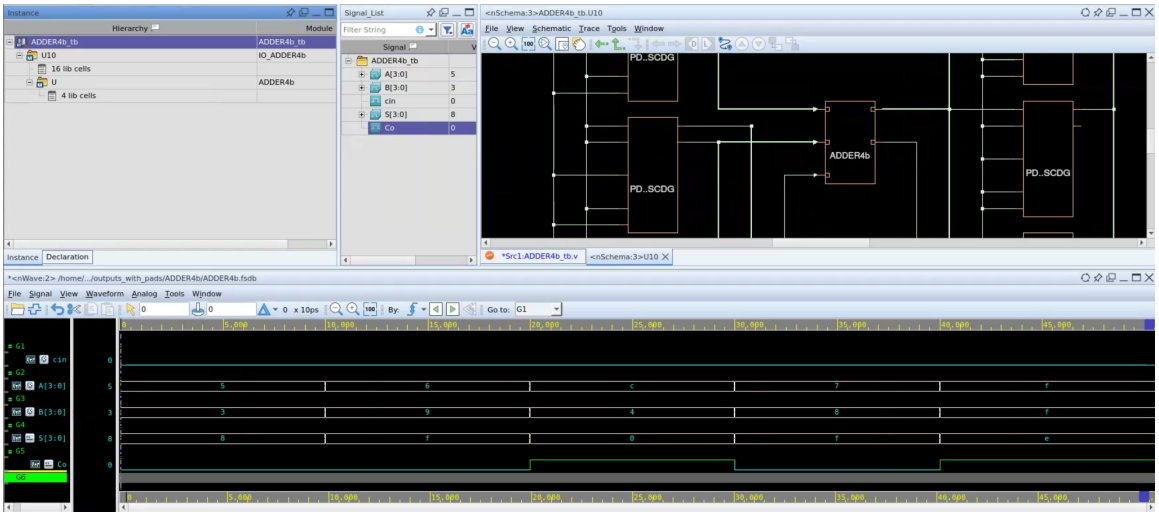
Nota. La imagen muestra los archivos que resultan de la síntesis del CLA con pads.

Luego se simula el CLA sintetizado con pads:

Cuadro 41. Comandos para la compilación y simulación del CLA sintetizado con pads

```
vcs -full64 -debug_access+all -kdb ADDER4b_tb.v ADDER4b_mapped.
   v -v ../../tcbn65lp.v -v ../../tpdn65lpnv2od3.v
./simv
verdi -ssf ADDER4b.fsdb
```

Figura 66. Simulación del CLA sintetizado con pads



Nota. La imagen muestra la simulación del CLA sintetizado con pads en Verdi.

La figura anterior muestra que la suma y el funcionamiento del CLA con pads es correcto, además el esquemático de Verdi coincide con el de Design Vision.

11.3. Síntesis lógica de un *Ripple Carry Counter* de 4 bits

Luego de haber hecho la síntesis de circuitos combinacionales, se procedió a sintetizar un circuito secuencial con señal de reloj.

En este caso se realizó la síntesis de un *Ripple Carry Counter* de 4 bits. La nomenclatura para este caso es: `rc_counter`.

11.3.1. Módulo *core* del circuito, *testbench* y módulo del circuito con pads

Para comenzar con la síntesis se diseñaron los tres módulos principales que se necesitan en todo momento. El Cuadro 42 muestra el *core* del RCC, el Cuadro 43 muestra el *testbench* y el Cuadro 44 muestra el circuito con pads.

Cuadro 42. Archivo rc_counter.v

```
module rc_counter(q, clk, reset);

output [3:0] q;
input clk, reset;

T_FF tff0(q[0], clk, reset);
T_FF tff1(q[1], q[0], reset);
T_FF tff2(q[2], q[1], reset);
T_FF tff3(q[3], q[2], reset);

endmodule

//Flip-Flop T_FF
module T_FF(q, clk, reset);
    output q;
    input clk, reset;
    wire d;

    D_FF dff0(q, d, clk, reset);
    assign d = ~q; //toggle
endmodule

//Flip-Flop D_FF
module D_FF(q, d, clk, reset);

output reg q;
input d, clk, reset;

always @(posedge reset or negedge clk)
if(reset)
    q <= 1'b0;
```

```
else
    q <= d;

endmodule
```

Cuadro 43. Archivo rc_counter_tb.v

```
`timescale 1ns/10ps
module rc_counter_tb;
    reg reset, clk;
    wire [3:0] q;
    rc_counter r1(q, clk, reset);
    //simulate clock signal
    initial
    clk = 1'b0; //set clk to 0
    always
    #5 clk = ~clk;

    //control
    initial begin
        reset = 1'b1;
        #15 reset = 1'b0;
        #180 reset = 1'b1;
        #10 reset = 1'b0;
        #20 $finish; //terminate the sim
    end

    initial begin
        $fsdbDumpfile("rc_counter.fsdb");
        $fsdbDumpvars(0, rc_counter_tb);
        $dumpfile("rc_counter.vcd");
        $dumpvars(0, rc_counter_tb);
    end
endmodule
```

Cuadro 44. Archivo IO_rc_counter.v

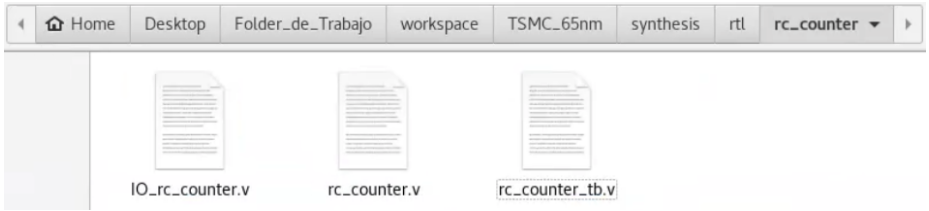
```
module IO_rc_counter(q, clk, reset);

output [3:0] q;
input clk, reset;
wire [3:0] q_w;
wire clk_w, reset_w;


rc_counter U10(q_w, clk_w, reset_w);

//Inputs:
```

```
PDDW0204SCDG U_A0 (.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(clk),
    .C(clk_w), .PE(1'b0), .IE(1'b1));
PDDW0204SCDG U_A1 (.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(reset)
    , .C(reset_w), .PE(1'b0), .IE(1'b1));

//Outputs:
PDDW0204SCDG U_q0 (.I(q_w[0]), .DS(1'b0), .OEN(1'b0), .PAD(q
    [0]), .C(1'bx), .PE(1'b0), .IE(1'b0));
PDDW0204SCDG U_q1 (.I(q_w[1]), .DS(1'b0), .OEN(1'b0), .PAD(q
    [1]), .C(1'bx), .PE(1'b0), .IE(1'b0));
PDDW0204SCDG U_q2 (.I(q_w[2]), .DS(1'b0), .OEN(1'b0), .PAD(q
    [2]), .C(1'bx), .PE(1'b0), .IE(1'b0));
PDDW0204SCDG U_q3 (.I(q_w[3]), .DS(1'b0), .OEN(1'b0), .PAD(q
    [3]), .C(1'bx), .PE(1'b0), .IE(1'b0));

endmodule
```

11.3.2. Simulación del módulo *core* del circuito

Para realizar la simulación del *core* del RCC, primero verificar que los archivos necesarios se encuentren en su respectivo directorio como ya se ha estado explicando con los circuitos anteriores.

Figura 67. Archivos para el RCC de 4 bits



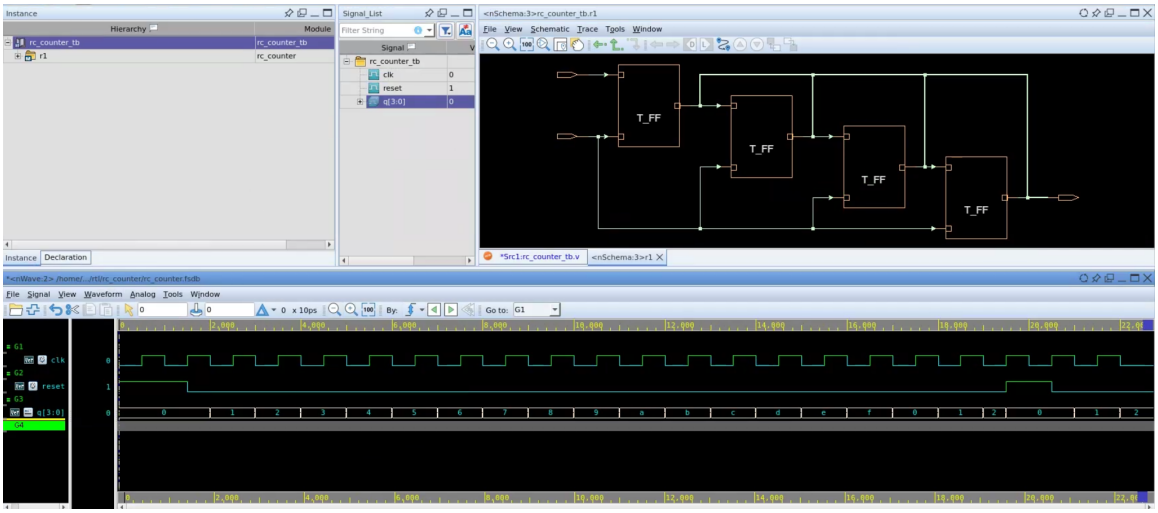
Nota. La imagen muestra los archivos para el RCC de 4 bits en su carpeta respectiva.

Luego ejecutar los siguientes comandos para simular el *core* del RCC:

Cuadro 45. Comandos para la compilación y simulación de la NOT con pads

```
vcs -full64 -debug_access+all -kdb rc_counter_tb.v rc_counter.v
./simv
verdi -ssf rc_counter.fsdb
```

Figura 68. Simulación del RCC de 4 bits



Nota. La imagen muestra la simulación del RCC de 4 bits en Verdi.

En la Figura 68 se observa el funcionamiento del contador. Dicho contador está conformado por cuatro *Toggle Flip-Flops*, los cuales se van activando y desactivando según la conmutación de la señal del reloj. El funcionamiento del contador es el esperado, ya que en cada flanco negativo de reloj la cuenta aumenta. Al llegar a 15 (f en hexadecimal), la cuenta se reinicia. Además, como se puede notar, el contador se reinicia al activarse la señal de *reset*.

11.3.3. Síntesis del módulo *core* del circuito

Luego se realiza la síntesis del *core* al verificar el funcionamiento del código.

Primeramente, no olvidar cambiar los parámetros del *script*:

Figura 69. Cambio de parámetros para la síntesis del RCC

```
# Author: Carlos Julio Valdes Oajaca
# Date: July 2025

# Parameters the user should modify
# PARAMETRO -> name of the circuit and instance (verilog file and instance must have the same name)
# USER_OPTION -> select between 1 and 2 only
# 1 -> synthesis without PADS
# 2 -> synthesis with PADS

set PARAMETRO "rc_counter"
set USER_OPTION 1

# Depending on user's selection do the synthesis without or with pads
if {$USER_OPTION == 1} { # Option without pads
        set PARAMETRO_2 "no"
        set rtl_files "${PARAMETRO}.v"
        set top_module "${PARAMETRO}"
} elseif {$USER_OPTION == 2} { # Option with pads
        set PARAMETRO_2 "with"
        set rtl_files [list "IO_${PARAMETRO}.v" "${PARAMETRO}.v"]
        set top_module "IO_${PARAMETRO}"
} else {
        puts "Error: please select 1 or 2"
        exit
}
```

Nota. La imagen muestra el cambio de parámetros para la síntesis del RCC.

Ahora ejecutar el archivo TCL:

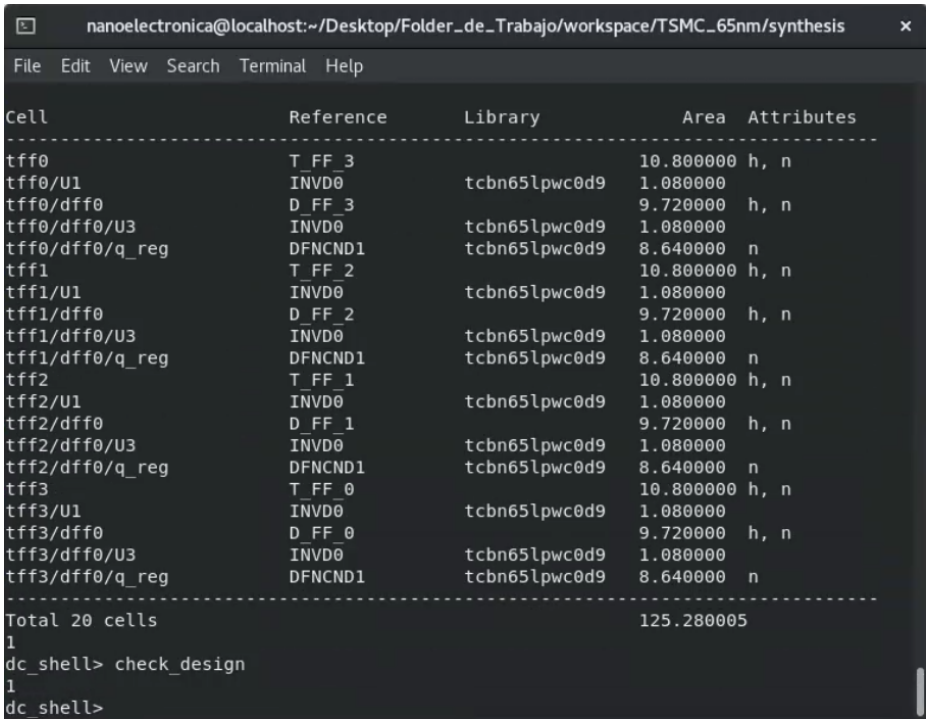
Figura 70. Ejecución del *script* desde la terminal para realizar la síntesis del *core* del RCC

```
nanoelectronica@localhost:~/Desktop/Folder_de_Trabajo/workspace/TSMC_65nm/synthesis
File  Edit  View  Search  Terminal  Help
if {[file exists command.log]} {
        file rename -force command.log logs/$PARAMETRO/command.log
}
if {[file exists filenames.log]} {
        file rename -force filenames.log logs/$PARAMETRO/filenames.log
}
if {[file exists default.svf]} {
        file rename -force default.svf logs/$PARAMETRO/default.svf
}
# Copy testbench to outputs
if {![file exists outputs/outputs_${PARAMETRO_2}_pads/$PARAMETRO/${PARAMETRO}_tb.v]
} {
file copy -force rtl/$PARAMETRO/${PARAMETRO}_tb.v outputs/outputs_${PARAMETRO_2}_pa
ds/$PARAMETRO/${PARAMETRO}_tb.v
}
# Delete unnecessary files
file delete -force {*}[glob -nocomplain *.mr]
file delete -force {*}[glob -nocomplain *-verilog.pvl]
file delete -force {*}[glob -nocomplain *-verilog.syn]
return
dc_shell> exit

Memory usage for this session 569 Mbytes.
Memory usage for this session including child processes 569 Mbytes.
CPU usage for this session 8 seconds ( 0.00 hours ).
Elapsed time for this session 1843 seconds ( 0.51 hours ).

Thank you...
[nanoelectronica@localhost synthesis]$ dc_shell -f scripts/script_dc.tcl
```

Nota. La imagen muestra la ejecución del *script* desde la terminal.

Posteriormente revisar las celdas y si existe algún problema con el diseño sintetizado:

Figura 71. Celdas de la síntesis del *core* del RCC



Nota. La imagen muestra las celdas de la síntesis del *core* del RCC.

Como se puede apreciar en la figura anterior no existen problemas con el diseño sintetizado del RCC.

A continuación, se presenta el *gate-level netlist* con el esquemático de Design Vision:

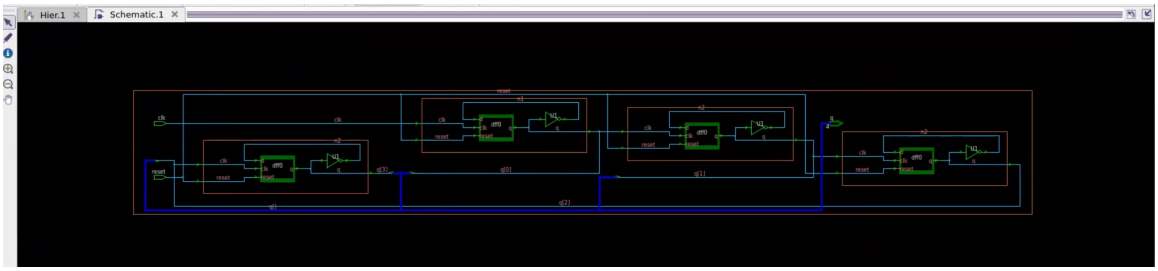
Cuadro 46. Archivo rc_counter_mapped.v en la carpeta `outputs_no_pads/`

```
////////////////////////////////////////////////////////////
// Created by: Synopsys DC Ultra(TM) in wire load mode
// Version   : W-2024.09-SP4
// Date      : Fri Nov  7 17:58:55 2025
////////////////////////////////////////////////////////////


module D_FF_3 ( q, d, clk, reset );
  input d, clk, reset;
  output q;
  wire   n1;

  DFNCND1 q_reg ( .D(d), .CPN(clk), .CDN(n1), .Q(q) );
  INVD0 U3 ( .I(reset), .ZN(n1) );
endmodule
```

```
module T_FF_3 ( q, clk, reset );
  input clk, reset;
  output q;
  wire   n1;

  D_FF_3 dff0 ( .q(q), .d(n1), .clk(clk), .reset(reset) );
  INVD0 U1 ( .I(q), .ZN(n1) );
endmodule


module D_FF_0 ( q, d, clk, reset );
  input d, clk, reset;
  output q;
  wire   n2;

  DFNCND1 q_reg ( .D(d), .CPN(clk), .CDN(n2), .Q(q) );
  INVD0 U3 ( .I(reset), .ZN(n2) );
endmodule


module T_FF_0 ( q, clk, reset );
  input clk, reset;
  output q;
  wire   n2;

  D_FF_0 dff0 ( .q(q), .d(n2), .clk(clk), .reset(reset) );
  INVD0 U1 ( .I(q), .ZN(n2) );
endmodule


module D_FF_1 ( q, d, clk, reset );
  input d, clk, reset;
  output q;
  wire   n2;

  DFNCND1 q_reg ( .D(d), .CPN(clk), .CDN(n2), .Q(q) );
  INVD0 U3 ( .I(reset), .ZN(n2) );
endmodule


module T_FF_1 ( q, clk, reset );
  input clk, reset;
  output q;
  wire   n2;

  D_FF_1 dff0 ( .q(q), .d(n2), .clk(clk), .reset(reset) );
  INVD0 U1 ( .I(q), .ZN(n2) );
endmodule
```

```
module D_FF_2 ( q, d, clk, reset );
  input d, clk, reset;
  output q;
  wire   n2;

  DFNCND1 q_reg ( .D(d), .CPN(clk), .CDN(n2), .Q(q) );
  INVD0 U3 ( .I(reset), .ZN(n2) );
endmodule


module T_FF_2 ( q, clk, reset );
  input clk, reset;
  output q;
  wire   n2;

  D_FF_2 dff0 ( .q(q), .d(n2), .clk(clk), .reset(reset) );
  INVD0 U1 ( .I(q), .ZN(n2) );
endmodule


module rc_counter ( q, clk, reset );
  output [3:0] q;
  input clk, reset;


  T_FF_3 tff0 ( .q(q[0]), .clk(clk), .reset(reset) );
  T_FF_2 tff1 ( .q(q[1]), .clk(q[0]), .reset(reset) );
  T_FF_1 tff2 ( .q(q[2]), .clk(q[1]), .reset(reset) );
  T_FF_0 tff3 ( .q(q[3]), .clk(q[2]), .reset(reset) );
endmodule
```

Figura 72. Esquemático del *core* del RCC



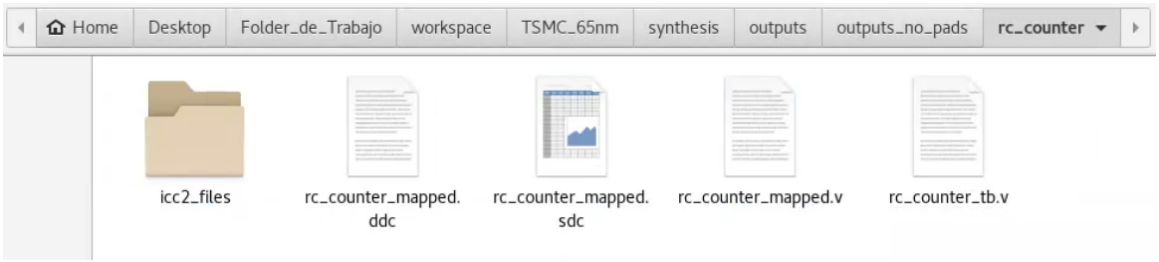
Nota. La imagen muestra el esquemático del *core* del RCC en Design Vision.

De igual manera que en Verdi se pueden observar los cuatro *Toggle Flip-Flops*. Ahora se debe verificar si el funcionamiento permanece siendo el esperado para el circuito.

11.3.4. Simulación del *core* del circuito sintetizado

Una vez realizada la síntesis se puede realizar la simulación del *core* del RCC de 4 bits. Primero confirmar que los archivos de salida estén presentes:

Figura 73. Archivos de salida al sintetizar el *core* del RCC de 4 bits



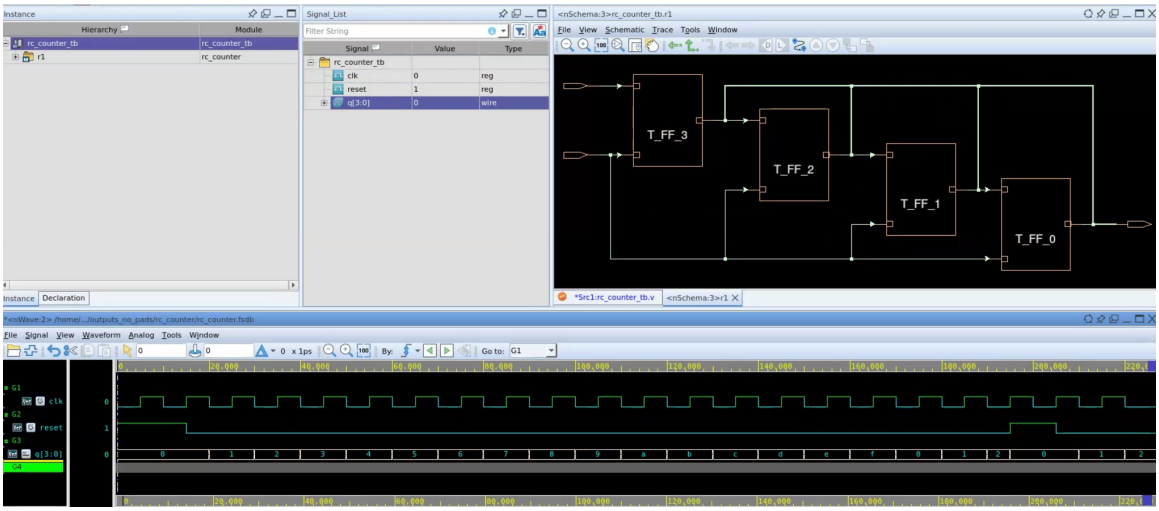
Nota. La imagen muestra archivos de salida al sintetizar el *core* del RCC de 4 bits.

Luego simular:

Cuadro 47. Comandos para la compilación y simulación de la NOT con pads

```
vcs -full64 -debug_access+all -kdb rc_counter_tb.v
   rc_counter_mapped.v -v ../../tcbn65lp.v
./simv
verdi -ssf rc_counter.fsdb
```

Figura 74. Simulación del *core* del RCC de 4 bits



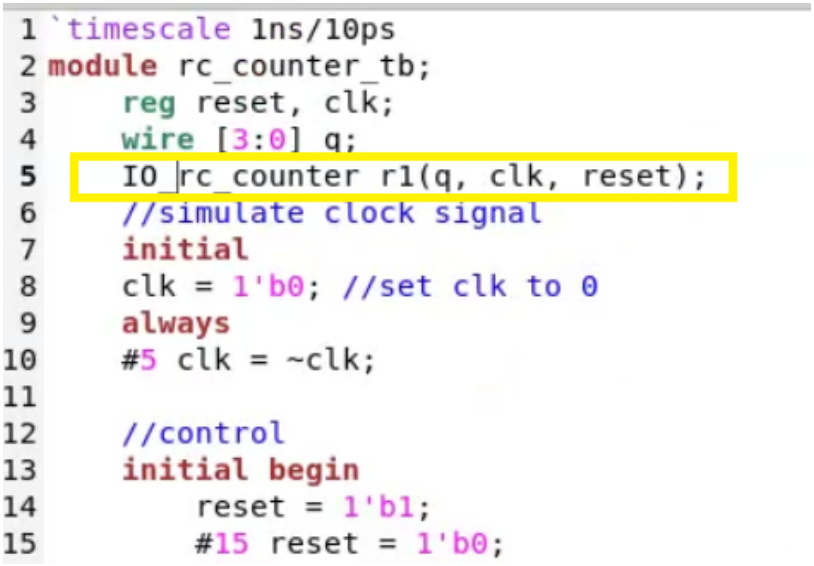
Nota. La imagen muestra la simulación del *core* del RCC de 4 bits en Verdi.

Al observar la figura anterior se puede confirmar que la síntesis se realizó de forma adecuada y correcta. El contador funciona de la misma manera que antes de ser sintetizado.

11.3.5. Simulación del módulo del circuito con pads

Como la síntesis del *core* del RCC de 4 bits tuvo éxito, ahora se comienza con el proceso de síntesis del mismo cicuito pero con pads incorporados. No olvidar cambiar la instancia en el archivo *testbench*.

Figura 75. Cambio de instancia del *testbench* del RCC de 4 bits



```
1  `timescale 1ns/10ps
2  module rc_counter_tb;
3      reg reset, clk;
4      wire [3:0] q;
5      IO_rc_counter r1(q, clk, reset);
6      //simulate clock signal
7      initial
8      clk = 1'b0; //set clk to 0
9      always
10     #5 clk = ~clk;
11
12     //control
13     initial begin
14         reset = 1'b1;
15         #15 reset = 1'b0;
```

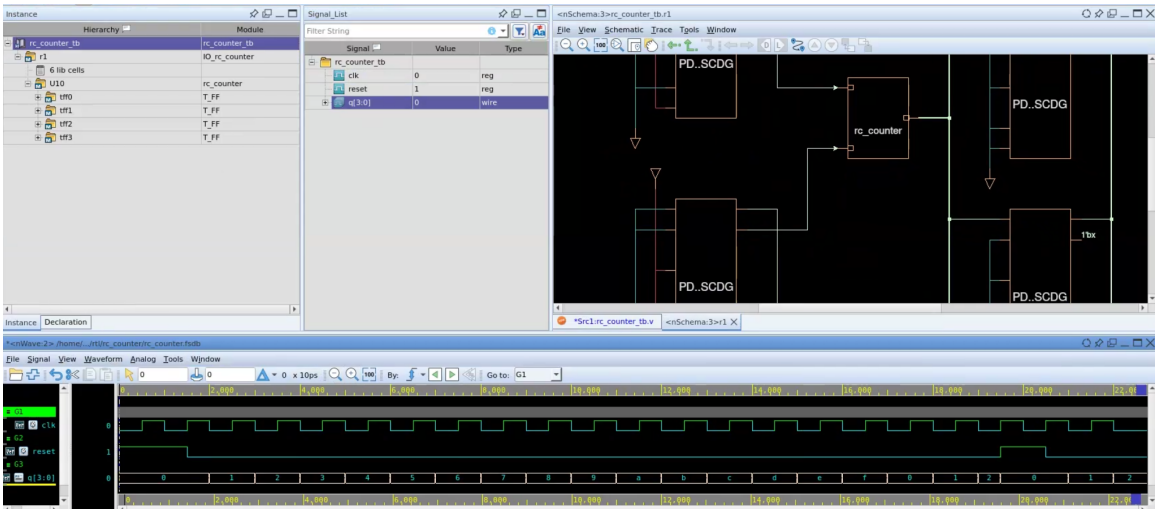
Nota. La imagen muestra el cambio de instancia del *testbench* del RCC de 4 bits en Verdi.

De primero se realiza la simulación del circuito con pads antes de sintetizar:

Cuadro 48. Comandos para la compilación y simulación de la NOT con pads

```
vcs -full64 -debug_access+all -kdb rc_counter_tb.v
   IO_rc_counter.v rc_counter.v -v ../../outputs/tpdn65lpnv2od3
   .v
./simv
verdi -ssf rc_counter.fsdb
```

Figura 76. Simulación del RCC de 4 bits con pads



Nota. La imagen muestra la simulación del RCC de 4 bits en Verdi.

Se puede observar en la figura anterior que el funcionamiento del RCC de 4 bits con pads coincide con el funcionamiento sin pads, por lo que la simulación demuestra que el código con pads está bien.

11.3.6. Síntesis del módulo del circuito con pads

Como siguiente punto, se realiza la síntesis del RCC de 4 bits con pads. Para ello, cambiar el segundo parámetro del *script*:

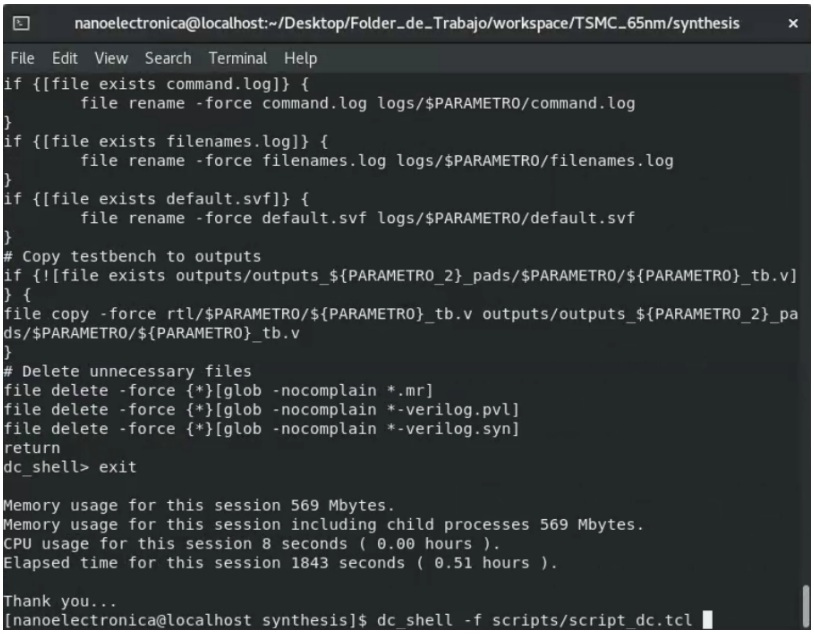
Figura 77. Cambio del segundo parámetro para realizar la síntesis del RCC de 4 bits con pads

```
1 # Author: Carlos Julio Valdes Oajaca
2 # Date: July 2025
3
4 # Parameters the user should modify
5 # PARAMETRO -> name of the circuit and instance (verilog file and instance must have the same name)
6 # USER_OPTION -> select between 1 and 2 only
7 # 1 -> synthesis without PADS
8 # 2 -> synthesis with PADS
9
10 set PARAMETRO "rc_counter"
11 set USER_OPTION 2
```

Nota. La imagen muestra el cambio del segundo parámtro para sintetizar el RCC de 4 bits con pads.

Ejecutar el *script*:

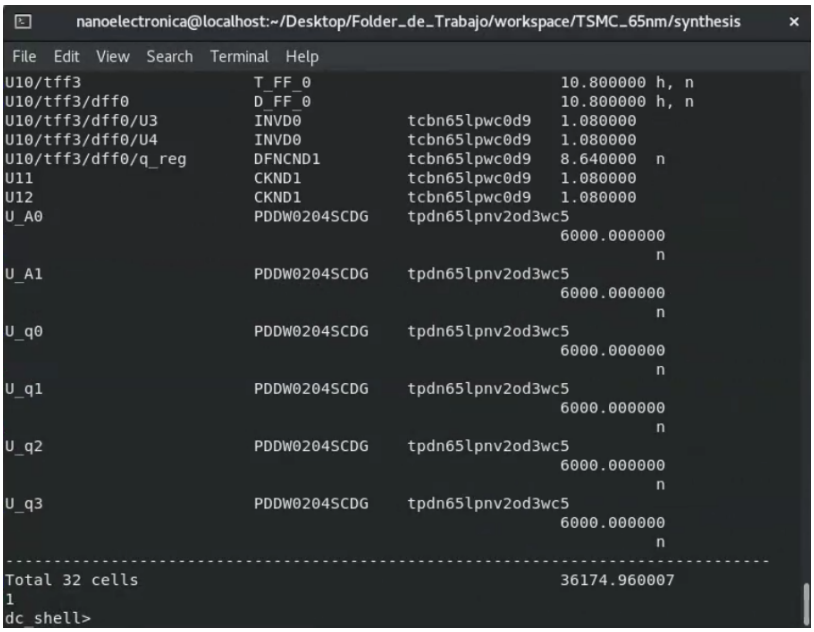
Figura 78. Ejecución del *script* desde la terminal para realizar la síntesis del RCC de 4 bits con pads



Nota. La imagen muestra la ejecución del *script* desde la terminal.

Verificar las celdas que fueron mapeadas:

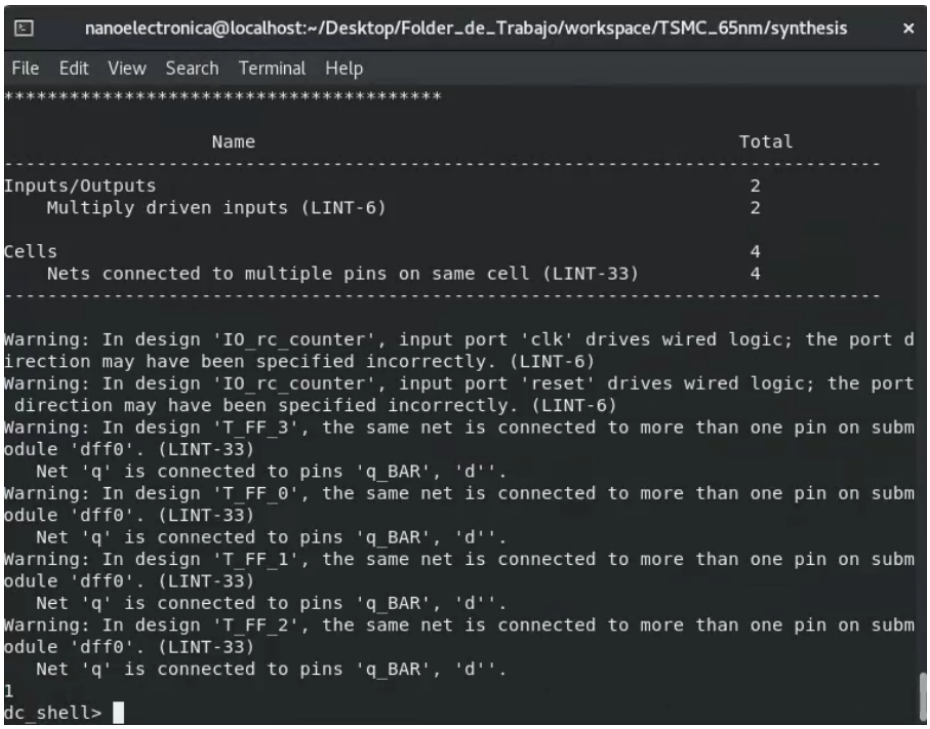
Figura 79. Celdas mapeadas del RCC de 4 bits con pads



Nota. La imagen muestra las celdas mapeadas del RCC de 4 bits con pads desde la terminal.

También verificar posibles problemas con la síntesis:

Figura 80. Probblemas asociados con la síntesis del RCC de 4 bits con pads



Nota. La imagen muestra los problemas asociados con la síntesis del RCC de 4 bits con pads desde la terminal.

Al observar la figura anterior se puede notar que aparte de las advertencias de *wired logic* aparecen otras cuatro advertencias donde se advierte de nodos conectados a más de un pin. Esta advertencia que muestra la herramienta es para que el diseñador revise si es correcto que ese nodo (*net*) especificado tenga más de una conexión. En este caso, el circuito está diseñado de esta manera, por lo que más adelante al simular se comprobará que no existen problemas de funcionalidad por estas advertencias.

Cuando se observen errores en lugar de advertencias ahí debe existir una mayor preocupación. Con las advertencias se trata de observarlas e interpretar si aplican o no al diseño que se sintetiza.

A continuación se muestra el archivo del RCC sintetizado con pads y su esquemático:

Cuadro 49. Archivo rc_counter_mapped.v en el directorio `outputs_with_pads/`

```
////////////////////////////////////////////////////////////
// Created by: Synopsys DC Ultra(TM) in wire load mode
// Version   : W-2024.09-SP4
// Date      : Fri Nov  7 18:42:11 2025
////////////////////////////////////////////////////////////
```

```
module D_FF_3 ( d, clk, reset, q_BAR );
  input d, clk, reset;
  output q_BAR;
  wire   n1;

  DFNCND1 q_reg ( .D(d), .CPN(clk), .CDN(n1), .QN(q_BAR) );
  INVD0 U3 ( .I(reset), .ZN(n1) );
endmodule


module T_FF_3 ( clk, reset, q_BAR );
  input clk, reset;
  output q_BAR;
  wire   q;
  assign q_BAR = q;

  D_FF_3 dff0 ( .d(q), .clk(clk), .reset(reset), .q_BAR(q) );
endmodule


module D_FF_0 ( d, reset, q_BAR, clk_BAR );
  input d, reset, clk_BAR;
  output q_BAR;
  wire   clk, n2, n4;
  assign clk = clk_BAR;

  DFNCND1 q_reg ( .D(d), .CPN(n2), .CDN(n4), .QN(q_BAR) );
  INVD0 U3 ( .I(reset), .ZN(n4) );
  INVD0 U4 ( .I(clk), .ZN(n2) );
endmodule


module T_FF_0 ( reset, q_BAR, clk_BAR );
  input reset, clk_BAR;
  output q_BAR;
  wire   q, clk;
  assign q_BAR = q;
  assign clk = clk_BAR;

  D_FF_0 dff0 ( .d(q), .reset(reset), .q_BAR(q), .clk_BAR(clk)
     );
endmodule


module D_FF_1 ( d, reset, q_BAR, clk_BAR );
  input d, reset, clk_BAR;
  output q_BAR;
  wire   clk, n2, n4;
```

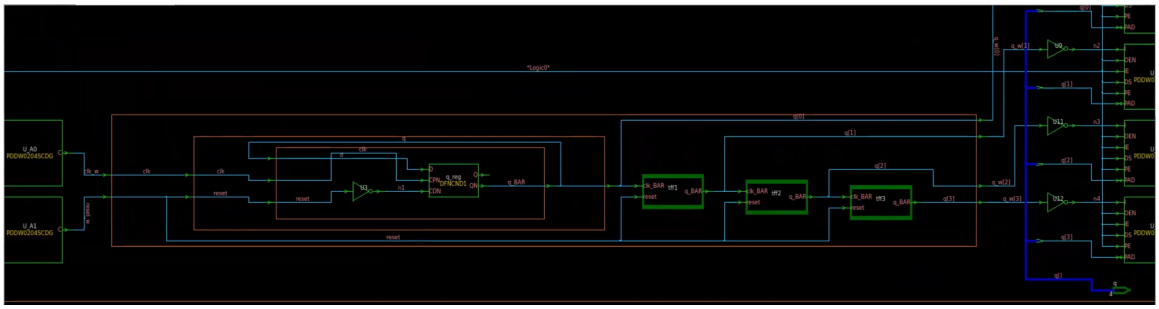
```
  assign clk = clk_BAR;

  DFNCND1 q_reg ( .D(d), .CPN(n2), .CDN(n4), .QN(q_BAR) );
  INVD0 U3 ( .I(reset), .ZN(n4) );
  INVD0 U4 ( .I(clk), .ZN(n2) );
endmodule


module T_FF_1 ( reset, q_BAR, clk_BAR );
  input reset, clk_BAR;
  output q_BAR;
  wire   q, clk;
  assign q_BAR = q;
  assign clk = clk_BAR;

  D_FF_1 dff0 ( .d(q), .reset(reset), .q_BAR(q), .clk_BAR(clk)
     );
endmodule


module D_FF_2 ( d, reset, q_BAR, clk_BAR );
  input d, reset, clk_BAR;
  output q_BAR;
  wire   clk, n2, n4;
  assign clk = clk_BAR;

  DFNCND1 q_reg ( .D(d), .CPN(n2), .CDN(n4), .QN(q_BAR) );
  INVD0 U3 ( .I(reset), .ZN(n4) );
  INVD0 U4 ( .I(clk), .ZN(n2) );
endmodule


module T_FF_2 ( reset, q_BAR, clk_BAR );
  input reset, clk_BAR;
  output q_BAR;
  wire   q, clk;
  assign q_BAR = q;
  assign clk = clk_BAR;

  D_FF_2 dff0 ( .d(q), .reset(reset), .q_BAR(q), .clk_BAR(clk)
     );
endmodule


module rc_counter ( clk, reset, \q[3]_BAR , \q[2]_BAR , \q[1]
   _BAR , \q[0]_BAR
 );
  input clk, reset;
```

```
  output \q[3]_BAR , \q[2]_BAR , \q[1]_BAR , \q[0]_BAR ;

  wire   [3:0] q;
  assign \q[3]_BAR  = q[3];
  assign \q[2]_BAR  = q[2];
  assign \q[1]_BAR  = q[1];
  assign \q[0]_BAR  = q[0];

  T_FF_3 tff0 ( .clk(clk), .reset(reset), .q_BAR(q[0]) );
  T_FF_2 tff1 ( .reset(reset), .q_BAR(q[1]), .clk_BAR(q[0]) );
  T_FF_1 tff2 ( .reset(reset), .q_BAR(q[2]), .clk_BAR(q[1]) );
  T_FF_0 tff3 ( .reset(reset), .q_BAR(q[3]), .clk_BAR(q[2]) );
endmodule


module IO_rc_counter ( q, clk, reset );
  output [3:0] q;
  input clk, reset;
  wire   \*Logic1* , \*Logic0* , clk_w, reset_w, n1, n2, n3, n4
     ;
  wire   [3:0] q_w;

  rc_counter U10 ( .clk(clk_w), .reset(reset_w), .\q[3]_BAR (
     q_w[3]),
        .\q[2]_BAR (q_w[2]), .\q[1]_BAR (q_w[1]), .\q[0]_BAR (
           q_w[0]) );
  PDDW0204SCDG U_A0 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(
        clk), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(clk_w) );
  PDDW0204SCDG U_A1 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(
        reset), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(reset_w) )
           ;
  PDDW0204SCDG U_q0 ( .I(n1), .OEN(\*Logic0* ), .IE(\*Logic0* )
     , .PAD(q[0]),
        .DS(\*Logic0* ), .PE(\*Logic0* ) );
  PDDW0204SCDG U_q1 ( .I(n2), .OEN(\*Logic0* ), .IE(\*Logic0* )
     , .PAD(q[1]),
        .DS(\*Logic0* ), .PE(\*Logic0* ) );
  PDDW0204SCDG U_q2 ( .I(n3), .OEN(\*Logic0* ), .IE(\*Logic0* )
     , .PAD(q[2]),
        .DS(\*Logic0* ), .PE(\*Logic0* ) );
  PDDW0204SCDG U_q3 ( .I(n4), .OEN(\*Logic0* ), .IE(\*Logic0* )
     , .PAD(q[3]),
        .DS(\*Logic0* ), .PE(\*Logic0* ) );
  TIEH U6 ( .Z(\*Logic1* ) );
  TIEL U7 ( .ZN(\*Logic0* ) );
  CKND1 U8 ( .I(q_w[0]), .ZN(n1) );
```

```
  CKND1 U9 ( .I(q_w[1]), .ZN(n2) );
  CKND1 U11 ( .I(q_w[2]), .ZN(n3) );
  CKND1 U12 ( .I(q_w[3]), .ZN(n4) );
endmodule
```

Puede apreciarse la aparición de las celdas TIEL y TIEH para la conexión de las señales con VDD y VSS.

Figura 81. Esquemático del RCC de 4 bits sintetizado con pads



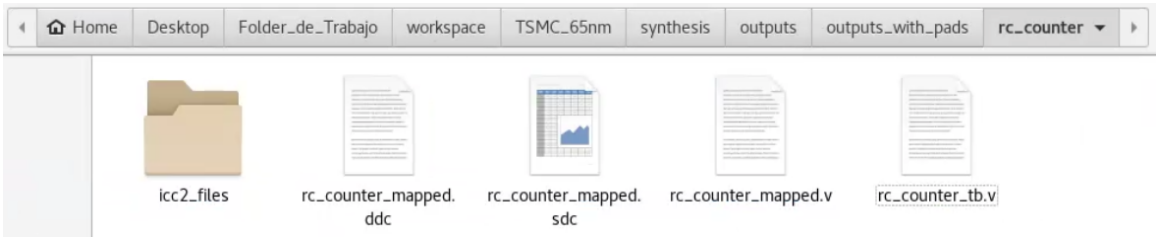
Nota. La imagen muestra el esquemático del RCC de 4 bits sintetizado con pads en Design Vision.

En la Figura 81 se puede apreciar las celdas de los pads en los extremos del *core* del RCC de 4 bits. También en el primer *Toggle Flip-Flop* del diseño se puede apreciar el nombre de la celda (`DFNCND1`), a la cual Design Compiler hizo el mapeo para cada *Flip-Flop*.

11.3.7. Simulación del módulo del circuito sintetizado con pads

Antes de simular el RCC de 4 bits sintetizado con pads se sugiere revisar que los archivos de salida estén presentes en la ubicación correcta:

Figura 82. Archivos producto de la síntesis del RCC de 4 bits con pads



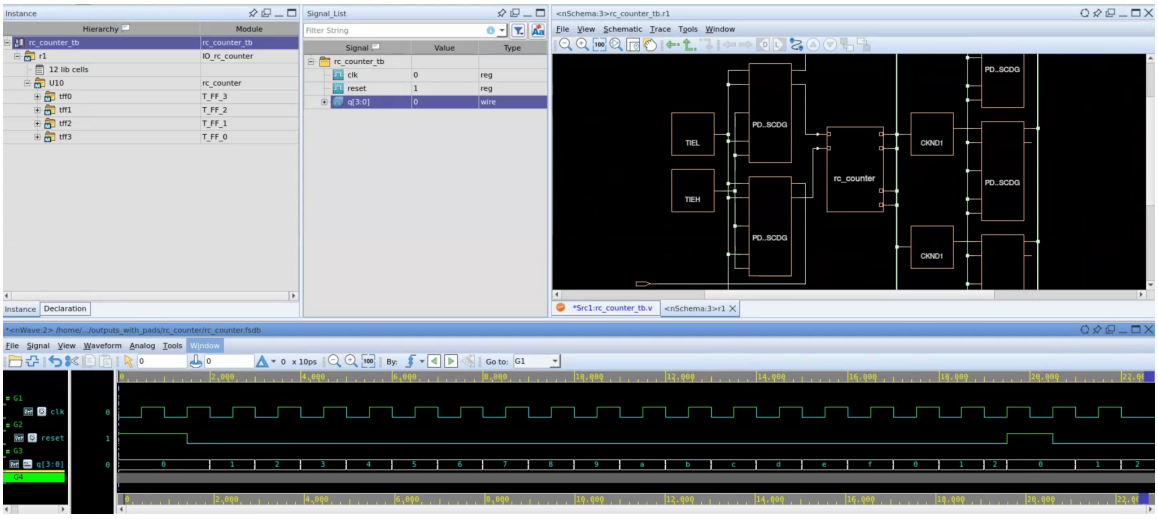
Nota. La imagen muestra los archivos producto de la síntesis del RCC de 4 bits con pads.

Luego simular el RCC de 4 bits sintetizado con pads:

Cuadro 50. Comandos para la compilación y simulación de la NOT con pads

```
vcs -full64 -debug_access+all -kdb rc_counter_tb.v
   rc_counter_mapped.v -v ../../tcbn65lp.v -v ../../
   tpdn65lpnv2od3.v
./simv
verdi -ssf rc_counter.fsdb
```

Figura 83. Simulación de la síntesis del RCC de 4 bits con pads



Nota. La imagen muestra la simulación de la síntesis del RCC de 4 bits con pads.

Como se mencionó anteriormente, los errores por tener varias conexiones al mismo nodo no afectaron al funcionamiento de la síntesis. Al observar la Figura 83 se concluye que la síntesis del RCC de 4 bits con pads fue exitosa.

11.4. Síntesis lógica de un *Ring Oscillator*

El nanochip "El Gran Jaguar" contiene en su diseño un *ring oscillator*. En aras a reducir la cantidad de errores que puedan ocurrir al realizar la síntesis del nanochip, se realizó la síntesis de un *ring oscillator* previamente. De esta manera, se consigue tener un diseño correcto de este módulo, el cual no funcionaba correctamente en trabajos previos.

11.4.1. Módulo *core* del circuito, *testbench* y módulo del circuito con pads

En este caso, la nomenclatura para los archivos y directorios propios del circuito es: `ring_osc`

El *ring oscillator* que se sintetizó contiene trece compuertas NOT como se puede notar en el Cuadro 51. En el Cuadro 52 se encuentra el *testbench* y en el Cuadro 53 se observa el diseño con pads.

Cuadro 51. Archivo ring_osc.v

```
module ring_osc(rst, out);
  input rst; // Reset signal
  output out; // Output signal

    wire W_1;
    wire W_2;
    wire W_3;
    wire W_4;
    wire W_5;
    wire W_6;
    wire W_7;
    wire W_8;
    wire W_9;
    wire W_10;
    wire W_11;
    wire W_12;
    wire W_13;

    assign  W_1 = rst | out;
    assign  W_2 = ~W_1;
    assign  W_3 = ~W_2;
    assign  W_4 = ~W_3;
    assign  W_5 = ~W_4;
    assign  W_6 = ~W_5;
    assign  W_7 = ~W_6;
    assign  W_8 = ~W_7;
    assign  W_9 = ~W_8;
    assign  W_10 = ~W_9;
    assign  W_11 = ~W_10;
    assign  W_12 = ~W_11;
    assign  W_13 = ~W_12;
    assign out = ~W_13;

endmodule
```

Cuadro 52. Archivo ring_osc_tb.v

```
`timescale 1ns/10ps

module ring_osc_tb;

  reg rst;     // Reset signal
  wire out;    // Output signal
```

```
  // Instantiate the ring oscillator
  ring_osc uut (
    .rst(rst),
    .out(out)
  );

  initial begin
    $dumpfile("ring_osc.vcd");
    $dumpvars(0, ring_osc_tb);
    $dumpvars(0, uut.W_1, uut.W_2, uut.W_3, uut.W_4, uut.W_5,
       uut.W_6);
    $fsdbDumpfile("ring_osc.fsdb");
    $fsdbDumpvars(0, ring_osc_tb);
  end

  initial begin
    // Initialize reset
    rst = 0;
    #20;  // Hold reset for 20ns
    rst = 1;  // Release reset - oscillation should start
       automatically
    #20;
    rst = 0;
    #50;  // Hold reset low for another 20ns
    rst = 1;
    #10;
    rst=0;
    #200;  // Run simulation for 200ns to see multiple
       oscillation cycles
    $finish;
  end

  // Monitor the signals
  initial begin
    $monitor("Time: %0t ns, Reset: %b, Output: %b", $time, rst,
        out);
  end

endmodule
```

Cuadro 53. Archivo IO_ring_osc.v

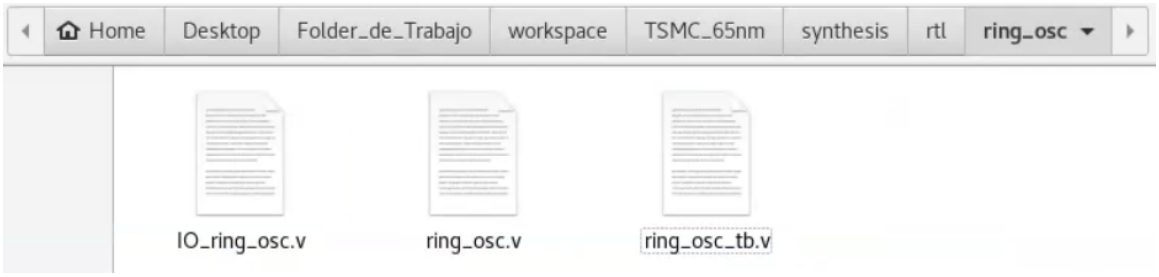
```
module IO_ring_osc(rst, out);
input rst; // Reset signal
output out; // Output signal

wire rst_w, out_w;
```

```
ring_osc U10(rst_w, out_w);

//Inputs:
PDDW0204SCDG U_R0(.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(rst), .
   C(rst_w), .PE(1'b0), .IE(1'b1));

//Outputs:
PDDW0204SCDG U_q0 (.I(out_w), .DS(1'b0), .OEN(1'b0), .PAD(out),
    .C(1'bx), .PE(1'b0), .IE(1'b1));

endmodule
```

11.4.2. Simulación del módulo *core* del circuito

Para comenzar con el proceso de síntesis, cerciorarse que los archivos de insumo estén en el directorio adecuado.

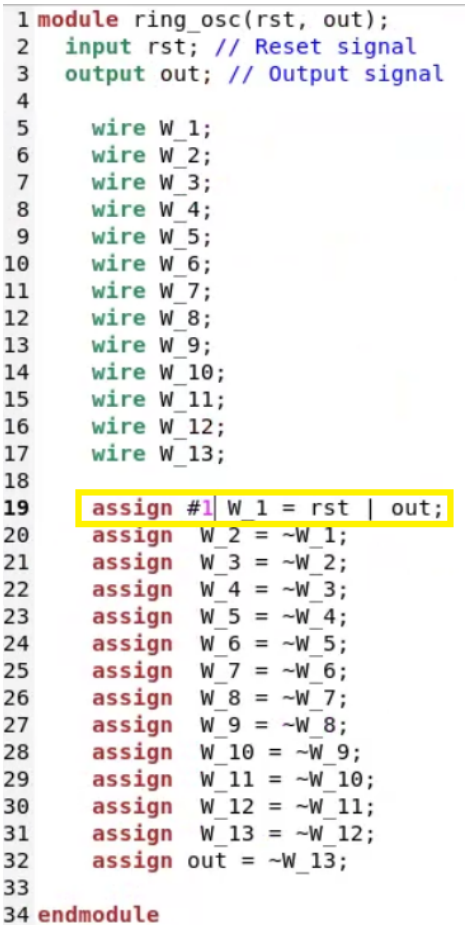
Figura 84. Archivos para llevar a cabo la síntesis del *ring oscillator*



Nota. La imagen muestra los archivos necesarios para la síntesis del *ring oscillator* en su directorio.

Debe destacarse que para el caso concreto de simulación del *ring oscillator* es de suma importancia agregar *delays*. Los simuladores son herramientas ideales que no toman en cuenta el *delay* propio de cualquier compuerta. Debido a que el funcionamiento de este tipo de circuito se basa en el *delay* natural de las compuertas que lo componen, de no agregarse, el simulador no mostrará nada.

Figura 85. Adición del *delay* en el archivo `ring_osc.v`



```
1 module ring_osc(rst, out);
2   input rst; // Reset signal
3   output out; // Output signal
4
5     wire W_1;
6     wire W_2;
7     wire W_3;
8     wire W_4;
9     wire W_5;
10    wire W_6;
11    wire W_7;
12    wire W_8;
13    wire W_9;
14    wire W_10;
15    wire W_11;
16    wire W_12;
17    wire W_13;
18
19    assign #1 W_1 = rst | out;
20    assign  W_2 = ~W_1;
21    assign  W_3 = ~W_2;
22    assign  W_4 = ~W_3;
23    assign  W_5 = ~W_4;
24    assign  W_6 = ~W_5;
25    assign  W_7 = ~W_6;
26    assign  W_8 = ~W_7;
27    assign  W_9 = ~W_8;
28    assign  W_10 = ~W_9;
29    assign  W_11 = ~W_10;
30    assign  W_12 = ~W_11;
31    assign  W_13 = ~W_12;
32    assign out = ~W_13;
33
34 endmodule
```

Nota. La imagen muestra la adición del *delay* para simular el circuito posteriormente.

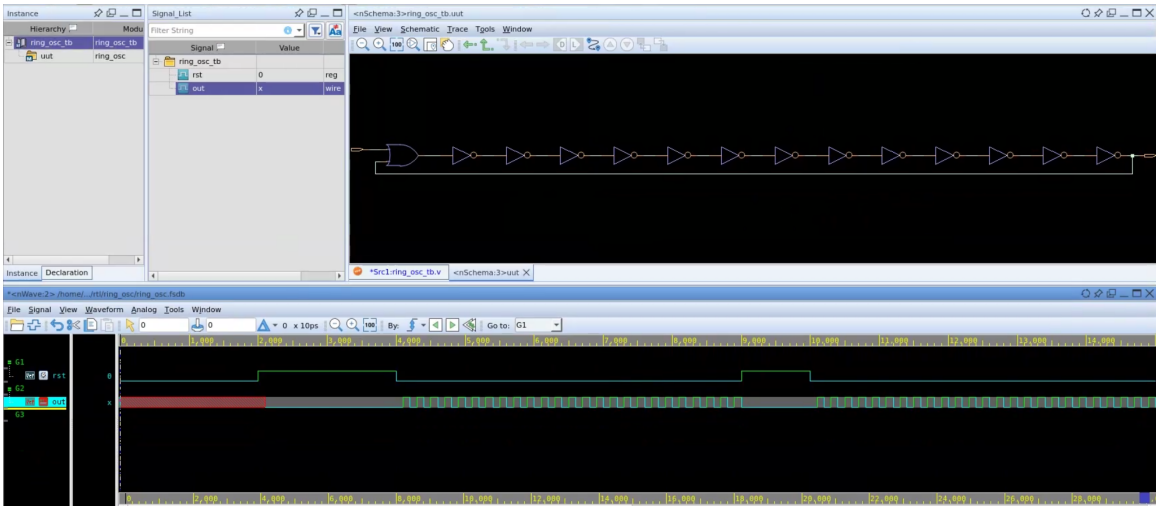
En este caso es suficientes con agregar *delay* en la primera etapa donde se encuentra la compuerta OR. No obstante, en otros casos puede ser necesario agregar *delay* en las etapas de inversión.

Luego ya se ejecutan los comandos para generar el archivo de simulación para Verdi:

Cuadro 54. Comandos para la compilación y simulación de la NOT con pads

```
vcs -full64 -debug_access+all -kdb ring_osc_tb.v ring_osc.v
./simv
verdi -ssf ring_osc.fsdb
```

Figura 86. Simulación de *core* del *ring oscillator*



Nota. La imagen muestra la simulación del *core* del *ring oscillator* en Verdi.

Como se puede apreciar en la Figura 86, la salida del circuito posee un estado indefinido (banda de color rojo en la vista de *Waveform*), hasta que el *reset* es presionado. Una vez el *reset* deja de estar activado, las oscilaciones en la salida comienzan. Mientras el *reset* se encuentre activo, la salida muestra el valor de 0.

Por otro lado, también se puede apreciar el diagrama del circuito, donde al inicio se encuentra una compuerta OR para indicar el momento en el que el circuito debe empezar oscilar. También se puede observar a las trece compuertas NOT en cascada.

11.4.3. Síntesis del módulo *core* del circuito

Antes de ejecutar el *script*, es importante modificar los parámetros de manera correcta para el caso de este circuito.

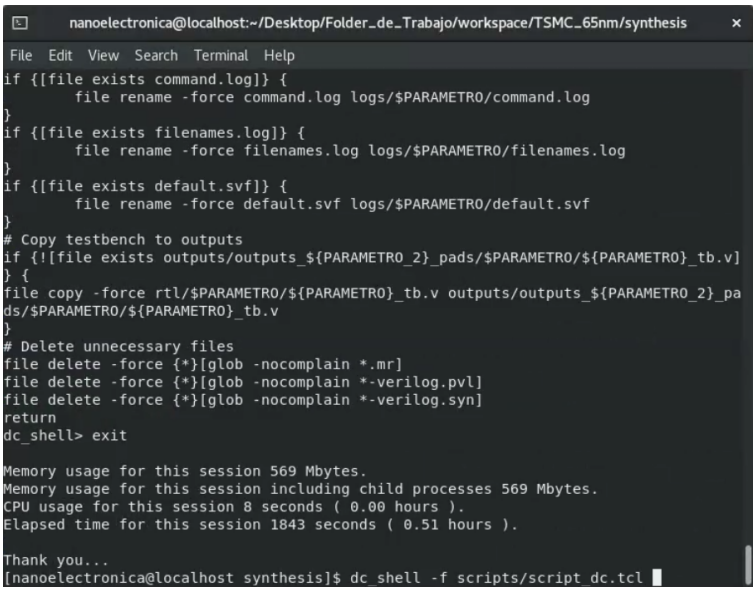
Figura 87. Parámetros para la síntesis del *core* del *ring oscillator*

```
1 # Author: Carlos Julio Valdes Oajaca
2 # Date: July 2025
3
4 # Parameters the user should modify
5 # PARAMETRO -> name of the circuit and instance (verilog file and instance must have the same name)
6 # USER_OPTION -> select between 1 and 2 only
7 # 1 -> synthesis without PADS
8 # 2 -> synthesis with PADS
9
10 set PARAMETRO "ring_osc"
11 set USER_OPTION 1
```

Nota. La imagen muestra los parámetros para la síntesis del *core* del *ring oscillator*.

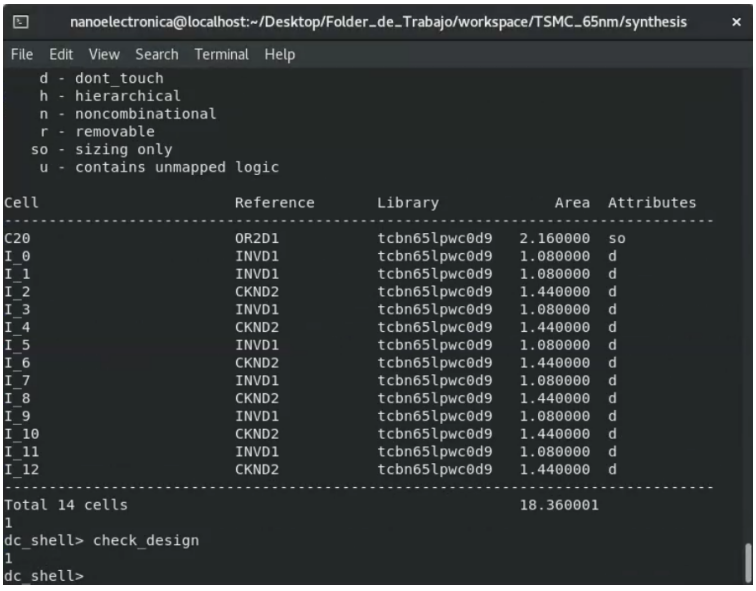
Luego, ejecutar el *script* como se ha realizado con circuitos anteriores, solicitar las celdas mapeadas y verificar posibles problemas en el diseño sintetizado:

Figura 88. Ejecución del *script* desde la terminal para realizar la síntesis del *core* del *ring oscillator*



Nota. La imagen muestra la ejecución del *script* desde la terminal.

Figura 89. Celdas mapeadas y reporte de problemas para la síntesis del *core* del *ring oscillator*



Nota. La imagen muestra las celdas mapeadas y posibles problemas para la síntesis del *core* del *ring oscillator*.

La Figura 89 muestra las celdas a las que fue mapeado el diseño del circuito, aunado a ello puede notarse que el diseño sintetizado no posee ningún problema.

Ahora se muestra el archivo sintetizado:

Cuadro 55. Archivo ring_osc_mapped.v en el directorio `outputs_no_pads/`

```
////////////////////////////////////////////////////////////////
// Created by: Synopsys DC Ultra(TM) in wire load mode
// Version   : W-2024.09-SP4
// Date      : Sat Nov  8 11:25:32 2025
////////////////////////////////////////////////////////////////


module ring_osc ( rst, out );
  input rst;
  output out;
  wire   W_1, W_2, W_3, W_4, W_5, W_6, W_7, W_8, W_9, W_10,
     W_11, W_12, W_13;

  CKND2 I_12 ( .I(W_13), .ZN(out) );
  CKND2 I_10 ( .I(W_11), .ZN(W_12) );
  CKND2 I_8 ( .I(W_9), .ZN(W_10) );
  CKND2 I_6 ( .I(W_7), .ZN(W_8) );
  CKND2 I_4 ( .I(W_5), .ZN(W_6) );
  CKND2 I_2 ( .I(W_3), .ZN(W_4) );
  OR2D1 C20 ( .A1(rst), .A2(out), .Z(W_1) );
  INVD1 I_0 ( .I(W_1), .ZN(W_2) );
  INVD1 I_1 ( .I(W_2), .ZN(W_3) );
  INVD1 I_3 ( .I(W_4), .ZN(W_5) );
  INVD1 I_5 ( .I(W_6), .ZN(W_7) );
  INVD1 I_7 ( .I(W_8), .ZN(W_9) );
  INVD1 I_9 ( .I(W_10), .ZN(W_11) );
  INVD1 I_11 ( .I(W_12), .ZN(W_13) );
endmodule
```

Este es el esquemático en Design Vision ejecutando el comando `start_gui`:

Figura 90. Esquemático de la síntesis del *core* del *ring oscillator*

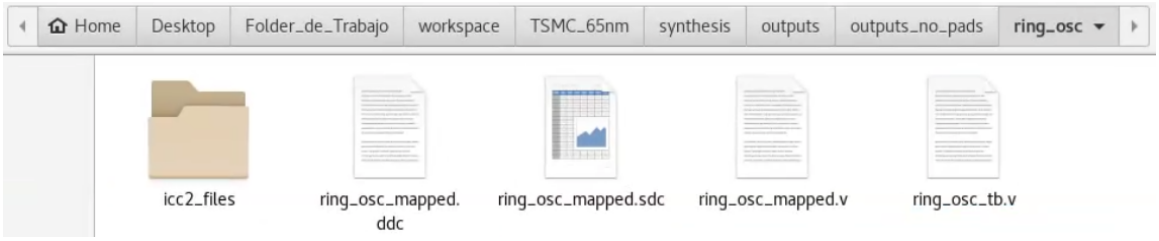
Nota. La imagen muestra el esquemático de la síntesis del *core* del *ring oscillator* en Design Vision.

El esquemático muestra las celdas del diseño sintetizado junto con el nombre propio de cada una según la librería de TSMC. Debido al espacio se puede ver la mayoría, mas no todas las celdas en la figura.

11.4.4. Simulación del *core* del circuito sintetizado

Luego de haber realizado la síntesis del *core*, se procede a simular el diseño. Primero verificar que los archivos de salida se hayan generado correctamente:

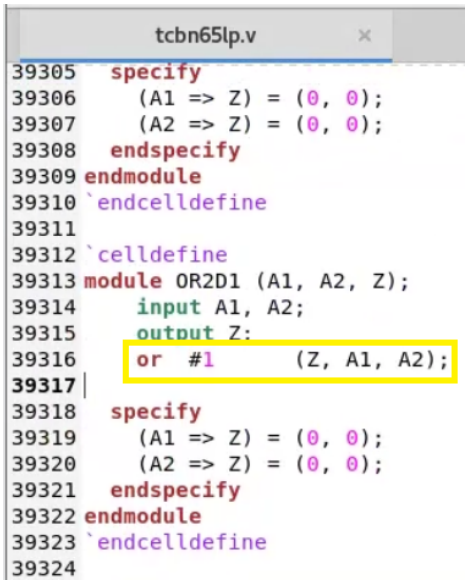
Figura 91. Archivos de salida como producto de la síntesis del *core* del *ring oscillator*



Nota. La imagen muestra los archivos de salida como producto de la síntesis del *core* del *ring oscillator*.

Es de importancia mencionar que para la simulación del *ring oscillator* sintetizado es necesario agregar el *delay* correspondiente desde la librería de las celdas, en este caso el archivo: `tcbn65lp.v` ubicado en el directorio `outputs` (ver Capítulo 8). Debe buscarse el nombre de la celda (OR2D1) que coincida con la del diseño sintetizado en el archivo.

Figura 92. Adición de *delay* a la celda OR del diseño sintetizado del *core* del *ring oscillator*



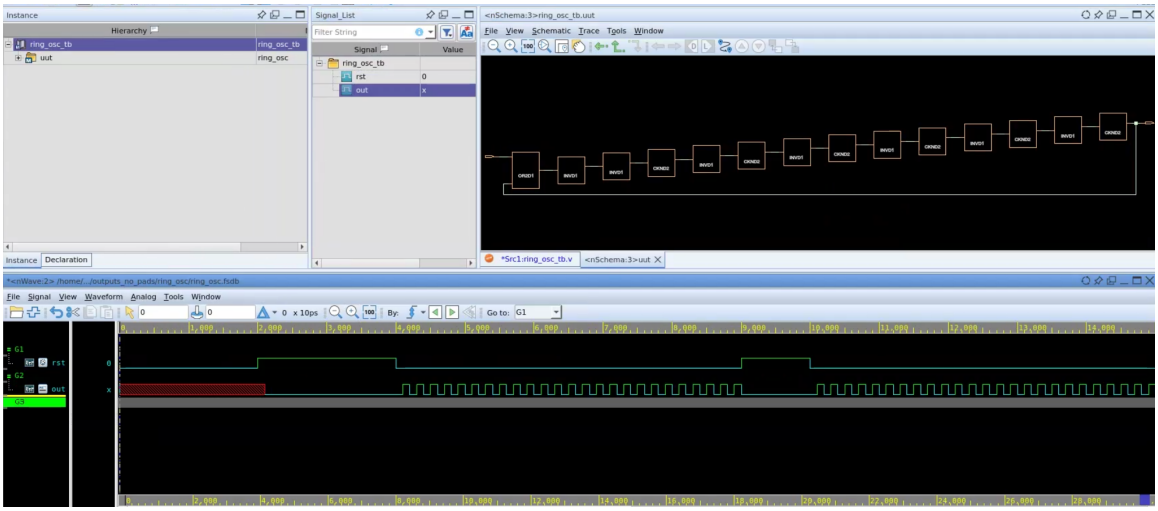
Nota. La imagen muestra la adición de *delay* a la compuerta OR2D1.

Ahora ejecutar los comandos de simulación y analizar en Verdi:

Cuadro 56. Comandos para la compilación y simulación del *ring oscillator*

```
vcs -full64 -debug_access+all -kdb ring_osc_tb.v
   ring_osc_mapped.v -v ../../tcbn65lp.v
./simv
verdi -ssf ring_osc.fsdb
```

Figura 93. Simulación de la síntesis del *core* del *ring oscillator*



Nota. La imagen muestra la simulación de la síntesis del *core* del *ring oscillator*.

En la Figura 93 se observa que el funcionamiento del circuito es el correcto, adicionalmente se muestra el diagrama con el nombre de las celdas estándar de la librería de TSMC, el cual coincide con el diagrama de Design Vision.

11.4.5. Simulación del módulo del circuito con pads

Luego de validar la síntesis del diseño sin pads, se realizó la síntesis del *ring oscillator* con pads. Para ello, es importante modificar la instancia del archivo *testbench* (recuadro amarillo), pero adicionalmente para este caso, también comentar la linea donde se almacenan los valores de las etapas de inversión (recuadro naranja). Esto porque el archivo con pads no posee estos *wires* definidos en su propia instancia:

Figura 94. Cambios en el *testbench* para la simulación del *ring oscillator* con pads

```
1  `timescale 1ns/10ps
2
3  module ring_osc_tb;
4
5    reg rst;     // Reset signal
6    wire out;    // Output signal
7
8    // Instantiate the ring oscillator
9    IO_ring_osc uut (
10     .rst(rst),
11     .out(out)
12   );
13
14   initial begin
15     $dumpfile("ring_osc.vcd");
16     $dumpvars(0, ring_osc_tb);
17     //$dumpvars(0, uut.W_1, uut.W_2, uut.W_3, uut.W_4, uut.W_5, uut.W_6);
18     $fsdbDumpfile("ring_osc.fsdb");
19     $fsdbDumpvars(0, ring_osc_tb);
20   end
```

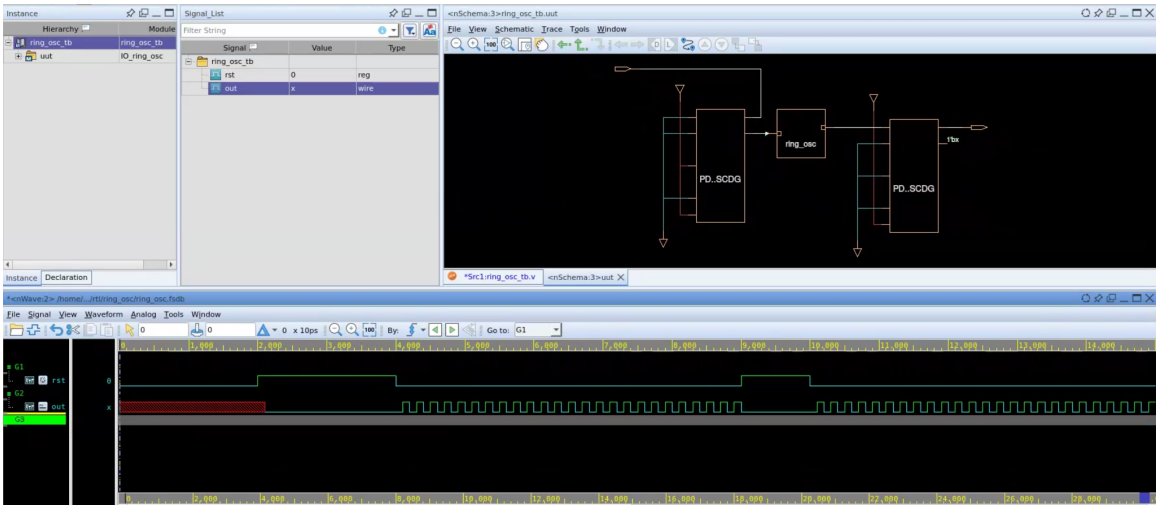
Nota. La imagen muestra cambios en el *testbench* del *ring oscillator*.

Posteriormnente, ejecutar los respectivos comandos para simular en Verdi:

Cuadro 57. Comandos para la compilación y simulación de la NOT con pads

```
vcs -full64 -debug_access+all -kdb ring_osc_tb.v IO_ring_osc.v
   ring_osc.v -v ../../outputs/tpdn65lpnv2od3.v
./simv
verdi -ssf ring_osc.fsdb
```

Figura 95. Simulación del *ring oscillator* con pads



Nota. La imagen muestra la simulación del *ring oscillator* con pads en Verdi.

La Figura 95 muestra que el *ring oscillator* funciona de manera esperada aún con pads. Por otro lado, en el esquemático es posible ver las instancias de dichos pads.

11.4.6. Síntesis del módulo del circuito con pads

Para realizar la síntesis del *ring oscillator* con pads, primero modificar el *script* de forma acorde:

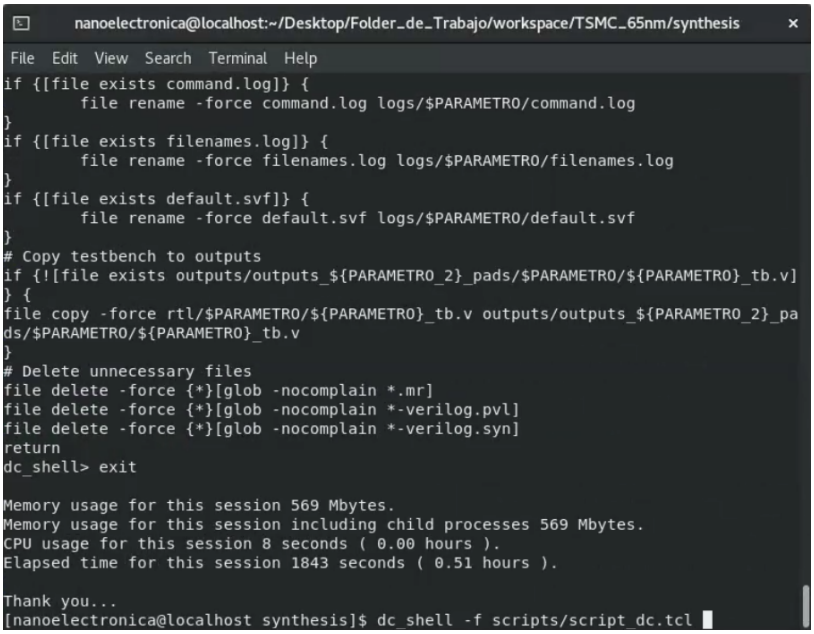
Figura 96. Modificaión de parámetros para la síntesis del *ring oscillator* con pads

```
# Author: Carlos Julio Valdes Oajaca
# Date: July 2025

# Parameters the user should modify
# PARAMETRO -> name of the circuit and instance (verilog file and instance must have the same name)
# USER_OPTION -> select between 1 and 2 only
# 1 -> synthesis without PADS
# 2 -> synthesis with PADS

set PARAMETRO "ring_osc"
set USER_OPTION 2
```

Nota. La imagen muestra el cambio en el *script* `script_dc.tcl`.

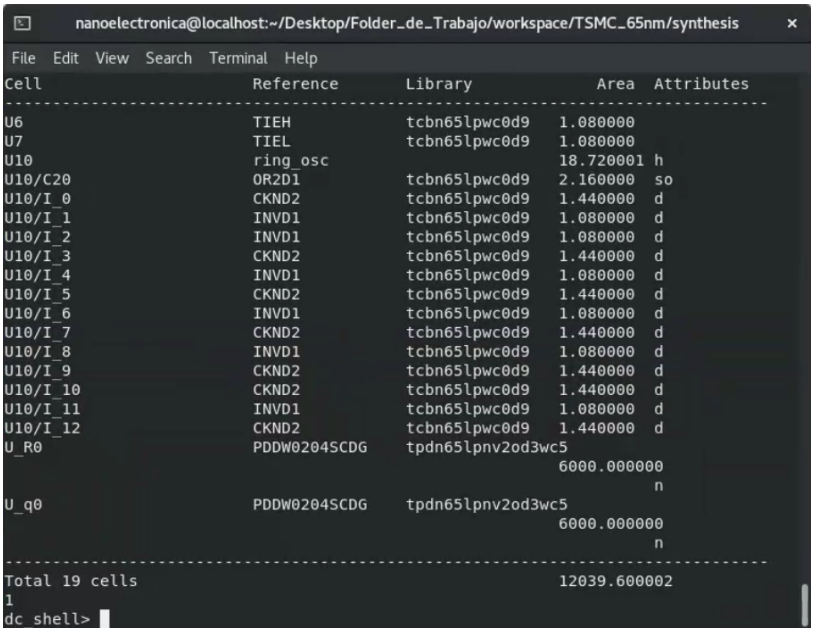
Ejecutar el *script*, reportar las celdas mapeadas y verificar posibles problemas en el diseño:

Figura 97. Ejecución del *script* desde la terminal para realizar la síntesis del *ring oscillator* con pads



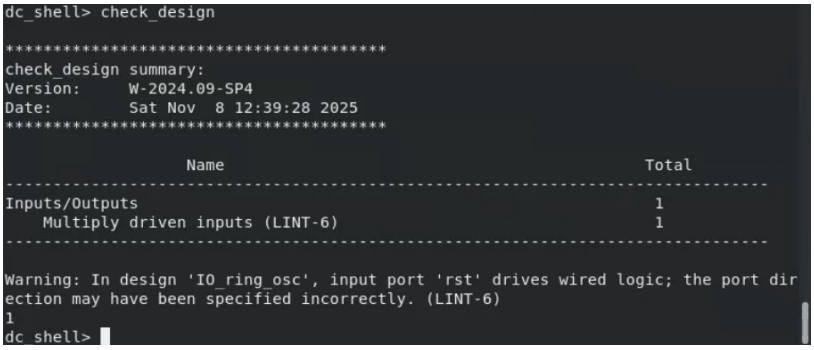
Nota. La imagen muestra la ejecución del *script* desde la terminal.

Figura 98. Ejecución del comando `report_cell [get_cells -hier *]` para la síntesis del *ring oscillator* con pads



Nota. La imagen muestra las celdas mapeadas del *ring oscillator* con pads.

Figura 99. Ejecución del comando `check_design` para la síntesis del *ring oscillator* con pads



Nota. La imagen muestra los posibles problemas de la síntesis del *ring oscillator* con pads.

En la Figura 99 se observa únicamente una advertencia acerca de la *wired logic*, debido a que solamente hay un pad de entrada.

A continuación se presenta el *gate-level netlist* del *ring oscillator* con pads:

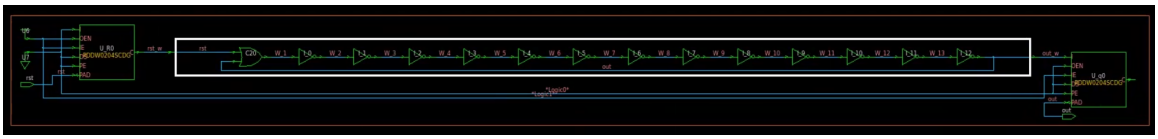
Cuadro 58. Archivo ring_osc_mapped.v en el directorio `outputs_with_pads`

```
////////////////////////////////////////////////////////////////
// Created by: Synopsys DC Ultra(TM) in wire load mode
// Version   : W-2024.09-SP4
// Date      : Sat Nov  8 11:25:59 2025
////////////////////////////////////////////////////////////////


module ring_osc ( rst, out );
  input rst;
  output out;
  wire   W_1, W_2, W_3, W_4, W_5, W_6, W_7, W_8, W_9, W_10,
     W_11, W_12, W_13;

  CKND2 I_12 ( .I(W_13), .ZN(out) );
  CKND2 I_10 ( .I(W_11), .ZN(W_12) );
  CKND2 I_9 ( .I(W_10), .ZN(W_11) );
  CKND2 I_7 ( .I(W_8), .ZN(W_9) );
  CKND2 I_5 ( .I(W_6), .ZN(W_7) );
  CKND2 I_3 ( .I(W_4), .ZN(W_5) );
  CKND2 I_0 ( .I(W_1), .ZN(W_2) );
  OR2D1 C20 ( .A1(rst), .A2(out), .Z(W_1) );
  INVD1 I_1 ( .I(W_2), .ZN(W_3) );
  INVD1 I_2 ( .I(W_3), .ZN(W_4) );
  INVD1 I_4 ( .I(W_5), .ZN(W_6) );
  INVD1 I_6 ( .I(W_7), .ZN(W_8) );
  INVD1 I_8 ( .I(W_9), .ZN(W_10) );
  INVD1 I_11 ( .I(W_12), .ZN(W_13) );
```

```
endmodule


module IO_ring_osc ( rst, out );
  input rst;
  output out;
  wire   \*Logic1* , \*Logic0* , rst_w, out_w;

  ring_osc U10 ( .rst(rst_w), .out(out_w) );
  PDDW0204SCDG U_R0 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(
        rst), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(rst_w) );
  PDDW0204SCDG U_q0 ( .I(out_w), .OEN(\*Logic0* ), .IE(\*Logic1
     * ), .PAD(out),
        .DS(\*Logic0* ), .PE(\*Logic0* ) );
  TIEH U6 ( .Z(\*Logic1* ) );
  TIEL U7 ( .ZN(\*Logic0* ) );
endmodule
```

Este es el esquemático del diseño visto con Design Vision:

Figura 100. Esquemático del *ring oscillator* con pads



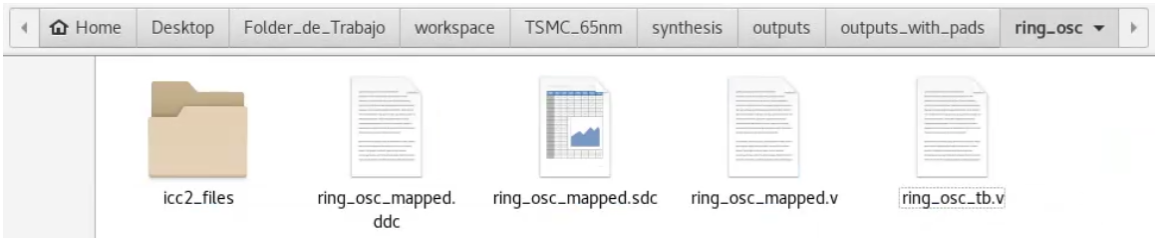
Nota. La imagen muestra el esquemático del *ring oscillator* con pads desde Design Vision.

En la figura anterior se puede observar el esquema del *ring oscillator* junto a los pads de entrada y salida respectivamente.

11.4.7. Simulación del módulo del circuito sintetizado con pads

Luego se comprueba la validez del diseño sintetizado, para confirmar que funcione de la forma esperada. Primero revisar que los archivos de salida se encuentren en el respectivo directorio:

Figura 101. Archivos de salida de la síntesis del *ring oscillator* con pads



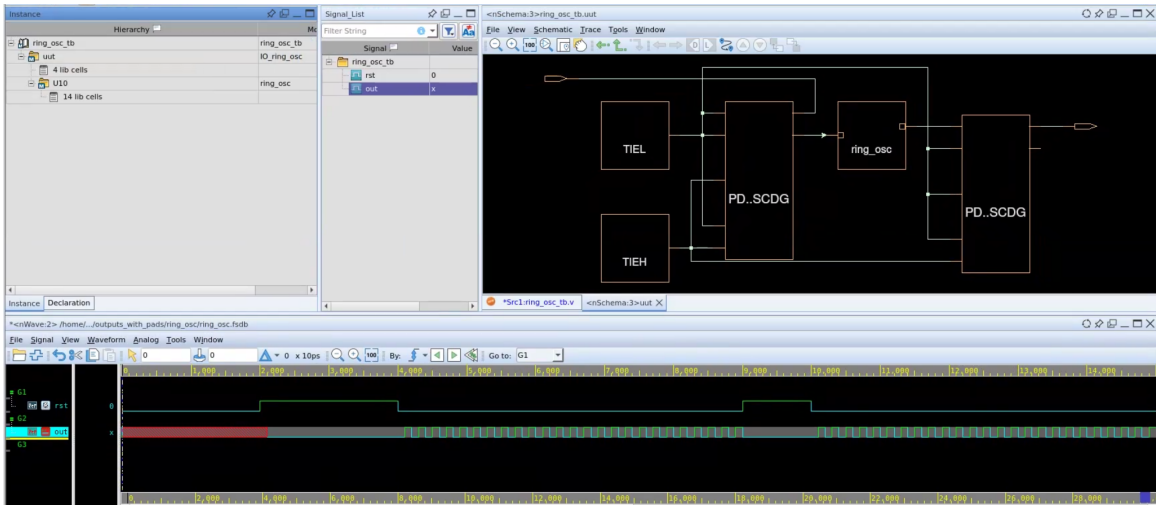
Nota. La imagen muestra archivos de salida de la síntesis del *ring oscillator* con pads.

Luego ejecutar los comandos para simular con Verdi:

Cuadro 59. Comandos para la compilación y simulación del *ring oscillator* con pads

```
vcs -full64 -debug_access+all -kdb ring_osc_tb.v
   ring_osc_mapped.v -v ../../tcbn65lp.v -v ../../
   tpdn65lpnv2od3.v
./simv
verdi -ssf ring_osc.fsdb
```

Figura 102. Simulación de la síntesis del *ring oscillator* con pads



Nota. La imagen muestra la simulación de la síntesis del *ring oscillator* con pads.

La figura anterior demuestra que el funcionamiento del *ring oscillator* con pads es correcto, las oscilasiones se presentan al igual que cuando el circuito no tiene pads. En el esquemático puede notarse las celdas de los pads y de alimentación (TIEL y TIEH).

11.5. Síntesis lógica de una *Arithmetic Logic Unit* (ALU) de 4 bits con un *Ring Oscillator*

Luego de sintetizar un *ring oscillator* de trece compuertas NOT, se realizó el proceso de síntesis de una *Arithmetic Logic Unit* (ALU) junto con un *ring oscillator*. Esto le agrega complejidad al circuito sin llegar al nanochip "El Gran Jaguar" todavía.

Para este diseño la nomenclatura de los archivos y directorios es: `ALU_with_ring_osc`

11.5.1. Módulo *core* del circuito, *testbench* y módulo del circuito con pads

Para comenzar con el proceso de síntesis, es importante tener los tres módulos principales listos. En el Cuadro 60 se observa el diseño del *core* que contiene la ALU y el *ring oscillator*, el Cuadro 61 muestra el *testbench* y el Cuadro 62 muestra el diseño con pads.

Cuadro 60. Archivo ALU_with_ring_osc.v

```
module ALU_with_ring_osc(a, b, select, q, reset_ring_osc,
   out_ring_osc);

input [3:0] a, b;
input reset_ring_osc;
input [2:0] select;
output out_ring_osc;
output reg [4:0] q;

// Wires para el ring oscillator (agregar delay a wires para la
    simulacion e.g. #1)
wire  W_1;
wire  W_2;
wire  W_3;
wire  W_4;
wire  W_5;
wire  W_6;
wire  W_7;
wire  W_8;
wire  W_9;
wire  W_10;
wire  W_11;
wire  W_12;
wire  W_13;

// Ring oscillator de 13 NOTs
assign #1 W_1 = reset_ring_osc | out_ring_osc;
assign W_2 = ~W_1;
assign W_3 = ~W_2;
```

```
assign W_4 = ~W_3;
assign W_5 = ~W_4;
assign W_6 = ~W_5;
assign W_7 = ~W_6;
assign W_8 = ~W_7;
assign W_9 = ~W_8;
assign W_10 = ~W_9;
assign W_11 = ~W_10;
assign W_12 = ~W_11;
assign W_13 = ~W_12;
assign out_ring_osc = ~W_13;

// ALU 4 BITS
    always @(select) // Always para que se ejecute siempre que
       cambie s (el selector)
    begin
        case(select) // Dependiendo del selector es la funcion
           de la ALU
            3'b000 : begin q = a; end
            3'b001 : begin q = a + b; end
            3'b010 : begin q = a - b; end
            3'b011 : begin q = a / b; end
            3'b100 : begin q = a % b; end // residuo
            3'b101 : begin q = a << 1; end
            3'b110 : begin q = a >> 1; end
            3'b111 : begin q = (a>b); end // comparar magnitud
        endcase
    end

endmodule
```

Cuadro 61. Archivo ALU_with_ring_osc_tb.v

```
`timescale 1ns/10ps

module ALU_with_ring_osc_tb;

reg [3:0] a, b;
reg reset_ring_osc;
reg [2:0] select;
wire out_ring_osc;
wire [4:0] q;

  // Instantiate
  ALU_with_ring_osc uut (
    .a(a),
    .b(b),
.select(select), .q(q),
```

```
.reset_ring_osc(reset_ring_osc), .out_ring_osc(out_ring_osc)
  );

  initial begin
    $dumpfile("ALU_with_ring_osc.vcd");
    $dumpvars(0, ALU_with_ring_osc_tb);
    $fsdbDumpfile("ALU_with_ring_osc.fsdb");
    $fsdbDumpvars(0, ALU_with_ring_osc_tb);
  end

  initial begin
        reset_ring_osc = 1'b0; a = 4'b1111; b = 4'b0000; select
            = 3'b000; // igualar
        #15;
        reset_ring_osc = 1'b1; a = 4'b1111; b = 4'b0001; select
            = 3'b001; // suma
        #15;
        reset_ring_osc = 1'b0; a = 4'b0011; b = 4'b0001; select
            = 3'b010; // resta
        #5;
        a = 4'b0100; b = 4'b0010; select = 3'b011; // division
        #5;
        a = 4'b0101; b = 4'b0010; select = 3'b100; //
           modulacion
        #5;
        a = 4'b1111; b = 4'b0001; select = 3'b101; //
           corrimiento a la izquierda
        #5;
        a = 4'b1111; b = 4'b1101; select = 3'b110; //
           corrimiento a la derecha
        #5;
        a = 4'b0011; b = 4'b0001; select = 4'b111; // comparar
           magnitud: a>b

        #50; $finish;
  end

  // Monitor the signals
  initial begin
    $monitor("Time: %0t ns, A: %b, B: %b,  Q: %b", $time, a, b,
        q);
  end

endmodule
```

Cuadro 62. Archivo IO_ALU_with_ring_osc.v

```
module IO_ALU_with_ring_osc(a, b, select, q, reset_ring_osc,
```

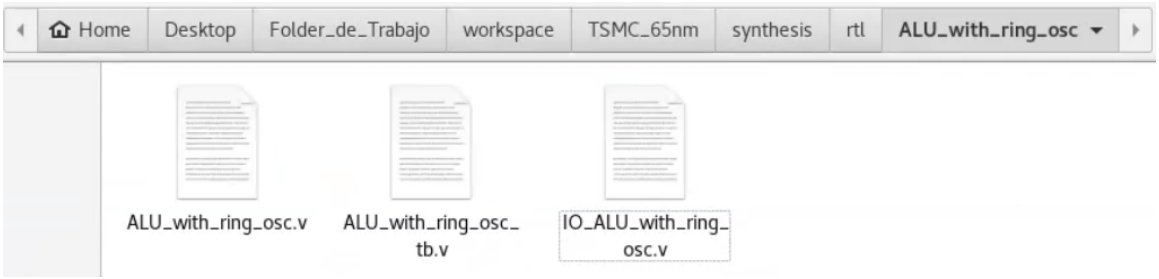
```
    out_ring_osc);

input [3:0] a, b;
input reset_ring_osc;
input [2:0] select;
output out_ring_osc;
output [4:0] q;

wire [3:0] a_w, b_w;
wire reset_ring_osc_w;
wire [2:0] select_w;
wire out_ring_osc_w;
wire [4:0] q_w;

//Entradas:
PDDW0204SCDG U9(.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(
    reset_ring_osc), .C(reset_ring_osc_w), .PE(1'b0), .IE(1'b1))
    ;
PDDW0204SCDG U10(.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(a[0]), .
    C(a_w[0]), .PE(1'b0), .IE(1'b1));
PDDW0204SCDG U11(.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(a[1]), .
    C(a_w[1]), .PE(1'b0), .IE(1'b1));
PDDW0204SCDG U12(.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(a[2]), .
    C(a_w[2]), .PE(1'b0), .IE(1'b1));
PDDW0204SCDG U13(.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(a[3]), .
    C(a_w[3]), .PE(1'b0), .IE(1'b1));

PDDW0204SCDG U14(.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(b[0]), .
    C(b_w[0]), .PE(1'b0), .IE(1'b1));
PDDW0204SCDG U15(.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(b[1]), .
    C(b_w[1]), .PE(1'b0), .IE(1'b1));
PDDW0204SCDG U16(.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(b[2]), .
    C(b_w[2]), .PE(1'b0), .IE(1'b1));
PDDW0204SCDG U17(.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(b[3]), .
    C(b_w[3]), .PE(1'b0), .IE(1'b1));

PDDW0204SCDG U18(.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(select
    [0]), .C(select_w[0]), .PE(1'b0), .IE(1'b1));
PDDW0204SCDG U19(.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(select
    [1]), .C(select_w[1]), .PE(1'b0), .IE(1'b1));
PDDW0204SCDG U20(.I(1'b0), .DS(1'b0), .OEN(1'b1), .PAD(select
    [2]), .C(select_w[2]), .PE(1'b0), .IE(1'b1));

//Salidas:
PDDW0204SCDG A1(.I(out_ring_osc_w), .DS(1'b0), .OEN(1'b0), .PAD
    (out_ring_osc), .C(1'bx), .PE(1'b0), .IE(1'b1));

PDDW0204SCDG A2(.I(q_w[0]), .DS(1'b0), .OEN(1'b0), .PAD(q[0]),
```

```
    .C(1'bx), .PE(1'b0), .IE(1'b1));
PDDW0204SCDG A3(.I(q_w[1]), .DS(1'b0), .OEN(1'b0), .PAD(q[1]),
    .C(1'bx), .PE(1'b0), .IE(1'b1));
PDDW0204SCDG A4(.I(q_w[2]), .DS(1'b0), .OEN(1'b0), .PAD(q[2]),
    .C(1'bx), .PE(1'b0), .IE(1'b1));
PDDW0204SCDG A5(.I(q_w[3]), .DS(1'b0), .OEN(1'b0), .PAD(q[3]),
    .C(1'bx), .PE(1'b0), .IE(1'b1));
PDDW0204SCDG A6(.I(q_w[4]), .DS(1'b0), .OEN(1'b0), .PAD(q[4]),
    .C(1'bx), .PE(1'b0), .IE(1'b1));

//Modulo:
ALU_with_ring_osc suma(.a(a_w), .b(b_w), .select(select_w), .q(
    q_w), .reset_ring_osc(reset_ring_osc_w), .out_ring_osc(
    out_ring_osc_w));

endmodule
```

11.5.2. Simulación del módulo *core* del circuito

Como se ha estado realizando, el primer paso es verificar que estén los archivos correspondientes en el folder `ALU_with_ring_osc`, en este caso concretamente.

Figura 103. Archivos para la síntesis del *core* de la ALU con *ring oscillator*



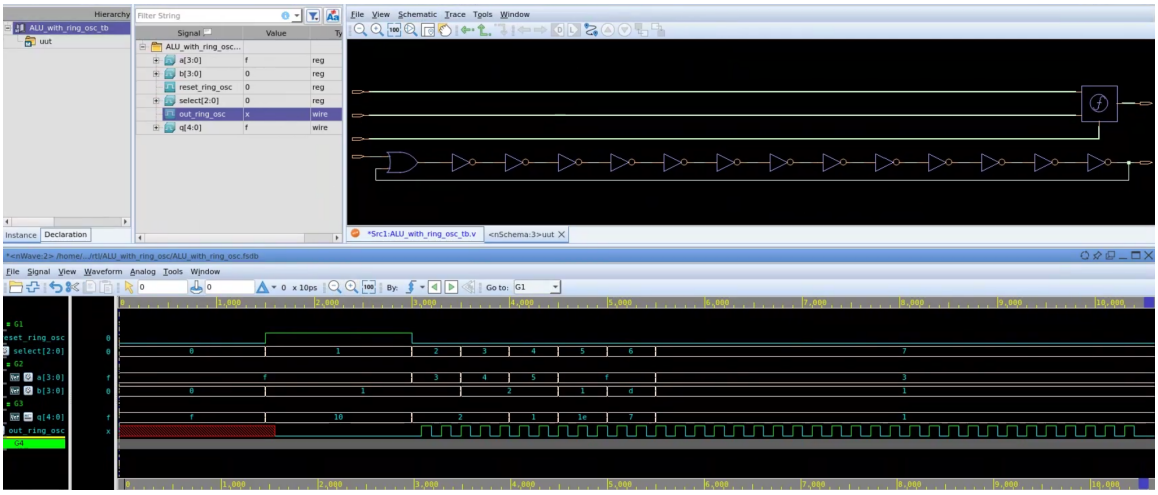
Nota. La imagen muestra los archivos necesarios para la síntesis del *core* de la ALU con *ring oscillator*.

Una vez estén los archivos diseñados en el direcetorio correspondiente simular:

Cuadro 63. Comandos para la compilación y simulación de la ALU con *ring oscillator*

```
vcs -full64 -debug_access+all -kdb ALU_with_ring_osc_tb.v
    ALU_with_ring_osc.v
./simv
verdi -ssf ALU_with_ring_osc.fsdb
```

Figura 104. Simulación del *core* de la ALU con *ring oscillator*



Nota. La imagen muestra la simulación del *core* de la ALU con *ring oscillator*.

Como se puede observar en la figura anterior, el diagrama del *ring oscillator* está separado de la ALU, ya que son partes distintas, pero conforman el mismo módulo en este caso.

El funcionamiento de la ALU es correcto y puede verificarse al ver los casos del código del *testbench* (ver Cuadro 61). Aunado a ello, el *ring oscillator* sí muestra oscilaciones en su salida.

11.5.3. Síntesis del módulo *core* del circuito

Ya que el funcionamiento del código de la ALU con *ring oscillator* es correcto, se procede a realizar la síntesis del mismo. Para ello, primero modificar los parámetros del *script*:

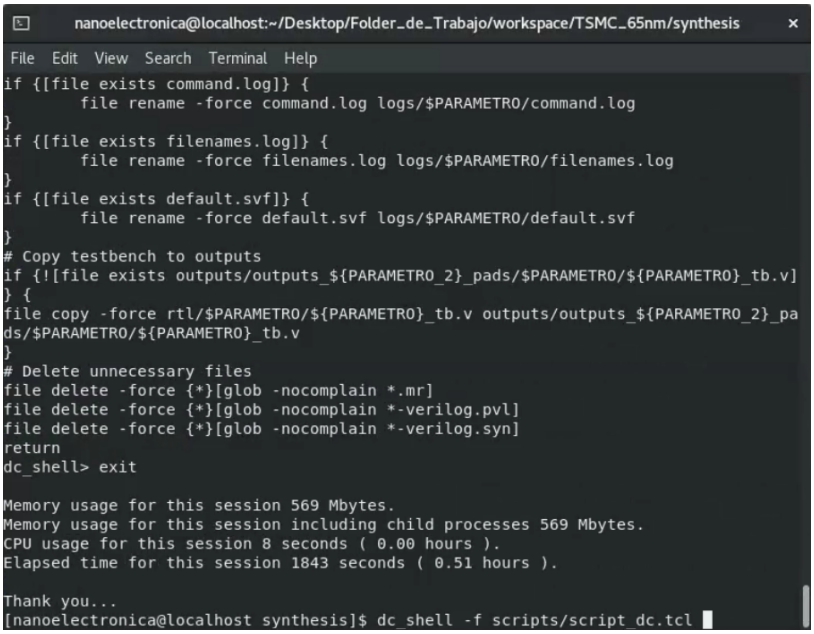
Figura 105. Parámetros para la síntesis del *core* de la ALU con *ring oscillator*

```
1 # Author: Carlos Julio Valdes Oajaca
2 # Date: July 2025
3
4 # Parameters the user should modify
5 # PARAMETRO -> name of the circuit and instance (verilog file and instance must have the same name)
6 # USER_OPTION -> select between 1 and 2 only
7 # 1 -> synthesis without PADS
8 # 2 -> synthesis with PADS
9
10 set PARAMETRO "ALU_with_ring_osc"
11 set USER_OPTION 1
```

Nota. La imagen muestra los parámetros para la síntesis del *core* de la ALU con *ring oscillator*.

Ahora se ejecuta el *script*:

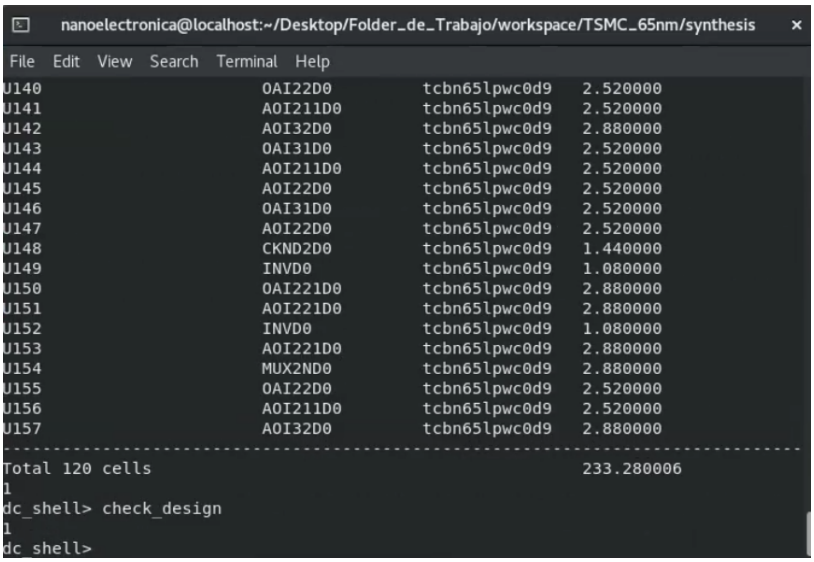
Figura 106. Ejecución del *script* desde la terminal para realizar la síntesis del *core* de la ALU con *ring oscillator*



Nota. La imagen muestra la ejecución del *script* desde la terminal.

Luego reportar las celdas y verificar si hay algún problema con el diseño sintetizado:

Figura 107. Celdas y problemas de la síntesis del *core* de la ALU con *ring oscillator*



Nota. La imagen muestra la ejecución del *script* desde la terminal.

Como se puede notar en la figura anterior, el diseño sintetizado consta de 120 celdas.

Esta cantidad va en aumento mientras el diseño se vuelve más complejo. Por otro lado, no existen adevertencias ni errores en el diseño del *core*.

A continación se muestra el *gate-level netlist* de este circuito:

Cuadro 64. Archivo ALU_with_ring_osc.v en el directorio `outputs_no_pads/`

```
////////////////////////////////////////////////////////////////
// Created by: Synopsys DC Ultra(TM) in wire load mode
// Version   : W-2024.09-SP4
// Date      : Sat Nov  8 17:05:07 2025
////////////////////////////////////////////////////////////////


module ALU_with_ring_osc ( a, b, select, q, reset_ring_osc,
    out_ring_osc );
  input [3:0] a;
  input [3:0] b;
  input [2:0] select;
  output [4:0] q;
  input reset_ring_osc;
  output out_ring_osc;
  wire   W_1, W_2, W_3, W_4, W_5, W_6, W_7, W_8, W_9, W_10,
     W_11, W_12, W_13,
         n52, n53, n54, n55, n56, n57, n58, n59, n60, n61, n62,
             n63, n64, n65,
         n66, n67, n68, n69, n70, n71, n72, n73, n74, n75, n76,
             n77, n78, n79,
         n80, n81, n82, n83, n84, n85, n86, n87, n88, n89, n90,
             n91, n92, n93,
         n94, n95, n96, n97, n98, n99, n100, n101, n102, n103,
            n104, n105,
         n106, n107, n108, n109, n110, n111, n112, n113, n114,
            n115, n116,
         n117, n118, n119, n120, n121, n122, n123, n124, n125,
            n126, n127,
         n128, n129, n130, n131, n132, n133, n134, n135, n136,
            n137, n138,
         n139, n140, n141, n142, n143, n144, n145, n146, n147,
            n148, n149,
         n150, n151, n152;

  CKND2 I_11 ( .I(W_12), .ZN(W_13) );
  CKND2 I_9 ( .I(W_10), .ZN(W_11) );
  CKND2 I_7 ( .I(W_8), .ZN(W_9) );
  CKND2 I_5 ( .I(W_6), .ZN(W_7) );
  CKND2 I_3 ( .I(W_4), .ZN(W_5) );
  CKND2 I_0 ( .I(W_1), .ZN(W_2) );
  OR2D1 C77 ( .A1(reset_ring_osc), .A2(out_ring_osc), .Z(W_1) )
```

```
   ;
INVD1 I_1 ( .I(W_2), .ZN(W_3) );
INVD1 I_2 ( .I(W_3), .ZN(W_4) );
INVD1 I_4 ( .I(W_5), .ZN(W_6) );
INVD1 I_6 ( .I(W_7), .ZN(W_8) );
INVD1 I_8 ( .I(W_9), .ZN(W_10) );
INVD1 I_10 ( .I(W_11), .ZN(W_12) );
INVD0 I_12 ( .I(W_13), .ZN(out_ring_osc) );
INVD0 U52 ( .I(b[0]), .ZN(n83) );
NR2D0 U53 ( .A1(n83), .A2(a[0]), .ZN(n138) );
INVD0 U54 ( .I(n138), .ZN(n136) );
NR2D0 U55 ( .A1(a[1]), .A2(n83), .ZN(n64) );
INVD0 U56 ( .I(b[1]), .ZN(n137) );
INVD0 U57 ( .I(a[2]), .ZN(n109) );
AOI211D0 U58 ( .A1(b[0]), .A2(n109), .B(b[3]), .C(b[2]), .ZN(
   n52) );
OAI21D0 U59 ( .A1(a[3]), .A2(n137), .B(n52), .ZN(n53) );
OAI21D0 U60 ( .A1(n83), .A2(n53), .B(a[2]), .ZN(n62) );
MAOI222D0 U61 ( .A(b[1]), .B(n64), .C(n62), .ZN(n56) );
INVD0 U62 ( .I(n53), .ZN(n93) );
NR2D0 U63 ( .A1(b[3]), .A2(b[2]), .ZN(n82) );
INVD0 U64 ( .I(a[3]), .ZN(n122) );
AO31D0 U65 ( .A1(n137), .A2(b[0]), .A3(n82), .B(n122), .Z(n54
   ) );
AOI21D0 U66 ( .A1(b[1]), .A2(n93), .B(n54), .ZN(n57) );
INVD0 U67 ( .I(b[2]), .ZN(n103) );
MAOI222D0 U68 ( .A(n56), .B(n57), .C(n103), .ZN(n55) );
NR2D0 U69 ( .A1(n55), .A2(b[3]), .ZN(n125) );
CKND2D0 U70 ( .A1(n103), .A2(n56), .ZN(n59) );
INVD0 U71 ( .I(n57), .ZN(n58) );
AOI21D0 U72 ( .A1(n125), .A2(n59), .B(n58), .ZN(n79) );
INVD0 U73 ( .I(b[3]), .ZN(n117) );
CKND2D0 U74 ( .A1(b[1]), .A2(n64), .ZN(n60) );
OAI211D0 U75 ( .A1(b[1]), .A2(n64), .B(n125), .C(n60), .ZN(
   n61) );
CKXOR2D0 U76 ( .A1(n62), .A2(n61), .Z(n107) );
INVD0 U77 ( .I(n107), .ZN(n106) );
INR2D0 U78 ( .A1(n125), .B1(n83), .ZN(n63) );
INVD0 U79 ( .I(a[1]), .ZN(n145) );
MAOI22D0 U80 ( .A1(n64), .A2(n125), .B1(n63), .B2(n145), .ZN(
   n140) );
MAOI222D0 U81 ( .A(b[1]), .B(n138), .C(n140), .ZN(n102) );
INVD0 U82 ( .I(n102), .ZN(n104) );
MAOI222D0 U83 ( .A(b[2]), .B(n106), .C(n104), .ZN(n78) );
MAOI222D0 U84 ( .A(n79), .B(n117), .C(n78), .ZN(n135) );
INVD0 U85 ( .I(select[1]), .ZN(n115) );
CKND2D0 U86 ( .A1(select[2]), .A2(n115), .ZN(n143) );
NR2D0 U87 ( .A1(select[2]), .A2(n115), .ZN(n126) );
```

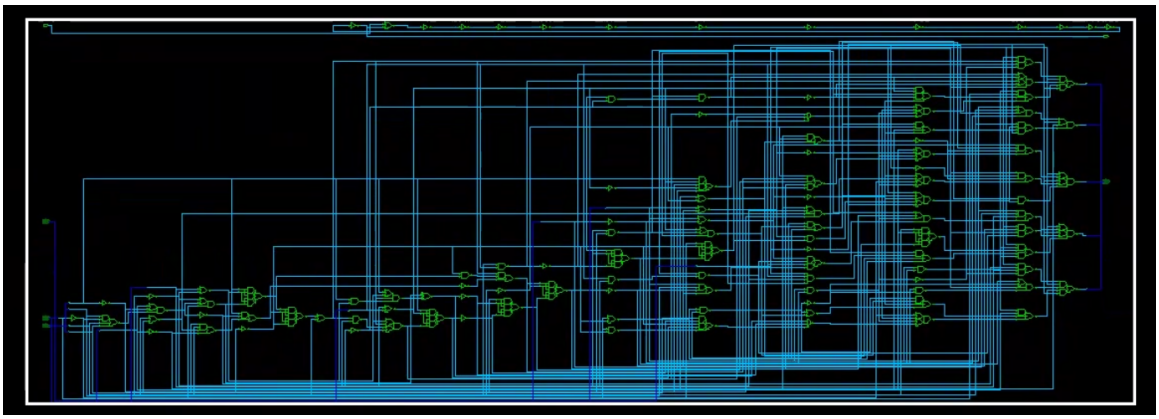
```
CKND2D0 U88 ( .A1(a[0]), .A2(n83), .ZN(n65) );
CKND2D0 U89 ( .A1(n136), .A2(n65), .ZN(n69) );
INVD0 U90 ( .I(n135), .ZN(n70) );
AOI31D0 U91 ( .A1(select[2]), .A2(b[0]), .A3(n70), .B(select
   [1]), .ZN(n66)
      );
AOI22D0 U92 ( .A1(n126), .A2(n69), .B1(a[0]), .B2(n66), .ZN(
   n67) );
OAI31D0 U93 ( .A1(n136), .A2(n135), .A3(n143), .B(n67), .ZN(
   n68) );
AOI31D0 U94 ( .A1(select[2]), .A2(select[1]), .A3(a[1]), .B(
   n68), .ZN(n76)
      );
INVD0 U95 ( .I(select[0]), .ZN(n149) );
NR2D0 U96 ( .A1(select[2]), .A2(select[1]), .ZN(n129) );
AOI22D0 U97 ( .A1(n126), .A2(n70), .B1(n129), .B2(n69), .ZN(
   n75) );
INR2D0 U98 ( .A1(select[2]), .B1(n115), .ZN(n148) );
CKND2D0 U99 ( .A1(a[2]), .A2(n103), .ZN(n94) );
MAOI222D0 U100 ( .A(b[3]), .B(n122), .C(n94), .ZN(n73) );
CKND2D0 U101 ( .A1(a[1]), .A2(n137), .ZN(n128) );
CKND2D0 U102 ( .A1(b[1]), .A2(n145), .ZN(n127) );
ND4D0 U103 ( .A1(select[0]), .A2(a[0]), .A3(n83), .A4(n127),
   .ZN(n71) );
NR2D0 U104 ( .A1(a[3]), .A2(n117), .ZN(n77) );
NR2D0 U105 ( .A1(a[2]), .A2(n103), .ZN(n95) );
AOI211D0 U106 ( .A1(n128), .A2(n71), .B(n77), .C(n95), .ZN(
   n72) );
AOI32D0 U107 ( .A1(select[0]), .A2(n148), .A3(n73), .B1(n72),
    .B2(n148),
     .ZN(n74) );
OAI221D0 U108 ( .A1(select[0]), .A2(n76), .B1(n149), .B2(n75)
   , .C(n74), .ZN(
     q[0]) );
IOA21D0 U109 ( .A1(n138), .A2(n128), .B(n127), .ZN(n100) );
AOI21D0 U110 ( .A1(n94), .A2(n100), .B(n95), .ZN(n118) );
AOI21D0 U111 ( .A1(a[3]), .A2(n117), .B(n77), .ZN(n89) );
INVD0 U112 ( .I(n89), .ZN(n86) );
INVD0 U113 ( .I(n129), .ZN(n144) );
AOI32D0 U114 ( .A1(n117), .A2(n79), .A3(n78), .B1(n135), .B2(
   n79), .ZN(n80)
      );
OAI22D0 U115 ( .A1(n122), .A2(n144), .B1(n143), .B2(n80), .ZN
   (n81) );
AOI31D0 U116 ( .A1(n126), .A2(n118), .A3(n86), .B(n81), .ZN(
   n92) );
CKND2D0 U117 ( .A1(b[0]), .A2(a[0]), .ZN(n130) );
INVD0 U118 ( .I(n130), .ZN(n131) );
```

```
MAOI222D0 U119 ( .A(b[1]), .B(a[1]), .C(n131), .ZN(n98) );
MAOI222D0 U120 ( .A(n109), .B(n103), .C(n98), .ZN(n116) );
INVD0 U121 ( .I(n116), .ZN(n87) );
INVD0 U122 ( .I(n126), .ZN(n132) );
OAI211D0 U123 ( .A1(a[3]), .A2(n83), .B(n82), .C(n137), .ZN(
   n84) );
OAI22D0 U124 ( .A1(n132), .A2(n84), .B1(n109), .B2(n143), .ZN
   (n85) );
AOI31D0 U125 ( .A1(n129), .A2(n87), .A3(n86), .B(n85), .ZN(
   n91) );
CKND2D0 U126 ( .A1(n129), .A2(n116), .ZN(n88) );
OAI32D0 U127 ( .A1(select[0]), .A2(n118), .A3(n132), .B1(n88)
   , .B2(n149),
      .ZN(n120) );
CKND2D0 U128 ( .A1(n89), .A2(n120), .ZN(n90) );
OAI221D0 U129 ( .A1(select[0]), .A2(n92), .B1(n149), .B2(n91)
   , .C(n90), .ZN(
      q[3]) );
INVD0 U130 ( .I(n143), .ZN(n124) );
AOI22D0 U131 ( .A1(n126), .A2(n93), .B1(a[1]), .B2(n124), .ZN
   (n114) );
IND2D0 U132 ( .A1(n95), .B1(n94), .ZN(n101) );
INVD0 U133 ( .I(n101), .ZN(n97) );
CKND2D0 U134 ( .A1(n98), .A2(n97), .ZN(n96) );
OAI211D0 U135 ( .A1(n98), .A2(n97), .B(n129), .C(n96), .ZN(
   n113) );
NR2D0 U136 ( .A1(n101), .A2(n100), .ZN(n99) );
AOI211D0 U137 ( .A1(n101), .A2(n100), .B(n132), .C(n99), .ZN(
   n111) );
AOI221D0 U138 ( .A1(b[2]), .A2(n104), .B1(n103), .B2(n102), .
   C(n135), .ZN(
      n105) );
MUX2ND0 U139 ( .I0(n107), .I1(n106), .S(n105), .ZN(n108) );
OAI22D0 U140 ( .A1(n109), .A2(n144), .B1(n143), .B2(n108), .
   ZN(n110) );
AOI211D0 U141 ( .A1(n148), .A2(a[3]), .B(n111), .C(n110), .ZN
   (n112) );
AOI32D0 U142 ( .A1(n114), .A2(select[0]), .A3(n113), .B1(n112
   ), .B2(n149),
      .ZN(q[2]) );
OAI31D0 U143 ( .A1(select[2]), .A2(b[3]), .A3(n116), .B(n115)
   , .ZN(n123) );
AOI211D0 U144 ( .A1(n118), .A2(n117), .B(a[3]), .C(n132), .ZN
   (n119) );
AOI22D0 U145 ( .A1(b[3]), .A2(n120), .B1(n119), .B2(n149), .
   ZN(n121) );
OAI31D0 U146 ( .A1(n123), .A2(n149), .A3(n122), .B(n121), .ZN
   (q[4]) );
```

```
  AOI22D0 U147 ( .A1(n126), .A2(n125), .B1(a[0]), .B2(n124), .
     ZN(n152) );
  CKND2D0 U148 ( .A1(n128), .A2(n127), .ZN(n134) );
  INVD0 U149 ( .I(n134), .ZN(n133) );
  OAI221D0 U150 ( .A1(n131), .A2(n134), .B1(n130), .B2(n133), .
     C(n129), .ZN(
        n151) );
  AOI221D0 U151 ( .A1(n138), .A2(n134), .B1(n136), .B2(n133), .
     C(n132), .ZN(
        n147) );
  INVD0 U152 ( .I(n140), .ZN(n141) );
  AOI221D0 U153 ( .A1(b[1]), .A2(n138), .B1(n137), .B2(n136), .
     C(n135), .ZN(
        n139) );
  MUX2ND0 U154 ( .I0(n141), .I1(n140), .S(n139), .ZN(n142) );
  OAI22D0 U155 ( .A1(n145), .A2(n144), .B1(n143), .B2(n142), .
     ZN(n146) );
  AOI211D0 U156 ( .A1(n148), .A2(a[2]), .B(n147), .C(n146), .ZN
     (n150) );
  AOI32D0 U157 ( .A1(n152), .A2(select[0]), .A3(n151), .B1(n150
     ), .B2(n149),
        .ZN(q[1]) );
endmodule
```

El esquemático de dicho diseño en Design Vision es el siguiente:

Figura 108. Esquemático del *core* de ALU con *ring oscillator*

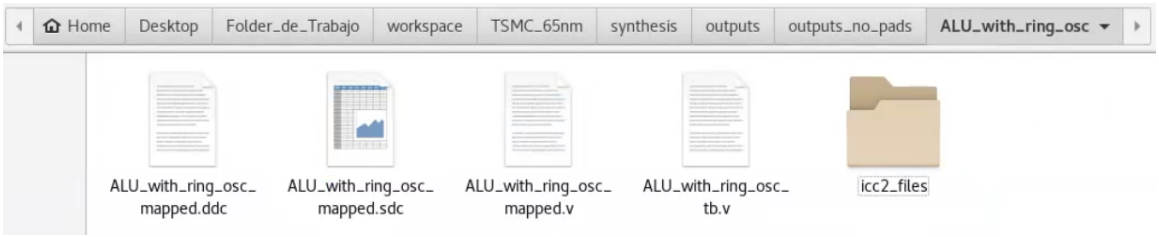


Nota. La imagen muestra el esquemático del *core* de la ALU con *ring oscillator* en Design Vision.

11.5.4. Simulación del *core* del circuito sintetizado

Antes de realizar la simulación se verifica si los archivos de salida se encuentran en el directorio apropiado:

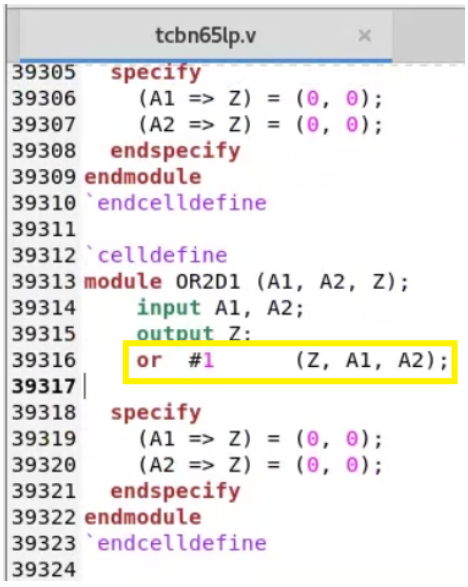
Figura 109. Archivos de salida de la síntesis del *core* de la ALU con *ring oscillator*



Nota. La imagen muestra archivos de salida de la síntesis del *core* de la ALU con *ring oscillator*.

Al igual que como se realizó la adición de *delay* de la OR2D1 con la síntesis del *ring oscillator*, para este caso también es importante realizarlo antes de simular:

Figura 110. Adición de *delay* a la celda OR del diseño sintetizado del *core* de la ALU con *ring oscillator*



Nota. La imagen muestra la adición de *delay* a la compuerta OR2D1.

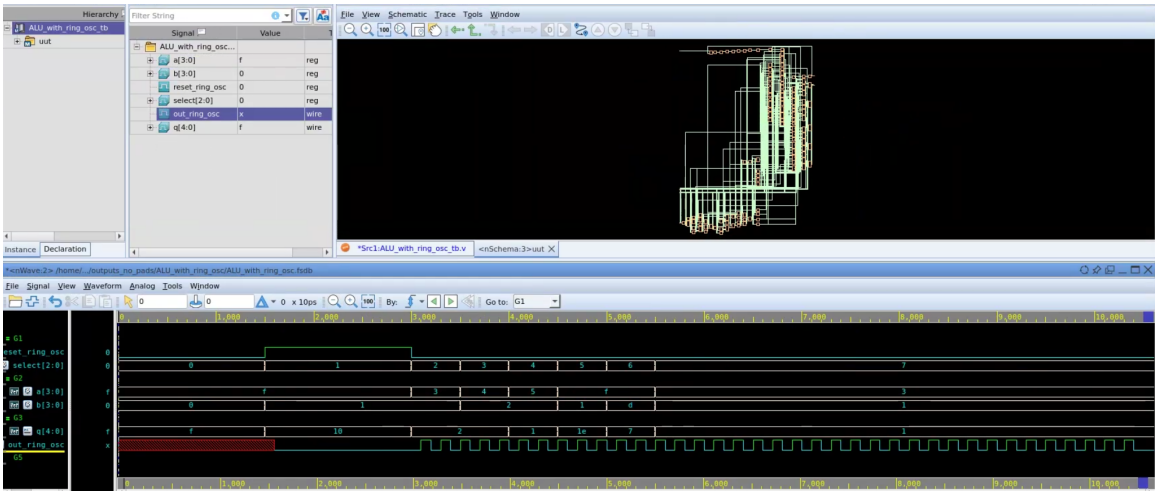
Luego ejectuar los comandos correspondientes para simular en Verdi:

Cuadro 65. Comandos para la compilación y simulación de la ALU con *ring oscillator*

```
vcs -full64 -debug_access+all -kdb ALU_with_ring_osc_tb.v
   ALU_with_ring_osc_mapped.v -v ../../tcbn65lp.v
```

```
./simv
verdi -ssf ALU_with_ring_osc.fsdb
```

Figura 111. Simulación de la síntesis del *core* de la ALU con *ring oscillator*



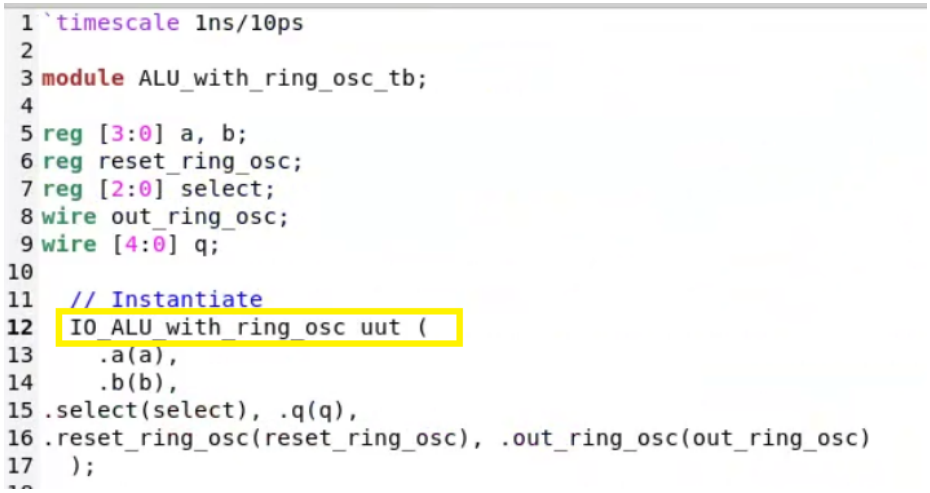
Nota. La imagen muestra la simulación de la síntesis del *core* de la ALU con *ring oscillator*.

Puede notarse en la Figura 111 que el funcionamiento del circuito sintetizado se conserva respecto al de la ALU con *ring oscillator* sin sintetizar. Esto indica que el diseño sintetizad es correcto y adecuado.

11.5.5. Simulación del módulo del circuito con pads

Como siguiente paso se simula el diseño de la ALU con *ring oscillator* con pads. Antes de ello, modificar el *testbench* de manera acorde.

Figura 112. Cambio de instancia en el *testbench* de la ALU con *ring oscillator*



```
1 `timescale 1ns/10ps
2
3 module ALU_with_ring_osc_tb;
4
5 reg [3:0] a, b;
6 reg reset_ring_osc;
7 reg [2:0] select;
8 wire out_ring_osc;
9 wire [4:0] q;
10
11  // Instantiate
12  IO_ALU_with_ring_osc uut (
13     .a(a),
14     .b(b),
15 .select(select), .q(q),
16 .reset_ring_osc(reset_ring_osc), .out_ring_osc(out_ring_osc)
17  );
```

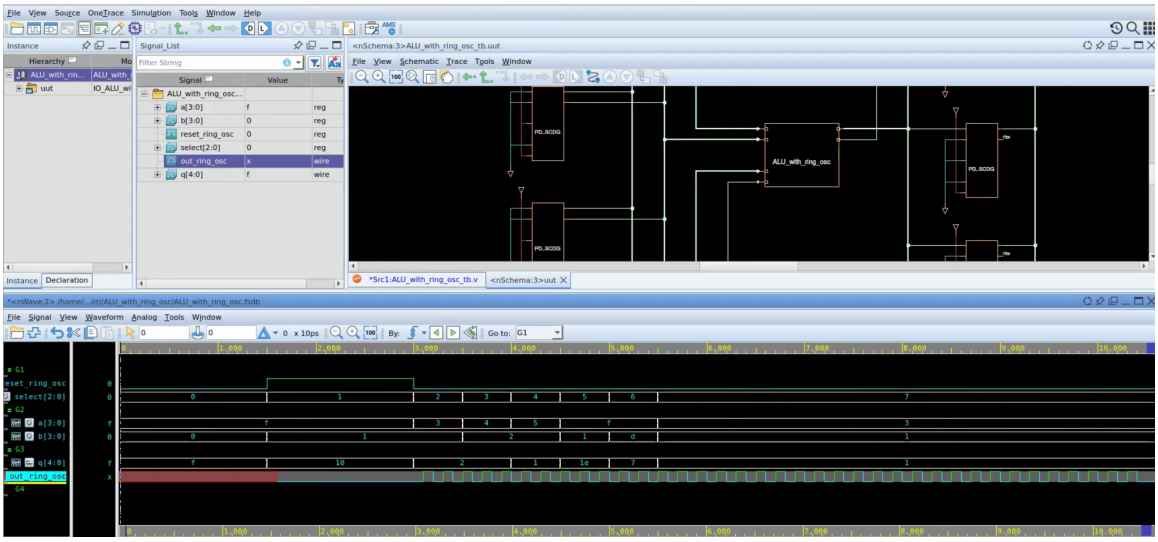
Nota. La imagen muestra el cambio de instancia en el *testbench* de la ALU con *ring oscillator*.

Luego, ejecutar los respectivos comandos para simular el diseño con pads antes de sintetizar.

Cuadro 66. Comandos para la compilación y simulación de la ALU con *ring oscillator* con pads

```
vcs -full64 -debug_access+all -kdb ALU_with_ring_osc_tb.v
   IO_ALU_with_ring_osc.v ALU_with_ring_osc.v -v ../../outputs/
   tpdn65lpnv2od3.v
./simv
verdi -ssf ALU_with_ring_osc.fsdb
```

Figura 113. Simulación de la ALU con *ring oscillator* con pads antes de sintetizar



Nota. La imagen muestra la simulación de la ALU con *ring oscillator* con pads en Verdi.

La Figura 113 muestra que el funcionamiento de la ALU con *ring oscillator* con pads es el mismo que el del circuito sin pads. La codificación demuestra ser correcta.

11.5.6. Síntesis del módulo del circuito con pads

Posteriormente se realiza la síntesis, se debe modificar antes el segundo parámetro del *script* para realizar la síntesis con pads.

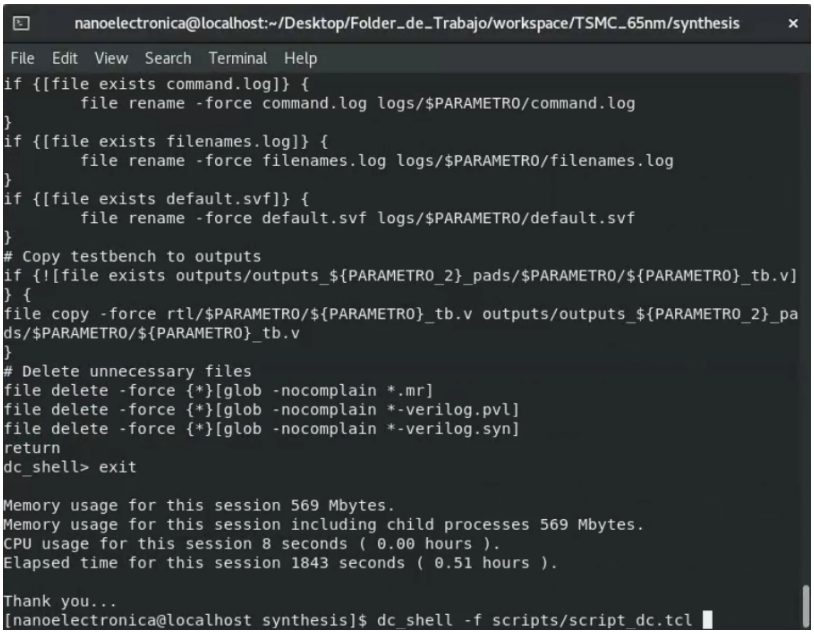
Figura 114. Parámetro modificado para la síntesis de la ALU con *ring oscillator* con pads

```
1 # Author: Carlos Julio Valdes Oajaca
2 # Date: July 2025
3
4 # Parameters the user should modify
5 # PARAMETRO -> name of the circuit and instance (verilog file and instance must have the same name)
6 # USER_OPTION -> select between 1 and 2 only
7 # 1 -> synthesis without PADS
8 # 2 -> synthesis with PADS
9
10 set PARAMETRO "ALU_with_ring_osc"
11 set USER_OPTION 2
```

Nota. La imagen muestra el parámetro modidicado para la síntesis de la ALU con *ring oscillator* con pads.

Ejecutar el *script*:

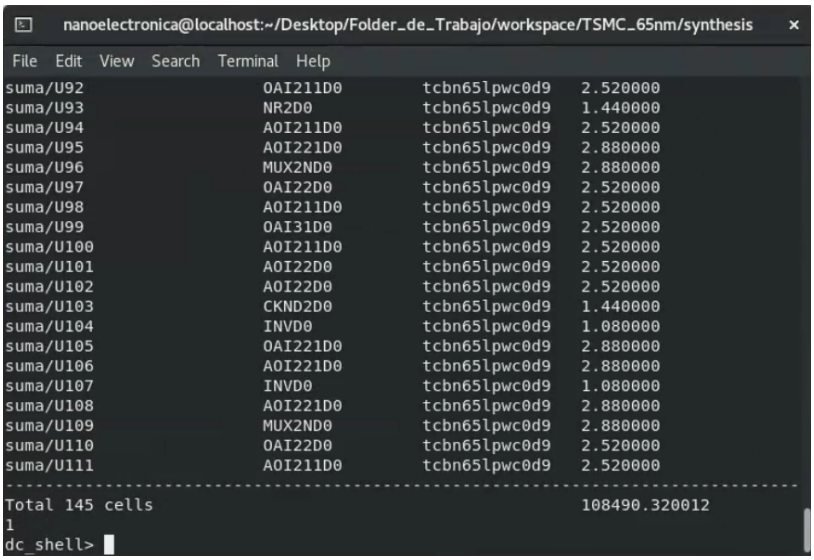
Figura 115. Ejecución del *script* desde la terminal para realizar la síntesis del *core* del *ring oscillator*



Nota. La imagen muestra la ejecución del *script* desde la terminal.

Reportar las celdas del diseño sintetizado:

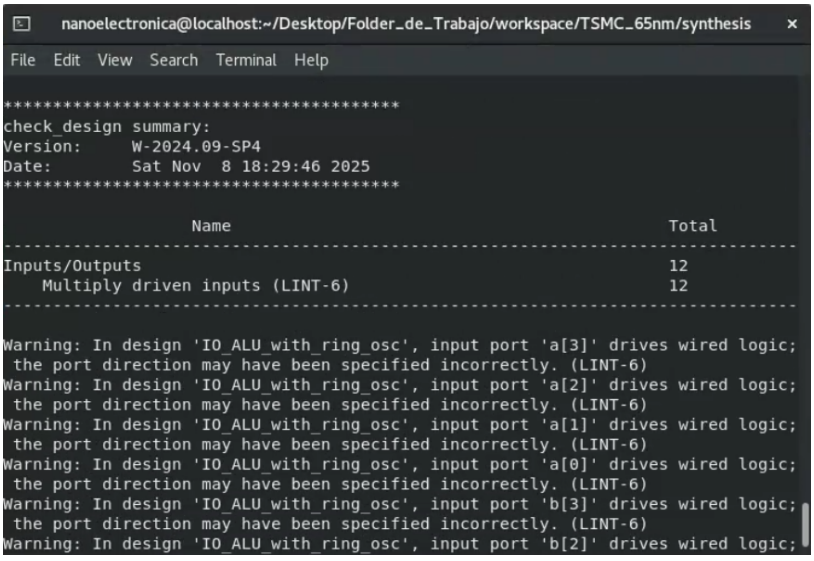
Figura 116. Celdas mapeadas de la síntesis de la ALU con *ring oscillator* con pads



Nota. La imagen muestra las celdas mapeadas de la síntesis de la ALU con *ring oscillator* con pads.

Verificar si existen problemas con el diseño sintetizado:

Figura 117. Posibles problemas de la síntesis de la ALU con *ring oscillator* con pads



Nota. La imagen muestra los posibles problemas de la síntesis de la ALU con *ring oscillator* con pads.

La Figura 117 muestra que existen doce advertencias relacionadas con la *wired logic*, esto se debe a las doce entradas del circuito. Como se ha visto con los circuitos anteriores esto no representa ningún problema para el funcionamiento adecuado del circuito sintetizado.

A continuación se muestra el *gate-level netlist*:

Cuadro 67. Archivo ALU_with_ring_osc_mapped.v en el directorio `otuputs_with_pads`

```
////////////////////////////////////////////////////////////////
// Created by: Synopsys DC Ultra(TM) in wire load mode
// Version   : W-2024.09-SP4
// Date      : Sat Nov  8 18:27:26 2025
////////////////////////////////////////////////////////////////


module ALU_with_ring_osc ( a, b, select, q, reset_ring_osc,
   out_ring_osc );
  input [3:0] a;
  input [3:0] b;
  input [2:0] select;
  output [4:0] q;
  input reset_ring_osc;
  output out_ring_osc;
  wire   n106, n107, n108, n109, W_1, W_2, W_3, W_4, W_5, W_6,
     W_7, W_8, W_9,
         W_10, W_11, W_12, W_13, n5, n6, n7, n8, n9, n10, n11,
            n12, n13, n14,
         n15, n16, n17, n18, n19, n20, n21, n22, n23, n24, n25,
```

```
            n26, n27, n28,
        n29, n30, n31, n32, n33, n34, n35, n36, n37, n38, n39,
            n40, n41, n42,
        n43, n44, n45, n46, n47, n48, n49, n50, n51, n52, n53,
            n54, n55, n56,
        n57, n58, n59, n60, n61, n62, n63, n64, n65, n66, n67,
            n68, n69, n70,
        n71, n72, n73, n74, n75, n76, n77, n78, n79, n80, n81,
            n82, n83, n84,
        n85, n86, n87, n88, n89, n90, n91, n92, n93, n94, n95,
            n96, n97, n98,
        n99, n100, n101, n102, n103, n104, n105;

CKND2 I_12 ( .I(W_13), .ZN(out_ring_osc) );
CKND2 I_10 ( .I(W_11), .ZN(W_12) );
CKND2 I_9 ( .I(W_10), .ZN(W_11) );
CKND2 I_7 ( .I(W_8), .ZN(W_9) );
CKND2 I_5 ( .I(W_6), .ZN(W_7) );
CKND2 I_3 ( .I(W_4), .ZN(W_5) );
CKND2 I_0 ( .I(W_1), .ZN(W_2) );
OR2D1 C77 ( .A1(reset_ring_osc), .A2(out_ring_osc), .Z(W_1) )
   ;
INVD1 I_1 ( .I(W_2), .ZN(W_3) );
INVD1 I_2 ( .I(W_3), .ZN(W_4) );
INVD1 I_4 ( .I(W_5), .ZN(W_6) );
INVD1 I_6 ( .I(W_7), .ZN(W_8) );
INVD1 I_8 ( .I(W_9), .ZN(W_10) );
INVD1 I_11 ( .I(W_12), .ZN(W_13) );
OA221D0 U2 ( .A1(select[0]), .A2(n29), .B1(n102), .B2(n28), .
   C(n27), .Z(n109) );
AO32D0 U3 ( .A1(n67), .A2(select[0]), .A3(n66), .B1(n65), .B2
   (n102), .Z(n107) );
AO32D0 U4 ( .A1(n105), .A2(select[0]), .A3(n104), .B1(n103),
   .B2(n102), .Z(
      n108) );
OA31D0 U5 ( .A1(n76), .A2(n102), .A3(n75), .B(n74), .Z(n106)
   );
CKND1 U6 ( .I(n107), .ZN(q[2]) );
CKND1 U7 ( .I(n108), .ZN(q[1]) );
CKND1 U8 ( .I(n106), .ZN(q[4]) );
CKND1 U9 ( .I(n109), .ZN(q[0]) );
OAI221D2 U10 ( .A1(select[0]), .A2(n45), .B1(n102), .B2(n44),
    .C(n43), .ZN(
      q[3]) );
INVD0 U11 ( .I(b[0]), .ZN(n36) );
NR2D0 U12 ( .A1(n36), .A2(a[0]), .ZN(n91) );
INVD0 U13 ( .I(n91), .ZN(n89) );
NR2D0 U14 ( .A1(a[1]), .A2(n36), .ZN(n17) );
```

```
  INVD0 U15 ( .I(b[1]), .ZN(n90) );
  INVD0 U16 ( .I(a[2]), .ZN(n62) );
  AOI211D0 U17 ( .A1(b[0]), .A2(n62), .B(b[3]), .C(b[2]), .ZN(
     n5) );
  OAI21D0 U18 ( .A1(a[3]), .A2(n90), .B(n5), .ZN(n6) );
  OAI21D0 U19 ( .A1(n36), .A2(n6), .B(a[2]), .ZN(n15) );
  MAOI222D0 U20 ( .A(b[1]), .B(n17), .C(n15), .ZN(n9) );
  INVD0 U21 ( .I(n6), .ZN(n46) );
  NR2D0 U22 ( .A1(b[3]), .A2(b[2]), .ZN(n35) );
  INVD0 U23 ( .I(a[3]), .ZN(n75) );
  AO31D0 U24 ( .A1(n90), .A2(b[0]), .A3(n35), .B(n75), .Z(n7) )
     ;
  AOI21D0 U25 ( .A1(b[1]), .A2(n46), .B(n7), .ZN(n10) );
  INVD0 U26 ( .I(b[2]), .ZN(n56) );
  MAOI222D0 U27 ( .A(n9), .B(n10), .C(n56), .ZN(n8) );
  NR2D0 U28 ( .A1(n8), .A2(b[3]), .ZN(n78) );
  CKND2D0 U29 ( .A1(n56), .A2(n9), .ZN(n12) );
  INVD0 U30 ( .I(n10), .ZN(n11) );
  AOI21D0 U31 ( .A1(n78), .A2(n12), .B(n11), .ZN(n32) );
  INVD0 U32 ( .I(b[3]), .ZN(n70) );
  CKND2D0 U33 ( .A1(b[1]), .A2(n17), .ZN(n13) );
  OAI211D0 U34 ( .A1(b[1]), .A2(n17), .B(n78), .C(n13), .ZN(n14
     ) );
  CKXOR2D0 U35 ( .A1(n15), .A2(n14), .Z(n60) );
  INVD0 U36 ( .I(n60), .ZN(n59) );
  INR2D0 U37 ( .A1(n78), .B1(n36), .ZN(n16) );
  INVD0 U38 ( .I(a[1]), .ZN(n98) );
  MAOI22D0 U39 ( .A1(n17), .A2(n78), .B1(n16), .B2(n98), .ZN(
     n93) );
  MAOI222D0 U40 ( .A(b[1]), .B(n91), .C(n93), .ZN(n55) );
  INVD0 U41 ( .I(n55), .ZN(n57) );
  MAOI222D0 U42 ( .A(b[2]), .B(n59), .C(n57), .ZN(n31) );
  MAOI222D0 U43 ( .A(n32), .B(n70), .C(n31), .ZN(n88) );
  INVD0 U44 ( .I(select[1]), .ZN(n68) );
  CKND2D0 U45 ( .A1(select[2]), .A2(n68), .ZN(n96) );
  NR2D0 U46 ( .A1(select[2]), .A2(n68), .ZN(n79) );
  CKND2D0 U47 ( .A1(a[0]), .A2(n36), .ZN(n18) );
  CKND2D0 U48 ( .A1(n89), .A2(n18), .ZN(n22) );
  INVD0 U49 ( .I(n88), .ZN(n23) );
  AOI31D0 U50 ( .A1(select[2]), .A2(b[0]), .A3(n23), .B(select
     [1]), .ZN(n19)
         );
  AOI22D0 U51 ( .A1(n79), .A2(n22), .B1(a[0]), .B2(n19), .ZN(
     n20) );
  OAI31D0 U52 ( .A1(n89), .A2(n88), .A3(n96), .B(n20), .ZN(n21)
      );
  AOI31D0 U53 ( .A1(select[2]), .A2(select[1]), .A3(a[1]), .B(
     n21), .ZN(n29)
```

```
        );
INVD0 U54 ( .I(select[0]), .ZN(n102) );
NR2D0 U55 ( .A1(select[2]), .A2(select[1]), .ZN(n82) );
AOI22D0 U56 ( .A1(n79), .A2(n23), .B1(n82), .B2(n22), .ZN(n28
   ) );
INR2D0 U57 ( .A1(select[2]), .B1(n68), .ZN(n101) );
CKND2D0 U58 ( .A1(a[2]), .A2(n56), .ZN(n47) );
MAOI222D0 U59 ( .A(b[3]), .B(n75), .C(n47), .ZN(n26) );
CKND2D0 U60 ( .A1(a[1]), .A2(n90), .ZN(n81) );
CKND2D0 U61 ( .A1(b[1]), .A2(n98), .ZN(n80) );
ND4D0 U62 ( .A1(select[0]), .A2(a[0]), .A3(n36), .A4(n80), .
   ZN(n24) );
NR2D0 U63 ( .A1(a[3]), .A2(n70), .ZN(n30) );
NR2D0 U64 ( .A1(a[2]), .A2(n56), .ZN(n48) );
AOI211D0 U65 ( .A1(n81), .A2(n24), .B(n30), .C(n48), .ZN(n25)
    );
AOI32D0 U66 ( .A1(select[0]), .A2(n101), .A3(n26), .B1(n25),
   .B2(n101), .ZN(
      n27) );
IOA21D0 U67 ( .A1(n91), .A2(n81), .B(n80), .ZN(n53) );
AOI21D0 U68 ( .A1(n47), .A2(n53), .B(n48), .ZN(n71) );
AOI21D0 U69 ( .A1(a[3]), .A2(n70), .B(n30), .ZN(n42) );
INVD0 U70 ( .I(n42), .ZN(n39) );
INVD0 U71 ( .I(n82), .ZN(n97) );
AOI32D0 U72 ( .A1(n70), .A2(n32), .A3(n31), .B1(n88), .B2(n32
   ), .ZN(n33) );
OAI22D0 U73 ( .A1(n75), .A2(n97), .B1(n96), .B2(n33), .ZN(n34
   ) );
AOI31D0 U74 ( .A1(n79), .A2(n71), .A3(n39), .B(n34), .ZN(n45)
    );
CKND2D0 U75 ( .A1(b[0]), .A2(a[0]), .ZN(n83) );
INVD0 U76 ( .I(n83), .ZN(n84) );
MAOI222D0 U77 ( .A(b[1]), .B(a[1]), .C(n84), .ZN(n51) );
MAOI222D0 U78 ( .A(n62), .B(n56), .C(n51), .ZN(n69) );
INVD0 U79 ( .I(n69), .ZN(n40) );
INVD0 U80 ( .I(n79), .ZN(n85) );
OAI211D0 U81 ( .A1(a[3]), .A2(n36), .B(n35), .C(n90), .ZN(n37
   ) );
OAI22D0 U82 ( .A1(n85), .A2(n37), .B1(n62), .B2(n96), .ZN(n38
   ) );
AOI31D0 U83 ( .A1(n82), .A2(n40), .A3(n39), .B(n38), .ZN(n44)
    );
CKND2D0 U84 ( .A1(n82), .A2(n69), .ZN(n41) );
OAI32D0 U85 ( .A1(select[0]), .A2(n71), .A3(n85), .B1(n41), .
   B2(n102), .ZN(
      n73) );
CKND2D0 U86 ( .A1(n42), .A2(n73), .ZN(n43) );
INVD0 U87 ( .I(n96), .ZN(n77) );
```

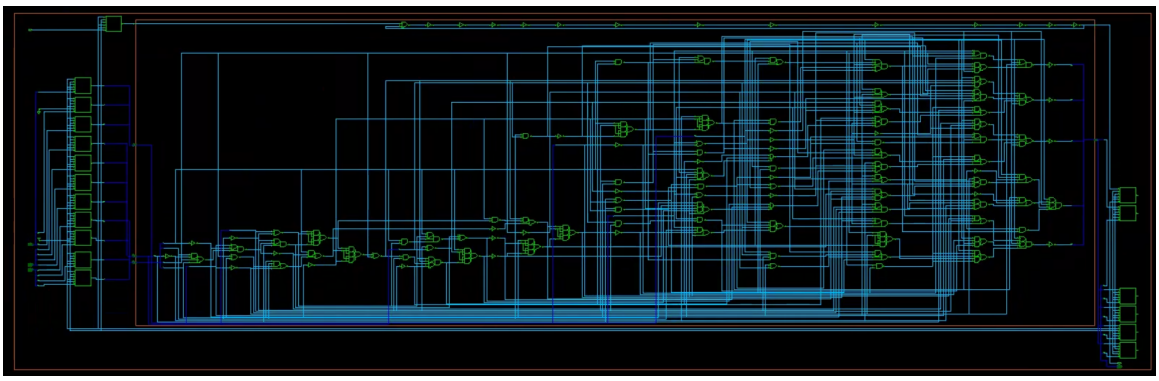
```
  AOI22D0 U88 ( .A1(n79), .A2(n46), .B1(a[1]), .B2(n77), .ZN(
     n67) );
  IND2D0 U89 ( .A1(n48), .B1(n47), .ZN(n54) );
  INVD0 U90 ( .I(n54), .ZN(n50) );
  CKND2D0 U91 ( .A1(n51), .A2(n50), .ZN(n49) );
  OAI211D0 U92 ( .A1(n51), .A2(n50), .B(n82), .C(n49), .ZN(n66)
      );
  NR2D0 U93 ( .A1(n54), .A2(n53), .ZN(n52) );
  AOI211D0 U94 ( .A1(n54), .A2(n53), .B(n85), .C(n52), .ZN(n64)
      );
  AOI221D0 U95 ( .A1(b[2]), .A2(n57), .B1(n56), .B2(n55), .C(
     n88), .ZN(n58) );
  MUX2ND0 U96 ( .I0(n60), .I1(n59), .S(n58), .ZN(n61) );
  OAI22D0 U97 ( .A1(n62), .A2(n97), .B1(n96), .B2(n61), .ZN(n63
     ) );
  AOI211D0 U98 ( .A1(n101), .A2(a[3]), .B(n64), .C(n63), .ZN(
     n65) );
  OAI31D0 U99 ( .A1(select[2]), .A2(b[3]), .A3(n69), .B(n68), .
     ZN(n76) );
  AOI211D0 U100 ( .A1(n71), .A2(n70), .B(a[3]), .C(n85), .ZN(
     n72) );
  AOI22D0 U101 ( .A1(b[3]), .A2(n73), .B1(n72), .B2(n102), .ZN(
     n74) );
  AOI22D0 U102 ( .A1(n79), .A2(n78), .B1(a[0]), .B2(n77), .ZN(
     n105) );
  CKND2D0 U103 ( .A1(n81), .A2(n80), .ZN(n87) );
  INVD0 U104 ( .I(n87), .ZN(n86) );
  OAI221D0 U105 ( .A1(n84), .A2(n87), .B1(n83), .B2(n86), .C(
     n82), .ZN(n104)
         );
  AOI221D0 U106 ( .A1(n91), .A2(n87), .B1(n89), .B2(n86), .C(
     n85), .ZN(n100)
         );
  INVD0 U107 ( .I(n93), .ZN(n94) );
  AOI221D0 U108 ( .A1(b[1]), .A2(n91), .B1(n90), .B2(n89), .C(
     n88), .ZN(n92)
         );
  MUX2ND0 U109 ( .I0(n94), .I1(n93), .S(n92), .ZN(n95) );
  OAI22D0 U110 ( .A1(n98), .A2(n97), .B1(n96), .B2(n95), .ZN(
     n99) );
  AOI211D0 U111 ( .A1(n101), .A2(a[2]), .B(n100), .C(n99), .ZN(
     n103) );
endmodule


module IO_ALU_with_ring_osc ( a, b, select, q, reset_ring_osc,
    out_ring_osc );
  input [3:0] a;
```

```
input [3:0] b;
input [2:0] select;
output [4:0] q;
input reset_ring_osc;
output out_ring_osc;
wire   \*Logic1* , \*Logic0* , reset_ring_osc_w,
   out_ring_osc_w;
wire   [3:0] a_w;
wire   [3:0] b_w;
wire   [2:0] select_w;
wire   [4:0] q_w;

PDDW0204SCDG U9 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
   Logic1* ), .PAD(
      reset_ring_osc), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(
         reset_ring_osc_w) );
PDDW0204SCDG U10 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
   Logic1* ), .PAD(
      a[0]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(a_w[0]) );
PDDW0204SCDG U11 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
   Logic1* ), .PAD(
      a[1]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(a_w[1]) );
PDDW0204SCDG U12 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
   Logic1* ), .PAD(
      a[2]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(a_w[2]) );
PDDW0204SCDG U13 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
   Logic1* ), .PAD(
      a[3]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(a_w[3]) );
PDDW0204SCDG U14 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
   Logic1* ), .PAD(
      b[0]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(b_w[0]) );
PDDW0204SCDG U15 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
   Logic1* ), .PAD(
      b[1]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(b_w[1]) );
PDDW0204SCDG U16 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
   Logic1* ), .PAD(
      b[2]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(b_w[2]) );
PDDW0204SCDG U17 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
   Logic1* ), .PAD(
      b[3]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(b_w[3]) );
PDDW0204SCDG U18 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
   Logic1* ), .PAD(
      select[0]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(
         select_w[0]) );
PDDW0204SCDG U19 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
   Logic1* ), .PAD(
      select[1]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(
         select_w[1]) );
```

```
  PDDW0204SCDG U20 ( .I(\*Logic0* ), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(
        select[2]), .DS(\*Logic0* ), .PE(\*Logic0* ), .C(
           select_w[2]) );
  PDDW0204SCDG A1 ( .I(out_ring_osc_w), .OEN(\*Logic0* ), .IE
     (\*Logic1* ),
        .PAD(out_ring_osc), .DS(\*Logic0* ), .PE(\*Logic0* ) );
  PDDW0204SCDG A2 ( .I(q_w[0]), .OEN(\*Logic0* ), .IE(\*Logic1*
      ), .PAD(q[0]),
        .DS(\*Logic0* ), .PE(\*Logic0* ) );
  PDDW0204SCDG A3 ( .I(q_w[1]), .OEN(\*Logic0* ), .IE(\*Logic1*
      ), .PAD(q[1]),
        .DS(\*Logic0* ), .PE(\*Logic0* ) );
  PDDW0204SCDG A4 ( .I(q_w[2]), .OEN(\*Logic0* ), .IE(\*Logic1*
      ), .PAD(q[2]),
        .DS(\*Logic0* ), .PE(\*Logic0* ) );
  PDDW0204SCDG A5 ( .I(q_w[3]), .OEN(\*Logic0* ), .IE(\*Logic1*
      ), .PAD(q[3]),
        .DS(\*Logic0* ), .PE(\*Logic0* ) );
  PDDW0204SCDG A6 ( .I(q_w[4]), .OEN(\*Logic0* ), .IE(\*Logic1*
      ), .PAD(q[4]),
        .DS(\*Logic0* ), .PE(\*Logic0* ) );
  ALU_with_ring_osc suma ( .a(a_w), .b(b_w), .select(select_w),
      .q(q_w),
        .reset_ring_osc(reset_ring_osc_w), .out_ring_osc(
           out_ring_osc_w) );
  TIEH U6 ( .Z(\*Logic1* ) );
  TIEL U7 ( .ZN(\*Logic0* ) );
endmodule
```

Puede notarse la aparición de las celdas TIEH y TIEL para conectar a VDD y VSS, respectivamente. El esquemático se presenta a continuación:

Figura 118. Esquemático de la síntesis de la ALU con *ring oscillator* con pads



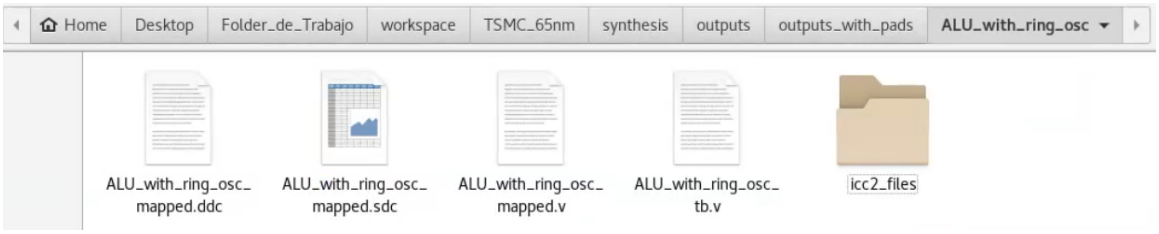
Nota. La imagen muestra el esquemático de la síntesis de la ALU con *ring oscillator* con pads en Design Vision.

Puede observarse en la Figura 118 el esquemático de la ALU con *ring oscillator* con pads a ambos lados. El diagrama del centro coincide con el del circuito sin pads (ver Subsección 11.5.3). Esto demuestra que el diseño es correcto.

11.5.7. Simulación del módulo del circuito sintetizado con pads

Antes de simular el diseño sintetizado con pads verificar que los archivos de salida se encuentren en su respectivo directorio.

Figura 119. Archivos de salida de la síntesis de la ALU con *ring oscillator* con pads



Nota. La imagen muestra archivos de salida de la síntesis de la ALU con *ring oscillator* con pads.

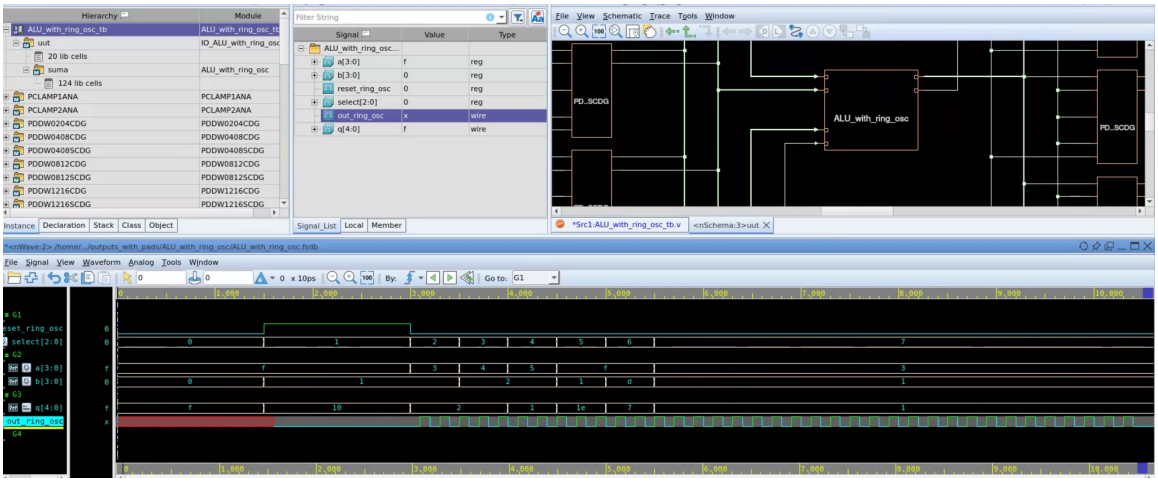
Posteriormente, ejecutar los comandos respectivos para simular en Verdi:

Cuadro 68. Comandos para la compilación y simulación de la ALU con *ring oscillator* con pads

```
vcs -full64 -debug_access+all -kdb ALU_with_ring_osc_tb.v
   ALU_with_ring_osc_mapped.v -v ../../tcbn65lp.v -v ../../
   tpdn65lpnv2od3.v
./simv
```

```
verdi -ssf ALU_with_ring_osc.fsdb
```

Figura 120. Simulación de la síntesis de la ALU con *ring oscillator* con pads



Nota. La imagen muestra la simulación de la síntesis de la ALU con *ring oscillator* con pads en Verdi.

La Figura 120 muestra la simulación del funcionamiento de la ALU con *ring oscillator* con pads. Puede notarse a través de las señales que el funcionamiento es el mismo al del circuito original. Por otro lado, el diagrama muestra que los pads se encuentran en ambos extremos del *core*. Esto semuestra que se realizó con éxito la síntesis de la ALU con *ring oscillator* con pads.

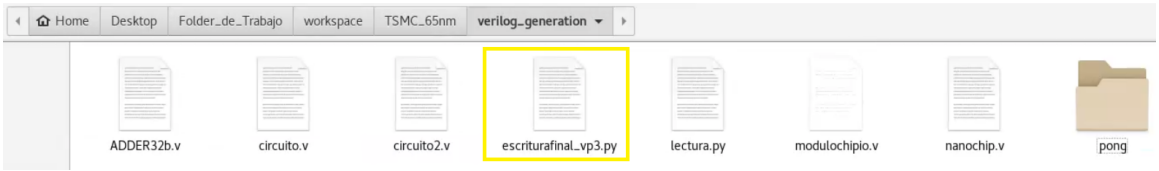
11.6. Síntesis lógica del nanochip "El Gran Jaguar"

En esta sección, se explica el proceso que se siguió para sintetizar el nanochip "El Gran Jaguar" con las librerías de TSMC para la tecnología de 65 nm. La nomenclatura para los archivos y carpetas propias para este caso es: `nanochip`.

11.6.1. Módulo *core* del circuito, *testbench* y módulo del circuito con pads

Primeramente, es importante contar con los archivos de entrada. El código del circuito *core* del nanochip se genera partir de un código de Python, el cual se encuentra en el Capítulo 18 de Anexos en la Sección 18.1. Este código se encuentra en el directorio llamado `verilog_generation/` (ver Capítulo 8).

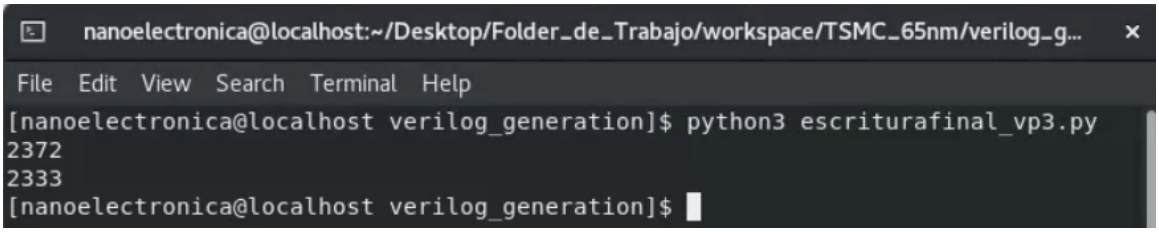
Figura 121. Archivo de Python para la generación del *core* del nanochip



Nota. La imagen muestra el archivo de Python para la generación del *core* del nanochip.

Luego de ubicar el archivo, abrir una terminal donde se encuentra el archivo y ejecutar lo siguiente:

Figura 122. Ejecución del Python para la generación del *core* del nanochip



Nota. La imagen muestra la ejecución del Python para la generación del *core* del nanochip desde la terminal.

De esta manera, se generó el archivo `nanochip.v` que se usa para la síntesis. Los números que aparecen en la Figura 122 son la cantidad de caracteres para los textos en español e inglés respectivamente. El archivo `nanochip.v` corresponde al código del diseño *core* del nanochip. Dicho archivo se incluye dentro de los entregables asociados a este trabajo, pero no se incorpora en este documento debido a su gran extensión.

A continuación, se muestra el *testbench* el Cuadro 69 y el nanochip con pads se muestra en el Cuadro 70.

Cuadro 69. Archivo nanochip_tb.v

```
//Test_Bench
// Las librerias de 65 nm poseen escala de tiempo, para simular
    todos los modulos deben tenerla
`timescale  1ns/10ps

module nanochip_tb;
reg clk, reset_nanochip, reset_ring_osc;
reg [1:0] select;
wire out_ring_osc;
wire [7:0] q_out;
```

```
nanochip chip(q_out, clk, reset_nanochip, select,
   reset_ring_osc, out_ring_osc);
initial
clk = 1'b0;
always
#1 clk=~clk;
initial begin
$fsdbDumpfile("nanochip.fsdb");
$fsdbDumpvars(0, nanochip_tb);
$dumpfile("nanochip.vcd");
$dumpvars(0, nanochip_tb);
reset_ring_osc = 1'b0;
reset_nanochip = 1'b0;
select = 2'b00;
#15; reset_nanochip =1'b1; reset_ring_osc = 1'b1;
#15; reset_nanochip =1'b0; reset_ring_osc = 1'b0;
//#15; reset_nanochip =1'b0; reset_ring_osc = 1'b0;
#5000; $finish;
end
endmodule
```

El nanochip codifica con texto ASCII tanto en español como en inglés, a través de la señal `select` en el *testbench* del Cuadro 69 es posible simular en el idioma que se indique.

En los casos que la señal `select` posee el valor de 2'b00 y 2'b11, el texto se codifica en español. Para los casos que la señal posee el valor de 2'b01 y 2'b10, el texto se codifica en inglés.

Cuadro 70. Archivo IO_nanochip.v

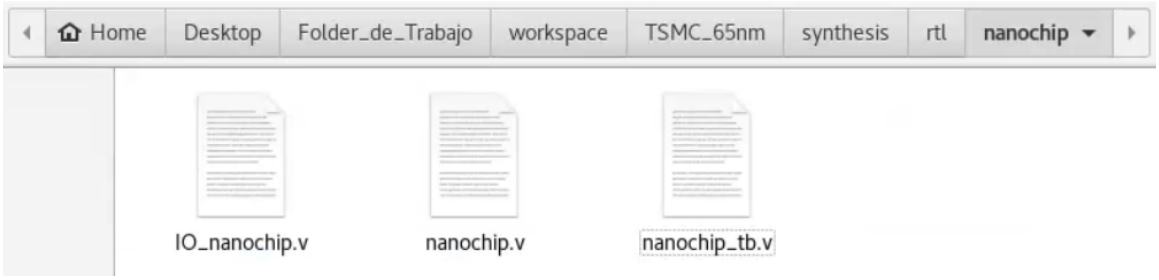
```
module IO_nanochip(q_out, clk, reset_nanochip, select,
   reset_ring_osc, out_ring_osc);

output [7:0] q_out;
output out_ring_osc;

input reset_nanochip;
input clk;
input [1:0]select;
input reset_ring_osc;

wire [7:0] q_out_w;
wire out_ring_osc_w;
wire reset_nanochip_w;
wire clk_w;
wire [1:0]select_w;
wire reset_ring_osc_w;

nanochip U10 (.q_out(q_out_w), .clk(clk_w), .reset_nanochip(
```

```
   reset_nanochip_w), .select(select_w), .reset_ring_osc(
   reset_ring_osc_w), .out_ring_osc(out_ring_osc_w));

// 14 pins de senales, 1 VDD y 1 VSS = 16
//Inputs (PAD->C):
PDDW0204SCDG IN0 (1'b0,1'b0,1'b1,reset_ring_osc,
   reset_ring_osc_w,1'b0,1'b1);
PDDW0204SCDG IN1 (1'b0,1'b0,1'b1,clk,clk_w,1'b0,1'b1);
PDDW0204SCDG IN2 (1'b0,1'b0,1'b1,reset_nanochip,
   reset_nanochip_w,1'b0,1'b1);
PDDW0204SCDG IN3 (1'b0,1'b0,1'b1,select[0],select_w[0],1'b0,1'
   b1);
PDDW0204SCDG IN4 (1'b0,1'b0,1'b1,select[1],select_w[1],1'b0,1'
   b1);
//Outputs (I->PAD):
PDDW0204SCDG OUT0 (out_ring_osc_w,1'b0,1'b0,out_ring_osc,1'bx
   ,1'b0,1'b1);
PDDW0204SCDG OUT1 ( q_out_w[0],1'b0,1'b0, q_out[0],1'bx,1'b0,1'
   b1);
PDDW0204SCDG OUT2 ( q_out_w[1],1'b0,1'b0, q_out[1],1'bx,1'b0,1'
   b1);
PDDW0204SCDG OUT3 ( q_out_w[2],1'b0,1'b0, q_out[2],1'bx,1'b0,1'
   b1);
PDDW0204SCDG OUT4 ( q_out_w[3],1'b0,1'b0, q_out[3],1'bx,1'b0,1'
   b1);
PDDW0204SCDG OUT5 ( q_out_w[4],1'b0,1'b0, q_out[4],1'bx,1'b0,1'
   b1);
PDDW0204SCDG OUT6 ( q_out_w[5],1'b0,1'b0, q_out[5],1'bx,1'b0,1'
   b1);
PDDW0204SCDG OUT7 ( q_out_w[6],1'b0,1'b0, q_out[6],1'bx,1'b0,1'
   b1);
PDDW0204SCDG OUT8 ( q_out_w[7],1'b0,1'b0, q_out[7],1'bx,1'b0,1'
   b1);

endmodule
```

11.6.2. Simulación del módulo *core* del circuito

Cuanta más complejidad exista en un diseño, más importante se convierte sintetizar primero el módulo *core* antes que el circuito con pads. Antes de simular el *core* del nanochip previo a sintetizar, verificar que los archivos se encuentren en la carpeta correcta.

Figura 123. Archivos de entrada para el nanochip "El Gran Jaguar"



Nota. La imagen muestra los archivos necesarios para la síntesis del nanochip "El Gran Jaguar".

Además, como se mencióno con el caso del *ring oscillator* es importante agregar el respectivo *delay* para que se pueda simular.

Figura 124. *Delay* para simular el *ring oscillator* del nanochip "El Gran Jaguar"

```
1 module nanochip(q_out, clk, reset_nanochip, select, reset_ring_osc, out_ring_osc);
2
3 // 14 pins de señales, 1 VDD y 1 VSS = 16
4 input clk, reset_nanochip, reset_ring_osc;
5 input [1:0] select;
6 output out_ring_osc;
7 output [7:0] q_out;
8
9 reg [7:0] q;
10
11 // Contador que llega hasta 4096 caracteres
12 reg [11:0] counter;
13
14 // Wires para el ring oscillator (agregar delay a wires para la simulacion e.g. #1)
15 wire /*#1*/ W_1;
16 wire /*#1*/ W_2;
17 wire /*#1*/ W_3;
18 wire /*#1*/ W_4;
19 wire /*#1*/ W_5;
20 wire /*#1*/ W_6;
21 wire /*#1*/ W_7;
22 wire /*#1*/ W_8;
23 wire /*#1*/ W_9;
24 wire /*#1*/ W_10;
25 wire /*#1*/ W_11;
26 wire /*#1*/ W_12;
27 wire /*#1*/ W_13;
28
29 // Ring oscillator de 13 NOTs
30 assign #1 W_1 = reset_ring_osc | out_ring_osc;
31 assign W_2 = ~W_1;
32 assign W_3 = ~W_2;
33 assign W_4 = ~W_3;
34 assign W_5 = ~W_4;
35 assign W_6 = ~W_5;
36 assign W_7 = ~W_6;
37 assign W_8 = ~W_7;
38 assign W_9 = ~W_8;
39 assign W_10 = ~W_9;
40 assign W_11 = ~W_10;
41 assign W_12 = ~W_11;
42 assign W_13 = ~W_12;
43 assign out_ring_osc = ~W_13;
```

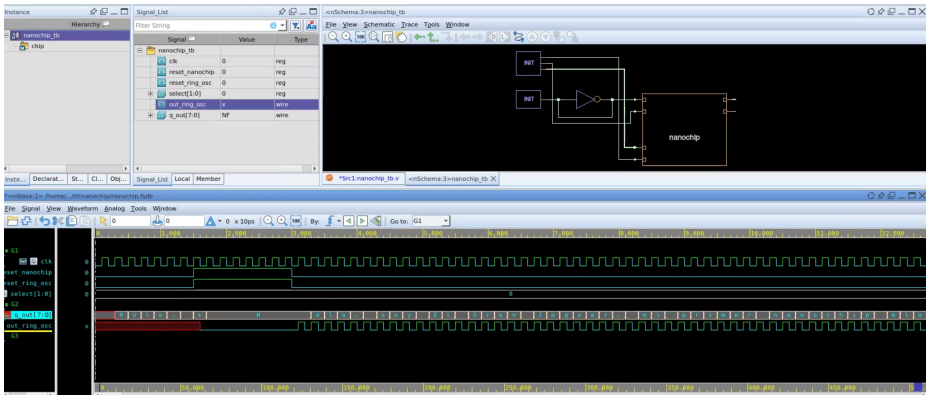
Nota. La imagen muestra el *delay* para simular el *ring oscillator* del nanochip "El Gran Jaguar".

Posteriormente, ejecutar los respectivos comandos para simular el *core* del nanochip previo a la síntesis en Verdi:

Cuadro 71. Comandos para la compilación y simulación del *core* del nanochip

```
vcs -full64 -debug_access+all -kdb nanochip_tb.v nanochip.v
./simv
verdi -ssf nanochip.fsdb
```

Figura 125. Simulación del *core* del nanochip



Nota. La imagen muestra la simulación del *core* del nanochip en Verdi.

Como se puede observar en la Figura 125, el texto se encuentra codificado en español en este caso. Además, el *ring oscillator* se encuentra funcionando y oscilando.

Ahora, para simular el circuito en el caso que se desee ver el texto en inglés es posible modificar el *textbench* desde Verdi. A continuación se describe cómo:

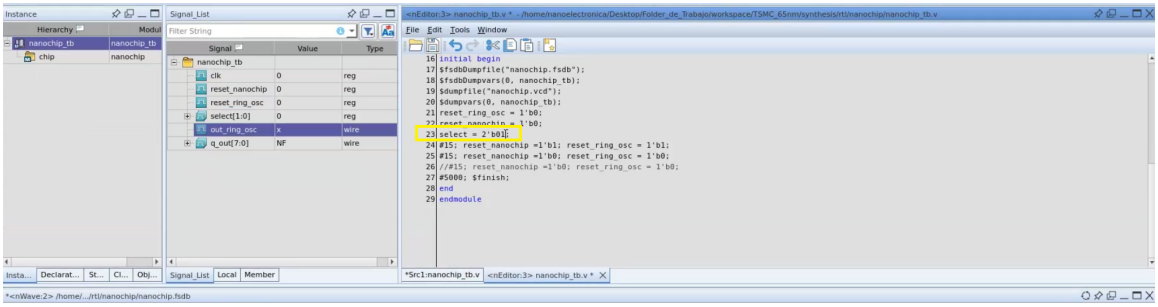
Primero hacer *click* en la opción *Edit Source File* (encerrada por el recuadro amarillo):

Figura 126. Opción para editar el *testbench* del nanochip desde Verdi

Nota. La imagen muestra la opción para editar el *testbench* del nanochip desde Verdi.

Luego, modificar la señal `select` y guardar, de forma que se codifique ahora en inglés:

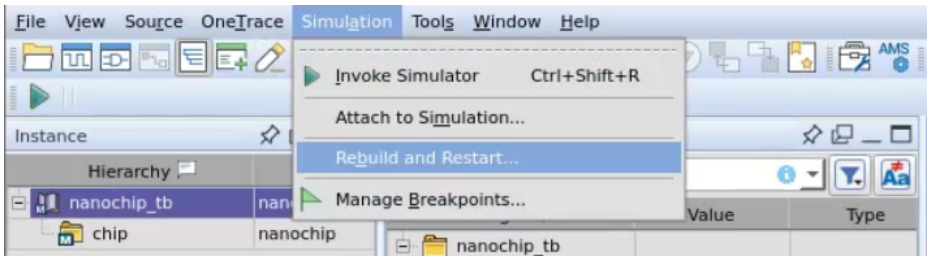
Figura 127. Edición de la señal `select` del *testbench* del nanochip



Nota. La imagen muestra la edición de la señal `select` del *testbench* del nanochip desde Verdi.

Hacer *click* en *Simulation*, y luego en *Rebuild and Restart*.

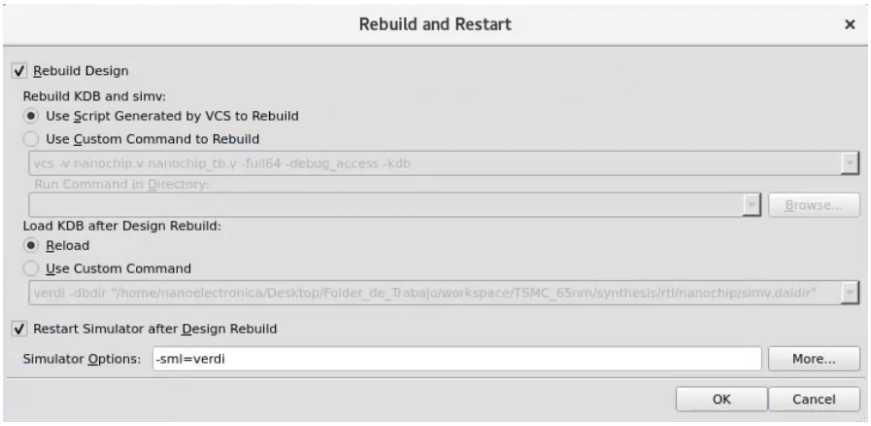
Figura 128. Opción para volver a simular el *testbench* del nanochip



Nota. La imagen muestra la opción para volver a simular el *testbench* del nanochip desde Verdi.

Verificar que se utilicen las siguientes opciones y hacer *click* en OK.

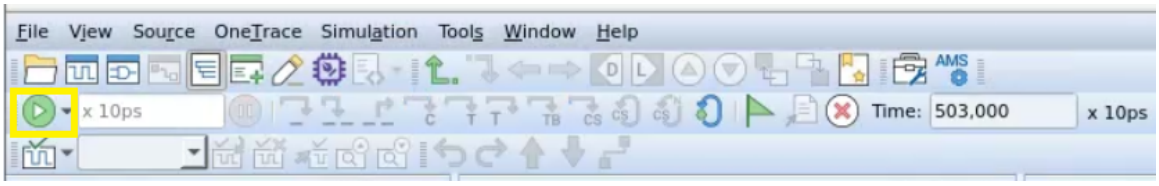
Figura 129. Configuración para volver a simular el *testbench* del nanochip



Nota. La imagen muestra la configuración para volver a simular el *testbench* del nanochip desde Verdi.

Se muestra un *prompt* y se hace *click* en *Yes*. Luego, hacer *click* en el siguiente botón encerrado en el recuadro amarillo:

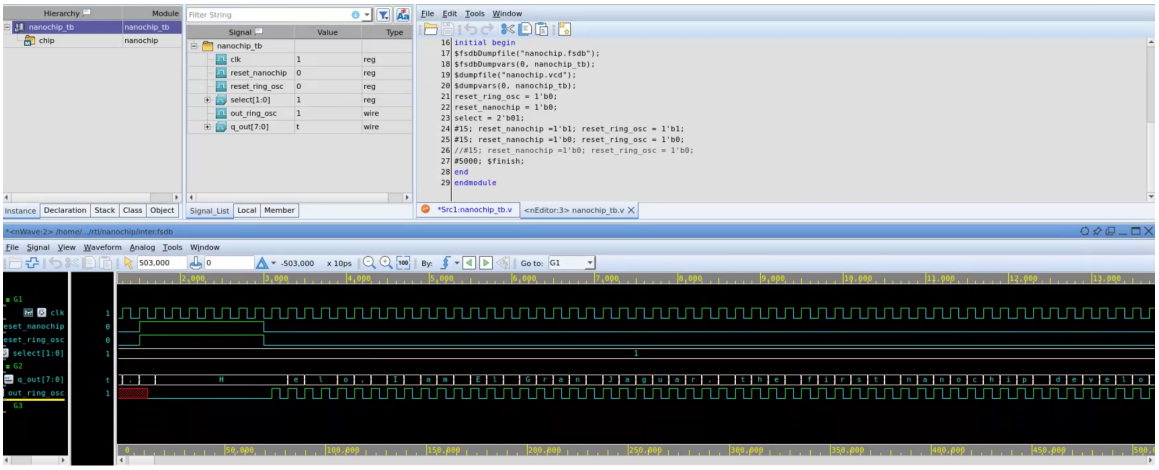
Figura 130. Botón para volver a simular el *testbench* del nanochip



Nota. La imagen muestra el botón para volver a simular el *testbench* del nanochip desde Verdi.

De esta forma aparece la nueva simulación del *core* del nanochip en inglés:

Figura 131. Simulación del *core* del nanochip previa a síntesis en inglés



Nota. La imagen muestra la simulación del *core* del nanochip previa a síntesis en inglés en Verdi.

11.6.3. Síntesis del módulo *core* del circuito

Luego de validar el funcionamiento del *core* del nanochip, se procedió con la síntesis del mismo. Para ello primero modificar los parámetros del *script*.

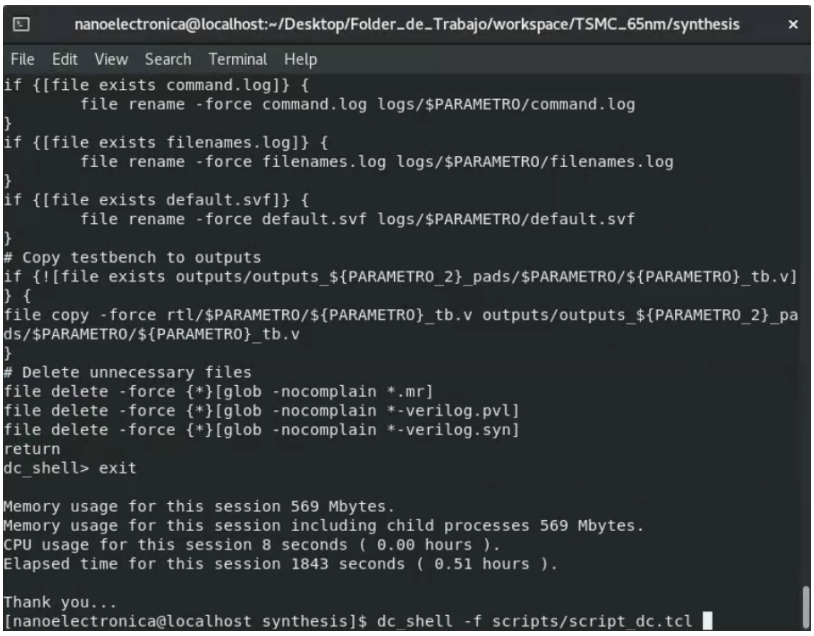
Figura 132. Parámetros para la síntesis del *core* del nanochip

```
# Author: Carlos Julio Valdes Oajaca
# Date: July 2025

# Parameters the user should modify
# PARAMETRO -> name of the circuit and instance (verilog file and instance must have the same name)
# USER_OPTION -> select between 1 and 2 only
# 1 -> synthesis without PADS
# 2 -> synthesis with PADS

set PARAMETRO "nanochip"
set USER_OPTION 1
```

Nota. La imagen muestra los parámetros para la síntesis del *core* del nanochip.

Ejecutar el *script*:

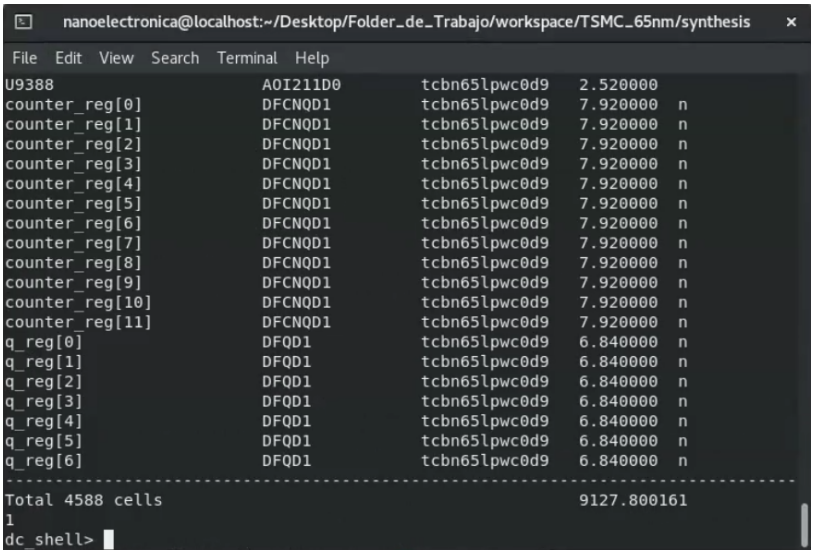
Figura 133. Ejecución del *script* desde la terminal para realizar la síntesis del *core* del nanochip



Nota. La imagen muestra la ejecución del *script* desde la terminal.

Para este caso la síntesis toma más tiempo ya que se trata de un circuito más complejo. Luego reportar las celdas que fueron mapeadas:

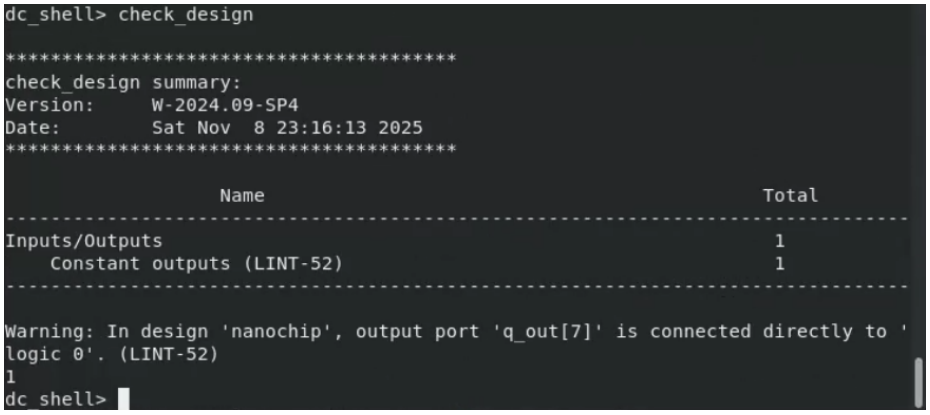
Figura 134. Celdas estándar mapeadas de la síntesis del *core* del nanochip



Nota. La imagen muestra las celdas mapeadas de la síntesis del *core* del nanochip.

Buscar si hay advertencias o errores en el diseño sintetizado:

Figura 135. Advertencias y/o errores de la síntesis del *core* del nanochip

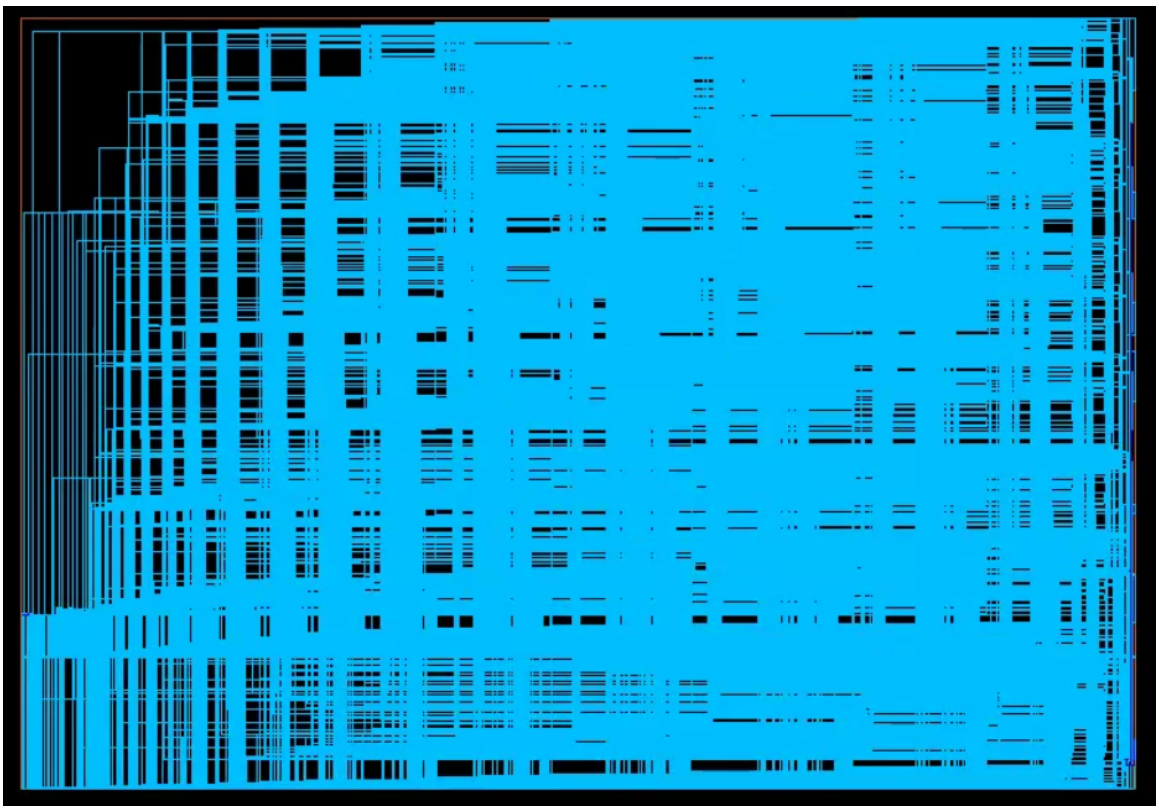


Nota. La imagen muestra las advertencias y/o errores de la síntesis del *core* del nanochip.

La Figura 135 muestra una advertencia acerca del último bit de la salida `q`. Sucede que este último siempre se encuentra con un valor de 1'b0, por lo que Design Compiler lo conecta al voltaje de referencia (0). La herramienta muestra la advertencia para que el diseñador esté enterado. En este caso, el funcionamiento es de esta forma, por lo que no hay ningún problema.

El *gate-level netlist* del *core* del nanochip se encuentra en la Subsección 18.2.1 de Anexos. El esquemático de dicho nanochip en Design Vision se muestra a continuación:

Figura 136. Esquemático del *gate-level netlist* del *core* del nanochip

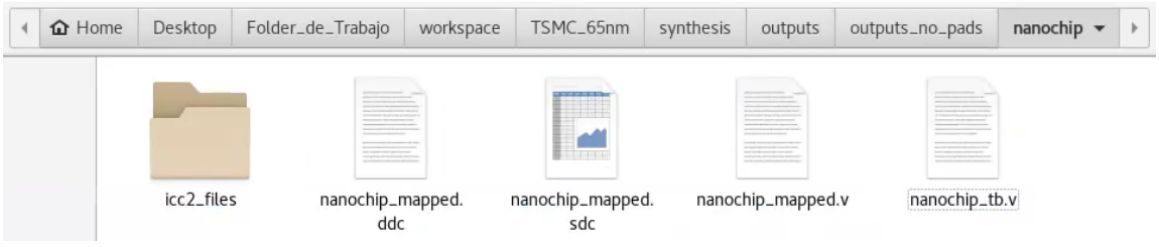


Nota. La imagen muestra el esquemático del *gate-level netlist* del *core* del nanochip en Design Vision.

11.6.4. Simulación del *core* del circuito sintetizado

Luego de la validación de la síntesis se simula el *gate-level netlist* del *core* del nanochip. Primero revisar que los archivos de salida se encuentren en su lugar.

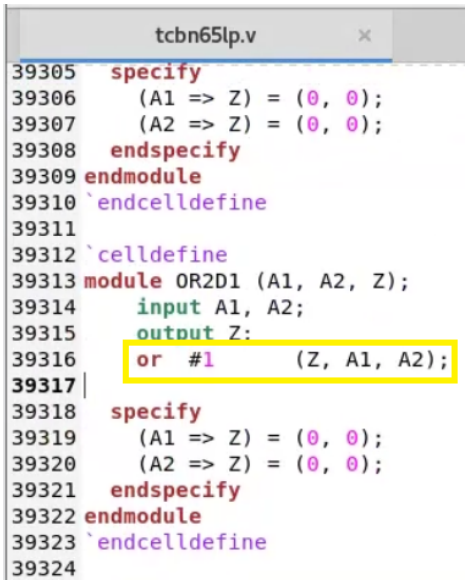
Figura 137. Archivos de salida de la síntesis del *core* del nanochip



Nota. La imagen muestra los archivos de salida de la síntesis del *core* del nanochip.

No olvidar que para el caso del *gate-level netlist* con un *ring oscillator* es necesario agregar el *delay* en el archivo de celdas estándar. En este caso, el nanochip contiene uno. Se debe modificar la celda OR2D1:

Figura 138. Adición de *delay* a la celda OR del diseño sintetizado del *core* del nanochip



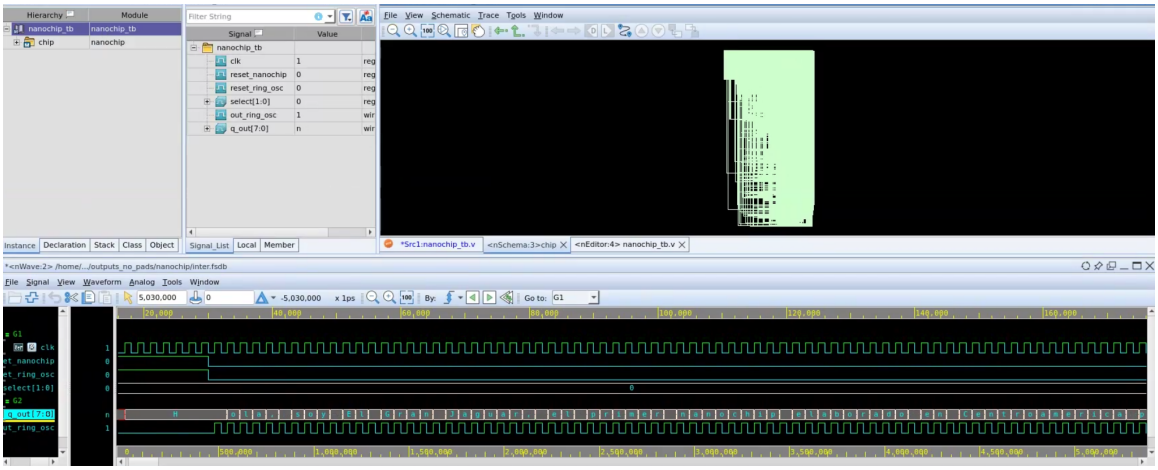
Nota. La imagen muestra la adición de *delay* a la compuerta OR2D1.

Posteriormente, ejecutar los siguientes comandos y simular en Verdi, desde una terminal donde se encuentren los archivos de salida:

Cuadro 72. Comandos para la compilación y simulación de la NOT con pads

```
vcs -full64 -debug_access+all -kdb nanochip_tb.v
   nanochip_mapped.v -v ../../tcbn65lp.v
./simv
verdi -ssf nanochip.fsdb
```

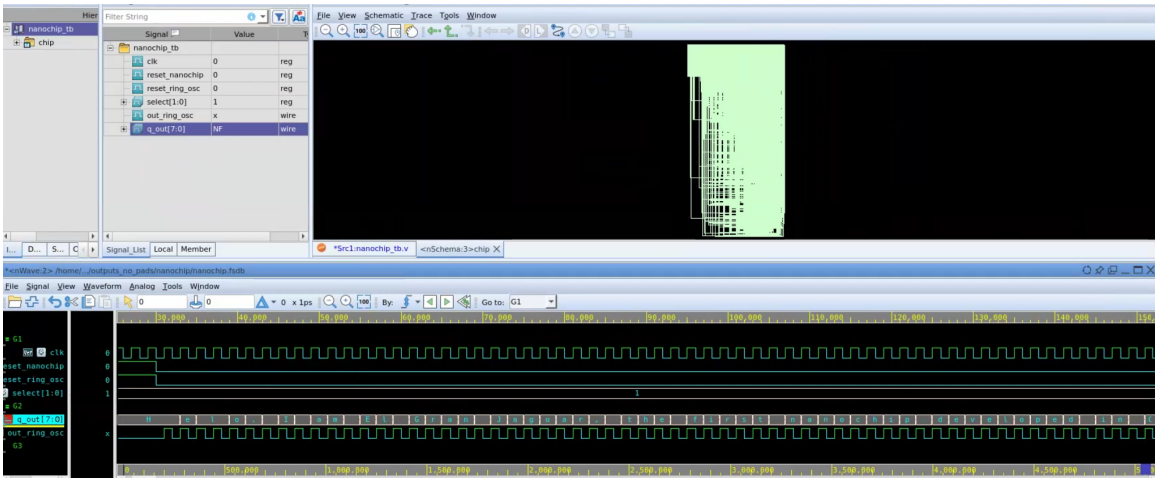
Figura 139. Simulación en español del *gate-level netlist* del *core* del nanochip



Nota. La imagen muestra la simulación del *gate-level netlist* del *core* del nanochip.

Luego, simular también para el caso que se desea codificar el texto en inglés. Puede utilizarse el mismo método que se usó previamente para modificar el *testbench* desde Verdi, o bien modificar el archivo `nanochip_tb.v` directamente.:

Figura 140. Simulación en inglés del *gate-level netlist* del *core* del nanochip



Nota. La imagen muestra la simulación del *gate-level netlist* del *core* del nanochip.

Como se puede observar el texto se codifica de forma correcta y el *ring oscillator* se comporta de forma esperada.

11.6.5. Simulación del módulo del circuito con pads

Luego de haber realizado la síntesis del *core* del nanochip se comienza el proceso para sintetizar el nanochip con pads. Primero se debe simular el nanochip con pads previo a la síntesis para verificar la validez del código, recordar que se modifica el *testbench* para instanciar el archivo con pads.

Figura 141. Modificación del *testbench* para simular el nanochip con pads

```
1 //Test_Bench
2 // Las librerias de 65 nm poseen escala de tiempo, para simular todos los modulos deben tenerla
3 `timescale  1ns/10ps
4
5 module nanochip_tb;
6 reg clk, reset_nanochip, reset_ring_osc;
7 reg [1:0] select;
8 wire out_ring_osc;
9 wire [7:0] q_out;
10
11 IO_nanochip chip(q_out, clk, reset_nanochip, select, reset_ring_osc, out_ring_osc);
12 initial
13 clk = 1'b0;
14 always
15 #1 clk=~clk;
```

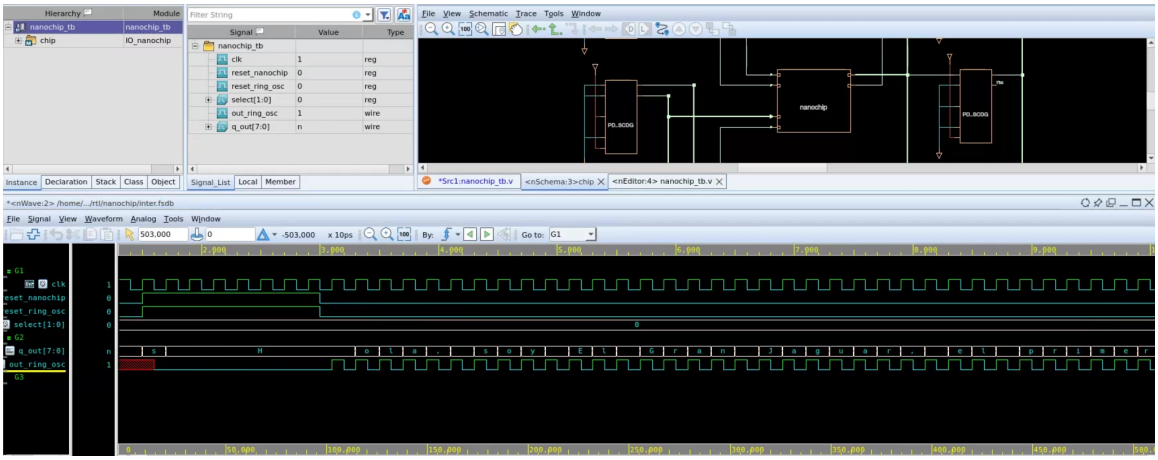
Nota. La imagen muestra la modificación del *testbench* para simular el nanochip con pads.

De esta manera ya se pueden ejecutar los siguientes comandos para simular en Verdi:

Cuadro 73. Comandos para la compilación y simulación del nanochip con pads

```
vcs -full64 -debug_access+all -kdb nanochip_tb.v IO_nanochip.v
   nanochip.v -v ../../outputs/tpdn65lpnv2od3.v
./simv
verdi -ssf nanochip.fsdb
```

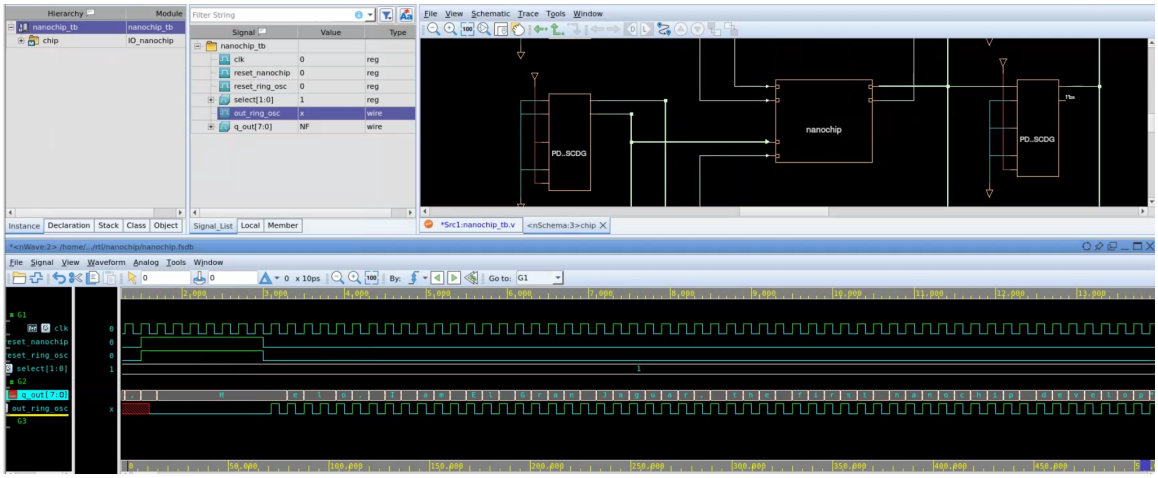
Figura 142. Simulación en español del nanochip con pads previo a síntesis



Nota. La imagen muestra la simulación en español del nanochip con pads previo a síntesis en Verdi.

Simular también para verificar el caso para el cual se codifica el texto en inglés:

Figura 143. Simulación en inglés del nanochip con pads previo a síntesis



Nota. La imagen muestra la simulación en inglés del nanochip con pads previo a síntesis.

Puede notarse en ambas figuras previas que el nanochip funciona correctamente. Se codifica el texto para el idioma correspondiente de forma esperada, de igual manera el *ring oscillator* funciona sin problema.

11.6.6. Síntesis del módulo del circuito con pads

El siguiente paso es realizar la síntesis del nanochip con pads. Antes de ejecutar el *script* modificar el segundo parámetro del mismo para realizar la síntesis con pads.

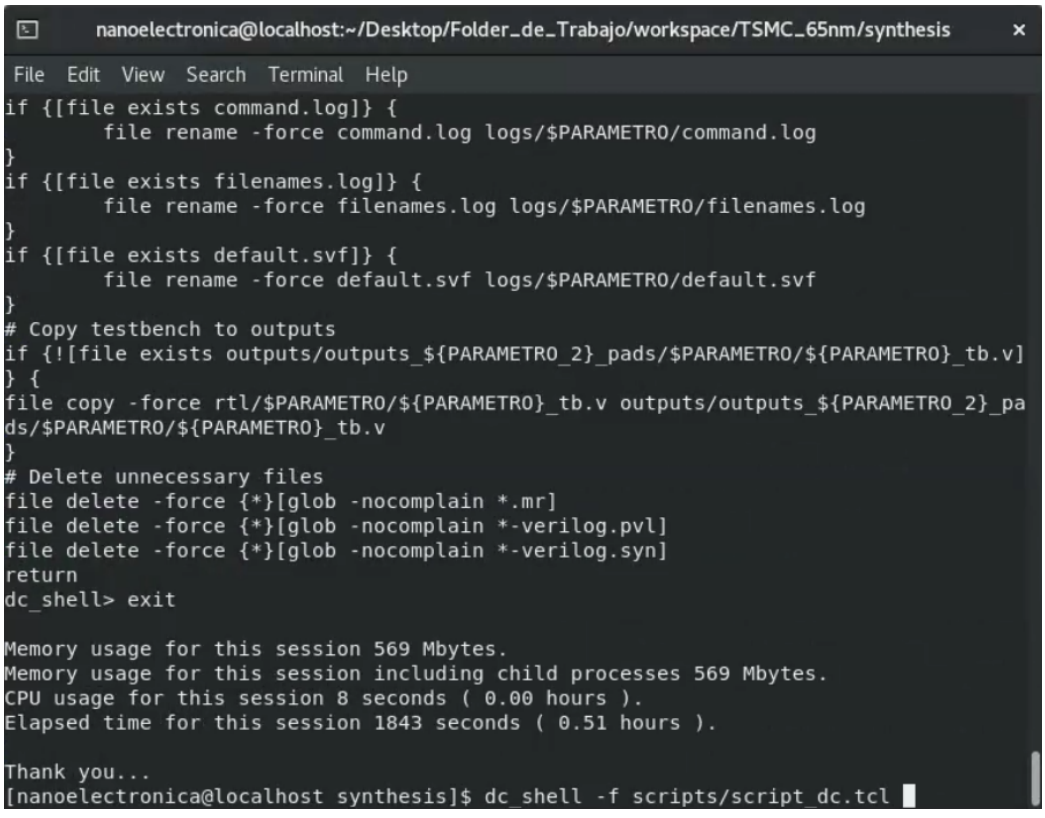
Figura 144. Cambio en el *script* para la síntesis del nanochip con pads

```
# Author: Carlos Julio Valdes Oajaca
# Date: July 2025

# Parameters the user should modify
# PARAMETRO -> name of the circuit and instance (verilog file and instance must have the same name)
# USER_OPTION -> select between 1 and 2 only
# 1 -> synthesis without PADS
# 2 -> synthesis with PADS

set PARAMETRO "nanochip"
set USER_OPTION 2

```

Nota. La imagen muestra el cambio en el *script* para la síntesis del nanochip con pads.

Luego ejecutar el *script*:

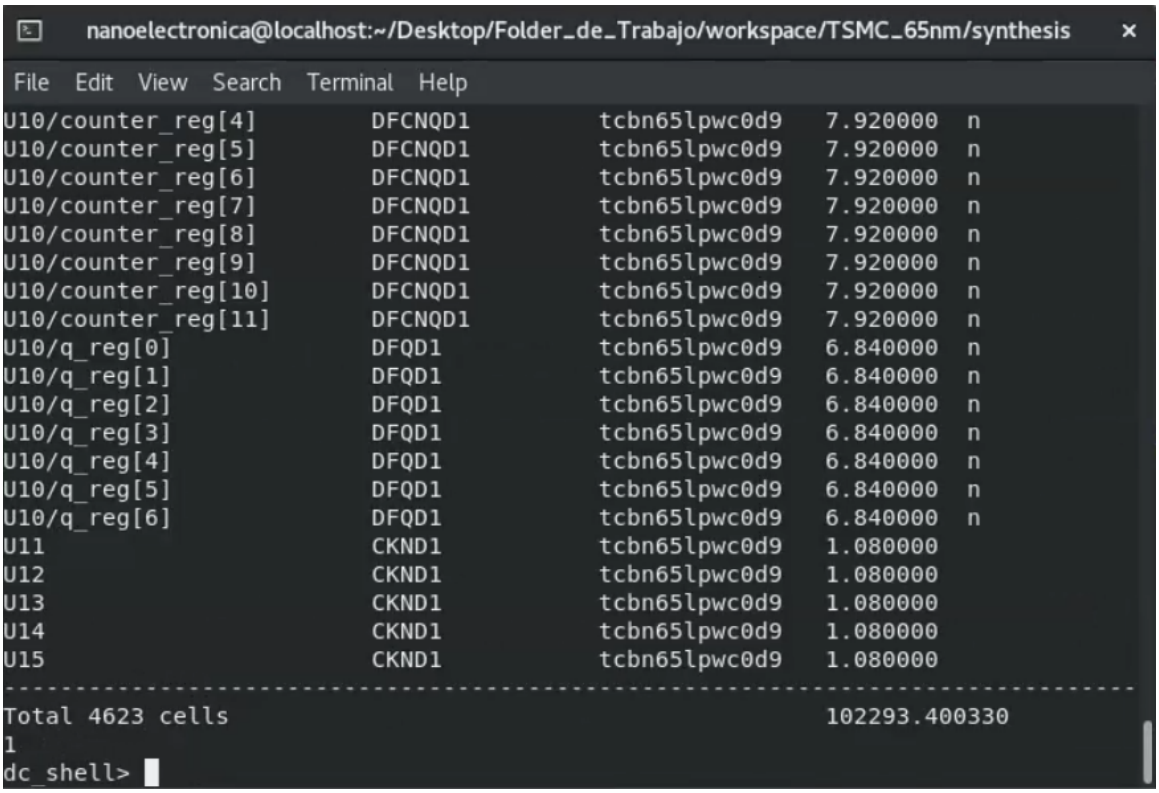
Figura 145. Ejecución del *script* desde la terminal para realizar la síntesis del nanochip con pads



Nota. La imagen muestra la ejecución del *script* desde la terminal.

Reportar las celdas estándar mapeadas:

Figura 146. Celdas estándar de la síntesis del nanochip con pads

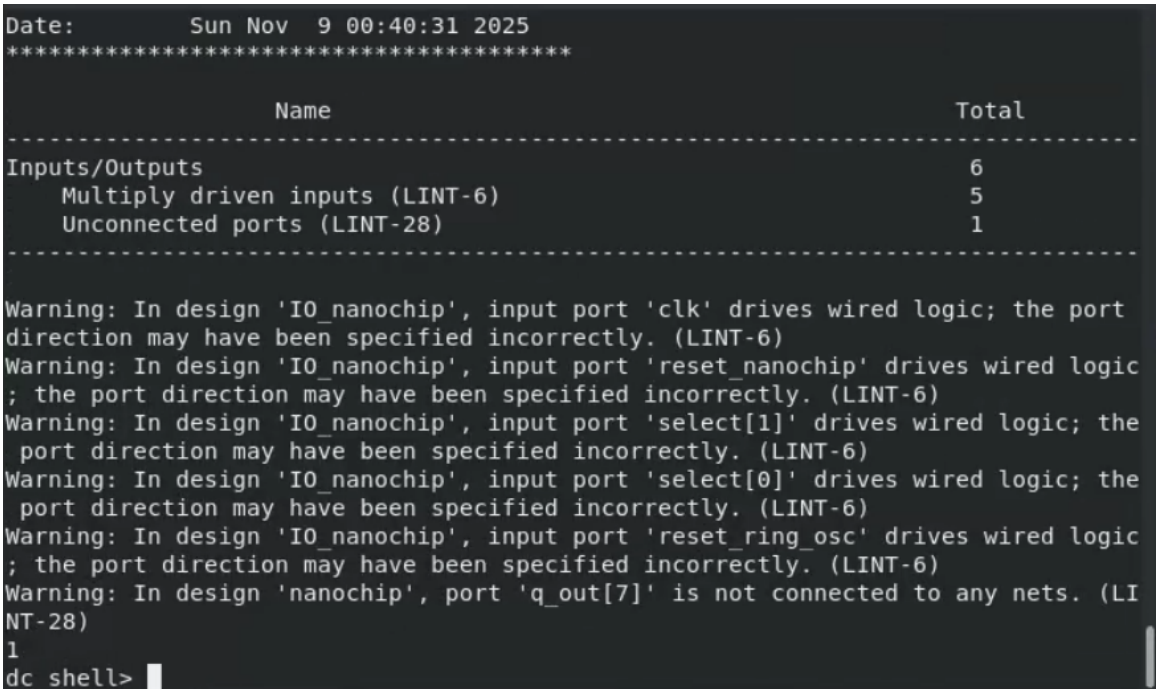


Nota. La imagen muestra las celdas estándar de la síntesis del nanochip con pads.

Como se ha mencionado con circuitos anteriores, se recomienda ver los reportes en el directorio `reports/`, donde hay información completa y detallada acerca de la síntesis del diseño.

Averiguar si existen advertencias y/o errores en el diseño sintetizado:

Figura 147. Advertencias y/o errores de la síntesis del nanochip con pads



Nota. La imagen muestra las advertencias y/o errores de la síntesis del nanochip con pads.

La Figura 147 muestra advertencias por *wired logic* y el último bit de la salida `q`. Como se ha visto anteriormente, este tipo de advertencias no representan ningún problema para el diseño. De igual forma se recomienda analizar las advertencias sea cuales sean para cerciorarse acerca de la validez de la síntesis realizada.

El *gate-level netlist* del nanochip con pads se encuentra en la Subsección 18.2.2 de los Anexos. Al igual que en los circuitos anteriores aparecen las celdas TIEH y TIEL para alimentar con VDD y VSS las señales del circuito que así lo requieran.

Al ser el nanochip el circuito alrededor del cual gira este trabajo, se presentan a continuación algunos detalles importantes del diseño sintetizado con pads:

Cuadro 74. Área y cantidad de celdas del nanochip con pads sintetizado

Tipo	**Cantidad de celdas**	**Área (μm^2)**
Combinacionales	4589	9008.64
Secuenciales	19	142.92
Macro/Black Box	14	84000
Total	4622	93151.56

Al comparar el Cuadro 74 con la Figura 146, se puede notar una diferencia en el total de celdas. Esto se debe a que en la Figura 146 se contabiliza la instancia del nanochip como

una celda más, por lo que en realidad el total de celdas que compone a "El Gran Jaguar" sintetizado con pads en 65 nm es de 4622 celdas.

Mencionar que la mayor parte de área ocupada se debe a los pads (*Macro/Black Box*), los cuales deben poseer un área mayor para poder ser conectados a través del *wirebonding*. Además, al tener un área mayor, los transistores internos son más grandes, lo que permite que los pads suministren una potencia mayor al exterior.

El total del área posterior a la síntesis lógica del nanochip con pads llega a ser aproximadamente $0.1mm^2$. Lo cual es alrededor del 10 % del área total disponible para el *die* ($1mm^2$).

Cuadro 75. Consumo de potencia del nanochip con pads sintetizado

Tipo	***Internal Power* (mW)**	***Switching Power* (mW)**	***Leakage Power* (nW)**	***Total Power* (mW)**
Combinacionales	4.4853e-02	4.8126e-02	951.3156	9.3930e-02
Secuenciales	5.8250e-03	3.9229e-04	20.3402	6.2376e-03
IO Pads	8.0677	5.3425	330.12	13.4105
Total	8.1183	5.3911	1301.8	13.5107

Por otro lado, al observar el Cuadro 75, se puede notar que la mayor parte del consumo de potencia se debe a los pads. Esto es lógico, ya que operan a un voltaje mayor que las celdas internas del nanochip y además deben sumministrar una potencia mayor para interactuar con el exterior.

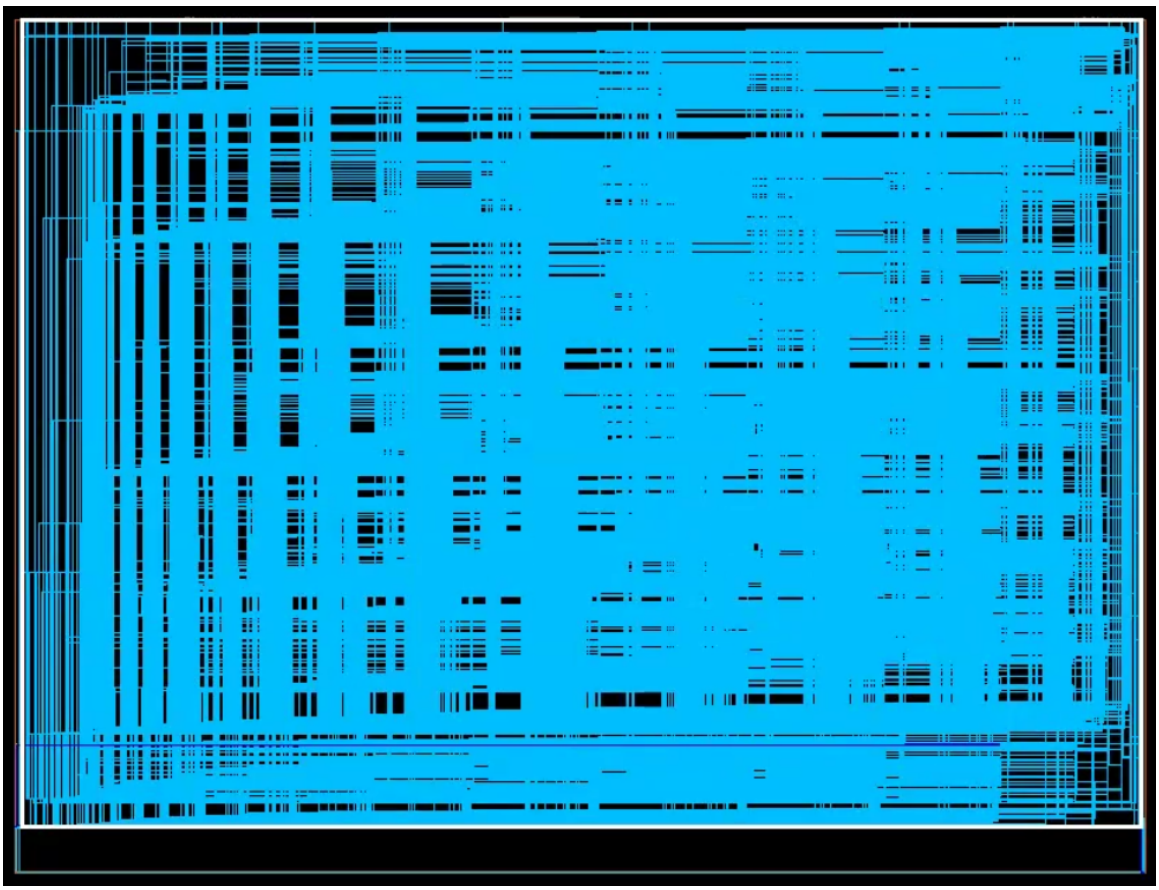
Tanto en pads como en celdas secuenciales, la potencia interna es mayor a la de conmutación. Esto se debe a que estas celdas esta compuestas por más transistores internamente, lo que genera mayor consumo interno.

En cambio, las celdas combinacionales poseen una potencia de conmutación mayor a la interna, ya que la red de estas celdas es más compleja y extensa, lo que agrega mayor capacitancia y por ende mayor consumo de potencia al conmutar.

La potencia de fuga es mayor en las celdas combinacionales, ya que a pesar de que la potencia de fuga de cada celda sea baja, el gran número de celdas provoca que esta potencia sea más significativa.

Como último punto, se presenta el esquemático con Design Vision:

Figura 148. Esquemático del *gate-level netlist* del nanochip con pads



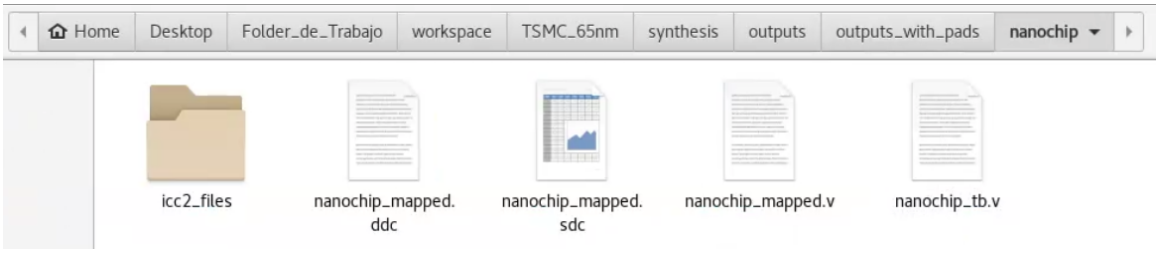
Nota. La imagen muestra el esquemático del *gate-level netlist* del nanochip con pads en Design Vision.

11.6.7. Simulación del módulo del circuito sintetizado con pads

Finalmente, el último paso es verificar que el *gate-level netlist* del nanochip con pads funcione como se espera.

Para ello, primero verificar que los archivos de salida se encuentren en su lugar:

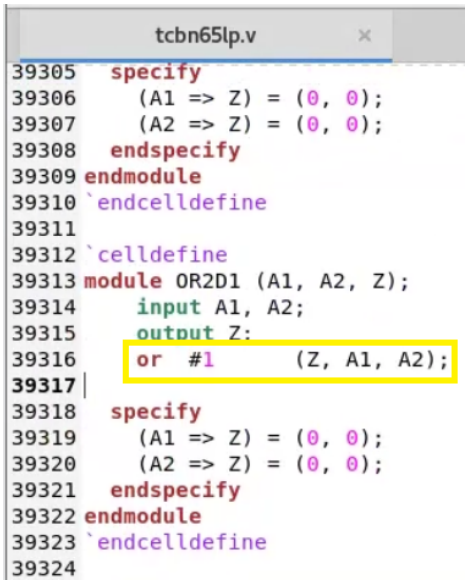
Figura 149. Archivos de salida por la síntesis del nanochip con pads



Nota. La imagen muestra los archivos de salida por la síntesis del nanochip con pads.

Recordar que es necesario agregar el *delay* en el archivo de celdas estándar. Se debe modificar la celda OR2D1:

Figura 150. Adición de *delay* a la celda OR del diseño sintetizado del nanochip con pads



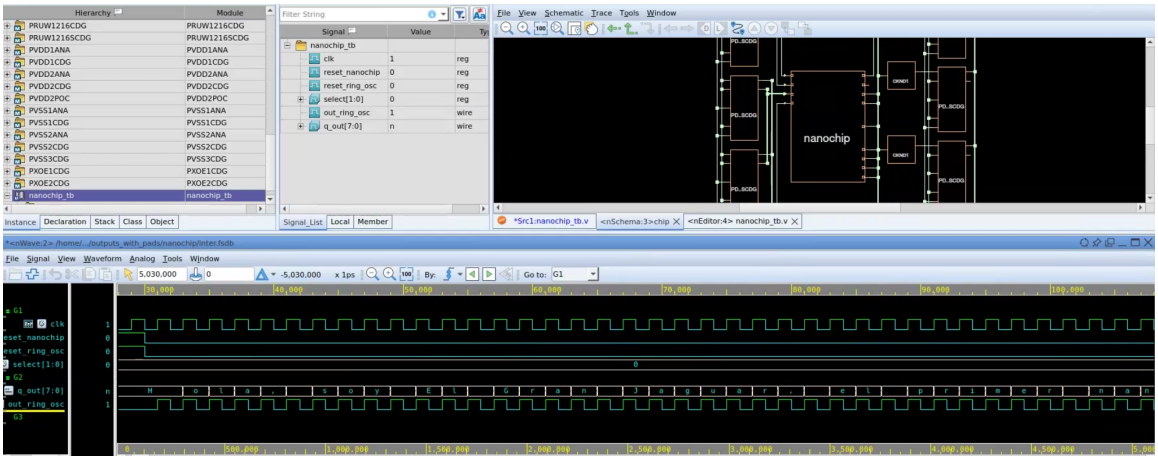
Nota. La imagen muestra la adición de *delay* a la compuerta OR2D1.

Ejecutar los respectivos comandos y simular en Verdi:

Cuadro 76. Comandos para la compilación y simulación del nanochip con pads

```
vcs -full64 -debug_access+all -kdb naochip_tb.v nanochip_mapped
   .v -v ../../tcbn65lp.v -v ../../tpdn65lpnv2od3.v
./simv
verdi -ssf nanochip.fsdb
```

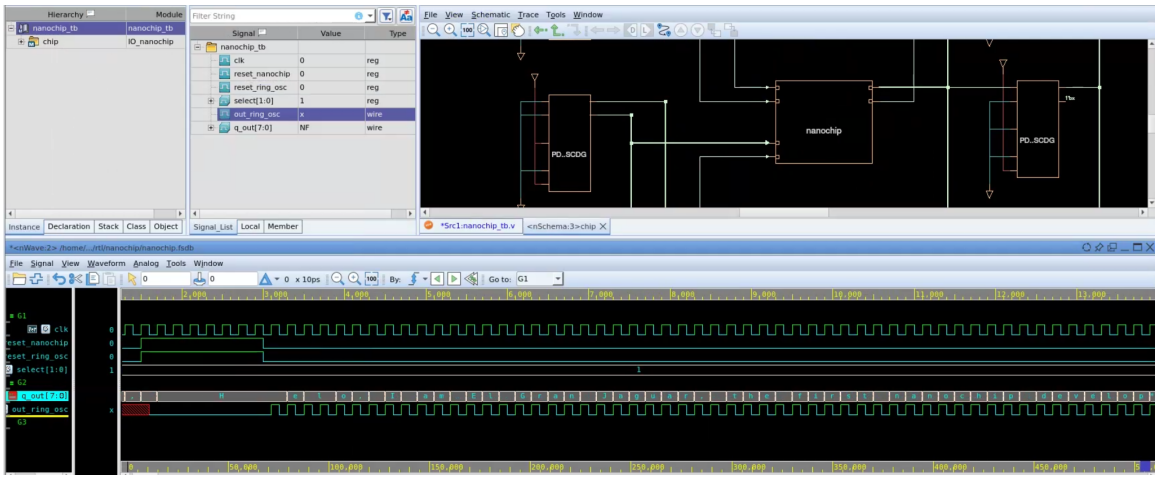
Figura 151. Simulación del *gate-level netlist* en español del nanochip con pads



Nota. La imagen muestra la simulación del *gate-level netlist* en español del nanochip con pads en Verdi.

Simlar también para el caso donde el texto se codifica en inglés:

Figura 152. Simulación del *gate-level netlist* en inglés del nanochip con pads



Nota. La imagen muestra la simulación del *gate-level netlist* en inglés del nanochip con pads.

Como se puede observar en las figuras 151 y 152 el *gate-level netlist* del nanochip "El Gran Jaguar" funciona de la manera correcta y esperada. El texto es codificado en ambos idiomas y el *ring oscillator* funciona apropiadamente.

De esta manera, se comprueba que todos los circuitos que se sintetizaron de forma lógica en este trabajo son válidos. Los *gate-level netlists* se compartieron con el equipo de *Back-End* para el desarrollo de los *layouts*.

CAPÍTULO 12

ISE Design Suite

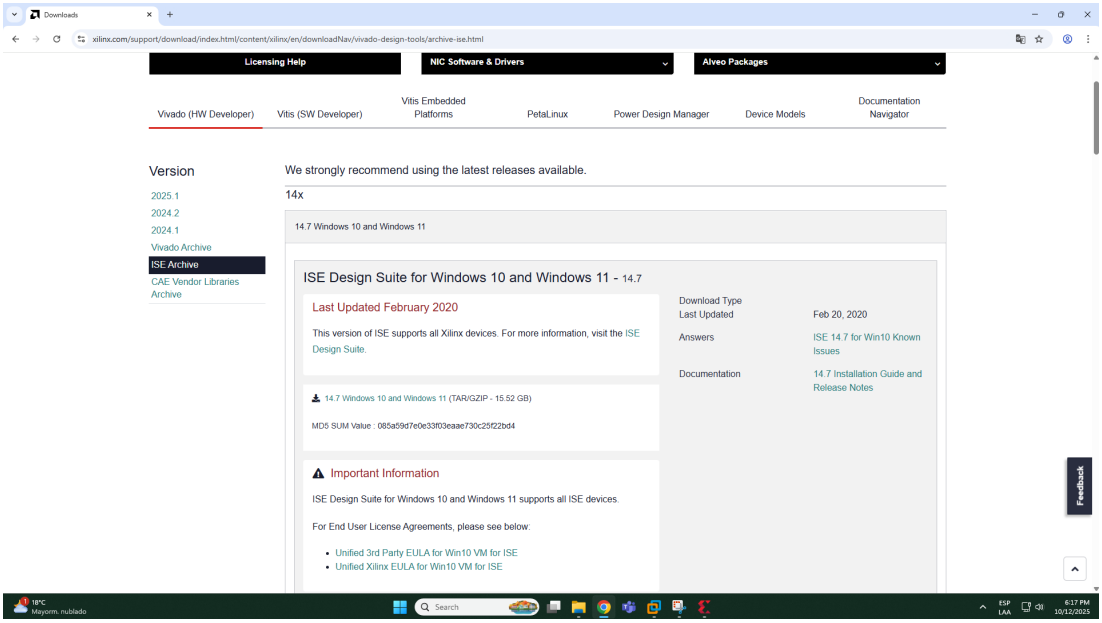
En este capítulo se detalla el procedimiento para la simulación de el *core* del nanochip "El Gran Jaguar" en ISE Design Suite. Luego de haber realizado la síntesis y simulación con las herramientas de Synopsys, el *gate-level netlist* se simula e implementa utilizando el FPGA Xilinx Virtex-5 Genesys de Digilent junto con ISE Design Suite. Esto con el propósito de verificar el funcionamiento del diseño sintetizado en un entorno con *hardware* real.

Como primer punto, se expone la simulación en ISE Design Suite a través de ISim. En el próximo capítulo se detalla la implementación física con el FPGA.

12.1. Instalación

Para comenzar, es necesario instalar el *software* ISE Design Suite. Para ello dirigirse a: `https://www.xilinx.com/support/download/index.html/content/xilinx/en/downloadNav/vivado-design-tools/archive-ise.html` Seleccionar la versión 14.7 para Windows 10 y 11:

Figura 153. Descarga de ISE Design Suite 14.7



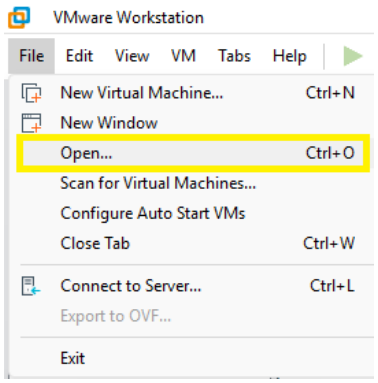
Nota. La imagen muestra la descarga de ISE Design Suite 14.7.

Seguir los pasos que se solicitan para la descarga del .zip. Una vez realizada la descarga, se descomprime el archivo.

La descarga que se realiza es una máquina virtual preconfigurada con el programa de ISE Design Suite incluido. Para poder utilizarla, es necesario importar la máquina virtual en un programa de virtualización. Se recomienda utilizar VMware Workstation debido a su facilidad de uso y compatibilidad.

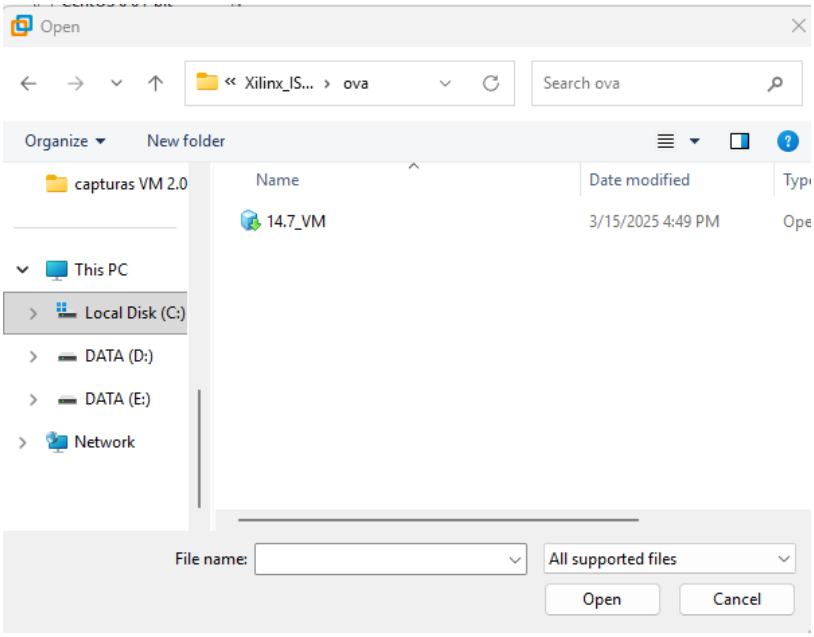
El siguiente paso es importar el archivo .ova que se encuentra en la carpeta descomprimida.

Figura 154. Primera parte de la importación del archivo .ova en VMware Workstation



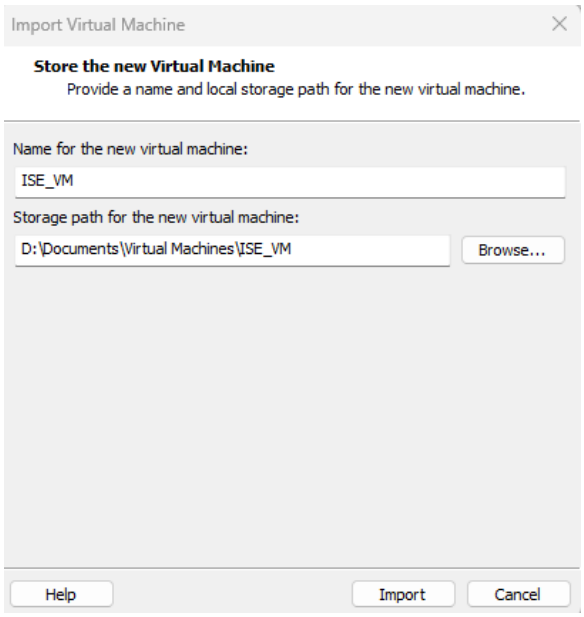
Nota. La imagen muestra la primera parte de la importación del archivo .ova en VMware Workstation.

Figura 155. Segunda parte de la importación del archivo .ova en VMware Workstation



Nota. La imagen muestra la segunda parte de la importación del archivo .ova en VMware Workstation.

Figura 156. Última parte de la importación del archivo .ova en VMware Workstation

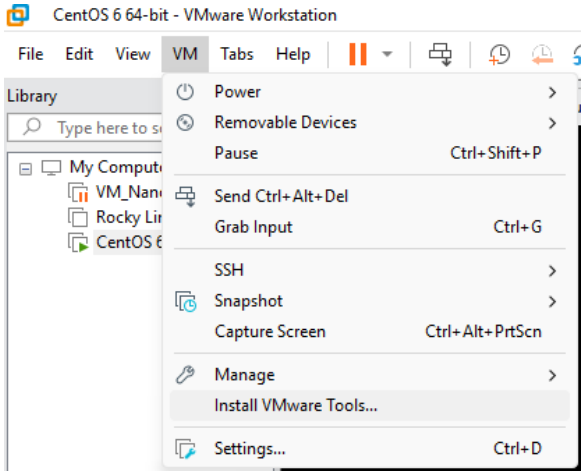


Nota. La imagen muestra la última parte de la importación del archivo .ova en VMware Workstation.

Seleccionar la opción de *Import* y seguir los pasos para crear la máquina virtual. Luego

de crear la máquina virtual, se instalan los *VMware Tools* para poder compartir archivos entre la máquina virtual y el equipo anfitrión. Para esto, buscar la opción de *Install VMware Tools* en el menú de *VM* en la parte superior de la ventana de VMware Workstation.

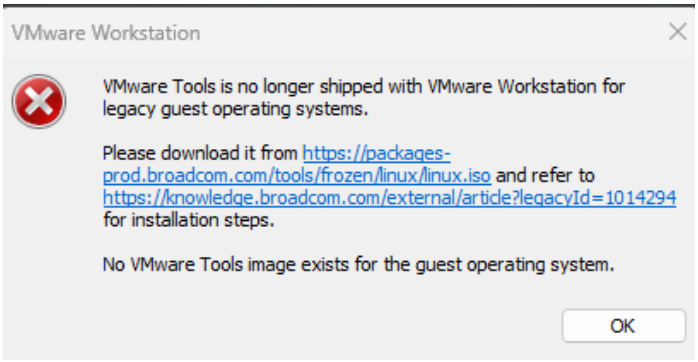
Figura 157. Primera parte de la instalación de los VMware Tools



Nota. La imagen muestra la primera parte de la instalación de los VMware Tools.

Como la máquina virtual ha sido creada de manera independiente, es necesario descargar la imagen de disco `linux.iso` desde el siguiente enlace: `https://packages-prod.broadcom.com/tools/frozen/linux/linux.iso`

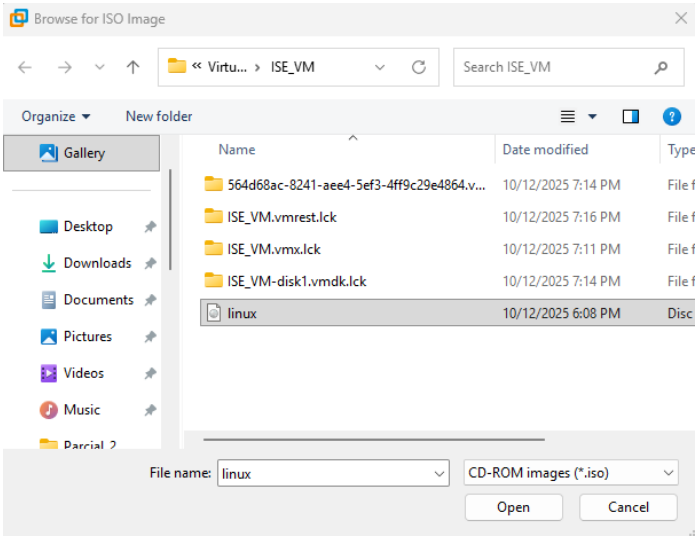
Figura 158. Ventana donde se solicita la descarga de la imagen de disco linux.iso



Nota. La imagen muestra la ventana donde se solicita la descarga de la imagen de disco linux.iso.

Luego de descargar la imagen de disco, en la configuración de la máquina virtual, seleccionar la unidad de CD/DVD e indicar la ubicación del archivo `linux.iso` descargado.

Figura 159. Selección de la imagen de disco linux.iso en la configuración de la máquina virtual



Nota. La imagen muestra la selección de la imagen de disco linux.iso en la configuración de la máquina virtual.

Cuadro 77. Comandos para instalar VMware Tools

```
# Crear punto de montaje
sudo mkdir -p /mnt/cdrom

# Realizar el montaje
sudo mount /dev/sr0/mnt/cdrom

# Verificar que se haya montado correctamente
ls /mnt/cdrom

# Extraer e instalar VMware Tools
cd /tmp
sudo tar xzf /mnt/cdrom/VMwareTools-*.tar.gz
cd vmware-tools-distrib
sudo ./vmware-install.pl
```

Figura 160. Instalación de los VMware Tools desde la terminal de la máquina virtual

Nota. La imagen muestra la instalación de los VMware Tools.

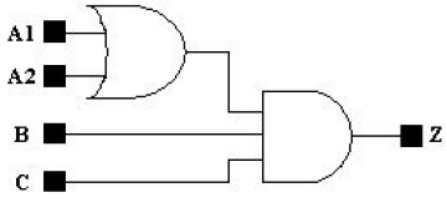
12.2. Celdas estándar (*Blackboxes*)

Posteriormente a la instalación de ISE Design Suite, es necesario crear un código de Verilog que decribe el comportamiento de las celdas estándar (*blackboxes*) del diseño sintetizado del *core* del nanochip "El Gran Jaguar". Estas celdas son las mismas del *gate-level netlist* creado con celdas de 65 nm de TSMC (ver la Subsección 18.2.1 de los Anexos). El archivo que describe el comportamiento de las celdas se llama `Blackboxes.v` y se encuentra en la Subsección 18.3.1 de los Anexos. Dicho archivo debe ser utilizado en ISE Design Suite para poder simular y emular el diseño sintetizado en el FPGA. El código fue construido en conjunto con parte del equipo de *Back-End*.

A continuación, se muestran y describen las celdas que fueron definidas en este trabajo:

- **OA211Dx**

Figura 161. Esquemático de la celda OA211Dx



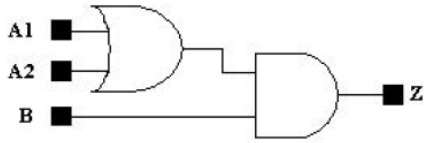
Nota. La imagen muestra el esquemático de la celda OA211Dx.

Cuadro 78. Tabla de verdad de la celda OA211Dx

INPUT				OUTPUT
A_1	A_2	B	C	Z
0	0	x	x	0
x	x	0	x	0
x	x	x	0	0
1	x	1	1	1
x	1	1	1	1

- **OA21Dx**

Figura 162. Esquemático de la celda OA21Dx



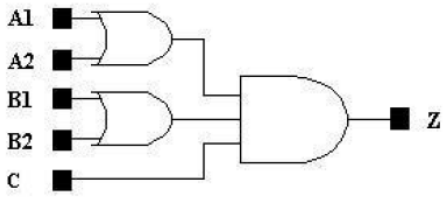
Nota. La imagen muestra el esquemático de la celda OA21Dx.

Cuadro 79. Tabla de verdad de la celda OA21Dx

INPUT			OUTPUT
A_1	A_2	B	Z
0	0	x	0
x	x	0	0
1	x	1	1
x	1	1	1

- **OA221Dx**

Figura 163. Esquemático de la celda OA221Dx



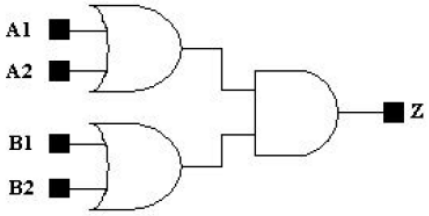
Nota. La imagen muestra el esquemático de la celda OA221Dx.

Cuadro 80. Tabla de verdad de la celda OA221Dx

INPUT					OUTPUT
A_1	A_2	B_1	B_2	C	Z
0	0	x	x	x	0
x	x	0	0	x	0
x	x	x	x	0	0
1	x	x	1	1	1
x	1	x	1	1	1
1	x	1	x	1	1
x	1	1	x	1	1

- **OA22Dx**

Figura 164. Esquemático de la celda OA22Dx



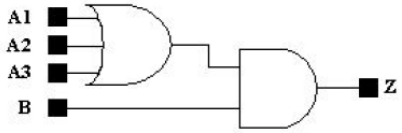
Nota. La imagen muestra el esquemático de la celda OA22Dx.

Cuadro 81. Tabla de verdad de la celda OA22Dx

INPUT				OUTPUT
A_1	A_2	B_1	B_2	Z
0	0	x	x	0
x	x	0	0	0
1	x	x	1	1
x	1	x	1	1
1	x	1	x	1
x	1	1	x	1

- **OA31Dx**

Figura 165. Esquemático de la celda OA31Dx



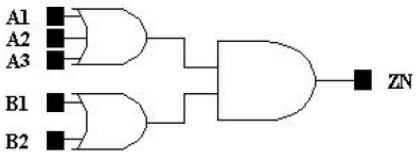
Nota. La imagen muestra el esquemático de la celda OA31Dx.

Cuadro 82. Tabla de verdad de la celda OA31Dx

INPUT				OUTPUT
A_1	A_2	A_3	B	Z
0	0	0	x	0
x	x	x	0	0
1	x	x	1	1
x	1	x	1	1
x	x	1	1	1

- **OA32Dx**

Figura 166. Esquemático de la celda OA32Dx



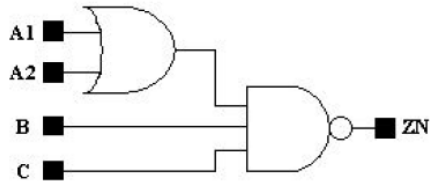
Nota. La imagen muestra el esquemático de la celda OA32Dx.

Cuadro 83. Tabla de verdad de la celda OA32Dx

INPUT					OUTPUT
A_1	A_2	A_3	B_1	B_2	Z
0	0	0	x	x	0
x	x	x	0	0	0
1	x	x	x	1	1
x	1	x	x	1	1
x	x	1	x	1	1
1	x	x	1	x	1
x	1	x	1	x	1
x	x	1	1	x	1

- **OAI211Dx**

Figura 167. Esquemático de la celda OAI211Dx



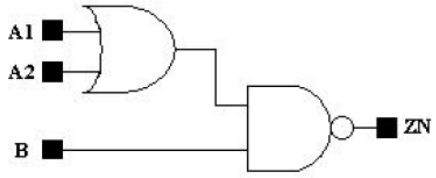
Nota. La imagen muestra el esquemático de la celda OAI211Dx.

Cuadro 84. Tabla de verdad de la celda OAI211Dx

INPUT				OUTPUT
A_1	A_2	B	C	ZN
0	0	x	x	1
x	x	0	x	1
x	x	x	0	1
1	x	1	1	0
x	1	1	1	0

- **OAI21Dx**

Figura 168. Esquemático de la celda OAI21Dx



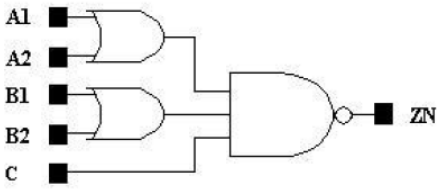
Nota. La imagen muestra el esquemático de la celda OAI21Dx.

Cuadro 85. Tabla de verdad de la celda OAI21Dx

INPUT			OUTPUT
A_1	A_2	B	ZN
0	0	x	1
x	x	0	1
1	x	1	0
x	1	1	0

- **OAI221Dx**

Figura 169. Esquemático de la celda OAI221Dx



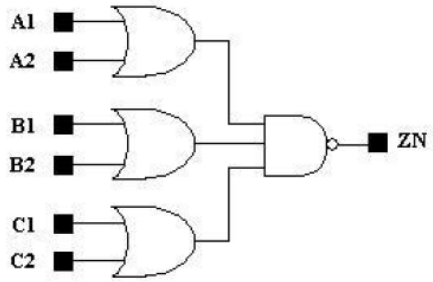
Nota. La imagen muestra el esquemático de la celda OAI221Dx.

Cuadro 86. Tabla de verdad de la celda OAI221Dx

INPUT					OUTPUT
A_1	A_2	B_1	B_2	C	ZN
0	0	x	x	x	1
x	x	0	0	x	1
x	x	x	x	0	1
1	x	x	1	1	0
x	1	x	1	1	0
1	x	1	x	1	0
x	1	1	x	1	0

- **OAI222Dx**

Figura 170. Esquemático de la celda OAI222Dx



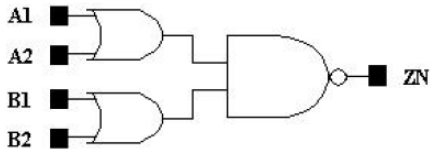
Nota. La imagen muestra el esquemático de la celda OAI222Dx.

Cuadro 87. Tabla de verdad de la celda OAI222Dx

INPUT						OUTPUT
A_1	A_2	B_1	B_2	C_1	C_2	ZN
0	0	x	x	x	x	1
x	x	0	0	x	x	1
x	x	x	x	0	0	1
1	x	x	1	x	1	0
x	1	x	1	x	1	0
1	x	1	x	x	1	0
x	1	1	x	x	1	0
1	x	x	1	1	x	0
x	1	x	1	1	x	0
1	x	1	x	1	x	0
x	1	1	x	1	x	0

- **OAI22Dx**

Figura 171. Esquemático de la celda OAI22Dx



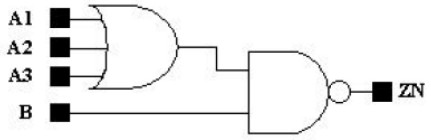
Nota. La imagen muestra el esquemático de la celda OAI22Dx.

Cuadro 88. Tabla de verdad de la celda OAI22Dx

INPUT				OUTPUT
A_1	A_2	B_1	B_2	ZN
0	0	x	x	1
x	x	0	0	1
1	x	x	1	0
x	1	x	1	0
1	x	1	x	0
x	1	1	x	0

- **OAI31Dx**

Figura 172. Esquemático de la celda OAI31Dx



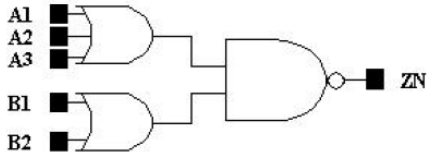
Nota. La imagen muestra el esquemático de la celda OAI31Dx.

Cuadro 89. Tabla de verdad de la celda OAI31Dx

INPUT				OUTPUT
A_1	A_2	A_3	B	ZN
0	0	0	x	1
x	x	x	0	1
1	x	x	1	0
x	1	x	1	0
x	x	1	1	0

- **OAI32Dx**

Figura 173. Esquemático de la celda OAI32Dx



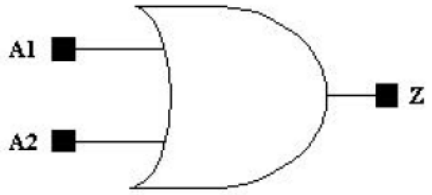
Nota. La imagen muestra el esquemático de la celda OAI32Dx.

Cuadro 90. Tabla de verdad de la celda OAI32Dx

INPUT					OUTPUT
A_1	A_2	A_3	B_1	B_2	ZN
0	0	0	x	x	1
x	x	x	0	0	1
1	x	x	x	1	0
x	1	x	x	1	0
x	x	1	x	1	0
1	x	x	1	x	0
x	1	x	1	x	0
x	x	1	1	x	0

- **OR2Dx**

Figura 174. Esquemático de la celda OR2Dx



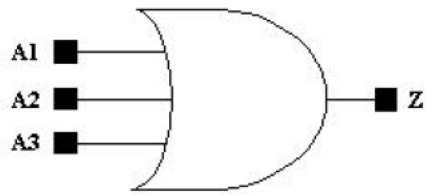
Nota. La imagen muestra el esquemático de la celda OR2Dx.

Cuadro 91. Tabla de verdad de la celda OR2Dx

INPUT		OUTPUT
A	B	Z
0	0	0
x	1	1
1	x	1

- **OR3Dx**

Figura 175. Esquemático de la celda OR3Dx



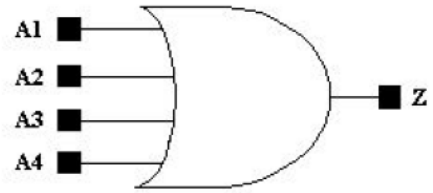
Nota. La imagen muestra el esquemático de la celda OR3Dx.

Cuadro 92. Tabla de verdad de la celda OR3Dx

INPUT			OUTPUT
A_1	A_2	A_3	Z
0	0	0	0
1	x	x	1
x	1	x	1
x	x	1	1

- **OR4Dx**

Figura 176. Esquemático de la celda OR4Dx



Nota. La imagen muestra el esquemático de la celda OR4Dx.

Cuadro 93. Tabla de verdad de la celda OR4Dx

INPUT				OUTPUT
A_1	A_2	A_3	A_4	Z
0	0	0	0	0
1	x	x	x	1
x	1	x	x	1
x	x	1	x	1
x	x	x	1	1

- **TIEL**

Figura 177. Esquemático de la celda TIEL

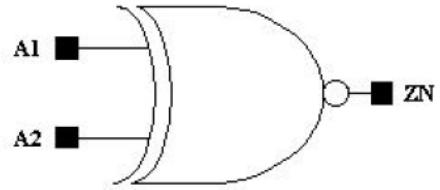
Nota. La imagen muestra el esquemático de la celda TIEL.

Cuadro 94. Tabla de verdad de la celda TIEL

OUTPUT
Z
0

- **XNR2Dx**

Figura 178. Esquemático de la celda XNR2Dx



Nota. La imagen muestra el esquemático de la celda XNR2Dx.

Cuadro 95. Tabla de verdad de la celda XNR2Dx

INPUT		OUTPUT
A	B	ZN
0	0	1
0	1	0
1	0	0
1	1	1

El funcionamiento de cualquiera de las celdas se puede verificar en el archivo PDF de la librería de 65 nm de TSMC utilizada en este caso (ver Capítulo 7). El código de todas las celdas como se mencionó anteriormente se encuentra en el archivo `Blackboxes.v` (ver la Subsección 18.3.1 de los Anexos).

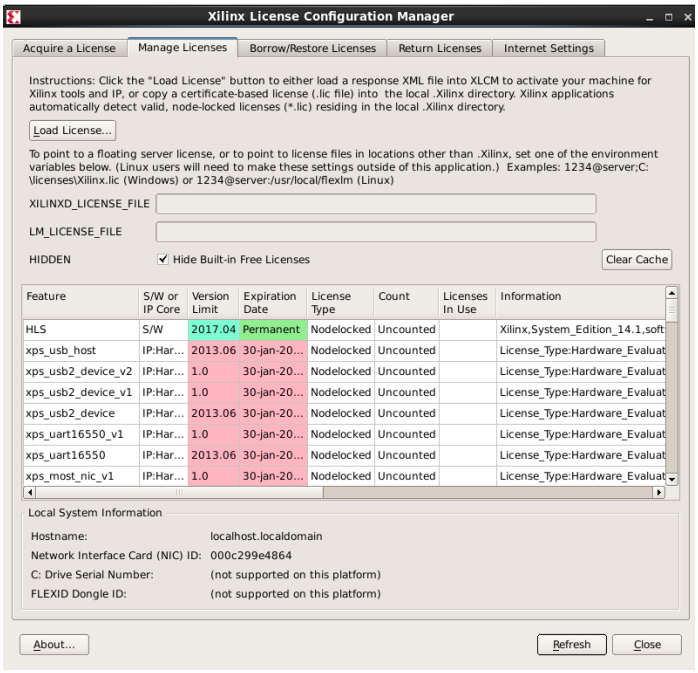
Ya con las *blackboxes* creadas, se puede proceder a simular y emular el diseño sintetizado del *core* del nanochip "El Gran Jaguar" en ISE Design Suite.

12.3. Simulación del nanochip "El Gran Jaguar" en ISE Design Suite

Una vez creadas las *blackboxes*, se procede a crear un nuevo proyecto en ISE Design Suite para simular el comportamiento del diseño sintetizado del *core* del nanochip "El Gran Jaguar". Mencionar que también es imposible implementar el diseño en el FPGA sin una licencia. Para ello, se utiliza la licencia de tipo WebPack que ofrece Xilinx de forma gratuita. Dirigirse a: `http://www.xilinx.com/getlicense` para obtener el archivo de licencia llamado `Xilinx.lic`.

Al tener el archivo de licencia, abrir Project Navigator en la máquina virtual y dirigirse a *Help* > *Manage License*.

Figura 179. Opción para administrar licencias en ISE Design Suite usando Project Navigator



Nota. La imagen muestra la opción para administrar licencias en ISE Design Suite.

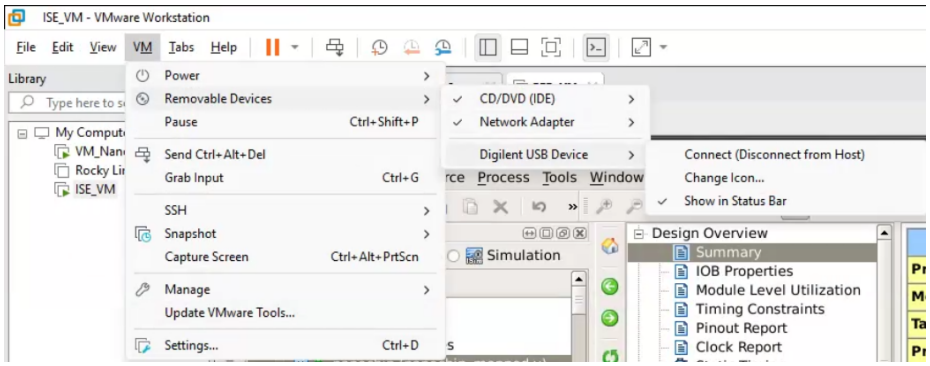
Hacer *click* en *Load License* y seleccionar el archivo `Xilinx.lic`. Para este paso es útil tener una carpeta compartida entre la máquina virtual y el equipo anfitrión.

Figura 180. Carga de la licencia en ISE Design Suite usando Project Navigator

Nota. La imagen muestra la carga de la licencia en ISE Design Suite.

Una vez cargadda la licencia, asegurarse que el FPGA se encuentre conectado a la máquina virtual a través de USB 2.0.

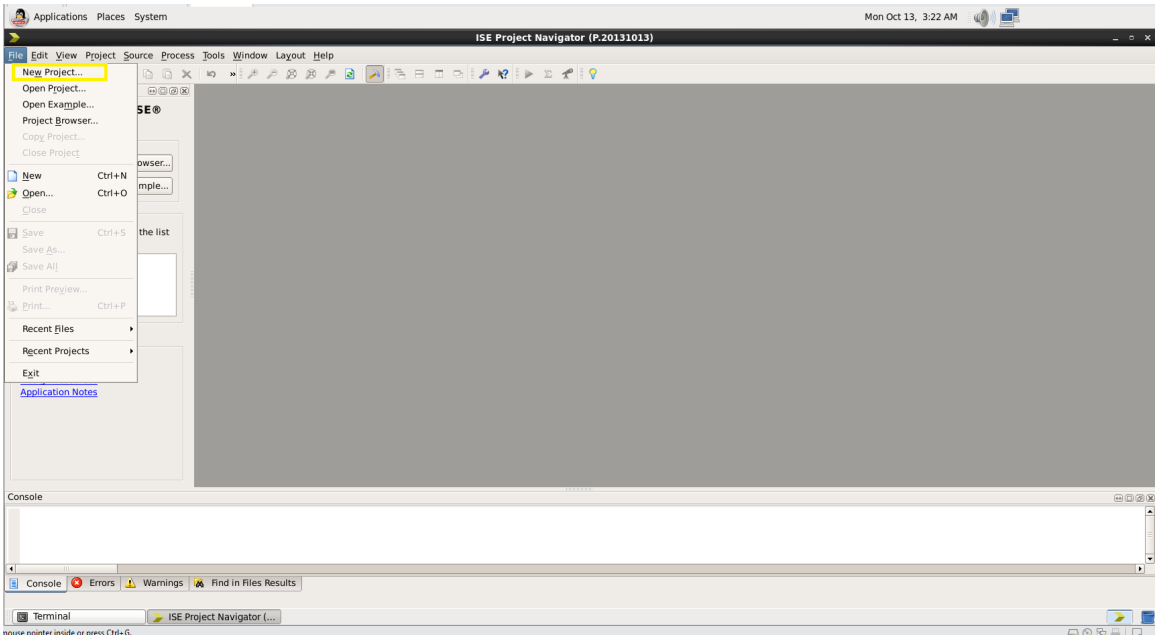
Figura 181. Verificación de la conexión del FPGA a la máquina virtual



Nota. La imagen muestra la verificación de la conexión del FPGA a la máquina virtual.

Con todo listo, se procede a crear un nuevo proyecto en ISE Design Suite.

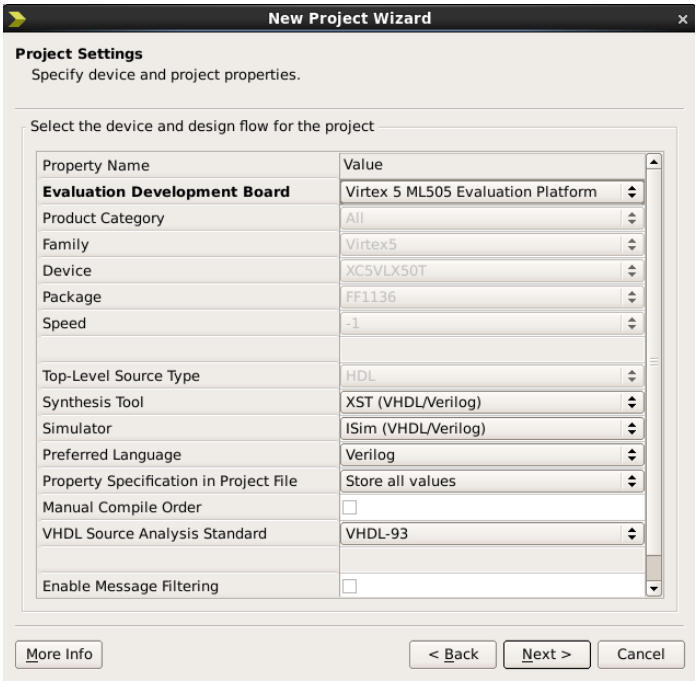
Figura 182. Creación de un nuevo proyecto en ISE Design Suite



Nota. La imagen muestra la creación de un nuevo proyecto en ISE Design Suite.

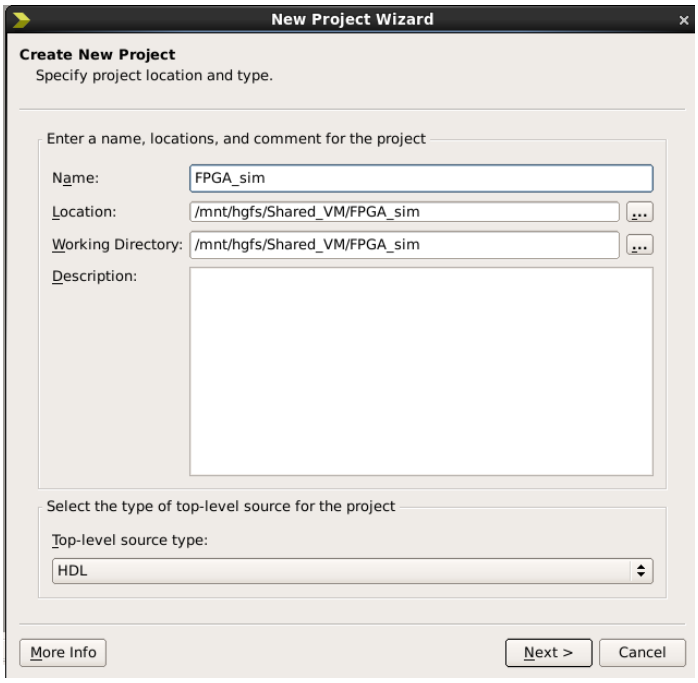
Luego se siguen los pasos para crear el proyecto.

Figura 184. Configuración del proyecto en ISE Design Suite



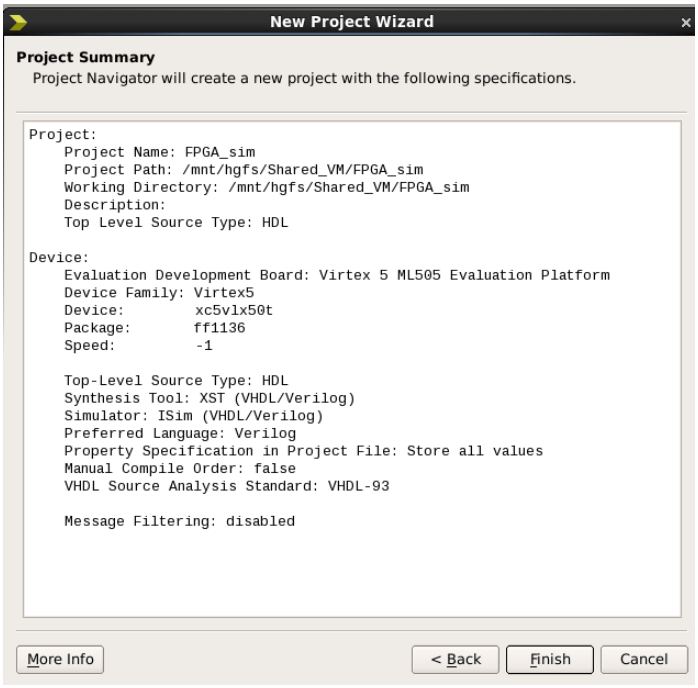
Nota. La imagen muestra la configuración del proyecto en ISE Design Suite.

Figura 183. Selección del proyecto en ISE Design Suite



Nota. La imagen muestra la selección del proyecto en ISE Design Suite.

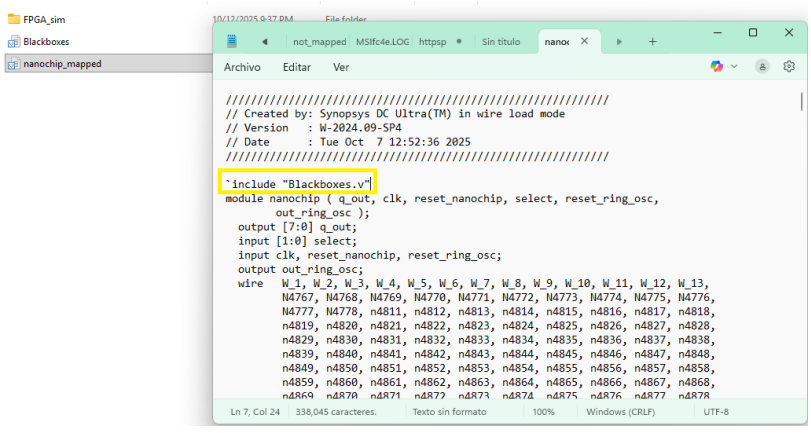
Figura 185. Creación del proyecto en ISE Design Suite



Nota. La imagen muestra la creación del proyecto en ISE Design Suite.

Una vez creado el proyecto, se deben agregar los archivos Verilog del diseño sintetizado a 65 nm del *core* del nanochip "El Gran Jaguar" (ver la Subsección 18.2.1 de los Anexos). En el archivo del *gate-level netlist* agregar la siguiente línea de código para enlazar este archivo al de `Blackboxes.v` (ver la Subsección 18.3.1 de los Anexos).

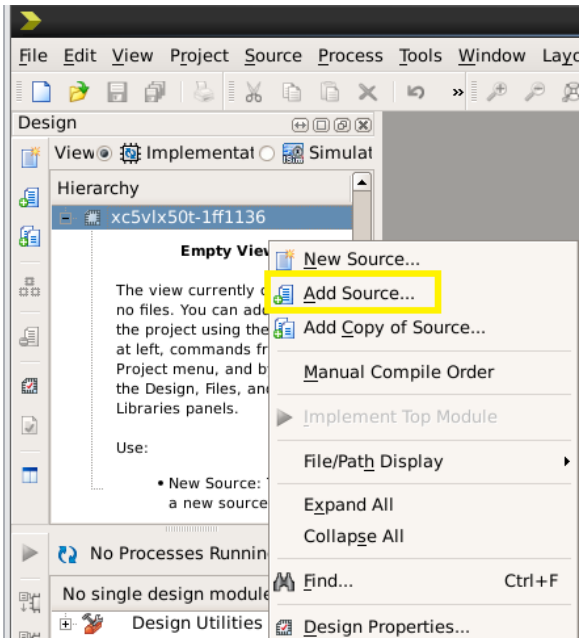
Figura 186. Vinculación del archivo del *gate-level netlist* del *core* del nanochip con el de *blackboxes*



Nota. La imagen muestra la vinculación del archivo del *gate-level netlist* del *core* del nanochip con el de *blackboxes*.

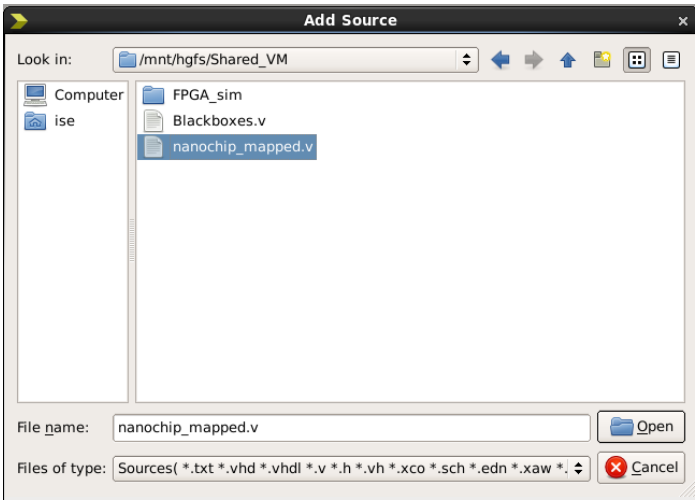
De esta manera, al agregar el archivo `nanochip_mapped.v` al proyecto, ISE Design Suite importará automáticamente el archivo de `Blackboxes.v`.

Figura 187. Paso 1 para agregar los archivos de Verilog al proyecto



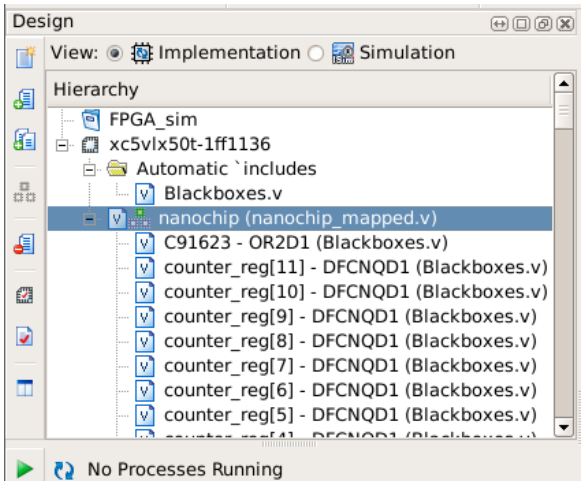
Nota. La imagen muestra el paso 1 para agregar los archivos de Verilog al proyecto.

Figura 188. Paso 2 para agregar los archivos de Verilog al proyecto



Nota. La imagen muestra el paso 2 para agregar los archivos de Verilog al proyecto.

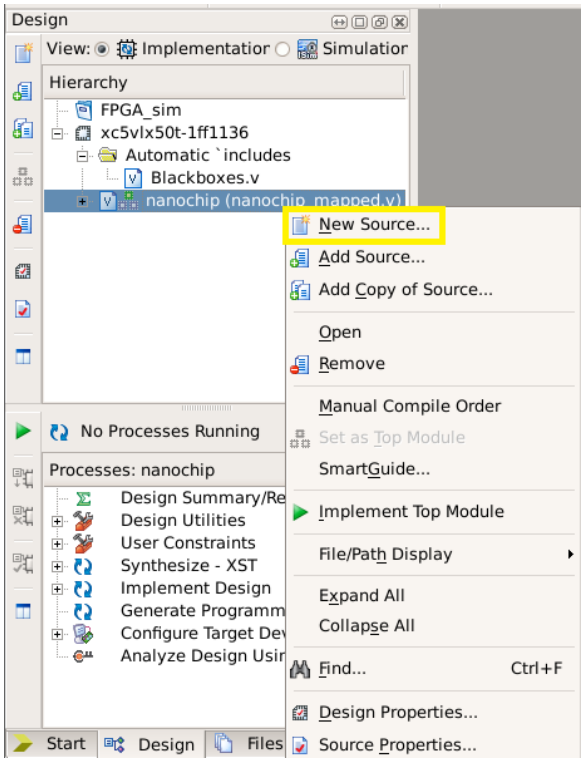
Figura 189. Archivos de Verilog agregados al proyecto



Nota. La imagen muestra los archivos de Verilog agregados al proyecto.

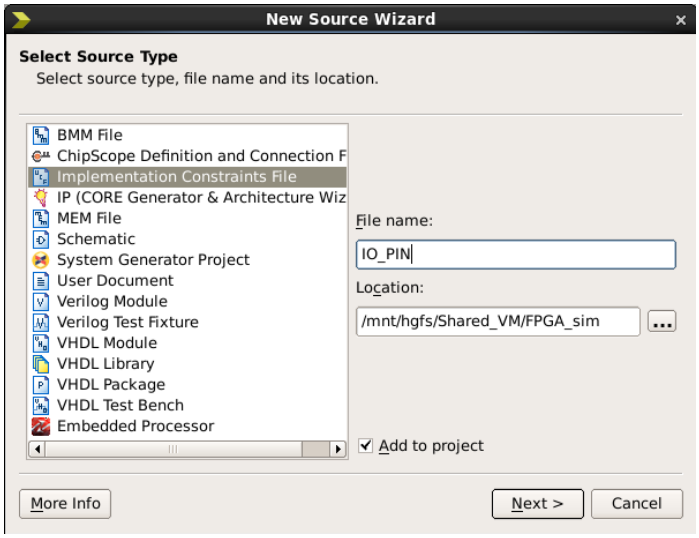
Además, se debe agregar un archivo UCF para definir los pines del FPGA que se van a utilizar.

Figura 190. Primer paso para agregar el archivo UCF al proyecto



Nota. La imagen muestra el primer paso para agregar el archivo UCF al proyecto.

Figura 191. Archivo UCF agregado al proyecto

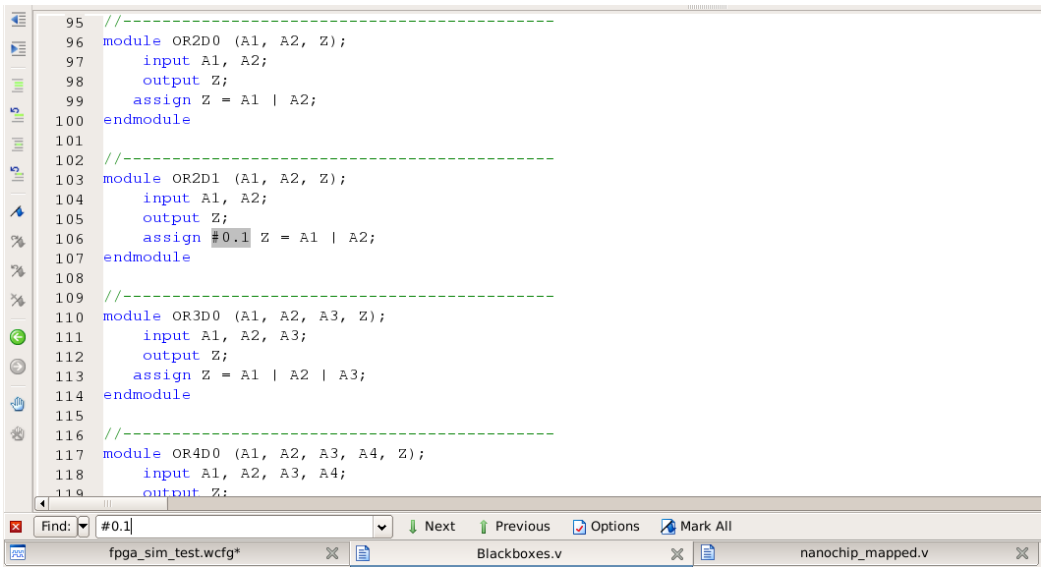


Nota. La imagen muestra el archivo UCF agregado al proyecto.

El archivo UCF utilizado se encuentra en la Subsección 18.3.2 de los Anexos.

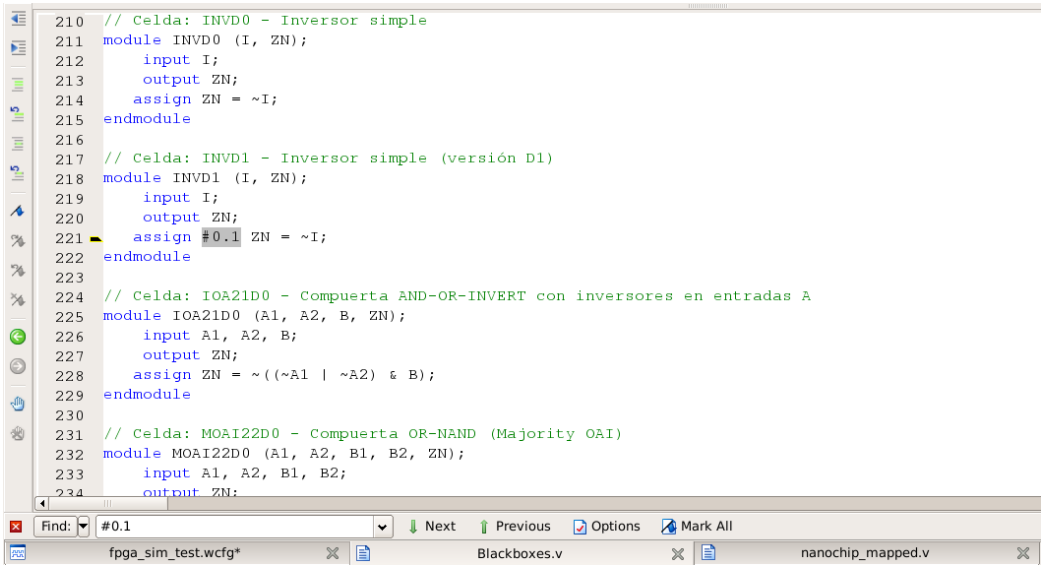
Es importante mencionar que al igual que para la simulación de un *ring oscillator* en Verdi es necesario agregar un *delay*, para este caso también es importante agregarlo. Se debe agregar el *delay* en el archivo `Blackboxes.v` a las celdas OR2D1 e INVD1.

Figura 192. Adición de *delay* a la celda OR en el archivo `Blackboxes.v`



Nota. La imagen muestra la adición de *delay* a la celda OR en el archivo `Blackboxes.v`.

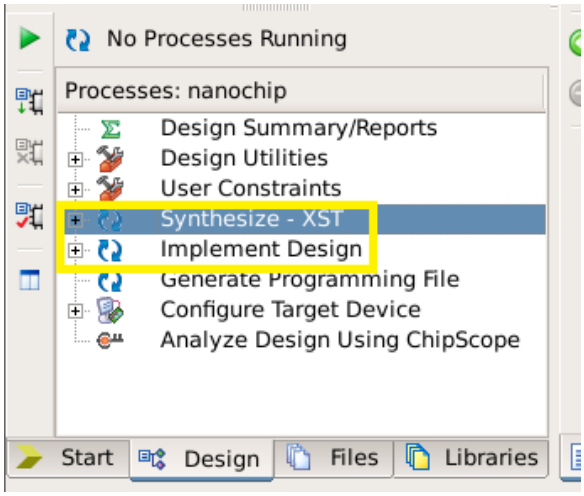
Figura 193. Adición de *delay* a la celda INV en el archivo `Blackboxes.v`



Nota. La imagen muestra la adición de *delay* a la celda INV en el archivo `Blackboxes.v`.

Posteriormente, se procede a sintetizar y mapear el diseño con las opciones *Synthesize* e *Implement Design*.

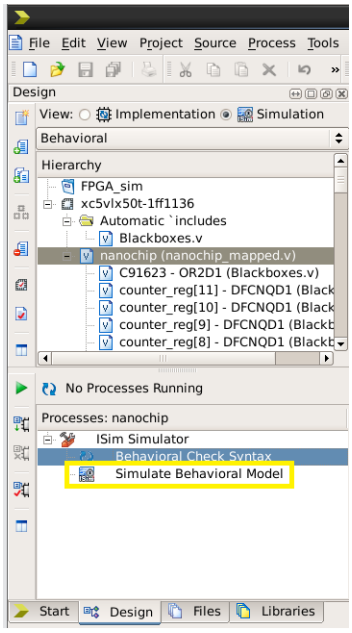
Figura 194. Opciones para sintetizar y mapear el diseño en ISE Design Suite



Nota. La imagen muestra las opciones para sintetizar y mapear el diseño en ISE Design Suite.

Este proceso puede tardar varios minutos dependiendo de la complejidad del diseño. Una vez finalizado el mapeo, se procede a simular el diseño utilizando ISim.

Figura 195. Opción para simular el diseño en ISE Design Suite usando ISim

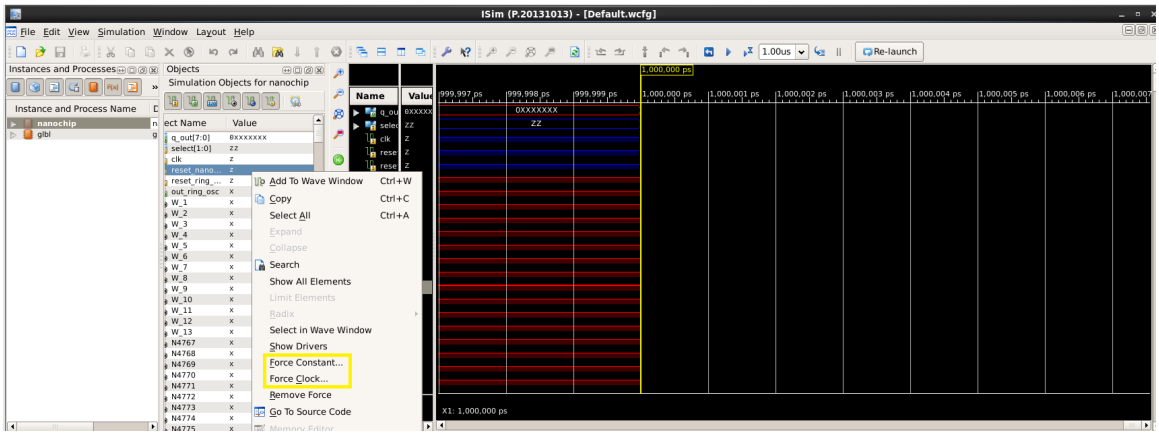


Nota. La imagen muestra la opción para simular el diseño en ISE Design Suite usando ISim.

ISim se abrirá en una nueva ventana. Para comenzar a simular asignarle constantes tanto al `reset_nanochip` como al `reset_ring_osc`. Además, asignarle una constante al `select` y crear un reloj para la señal `clk`.

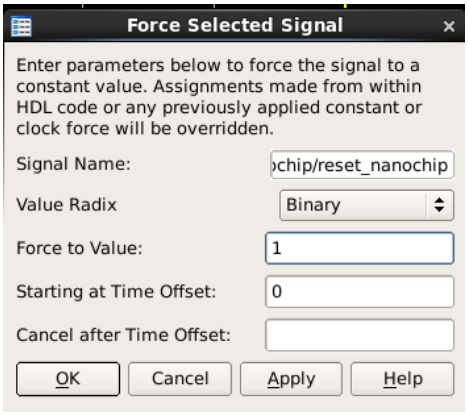
En este caso, se inicia con los *resets* encendidos (1'b1) y con la selección del texto en español.

Figura 196. Primer paso para asignar una constante o crear un reloj en ISim



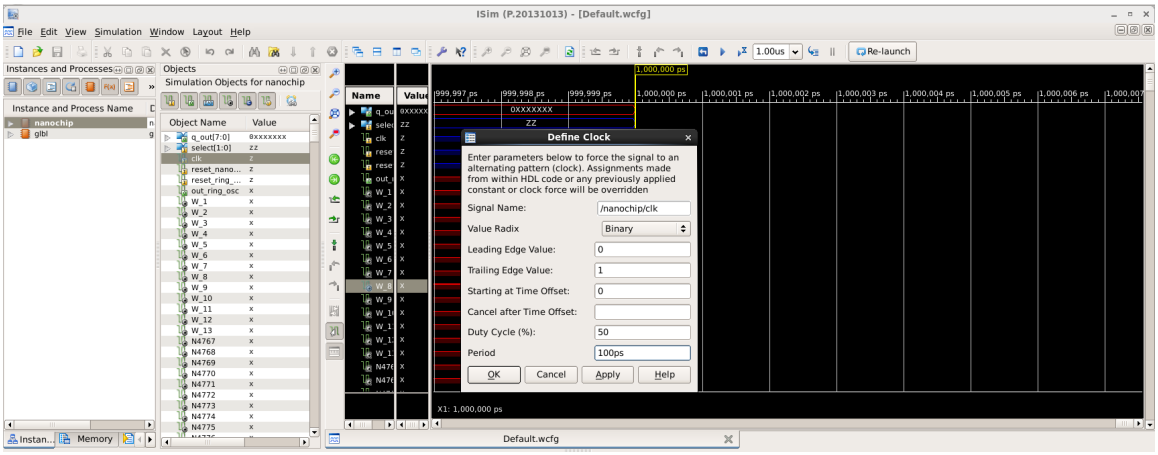
Nota. La imagen muestra el primer paso para asignar una constante o crear un reloj en ISim.

Figura 197. Asignación de una constante a una señal en ISim



Nota. La imagen muestra el segundo paso para asignar una constante a una señal en ISim.

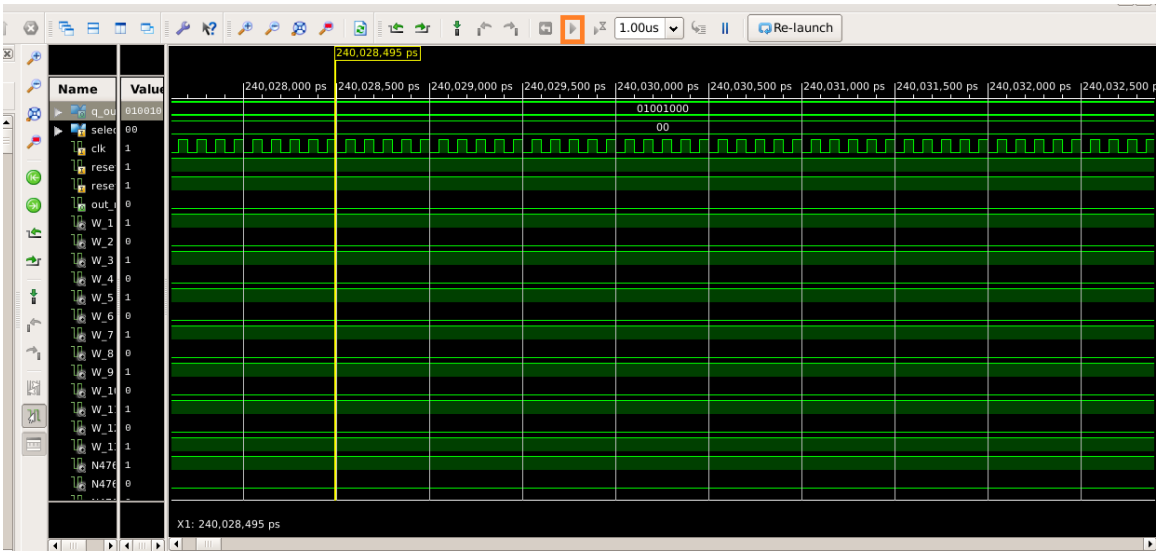
Figura 198. Creación de un reloj en ISim



Nota. La imagen muestra la creación de un reloj en ISim.

Luego se procede a correr la simulación. Para ello, hacer *click* en el botón encerrado en el recuadro naranja:

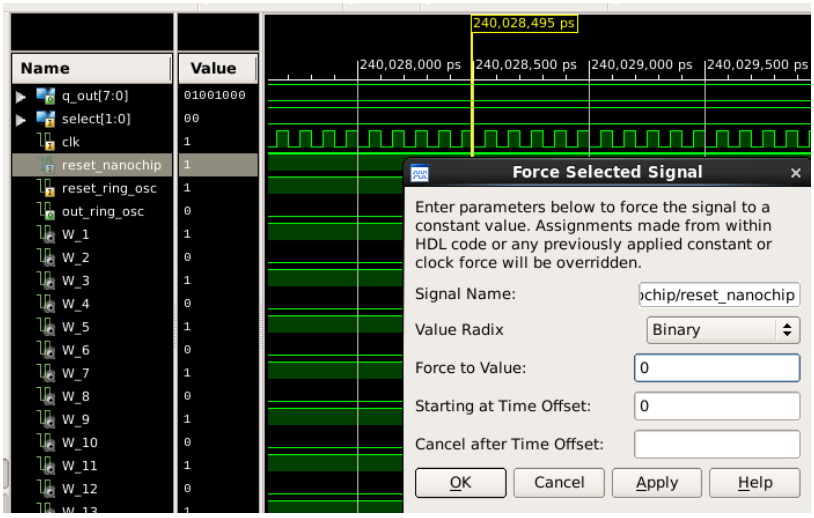
Figura 199. Ejecución de la simulación en ISim



Nota. La imagen muestra la ejecución de la simulación en ISim.

En la Figura 199 puede observarse la simulación corriendo, ya que las señales aparecen en verde. Para que se sigan codificando los caracteres, pausar la simulación y desactivar los *resets*.

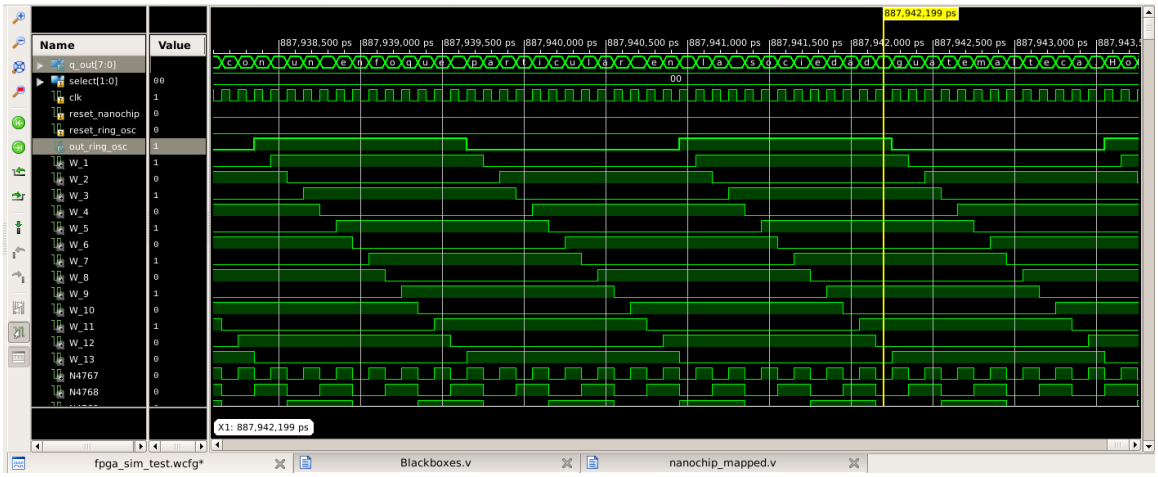
Figura 200. Desactivación de los *resets* en ISim



Nota. La imagen muestra la desactivación de los *resets* en ISim.

Luego de desactivar los *resets*, reanudar la simulación para observar el funcionamiento.

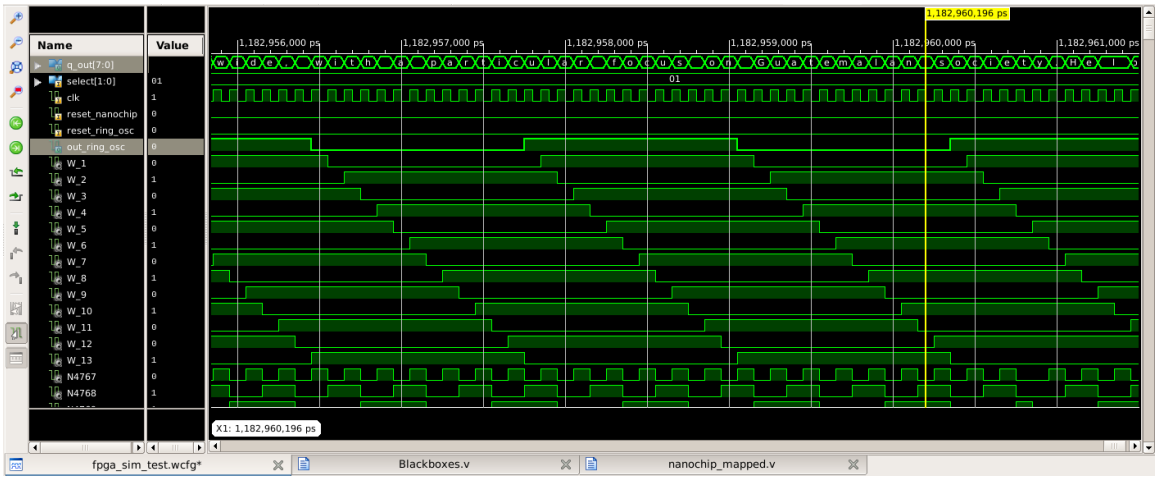
Figura 201. Simulación del diseño sintetizado del *core* del nanochip "El Gran Jaguar" en ISim para el texto en español



Nota. La imagen muestra la simulación del diseño sintetizado del *core* del nanochip "El Gran Jaguar" en ISim para el texto en español.

La Figura 201 muestra la simulación para el caso del texto en español. Para verificar el caso del texto en inglés, se debe cambiar la señal *select* y volver a ejecutar la simulación.

Figura 202. Simulación del diseño sintetizado del *core* del nanochip "El Gran Jaguar" en ISim para el texto en inglés



Nota. La imagen muestra la simulación del diseño sintetizado del *core* del nanochip "El Gran Jaguar" en ISim para el texto en inglés.

La Figura 202 muestra la simulación para el caso del texto en inglés.

Tanto la simulación del texto en español como en inglés se ejecutan correctamente en

ISim. Además, el *ring oscillator* oscila y funciona de manera adecuada. Ambas simulaciones coinciden con las realizadas en Verdi en la Subsección 11.6.2 del Capítulo 11, lo que brinda mayor validez al diseño sintetizado del *core* del nanochip "El Gran Jaguar".

El próximo paso es emular el diseño sintetizado en el FPGA Xilinx Virtex-5 Genesys de Digilent, lo cual se detalla en el próximo capítulo.

CAPÍTULO 13

Verificación experimental del *gate-level netlist* del *core* del nanochip "El Gran Jaguar" (*FPGA prototyping*)

En este capítulo se presenta lo realizado para la verificación experimental de la síntesis lógica del nanochip "El Gran Jaguar" con librerías de 65 nm de TSMC. Se utilizó el diseño sintetizado del *core* de "El Gran Jaguar" que se encuenta en la Subsección 18.2.1 de los Anexos. Por otra parte, la verificación se hizo emulando el diseño en el FPGA Xilinx Virtex-5 Genesys de Digilent, conectando el FPGA a una Raspberry Pi 4B que recibe el texto ASCII y lo reproduce en audio utilizando una bocina estéreo.

La metodología seguida para la verificación experimental se divide en tres partes principales:

1. Conexiones físicas entre el FPGA, la Raspberry Pi 4B, el osciloscopio y todo lo necesario.
2. Programación del FPGA Xilinx Virtex-5 Genesys con el *bitstream* generado a partir del diseño sintetizado del *core* de "El Gran Jaguar".
3. Desarrollo y ejecución del *software* en la Raspberry Pi 4B para la recepción del texto ASCII y su reproducción en audio.

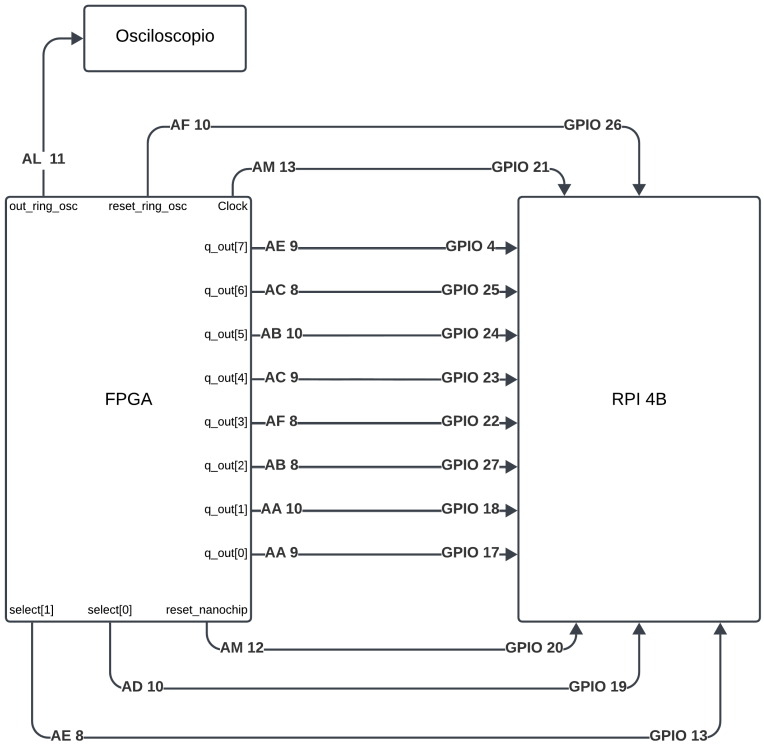
Por otro lado, para realizar la verificación experimental se utilizaron los siguintes materiales:

- FPGA Xilinx Virtex-5 Genesys de Digilent.
- Raspberry Pi 4B.
- Teclado y mouse USB.

- Cable HDMI.
- Monitor con entrada HDMI.
- Bocina estéreo.
- Cable de red UTP.
- 16 Cables *jumper* macho-macho.
- *Protoboard*.
- Extensión de GPIOs tipo T para Raspberry Pi 4B con cable plano para conectar todos los pines.
- Osciloscopio y puntas para medición.

Primeramente, se debe conectar el FPGA con la Raspberry Pi 4B utilizando los cables jumper macho-macho y la extensión de GPIOs tipo T. El diagrama de conexión se muestra a continuación:

Figura 203. Diagrama de conexión entre el FPGA y la Raspberry Pi 4B



Nota. La imagen muestra el diagrama de conexión entre el FPGA y la Raspberry Pi 4B.

Importante concer la distribución de pines del FPGA Xilinx Virtex-5 Genesys:

Figura 204. Distribución de pines del FPGA Xilinx Virtex-5 Genesys

Pmod A		Pmod B		Pmod C		Pmod D	
Signal	Pin	Signal	Pin	Signal	Pin	Signal	Pin
JA1	AD11	JB1	AE9	JC1	AL11	JD1	AN14
JA2	AD9	JB2	AC8	JC2	AJ10	JD2	AN13
JA3	AM13	JB3	AB10	JC3	AK9	JD3	AP12
JA4	AM12	JB4	AC9	JC4	AF9	JD4	AL10
JA7	AD10	JB7	AF8	JC7	AK11	JD7	AP14
JA8	AE8	JB8	AB8	JC8	AC10	JD8	AN12
JA9	AF10	JB9	AA10	JC9	AJ9	JD9	AM11
JA10	AJ11	JB10	AA9	JC10	AA8	JD10	AK8

Nota. La imagen muestra la distribución de pines del FPGA Xilinx Virtex-5 Genesys [39].

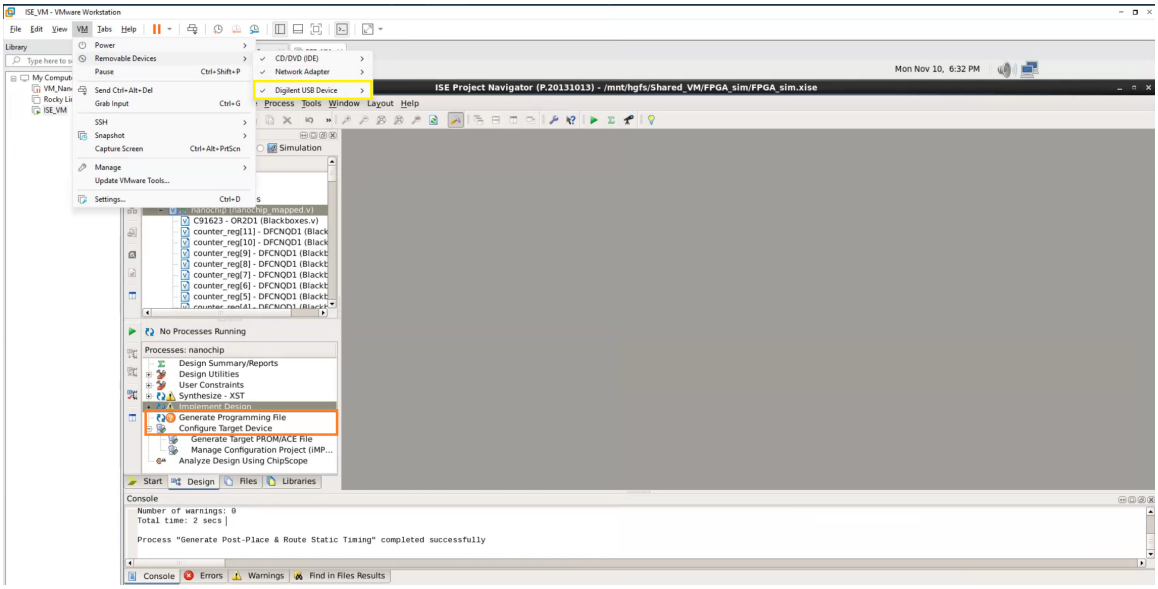
Además, también es importante recordar la colocación de estos pines en la PCB del FPGA (ver Figura 16 del Marco teórico).

13.1. Programación del FPGA Xilinx Virtex-5 Genesys

Como siguiente punto, se debe programar el FPGA con el *bitstream* generado a partir del diseño sintetizado del *core* de "El Gran Jaguar". Para ello, se utilizó la misma herramienta que se utilizó en el capítulo anterior para simular el diseño, es decir, Xilinx ISE Design Suite 14.7.

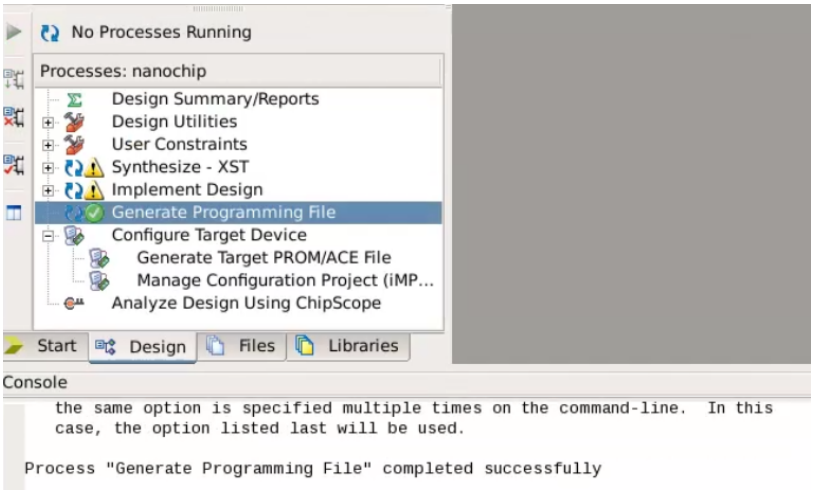
Con el mismo proyecto que se creó y utilizó en el capítulo anterior, se procede a generar el *bitstream* del diseño sintetizado. Se debe hacer doble *click* en la opción *Generate Programming File* en el menú de la izquierda dentro de ISE Design Suite. Antes de ello, asegurarse que el FPGA esté conectado a la máquina virtual y el diseño ya haya sido sintetizado correctamente.

Figura 205. Primer paso para implementar el *core* de "El Gran Jaguar" en el FPGA



Nota. La imagen muestra el primer paso para implementar el diseño sintetizado en el FPGA.

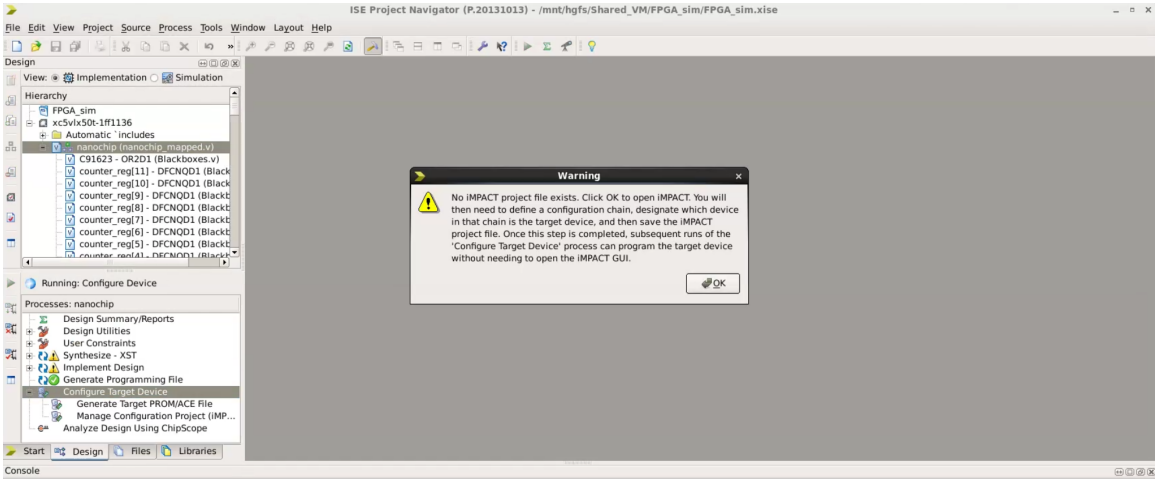
Figura 206. Generación del *bitstream* del diseño sintetizado del *core* de "El Gran Jaguar"



Nota. La imagen muestra la generación del *bitstream* del diseño sintetizado.

Posteriormente, se debe programar el FPGA con el *bitstream* generado. Para ello, se debe hacer doble *click* en la opción *Configure Target Device* en el menú de la izquierda dentro de ISE Design Suite.

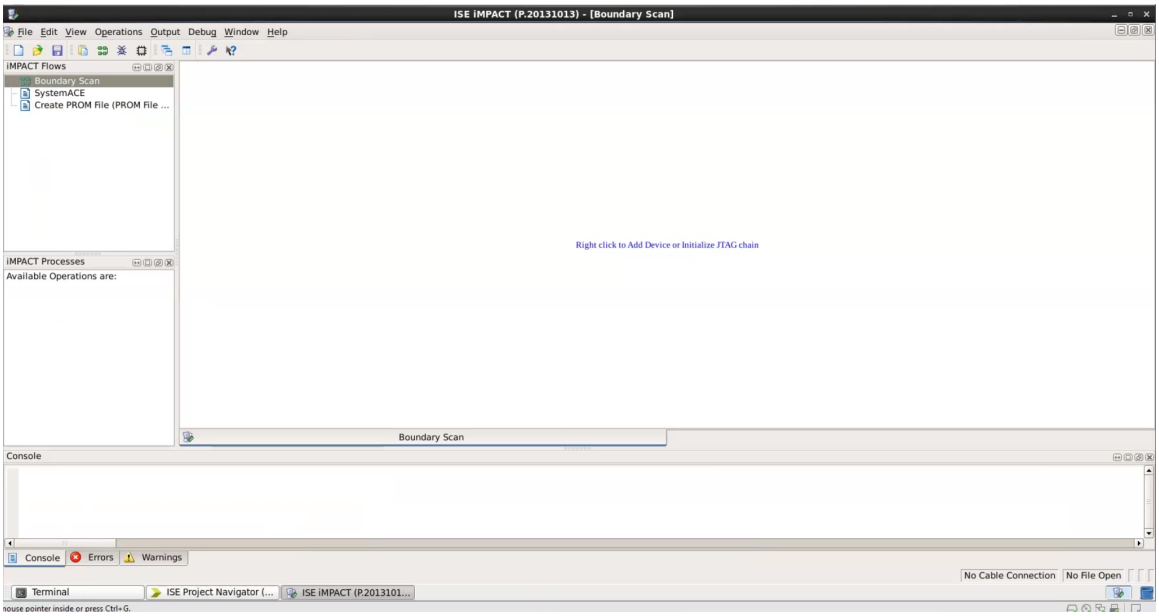
Figura 207. Creación del archivo de iMPACT para programar el FPGA



Nota. La imagen muestra la creación del archivo de iMPACT para programar el FPGA.

Se mostrará una advertencia donde no hay ningún archivo de iMPACT asociado al proyecto actual. Hacer *click* en *OK* para crear uno nuevo y así poder programar el FPGA. Luego, hacer doble *click* en *Boundary Scan* para detectar el FPGA conectado a la máquina virtual.

Figura 208. Archivo de iMPACT para programar el FPGA



Nota. La imagen muestra el archivo de iMPACT para programar el FPGA.

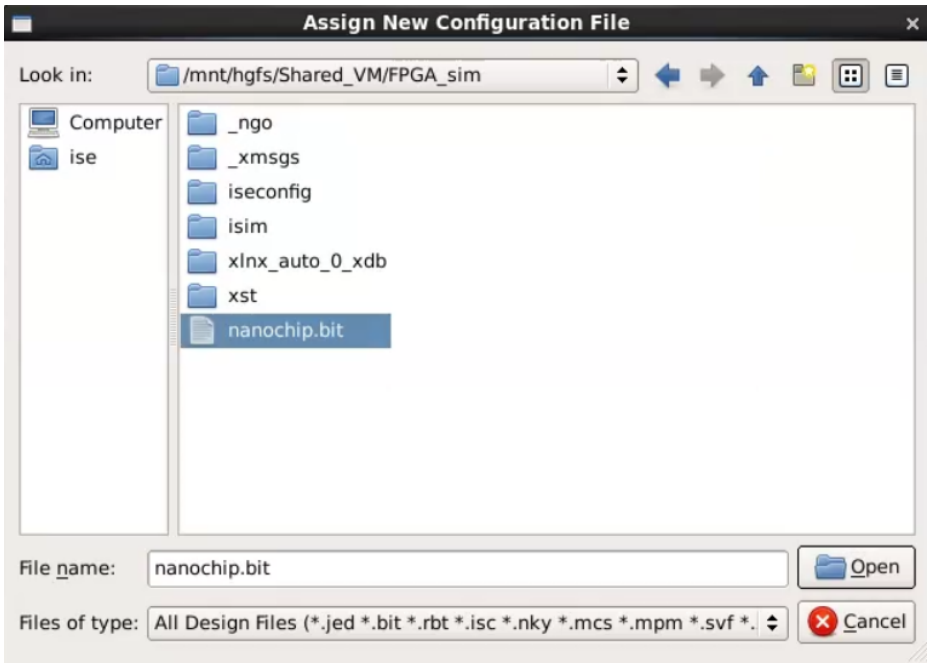
Después hacer *click* derecho en el medio y seleccionar *Add Xilinx Device*.

Figura 209. Opción para añadir el archivo *bitstream*

Nota. La imagen muestra la opción para añadir el archivo *bitstream*.

Seleccionar el archivo *bitstream* generado previamente y hacer *click* en *Open*.

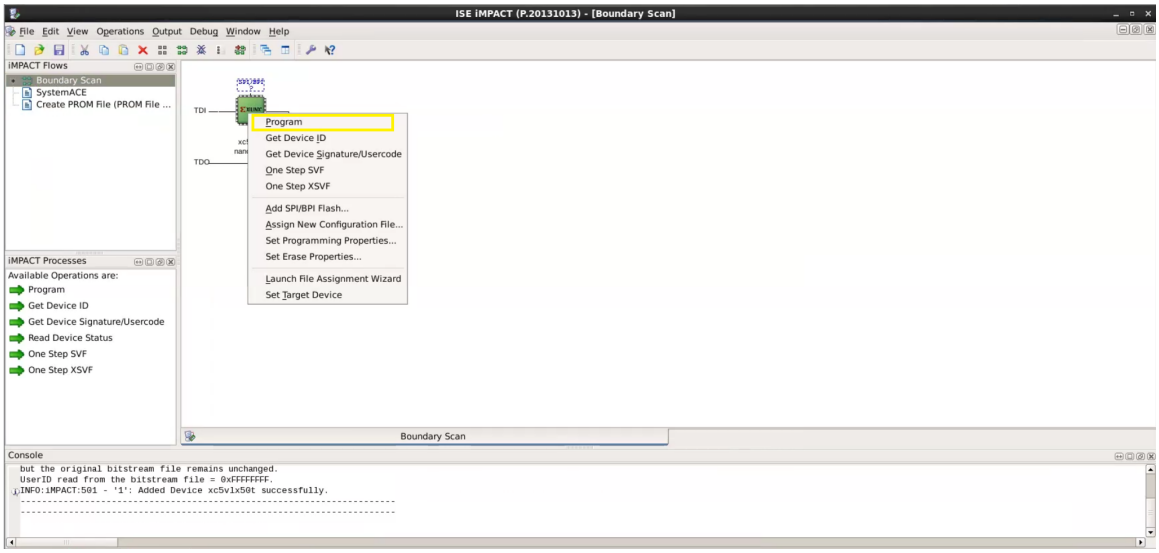
Figura 210. Selección del archivo *bitstream* para programar el FPGA



Nota. La imagen muestra la selección del archivo *bitstream* para programar el FPGA.

De esta manera, aparece como reconocido el dispositivo. Hacer *click* derecho sobre el mismo y seleccionar *Program*.

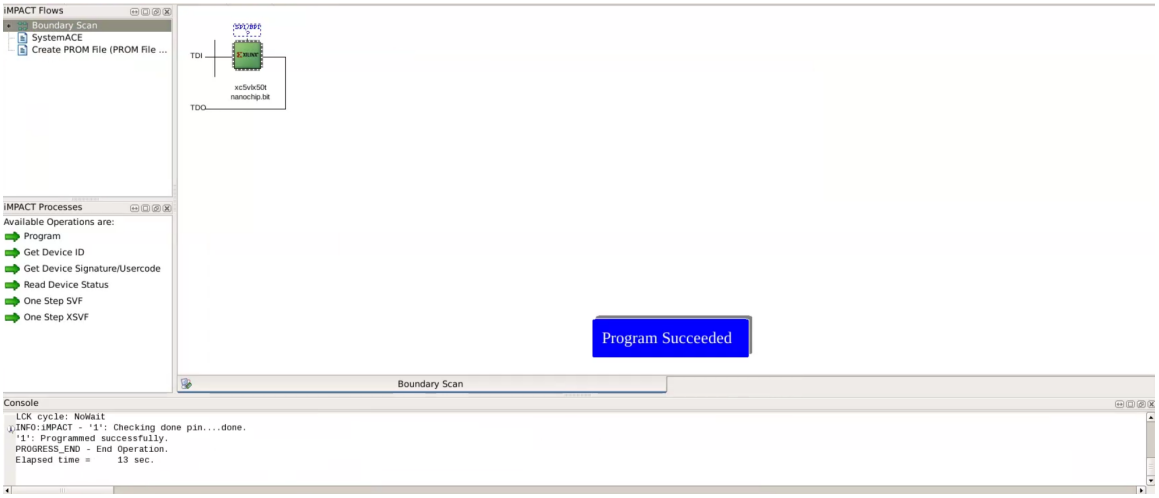
Figura 211. Opción para programar el FPGA



Nota. La imagen muestra la opción para programar el FPGA.

Finalmente, esperar a que el proceso de programación finalice correctamente y listo, ya está implementado el diseño sintetizado del *core* de "El Gran Jaguar" en el FPGA.

Figura 212. Programación del FPGA con el diseño sintetizado del *core* de "El Gran Jaguar"



Nota. La imagen muestra la programación del FPGA con el diseño sintetizado del *core* de "El Gran Jaguar".

13.2. Verificación experimental junto a la Raspberry Pi 4B

El siguiente paso es preparar la Raspberry Pi 4B para que pueda recibir el texto ASCII del FPGA y reproducir el audio en la bocina estéreo. Primero, se instaló la distribución de Raspberry Pi OS de 64 bits utilizando Raspberry Pi Imager. Luego se creó un código de Python v3 para recibir el texto ASCII, procesarlo y luego reproducirlo. El nombre del código es `nano_RPI_modified.py` y se encuentra en la Subsección 18.3.3 de los Anexos. El código está basado en el realizado por la estudiante Katherine Rac [27].

El código realiza las siguientes funciones:

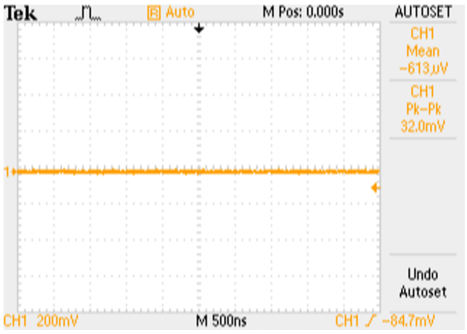
1. Recibir el texto ASCII del FPGA a través de 8 pines GPIO. Envía los pulsos de reloj para sincronizar la recepción de datos y convierte los *bytes* en caracteres.
2. Divide el texto en frases usando el punto (.) como separador, detecta automáticamente el idioma para reproducirlo de forma apropiada y se utiliza una heurística especial para forzar la pronunciación en español cuando se identifican los nombres.
3. Utiliza multiprocesos para leer datos de los GPIO y convertir cada frase en audio de forma paralela usando *Google Text-to-Speech* y guarda los archivos de audio en formato MP3.
4. Utiliza multihilos para monitoreo y reinicio de parámetros.

Luego de conectar todo y programar el FPGA, se ejecuta el código de Python para recibir el ASCII y reproducirlo. Como el nanochip consta de la parte que codifica el ASCII y el *ring oscillator*, también se utilizó un osciloscopio para verificar que el *ring oscillator* funcionara correctamente.

Para comprobar el funcionamiento del *ring oscillator* se realizará una comparación entre la señal que muestra el osciloscopio antes y después de ejecutar el código de Python para que empiece a funcionar la implementación. Para capturar dichas señales se utilizó el *software* de OpenChoice de Tektronix.

A continuación, se muestra la señal del *ring oscillator* antes de ejecutar el código en la Raspberry Pi 4B:

Figura 213. Señal del *ring oscillator* antes de ejecutar el código en la Raspberry Pi 4B



Nota. La imagen muestra la señal del *ring oscillator* antes de ejecutar el código en la Raspberry Pi 4B.

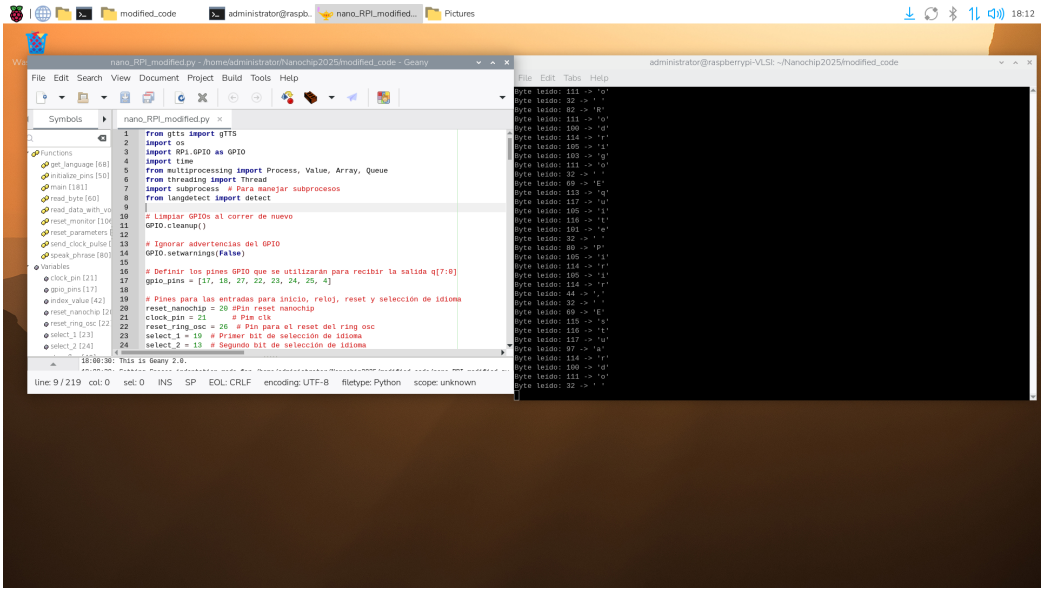
Se puede observar en la figura anterior que el osciloscopio no muestra nada, ya que el *ring oscillator* no está funcionando.

Ahora se procede a ejecutar el código. Se abre una terminal donde se encuentre el archivo de Python y correr el siguiente comando:

Cuadro 96. Comando para ejecutar el código de Python en la Raspberry Pi 4B

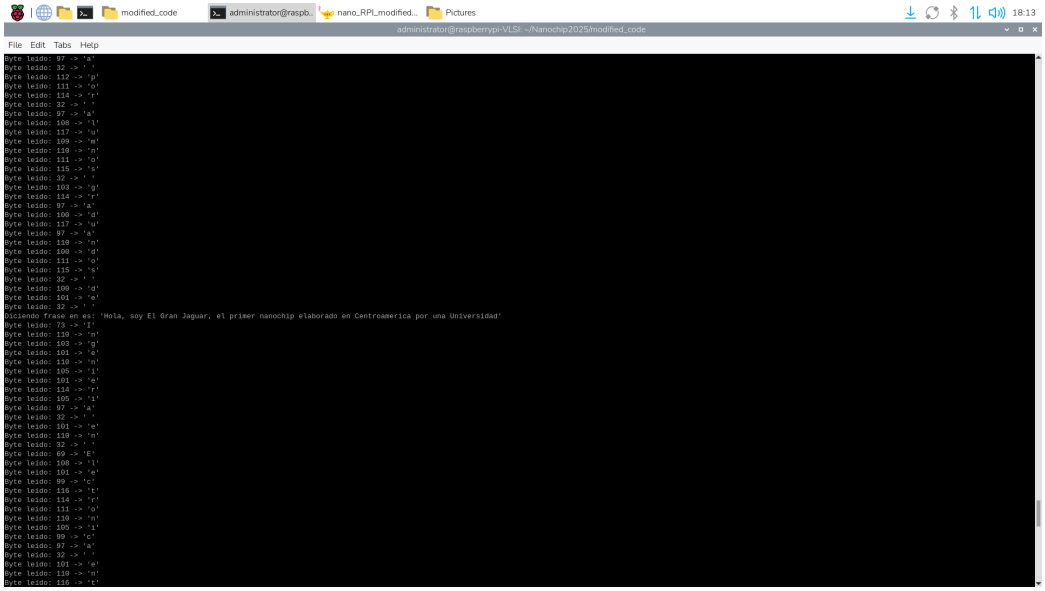
```
python3 nano_RPI_modified.py
```

Figura 214. Ejecución del programa de Python en la RPI 4B para recibir el ASCII del nanochip emulado



Nota. La imagen muestra la ejecución del programa de Python en la RPI 4B.

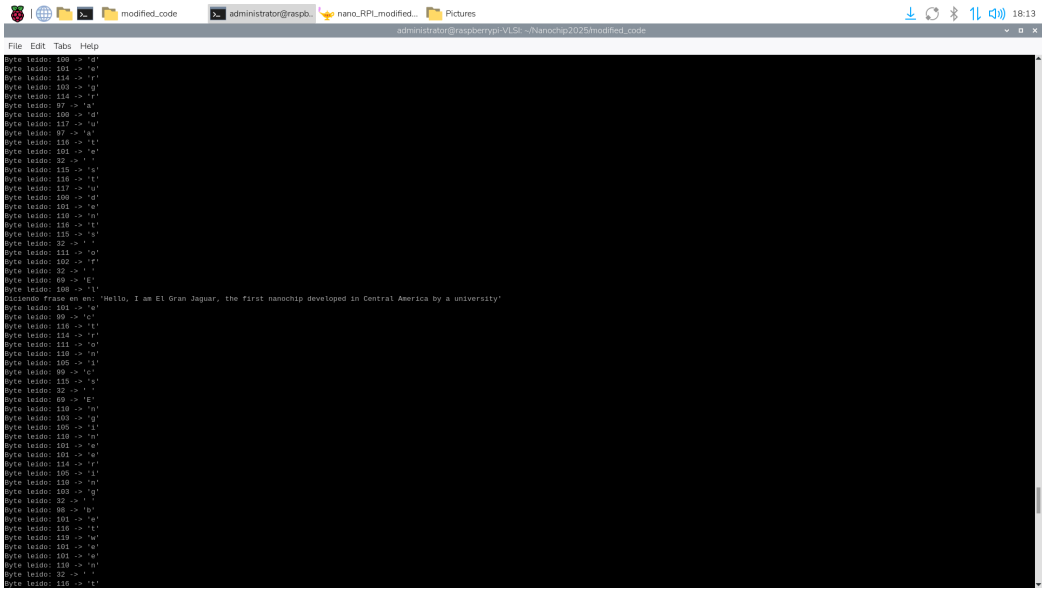
Figura 215. Terminal que muestra la cadena ASCII recibida en español



Nota. La imagen muestra la terminal que recibe la cadena ASCII en español.

También se hizo la prueba cambiando la señal `select` en el FPGA para codificar el texto en inglés:

Figura 216. Terminal que muestra la cadena ASCII recibida en inglés

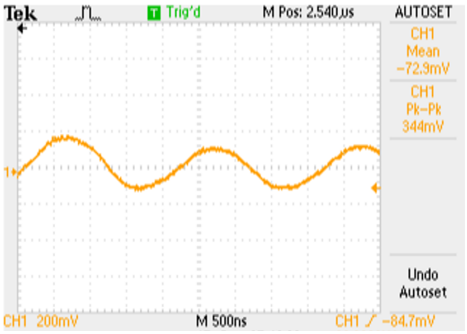


Nota. La imagen muestra la terminal que recibe la cadena ASCII en inglés.

Finalmente, se observa la señal del *ring oscillator* después de ejecutar el código en la

Raspberry Pi 4B:

Figura 217. Señal del *ring oscillator* después de ejecutar el código en la Raspberry Pi 4B



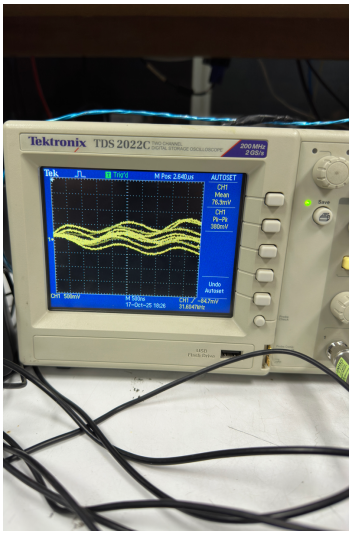
Nota. La imagen muestra la señal del *ring oscillator* después de ejecutar el código en la Raspberry Pi 4B.

La Figura 217 muestra que el *ring oscillator* está funcionando correctamente, ya que se observa una señal oscilatoria a comparación de la señal antes de ejecutar el código de la Figura 213.

En base a la Figura 217, puede obtenerse una frecuencia aproximada del *ring oscillator* de 235.29 kHz. Esta es una frecuencia alta de operación esperable, ya que el *ring oscillator* está compuesto por 13 inversores y 1 compuerta OR. Dichas compuertas introducen un pequeño *delay* que se refleja en la frecuencia de operación del *ring oscillator*.

No obstante, también se muestra a continuación una imagen del osciloscopio tomada desde fuera sin OpenChoice:

Figura 218. Señal del *ring oscillator* observada desde el osciloscopio

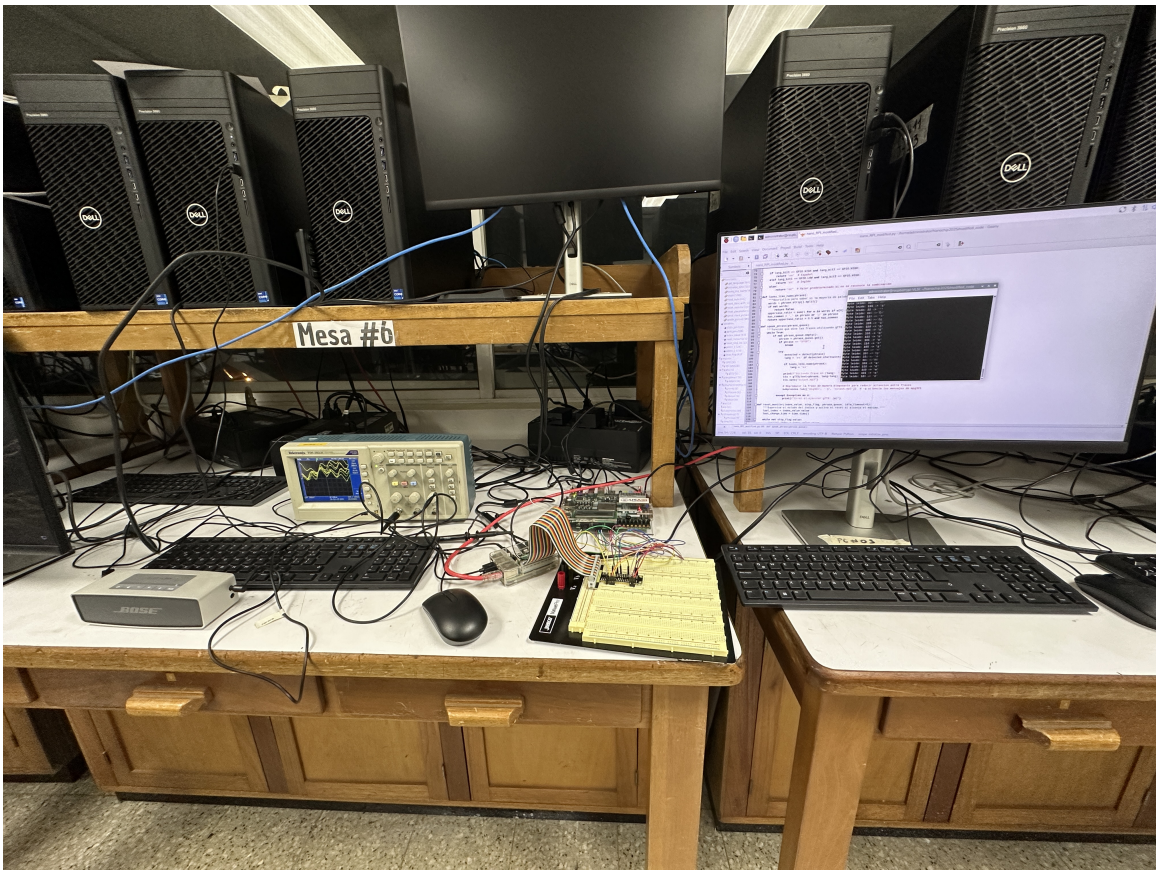


Nota. La imagen muestra la señal del *ring oscillator* observada desde el osciloscopio.

Al comparar la Figura 217 con la Figura 218 se puede observar que ambas señales son oscilatorias, no obstante, la señal de la Figura 218 muestra varias oscilaciones rápidas, lo que supone que el *ring oscillator* está funcionando a una frecuencia alta, como se mencionó anteriormente. Sin embargo, la señal de la Figura 218 indica que si bien el *ring oscillator* está funcionando, la señal no es tan limpia y suave como la de la Figura 217. Se recomienda que a futuro se profundice más en este aspecto para mejorar la calidad de la señal del *ring oscillator*.

Finalmente, el montaje completo en funcionamiento de la experimentación se ve a continuación:

Figura 219. Verificación experimental de la síntesis lógica del nanochip "El Gran Jaguar"



Nota. La imagen muestra la verificación experimental de la síntesis lógica del nanochip "El Gran Jaguar".

A través de las evidencias presentadas en este capítulo, se comprueba que la síntesis lógica del nanochip "El Gran Jaguar" en 65 nm con librerías de TSMC es válida, gracias a la emulación en un entorno físico como lo es el FPGA Xilinx Virtex-5 Genesys.

CAPÍTULO 14

Automatización para la instalación de las aplicaciones de Synopsys

En el capítulo de a continuación se presenta el *script* en Bash creado para eficientizar la descarga de las aplicaciones de Synopsys. Este *script* cuyo nombre es `Synopsys_Apps_Inst.sh`, trabaja en conjunto con otro creado en el año 2022 por la estudiante Paola Mendizábal (`APPS.sh`) [20].

El *script* `Synopsys_Apps_Inst.sh` contiene tres funcionalidades:

1. Creación de dependencias iniciales para la descarga e instalación de cualquier aplicación de Synopsys. Esto incluye la adición del servidor de licencias, la creación de carpetas importantes y configuración de permisos.

2. Descarga e instalación del Synopsys Installer. El instalador de Synopsys es necesario para poder instalar después cualquier aplicación de la empresa (Design Compiler, ICC2, Verdi, etc.). El *script* se encarga de buscar si ya hay alguna versión del instalador, pregunta si se desea instalar alguna otra versión, descarga el instalador solicitado del servidor de Synopsys a través de SSH y ejecuta el archivo para instalar el Synopsys Installer.

3. Descarga e instalación de alguna aplicación de Synopsys. En este caso se llama al *script* `APPS.sh` para cumplir con esta función donde se contacta al servidor de Synopsys a través de SSH, se descarga la aplicación y la versión especificada, se corre el Synopsys Installer para instalar la aplicación y se agrega la variable de entorno.

El *script* `APPS.sh` fue editado para corregir errores de comunicación SSH con el servidor de Synopsys. El código del *script* se puede encontrar en la Subsección 18.4.1 de los Anexos.

Por otra parte, el *script* `Synopsys_Apps_Inst.sh` se encuentra en la Subsección 18.4.2 de los Anexos. A continuación se explica su funcionamiento:

14.1. Explicación general del *script* Synopsys_Apps_Inst.sh

El *script* `Synopsys_Apps_Inst.sh` fue creado tomando como referencia el *script* `Instalador_APPs_Synopsys.sh` desarrollado por el estudiante Kevin Hernández y el *script* `APPS.sh` desarrollado por la estudiante Paola Mendizábal [24] [20].

El *script* de este trabajo `Synopsys_Apps_Inst.sh`, está estructurado en varias secciones principales:

- **Configuración:** define variables de usuario y control necesarias para la ejecución del *script*.
- **Funciones Generales:** contiene varias funciones que realizan tareas específicas como verificar entradas, descargar aplicaciones, verificar checksums, etc.
- **Función de Dependencias Iniciales:** función que configura el entorno necesario para la instalación de aplicaciones de Synopsys, incluyendo la adición del servidor de licencias y la creación de carpetas.
- **Función de Descarga e instalación del Synopsys Installer:** función que maneja la descarga e instalación del Synopsys Installer, incluyendo la verificación de versiones disponibles y la ejecución del instalador.
- **Función de Descarga e instalación de Synopsys *Apps*:** función que utiliza el *script* `APPS.sh` para descargar e instalar aplicaciones específicas de Synopsys.
- **Main Principal:** un bucle que presenta un menú al usuario para seleccionar las diferentes opciones disponibles (configurar dependencias, instalar el Synopsys Installer, instalar aplicaciones, salir).

Al realizar su ejecución, el *script* guía al usuario a través de un menú interactivo para seleccionar las acciones deseadas.

14.1.1. Función de dependencias iniciales

La función `Dependencias_Iniciales` realiza las siguientes tareas:

1. Detecta el usuario y grupo actual para configurar permisos correctamente.
2. Agrega la configuración del servidor de licencias de Synopsys al archivo `.bashrc`.
3. Crea carpetas necesarias en el sistema de archivos, como `/usr/synopsys` y `/Desktop/Folder_de_Trabajo`.
4. Configura permisos adecuados para las carpetas creadas.
5. Instala dependencias necesarias utilizando el gestor de paquetes `dnf`.
6. Crea un archivo de estado para verificar si ya se han instalado las dependencias iniciales con anterioridad.

14.1.2. Función de descarga e instalación del Synopsys Installer

La función `Install_Synopsys_Installer` se encarga de:

1. Verificar si el Synopsys Installer ya está instalado en el sistema.
2. Si no está instalado o el usuario desea reinstalarlo, muestra un listado de versiones disponibles del Synopsys Installer.
3. Solicita credenciales de acceso al servidor de Synopsys si no están definidas (SolvNetPlus).
4. Descarga los archivos necesarios desde el servidor utilizando la función `descarga_apps`.
5. Verifica la integridad de los archivos descargados mediante la función `chcksm`. Esta función compara los checksums de los archivos descargados con los proporcionados en el servidor.
6. Busca el archivo instalador `.run`, le asigna permisos de ejecución y crea el directorio de instalación.
7. Actualiza las variables de entorno en el archivo `.bashrc` para incluir el Synopsys Installer en el PATH.

14.1.3. Función de descarga e instalación de Synopsys *Apps*

La función `Install_Synopsys_Apps` se encarga de:

1. Llamar al *script* `APPS.sh` para gestionar la instalación de las aplicaciones de Synopsys.
2. Verificar y configurar los permisos necesarios para la carpeta de destino antes de ejecutar `APPS.sh` de instalación.
3. Si los permisos son adecuados, se ejecuta el *script* `APPS.sh`. Aparece un menú interactivo para que el usuario seleccione si desea instalar EDKs (*Educational Design Kits*), PDKs (*Process Design Kits*) o aplicaciones.
4. En base a la selección del usuario, `APPS.sh` procede a descargar la aplicación o kit seleccionado en la versión deseada.
5. En el caso de las aplicaciones, se ejecuta el Synopsys Installer para completar la instalación.
6. Luego, se agrega la variable de entorno correspondiente al archivo `.bashrc` para que la aplicación o librería instalada esté disponible en el sistema.
7. Finalmente, se notifica al usuario que la instalación ha sido completada.

14.2. Ejecución del *script* Synopsys_Apps_Inst.sh

Antes de ejecutar el *script* es importante crear un usuario con permisos de administrador (sudo) en la máquina virtual. Esto se debe a que el *script* requiere permisos elevados para crear carpetas en el sistema y modificar archivos de configuración.

Una vez se haya creado el usuario con permisos sudo, acceder y navegar a la carpeta de descargas de este usuario (`$HOME/Downloads`). Luego, copiar tanto el *script* `Synopsys_Apps_Inst.sh`, como el archivo `APPS.sh` en dicha ubicación.

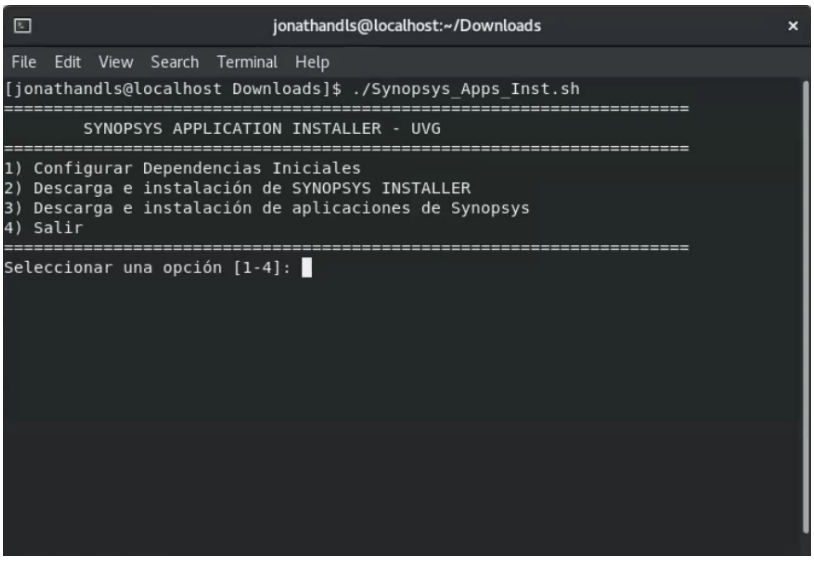
De ser necesario, dar permisos de ejecución a ambos archivos con el siguiente comando:

Cuadro 97. Comando para dar permisos de ejecución a los *scripts*

```
chmod +x Synopsys_Apps_Inst.sh APPS.sh
```

Posteriormente, abrir una terminal y ejecutar el *script* `Synopsys_Apps_Inst.sh`:

Figura 220. Ejecución del *script* `Synopsys_Apps_Inst.sh` desde la terminal



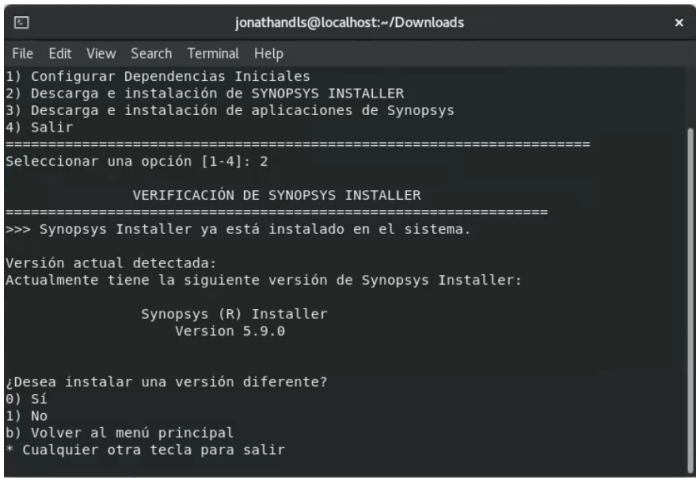
Nota. La imagen muestra la ejecución del *script* desde la terminal.

Al iniciar el *script* aparece un menú interactivo con las opciones disponibles:

1. Configurar dependencias iniciales
2. Descarga e instalación del Synopsys Installer
3. Descarga e instalación de Synopsys *Apps*
4. Salir

Para este caso, la primera opción ya fue realizada por lo que se mostrará el funcionamiento con las opciones 2 y 3. Seleccionar la opción 2 para descargar e instalar el Synopsys Installer:

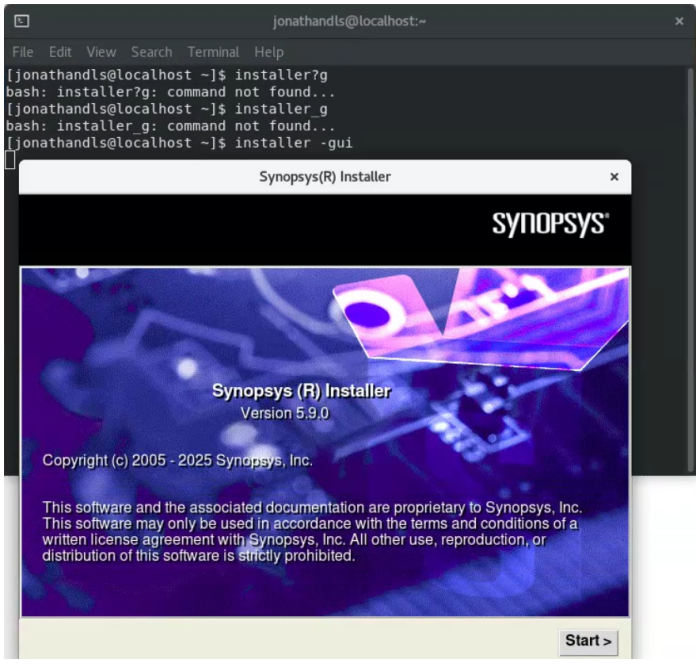
Figura 221. Selección de la opción 2 del *script*



Nota. La imagen muestra la selección de la opción 2 del *script*.

Como se puede notar en la figura anterior, el *script* detecta que existe una versión del Synopsys Installer ya instalada.

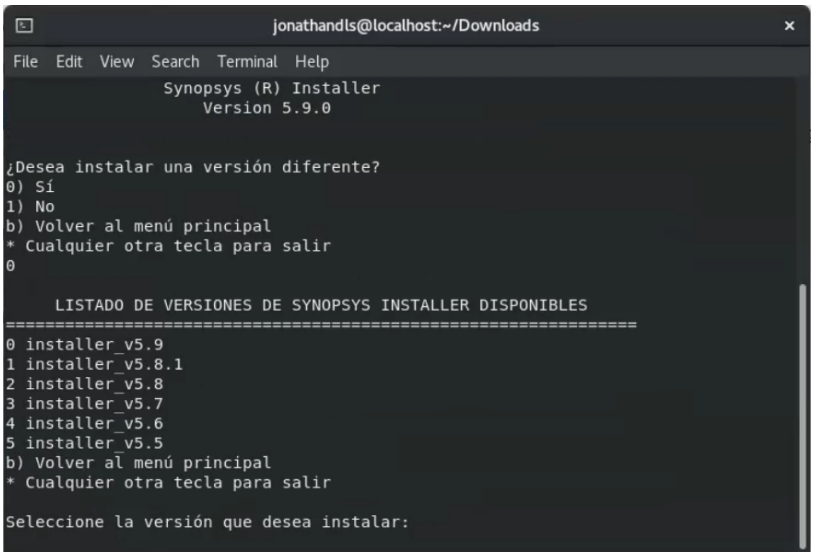
Figura 222. Verificación de la instalación previa del Synopsys Installer



Nota. La imagen muestra la verificación de la instalación previa del Synopsys Installer.

En este caso, se desea instalar otra versión del Synopsys Installer, por lo que se selecciona la opción 0 y presiona *enter*.

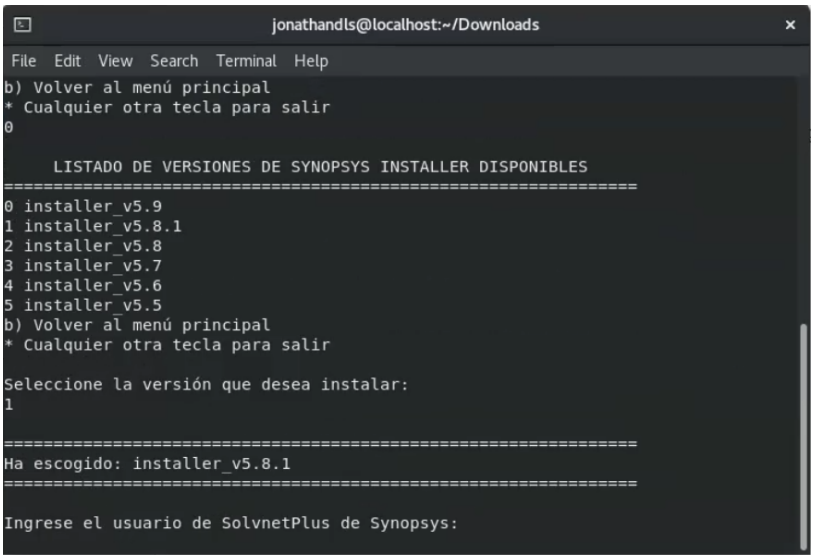
Figura 223. Versiones disponibles del Synopsys Installer



Nota. La imagen muestra las versiones disponibles del Synopsys Installer.

Para este caso que se desea demostrar la funcionalidad del *script*, se selecciona la versión 5.8.1:

Figura 224. Selección de la versión del Synopsys Installer

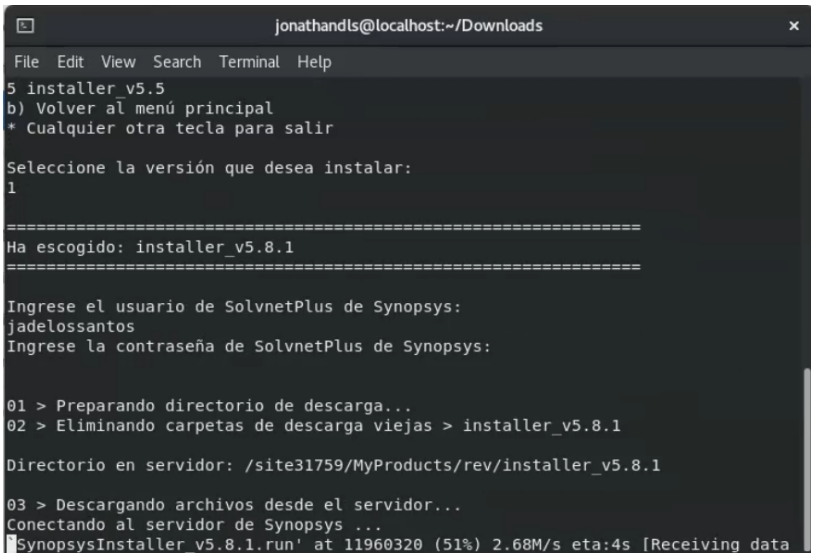


Nota. La imagen muestra la selección de la versión del Synopsys Installer.

Luego, se solicitan las credenciales de acceso al servidor de Synopsys (SolvNetPlus),

ingresarlas para continuar con la descaga e instalación.

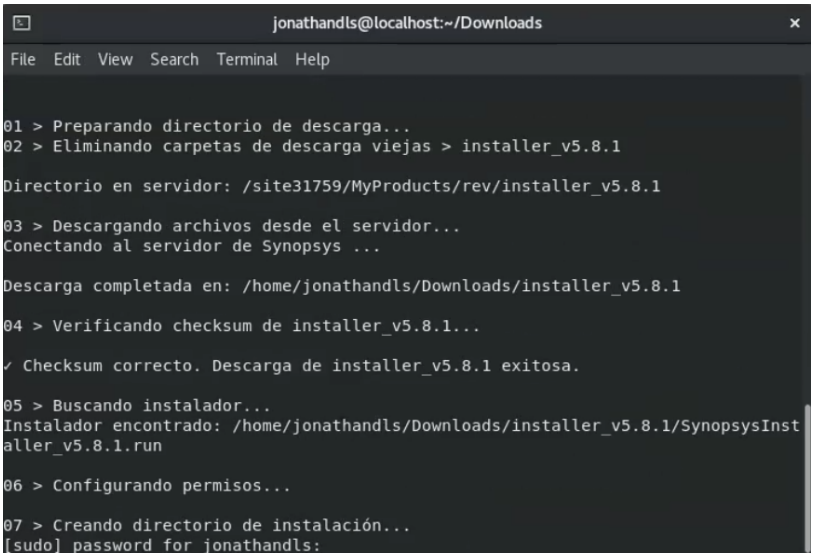
Figura 225. Descarga del Synopsys Installer



Nota. La imagen muestra la descarga del Synopsys Installer.

Tal y como se había mencionado, al instalar el Synopsys Installer se necesita permiso de administrador (sudo):

Figura 226. Descarga del Synopsys Installer

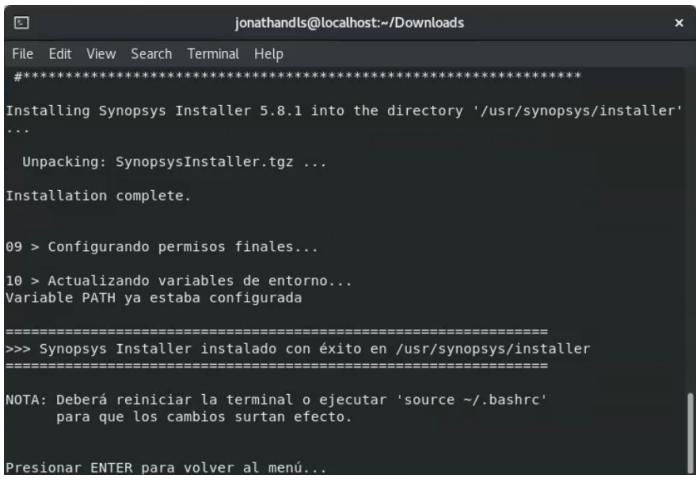


Nota. La imagen muestra la descarga del Synopsys Installer.

Una vez ingresada la contraseña del usuario con permisos sudo, el *script* finaliza con la

instalación del Synopsys Installer:

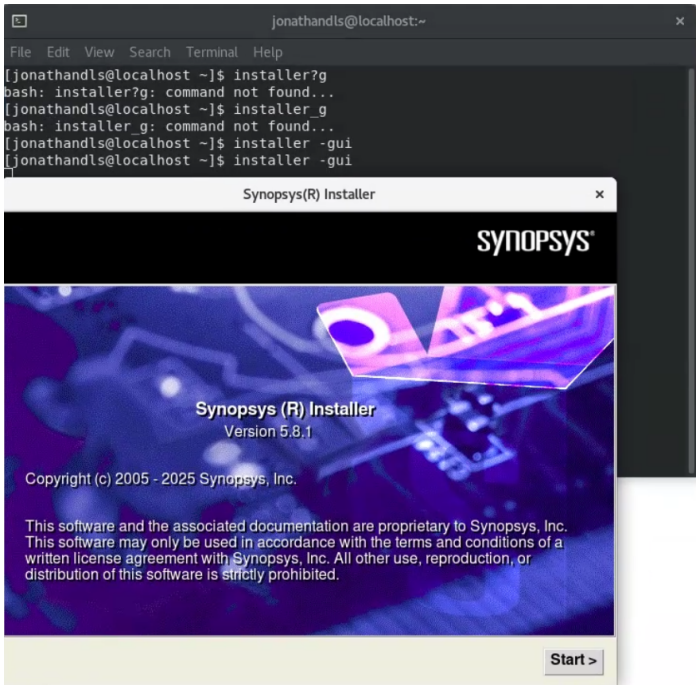
Figura 227. Descarga del Synopsys Installer



Nota. La imagen muestra la descarga del Synopsys Installer.

A continuación, se demuestra que la descarga e instalación de la nueva versión del Synopsys Installer fue exitosa:

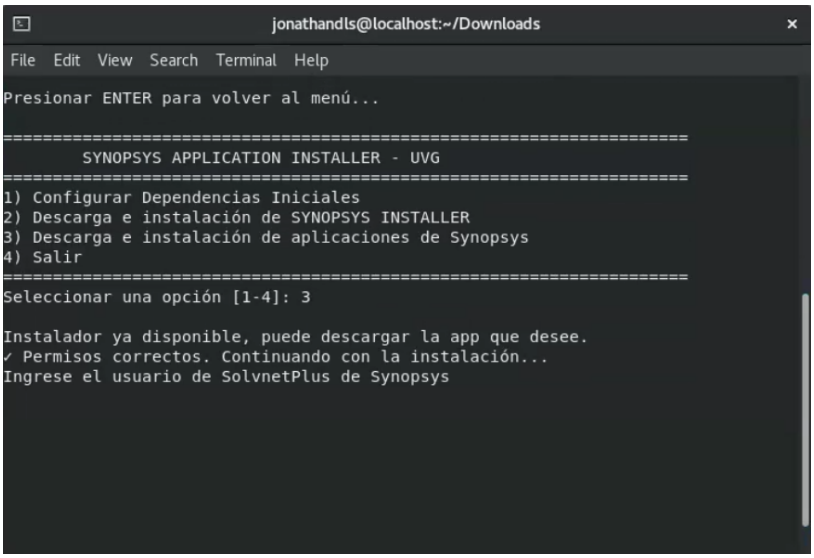
Figura 228. Ejecución de la nueva versión del Synopsys Installer



Nota. La imagen muestra la ejecución de la nueva versión del Synopsys Installer.

Ahora, seleccionar la opción 3 para descargar e instalar alguna aplicación de Synopsys:

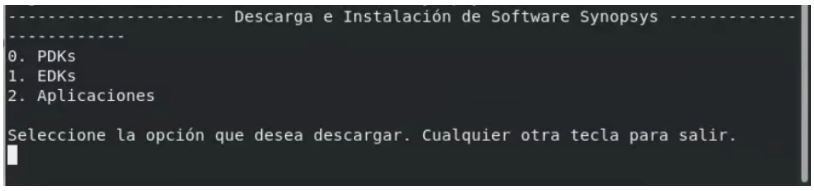
Figura 229. Selección de la opción 3 del *script*



Nota. La imagen muestra la selección de la opción 3 del *script*.

Al igual que en el caso anterior, se solicitan las credenciales de acceso al servidor de Synopsys (SolvNetPlus), ingresarlas para continuar.

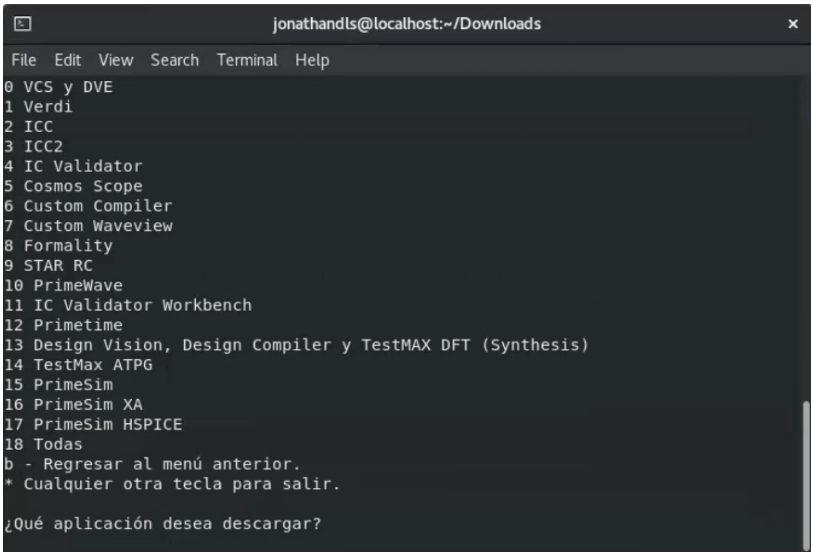
Figura 230. Opciones para descargar con la opción 3 del *script*



Nota. La imagen muestra las opciones para descargar con la opción 3 del *script*.

Seleccionar la opción 2 para descargar e instalar alguna aplicación de Synopsys:

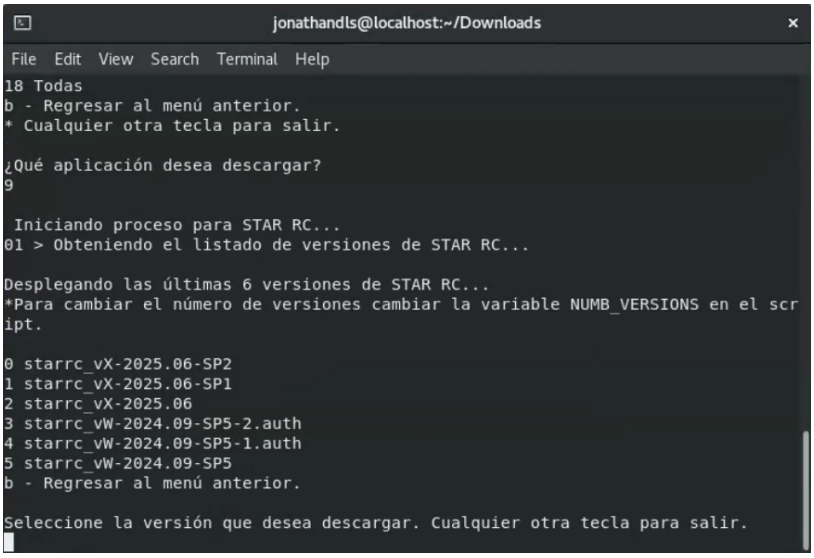
Figura 231. Selección de la opción 2 para descargar aplicaciones



Nota. La imagen muestra la selección de la opción 2 para descargar aplicaciones.

En este caso, se va a seleccionar la aplicación StarRC para descargar e instalar:

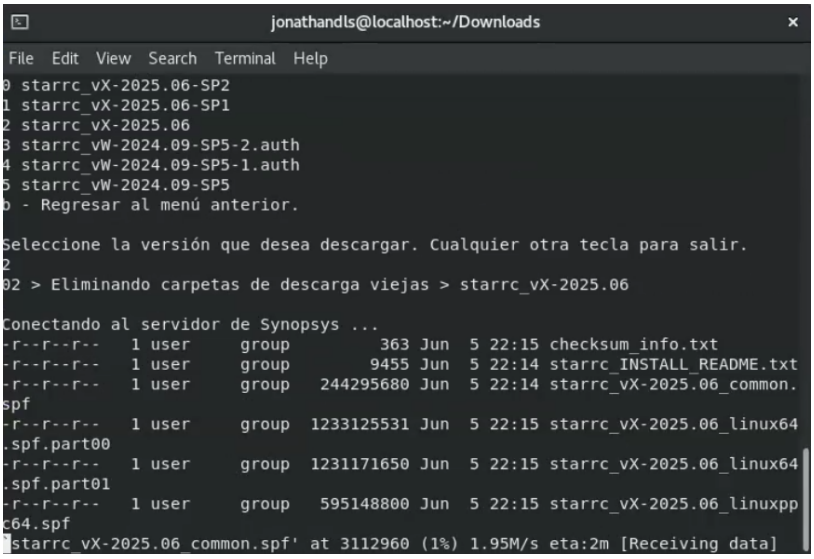
Figura 232. Versiones disponibles de StarRC



Nota. La imagen muestra las versiones disponibles de StarRC.

Seleccionar la versión X-2025.06 de StarRC (opción 2):

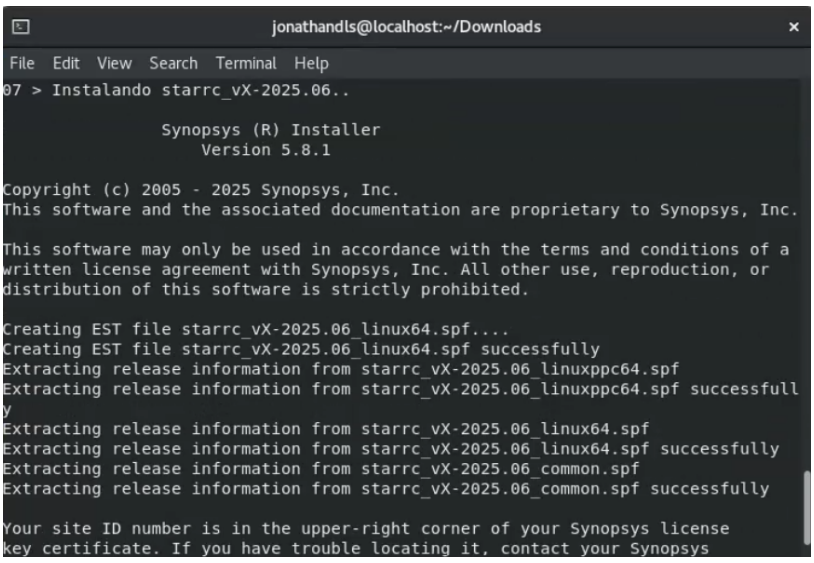
Figura 233. Descarga de StarRC



Nota. La imagen muestra la descarga de StarRC.

El proceso de los archivos de descarga puede tardar dependiendo de la aplicación. Una vez finalizada la descarga, se procede a instalar StarRC:

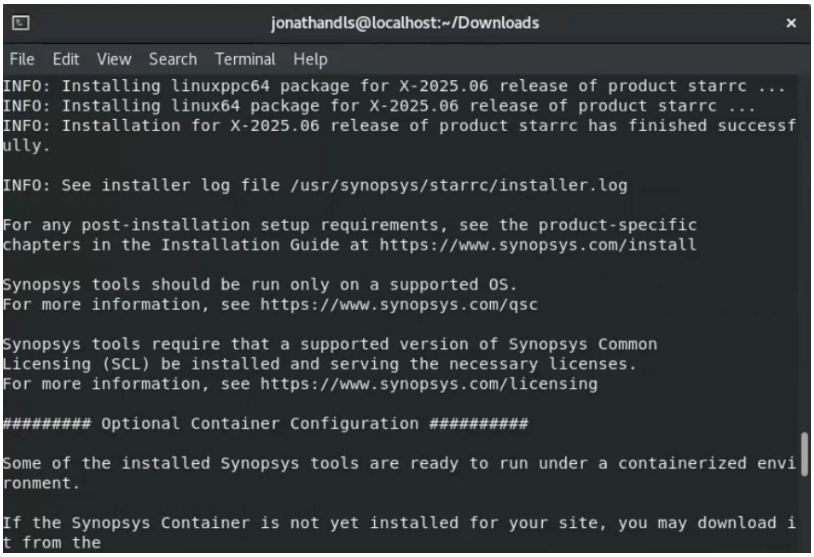
Figura 234. Instalación de StarRC



Nota. La imagen muestra la instalación de StarRC.

Es posible que para la instalación se solicite el *site ID number* del servidor de licencias. Ingresar el número correspondiente para continuar. Posteriormente, se pregunta si se desea continuar con la instalación, ingresar *yes* para proceder:

Figura 235. Finalización de la instalación de StarRC



Nota. La imagen muestra la finalización de la instalación de StarRC.

La figura anterior muestra que la instalación de StarRC fue completada exitosamente. De esta manera se demuestra el funcionamiento del *script* `Synopsys_Apps_Inst.sh` para la descarga e instalación del Synopsys Installer y alguna aplicación de Synopsys.

CAPÍTULO 15

Conclusiones

- Se desarrolló el *script* en TCL `script_dc.tcl`, el cual permite realizar la síntesis lógica de los diseños con o sin pads, a través de solamente un archivo. Al necesitar un solo archivo aprovechando los comandos al máximo, se optimiza el flujo de diseño, existe mayor facilidad de uso y se prescinde de la necesidad de sintetizar el *core* antes para poder sintetizar luego el diseño con pads. El *script* crea las carpetas necesarias de forma automática, analiza los archivos de diseño, sintetiza con optimizaciones de *timing* y área, genera reportes útiles y crea los archivos de salida necesarios para la etapa de *Back-End*. Se incluyeron *constraints* de ejemplo comentados que pueden ser útiles para diseños futuros. Mencionar que el nanochip con pads después de la síntesis ocupa aproximadamente el 10 % del área total disponible y la potencia total está alrededor de los 13.51 mW.
- Se implementó una nueva jerarquía de directorios para la organización de los archivos de los proyectos de síntesis lógica. Esta jerarquía permite una mejor organización y facilita la localización de todos los archivos en el flujo de diseño, siguiendo las recomendacion de Synopsys. Al tener directorios con nombres más intuitivos para cada tipo de archivo, se optimiza el proceso y se mejora la experiencia del diseñador.
- Se realizó exitosamente la síntesis lógica de los siguientes circuitos: compuerta NOT, sumador *Carry Look-Ahead* de 4 bits, *Ripple Carry Counter* de 4 bits, *ring oscillator*, ALU de 4 bits con un *ring oscillator* y el nanochip "El Gran Jaguar". Para el caso concreto del nanochip, se actualizó el código en Python `escriturafinal_vp3.py` para generar el archivo en Verilog del *core*, adaptándolo a las especificaciones del proyecto. A través de simulaciones antes y después de la síntesis, fue posible validar el comportamiendo de las señales y conocer los aspectos a tomar en cuenta tanto para los diseños combinacionales, como para los secuenciales. Además, las simulaciones permitieron validar el funcionamiento de los circuitos sintetizados con y sin pads de los circuitos.
- Se logró junto al equipo de *Back-End* describir en Verilog las celdas (*blackboxes*) del *core* sintetizado del nanochip "El Gran Jaguar" en su adaptación a la tecnología de 65 nm. Esto permitió simular e implementar el diseño del *core* del nanochip en el FPGA Xilinx Virtex-5 Genesys.

- Se verificó experimentalmente el funcionamiento del *core* sintetizado del nanochip "El Gran Jaguar" en su adaptación a la tecnología de 65 nm de TSMC. La validación se llevó a cabo mediante la conexión del FPGA Xilinx Virtex-5 Genesys con una Raspberry Pi 4B, lo que permitió evaluar el desempeño del sistema en condiciones reales de operación antes de mandar a fabricación el nanochip. Los resultados experimentales confirmaron el correcto funcionamiento del diseño, al observarse la recepción adecuada de los caracteres, la reproducción del audio en una bocina y el comportamiento esperado del *ring oscillator* operando a una frecuencia aproximada de 235.29 kHz, observado a través de mediciones realizadas con un osciloscopio.

- Se estandarizó el código de Python para la Raspberry Pi utilizando funciones para el reconocimiento automático del idioma y así ajustar la pronunciación de la voz del *Google Text-to-Speech*. Esto permite que para el caso donde se selecciona la codificación del texto en inglés, se pronuncien de forma correcta los nombres en español y las partes que se encuentran en inglés. Además, la conversión de texto a voz ya no se detiene al terminar el texto. El nanochip funciona de forma continua, por lo que se necesita que el código funcione hasta que se indique lo contrario. Aunado a lo anterior, el código sigue funcionando a través de las técnicas de multiprocesos y multihilos.

- Se consiguió mejorar el proceso para la descarga automática de las aplicaciones de Synopsys. Se creó el *script* en Bash `Synopsys_Apps_Inst.sh`, el cual crea las dependencias iniciales necesarias, logra la descarga e instalación del Synopsys Installer de forma automática y permite la descarga e instalación de las aplicaciones de Synopsys llamando al *script* `APPS.sh`. A diferencia del *script* del año anterior que no funcionaba, este año se logró la conexión con el servidor de Synopsys a través de SSH, la descarga de la versión de StarRC deseada y la automática instalación de la misma. Además, fue necesario editar el *script* `APPS.sh` debido a errores en la conexión SSH. Con el nuevo *script* se facilita la configuración de nuevos entornos de trabajo para el futuro.

- Se documentó todo el proceso de síntesis lógica, desde el diseño de los archivos RTL hasta la simulación de cada circuito sintetizado con pads. Se incluyó un diagrama de flujo que explica cómo se construye el *script* de síntesis lógica, así como también se crearon diagramas que brindan una explicación detallada de cada paso del proceso de síntesis. Además, se documentó todo el proceso de simulación e implementación del *core* del nanochip "El Gran Jaguar" en el FPGA Xilinx Virtex-5 Genesys. Esta documentación es esencial para futuros trabajos y para cualquier persona que desee comprender o replicar el proceso. Por último, se documentó el proceso de descarga e instalación automática de las aplicaciones de Synopsys.

CAPÍTULO 16

Recomendaciones

- Se recomienda basarse en el *script* `script_dc.tcl` para futuras síntesis de cualquier índole. La construcción del *script* es apoyada en base a recomendaciones ofrecidas en el *LATAM Digital IC Design Bootcamp* realizado en Ciudad de Guatemala en el 2025 y documentación disponible en SolvNet de Synopsys. Además, para futuras síntesis que requieran optimizaciones de tiempo, el *script* `script_dc.tcl` ya tiene algunos ejemplos comentados incluidos que se podrían implementar.
- Se sugiere profundizar en el estudio de las herramientas de Synopsys. Es de mencionar que la herramienta de Verdi posee extensas funcionalidades y capacidades de *debugging* que no se alcanzaron a explorar en su totalidad. Además, se recomienda investigar más a fondo la herramienta Formality para poder realizar la verificación de que el *gate-level netlist* sea matemáticamente equivalente al diseño RTL original para circuitos más complejos.
- Asimismo, dado que la herramienta de Synopsys PrimeTime puede ser utilizada para realizar análisis de *timing* pre-síntesis y post-síntesis. Se recomienda explorar la aplicación para su implementación en futuros circuitos que requieran su uso. Con ella se puede obtener reportes detallados sobre el rendimiento temporal de cualquier diseño.
- Por otro lado, más recientemente Synopsys ha lanzado una herramienta de síntesis lógica llamada DC NXT, la cual promete mejoras en el flujo de diseño al incorporar librerías y archivos de información física directamente para realizar una síntesis lógica *physically aware*. Esto mejora el posicionamiento de las celdas para optimizaciones varias, basándose en un *pre-floorplan*. Se recomienda explorar esta herramienta para futuros trabajos de síntesis lógica.
- Si bien la verificación experimental del *core* del nanochip "El Gran Jaguar" se realizó exitosamente en el FPGA Xilinx Virtex-5 Genesys, fue posible observar un comportamiento errático del *ring oscillator* con el osciloscopio. Se propone investigar más a fondo las posibles causas de este comportamiento, y su vez, diseñar un *ring oscillator* con un mayor número de compuertas NOT para aumentar el *delay* de la señal y que así sea posible visualizar mejor el comportamiento en el osciloscopio.

- Para el futuro sería útil implementar capacidades de simulación y verificación (VCS, Verdi, Formality, etc.) de forma automatizada en el mismo flujo de la síntesis lógica. Esto facilitaría la visualización de las señales en una misma interfaz y la detección de errores, evitando la necesidad de interactuar de forma manual con varias terminales y comandos.

- Es posible mejorar el *script* `Synopsys_Apps_Inst.sh` para incluir las funciones del *script* `APPS.sh`. De esta forma, se lograría tener en un solo *script* la capacidad de configuración del entorno y la descarga e instalación de cualquier herramienta de Synopsys de forma automática. Aunado a ello, sería ideal modificar el proceso de obtención de los nombres de las herramientas, ya que actualmente se define desde el inicio la variable `apps_real` con el nombre de las aplicaciones. Esto puede generar errores si Synopsys decide cambiar los nombres en el futuro.

CAPÍTULO 17

Referencias

[1] J. A. de los Santos, «Diseño de un sumador/restador completo de 32 bits con tecnología CMOS en un proceso de 28 nanómetros usando aplicaciones de diseño de la empresa Synopsys,» Tesis de Licenciatura en Ingeniería Electrónica, Universidad del Valle de Guatemala, 2014.

[2] S. H. Rubio, «Definición del Flujo de Diseño para Fabricación de un Chip con Tecnología VLSI CMOS,» Tesis de Licenciatura Ingeniería Electrónica, Universidad del Valle de Guatemala, 2019.

[3] L. A. Nájera, «Implementación de circuitos sintetizados a nivel netlist a partir de un diseño en lenguaje descriptivo de hardware como primer paso en el flujo de diseño de un circuito integrado,» Tesis de Licenciatura Ingeniería Electrónica, Universidad del Valle de Guatemala, 2019.

[4] C. Cruz, «Ejecución y utilización de un flujo de diseño para el desarrollo de un chip con tecnología nanométrica: Extracción de componentes parásitos y simulaciones en HSPICE,» Universidad del Valle de Guatemala, Trabajo de Graduación, 2020.

[5] S. Cardona, «Mejoramiento del proceso de síntesis lógica llevada a cabo para la elaboración de un circuito integrado a escala nanométrica,» Facultad de Ingeniería, Universidad del Valle de Guatemala, Trabajo de Graduación en modalidad de Tesis, 2020.

[6] M. G. Flores, «Corrección de anillo de entradas/salidas y pruebas de antenna y ERC para la definición del flujo de diseño del primer chip con tecnología nanométrica desarrollado en Guatemala,» Facultad de Ingeniería, Universidad del Valle de Guatemala, Trabajo de Graduación en modalidad de Tesis, 2020.

[7] M. Sibrian, «Verificación de reglas de diseño (DRC) para el desarrollo de un flujo funcional de un circuito integrado con tecnología nanométrica,» Universidad del Valle de Guatemala, Tesis de Graduación, 2020.

[8] J. Girón, «Etapa de verificación física de Diseño en Silicio vs. Esquemático (LVS) en el flujo de diseño para un chip a nanoescala,» Universidad del Valle de Guatemala, Trabajo de Graduación, 2020.

[9] E. O. Torres, «Diseño de un circuito integrado con tecnología de 180 nm usando librerías de diseño de TSMC: Ejecución y simulación para la etapa de síntesis lógica,» Universidad del Valle de Guatemala, Trabajo de Graduación en modalidad de Tesis, 2021.

[10] L. Abadía, «Posicionamiento e interconexión entre componentes de un circuito sintetizado para el flujo de diseño de un circuito a escala nanométrica utilizando la herramienta de IC Compiler,» Facultad de Ingeniería, Universidad del Valle de Guatemala, Tesis de Graduación, 2021.

[11] J. R. Orozco, «Diseño de un circuito integrado con tecnología de 180 nm usando librerías de diseño de TSMC: ejecución de la fase de verificación física Layout vs Schematic (LVS),» Universidad del Valle de Guatemala, Tesis de Graduación, 2021.

[12] J. E. Shin, «Diseño de un circuito integrado con tecnología de 180 nm usando librerías de diseño de TSMC: ejecución de la síntesis física, validación de reglas eléctricas y corrección de errores obtenidos,» Universidad del Valle de Guatemala, Tesis de Graduación, 2021.

[13] H. Ovalle, «Diseño de un circuito integrado con tecnología de 180 nm utilizando librerías de diseño de TSMC: Implementación de un alternativo flujo de diseño proporcionado por Synopsys con las herramientas de Formality y Design Compiler,» Universidad del Valle de Guatemala, Trabajo de Graduación, 2022.

[14] J. Ponce, «Diseño de un circuito integrado con tecnología de 180 nm utilizando librerías de diseño de TSMC: Implementación de un alternativo flujo de diseño proporcionado por Synopsys con las herramientas PrimeTime y TetraMAX,» Universidad del Valle de Guatemala, Tesis de Graduación, 2022.

[15] D. Equité, «Diseño e implementación de interfaz gráfica de la automatización de las fases de diseño de un circuito integrado y uso avanzado de IC Compiler II,» Universidad del Valle de Guatemala, Trabajo de Graduación, 2022.

[16] A. E. Aguilar, «Automatización de la etapa de síntesis lógica y creación de archivos Verilog para pruebas físicas en un FPGA Genesys Xilinx Virtex-5 LX50T y automatización de la verificación extracción de parásitos,» Facultad de Ingeniería, Universidad del Valle de Guatemala, Trabajo de Graduación, 2022.

[17] C. A. Letona, «Diseño de un circuito integrado con tecnología de 180 nm usando librerías de diseño de TSMC: uso avanzado de StarRC para la generación de un archivo HSPICE con componentes parásitos para su correcta simulación,» Universidad del Valle de Guatemala, Tesis de Graduación, 2022.

[18] I. A. Arévalo, «Diseño de un circuito integrado con tecnología de 180 nm utilizando librerías de diseño de TSMC: Simulación en PrimeSim y análisis de errores,» Universidad del Valle de Guatemala, Tesis de Graduación, 2022.

[19] S. Schwendener, «Automatización de las verificaciones físicas de un circuito integrado con tecnología de 180 nm utilizando librerías de diseño de TSMC,» Universidad del Valle de Guatemala, Trabajo de Graduación, 2022.

[20] P. A. Mendizábal, «Automatización del proceso de instalación de software de Synopsys e implementación de mejoras utilizando contenedores,» Facultad de Ingeniería, Universidad del Valle de Guatemala, Trabajo de Graduación en modalidad de Tesis, 2022.

[21] E. G. Mancio, «Implementación y verificación de la funcionalidad de código obtenido durante la síntesis lógica de arquitectura del nano chip Gran Jaguar utilizando una plataforma de desarrollo con un FPGA Xilinx Virtex-5 Genesys,» Universidad del Valle de Guatemala, Trabajo de Graduación, 2022.

[22] L. R. Gómez, «Procesamiento de las señales generadas por un circuito integrado con tecnología de 180nm usando librerías de diseño de TSMC montado en un FPGA Digilent Genesys Board demostrando el funcionamiento mediante una aplicación y sintetizador digital,» Universidad del Valle de Guatemala, Trabajo de Graduación, 2022.

[23] N. Prado, «Diseño de un circuito integrado con tecnología de 180 nm, utilizando las librerías de diseño de TSMC y las librerías educativas de Synopsys: corrección de errores de densidad y polisilicio, verificación DRC y antena,» Universidad del Valle de Guatemala, Tesis de Graduación, 2023.

[24] K. Hernández, «Automatización de la etapa de síntesis lógica, optimización del proceso de instalación de las aplicaciones de Synopsys y documentación de los pasos de Front-End y Back-End para el diseño del Circuito Integrado El Gran Jaguar,» Universidad del Valle de Guatemala, Trabajo de Graduación, 2024.

[25] L. Ruiz, «Diseño y fabricación de un circuito integrado con tecnología de 65 nm usando librerías de diseño de TSMC: pruebas finales de funcionamiento en HSPICE, generación y documentación de la síntesis física, verificaciones de Antena y DRC,» Universidad del Valle de Guatemala, Trabajo de Graduación, 2024.

[26] D. Alvarado, «Diseño de un circuito integrado con tecnología de 65 nm utilizando librerías de diseño de TSMC: Pruebas de LVS, ERC y extracción de parásitos,» Universidad del Valle de Guatemala, Tesis de Graduación, 2024.

[27] K. Rac, «Simplificación del hardware complementario de funcionamiento del Nanochip "El Gran Jaguar" y optimización de software de lectura de Python usando multihilos y multiprocesamiento,» Universidad del Valle de Guatemala, Trabajo de Graduación, 2024.

[28] J. Mora, «Automatización del Flujo de Diseño de un Nanochip Elaborado con Librerías de Diseño de TSMC de Tecnología de 65 Nanómetros,» Universidad del Valle de Guatemala, Trabajo de Graduación, 2024.

[29] V. Melikyan, *IC Design Flow*, Synopsys, Inc. - SolvNetPlus, 2019.

[30] V. Melikyan, *Synopsys Design Flow Tutorial*, Synopsys, Inc. - SolvNetPlus, 2024.

[31] R. Rao, *ASIC Design: A Step-by-Step Guide from Specification to Silicon*, `https://www.wevolver.com/article/the-ultimate-guide-to-asic-design-from-concept-to-production`, 2025.

[32] K. Chauhan, *ASIC Design Flow in VLSI Engineering Services - A Quick Guide*, `https://www.einfochips.com/blog/asic-design-flow-in-vlsi-engineering-services-a-quick-guide/`, 2023.

[33] V. Melikyan, *Synopsys EDA Tool Flow for Front-End Digital IC Design*, Synopsys, Inc. - SolvNetPlus, 2025.

[34] I. Synopsys, *VCS User Guide X-2025.06-SP1*, SolvNetPlus, 2025.

[35] I. Synopsys, *Library Compiler User Guide X-2025.06-SP1*, SolvNetPlus, 2025.

[36] I. Synopsys, *Design Compiler User Guide X-2025.06-SP2*, SolvNetPlus, 2025.

[37] I. Synopsys, *Design Vision User Guide X-2025.06-SP2*, SolvNetPlus, 2025.

[38] I. Synopsys, *Verdi User Guide X-2025.06-SP1*, SolvNetPlus, 2025.

[39] H. Pullman, *Genesys^{TM} FPGA Board Reference Manual.* https://digilent.com/reference/_media/reference/programmable-logic/genesys/genesys_rm.pdf, 2016.

[40] XILINX, *ISE In-Depth Tutorial.* https://docs.amd.com/v/u/en-US/ise_tutorial_ug695, 2012.

[41] XILINX, *Synthesis and Simulation Design Guide.* https://docs.amd.com/v/u/en-US/sim, 2012.

[42] XILINX, *INDUSTRY-PROVEN SOLUTIONS FOR XILINX ALL PROGRAMMABLE DEVICES*, https://www.xilinx.com/publications/prod_mktg/ISE_sellsheet.pdf, 2012.

[43] H. Pullman, *ISim User Guide.* https://docs.amd.com/v/u/en-US/plugin_ism, 2016.

[44] R. Pi, *Raspberry Pi 4 Model B*, https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/, 2020.

[45] R. Pi, *BCM2711 ARM Peripherals*, https://datasheets.raspberrypi.com/bcm2711/bcm2711-peripherals.pdf, 2020.

[46] S. Aldhaher, *Wire Bonding: Modern Applications, Technology Trends and Cost Considerations*, https://resources.altium.com/p/wire-bonding-modern-applications-technology-trends-and-cost-considerations, 2024.

CAPÍTULO 18

Anexos

En estos anexos se incluyen los códigos de mayor longitud del trabajo necesarios para entender el contexto completo.

1. De primero se presenta el código en Python v3 para generar el Verilog del *core* del nanochip "El Gran Jaguar".
2. Luego, se muestran los códigos en Verilog relacionados a la síntesis de "El Gran Jaguar" con las librerías de TSMC en 65 nm.
3. Además, se incluyen códigos relacionados con la simulación e implementación del nanochip sintetizado en el FPGA Xilinx Virtex-5 Genesys.
4. Por último, se proporcionan los *scripts* relacionados a la automatización del proceso de descarga e instalación de aplicaciones de Synopsys.

Esto guiará al lector, ya que no debe asumirse que sabe lo mismo que el autor, por algo está leyendo el trabajo. Así que sólo copiar y pegar una serie de figuras y cuadros sin un contexto es una falta de cortesía.

Los anexos incluyen material relevante para mayor claridad y profundidad de la investigación, pero distraen al lector si se colocan en el texto del informe. Pueden agregarse cartas, manuales, guías, etc. Es necesario asignar un número a cada anexo de acuerdo con el orden de mención en el desarrollo del trabajo.

18.1. Código de Python para la generación del Verilog del *core* del nanochip "El Gran Jaguar"

Cuadro 98. Archivo escriturafinal_vp3.py ubicado en el directorio `verilog_generation/`

```
# Archivo de Python v3 para generar el Verilog del nanochip "El
   ␣Gran␣Jaguar"
# Editado por: Carlos Valdes
# Fecha: Agosto 2025

import binascii
texto = "Hola,␣soy␣El␣Gran␣Jaguar,␣el␣primer␣nanochip␣elaborado
   ␣en␣Centroamerica␣por␣una␣Universidad.␣Desarrollado␣
   completamente␣en␣la␣Universidad␣del␣Valle␣de␣Guatemala␣por␣
   alumnos␣graduandos␣de␣Ingenieria␣en␣Electronica␣entre␣dos␣
   mil␣diecinueve␣y␣dos␣mil␣veinticinco.␣El␣equipo␣encargado␣de
   ␣mi␣creacion␣se␣conformo
por:␣Carlos␣Alberto␣Esquit␣Hernandez,␣Jonathan␣Alberto␣de␣los␣
   Santos␣Chonay,␣Kurt␣Emmanuel␣Kellner␣Juarez,␣Miguel␣Enrique␣
   Zea␣Arenales,␣Luis␣Alberto␣Rivera␣Estrada,␣Luis␣Arturo␣
   Najera␣Vasquez,␣Steven␣Hiram␣Rubio␣Vasquez,␣Marvin␣Geovanni␣
   Flores␣Espino,␣Matthias␣Sibrian␣Illescas.␣Julio␣Enrique␣Shin
   ␣Jo,␣Karol␣Sophia
Cardona␣Polanco,␣␣Luis␣Estuardo␣Abadia␣Lopez,␣Joel␣Andres␣
   Gonzalez␣Herrera,␣Jose␣Adrian␣Ayala␣Escobar,␣Jose␣Alejandro␣
   Ruiz␣Orozco,␣Juan␣Ricardo␣Giron␣Rubio,␣Charlie␣Ayenci␣Cruz␣
   Giron.␣Elmer␣Otoniel␣Torres␣Garza,␣Gerardo␣Cardoza,␣Antonio␣
   Altuna␣Hernandez,␣Carlos␣Andres␣Letona␣Torres,␣Helder␣
   Arnoldo␣Ovalle␣Barrios,
Jose␣David␣Ponce␣del␣Cid,␣Israel␣Antonio␣Arevalo␣Gonzalez,␣
   Allison␣Estuardo␣Aguilar␣Chocooj,␣Paola␣Alejandra␣Mendizabal
   ␣Batres,␣Diego␣Rodrigo␣Equite␣Pirir,␣Estuardo␣Geovany␣Mancio
   ␣Ruballos,␣Stefan␣Alexander␣Schwendener␣Farrington,␣Luis␣
   Ricardo␣Gomez␣Velasquez,␣Julio␣Emanuel␣Lopez␣Ozaeta.␣Noel␣
   Francisco␣Prado␣Bucaro,
Diana␣Sofia␣Alvarado␣Mota,␣Katherine␣Arletth␣Rac␣Pocon,␣Lourdes
   ␣Maria␣Ruiz␣Vasquez,␣Juan␣Pablo␣Mora␣Argueta,␣Kevin␣David␣
   Hernandez␣Guarcas.␣Jose␣Mario␣Mendez␣Rodriguez,␣Byron␣
   Estuardo␣Barrientos␣Perez,␣Luis␣Pablo␣Gustavo␣Carranza␣
   Vasquez,␣Miguel␣Alberto␣Chacon␣Ortiz␣y,␣Carlos␣Julio␣Valdes␣
   Oajaca.␣La␣idea␣de␣fabricar
un␣CHIP␣nacio␣a␣inicios␣del␣dos␣mil␣diecinueve,␣donde␣los␣
   alumnos␣graduandos␣consiguieron␣el␣apoyo␣de␣Euro-practice␣
   para␣financiar␣gran␣parte␣de␣la␣fabricacion␣de␣El␣Gran␣
   Jaguar.␣Tambien␣a␣traves␣de␣IMEC␣se␣tuvo␣acceso␣a␣librerias␣
   de␣fabricacion␣implementadas␣por␣la␣Taiwan␣Semiconductor␣
   Manufacturing␣Company␣en␣un
proceso␣de␣sesenta␣y␣cinco␣nanometros.␣Tambien␣se␣trabaja␣para␣
   automatizar␣todo␣el␣proceso␣logrado␣a␣lo␣largo␣de␣la␣linea␣
   de␣investigacion␣desde␣la␣instalacion␣y␣actualizacion␣de␣las
   ␣aplicaciones␣de␣Synopsys␣hasta␣la␣simulacion␣final␣del␣
   circuito.␣El␣Gran␣Jaguar␣es␣un␣proyecto␣ambicioso␣que␣dara␣
   paso␣a␣una␣infinidad
```

```
de␣oportunidades␣en␣la␣investigacion␣y␣desarrollo␣de␣nuevas␣
    tecnologias␣para␣beneficio␣de␣la␣sociedad␣global,␣con␣un␣
    enfoque␣particular␣en␣la␣sociedad␣guatemalteca."

texto2 = "Hello,␣I␣am␣El␣Gran␣Jaguar,␣the␣first␣nanochip␣
    developed␣in␣Central␣America␣by␣a␣university.␣Completely␣
    developed␣at␣Universidad␣del␣Valle␣de␣Guatemala␣by␣
    undergraduate␣students␣of␣Electronics␣Engineering␣between␣
    two␣thousand␣nineteen␣and␣two␣thousand␣twenty␣five.␣The␣team
    ␣in␣charge␣of␣my␣creation␣was␣made
up␣of:␣Carlos␣Alberto␣Esquit␣Hernandez,␣Jonathan␣Alberto␣de␣los
    ␣Santos␣Chonay,␣Kurt␣Emmanuel␣Kellner␣Juarez,␣Miguel␣Enrique
    ␣Zea␣Arenales,␣Luis␣Alberto␣Rivera␣Estrada,␣Luis␣Arturo␣
    Najera␣Vasquez,␣Steven␣Hiram␣Rubio␣Vasquez,␣Marvin␣Geovanni␣
    Flores␣Espino,␣Matthias␣Sibrian␣Illescas.␣Julio␣Enrique␣Shin
    ␣Jo,␣Karol␣Sophia
Cardona␣Polanco,␣Luis␣Estuardo␣Abadia␣Lopez,␣Joel␣Andres␣
    Gonzalez␣Herrera,␣Jose␣Adrian␣Ayala␣Escobar,␣Jose␣Alejandro␣
    Ruiz␣Orozco,␣Juan␣Ricardo␣Giron␣Rubio,␣Charlie␣Ayenci␣Cruz␣
    Giron.␣Elmer␣Otoniel␣Torres␣Garza,␣Gerardo␣Cardoza,␣Antonio␣
    Altuna␣Hernandez,␣Carlos␣Andres␣Letona␣Torres,␣Helder␣
    Arnoldo␣Ovalle␣Barrios,␣Jose
David␣Ponce␣del␣Cid,␣Israel␣Antonio␣Arevalo␣Gonzalez,␣Allison␣
    Estuardo␣Aguilar␣Chocooj,␣Paola␣Alejandra␣Mendizabal␣Batres,
    ␣Diego␣Rodrigo␣Equite␣Pirir,␣Estuardo␣Geovany␣Mancio␣
    Ruballos,␣Stefan␣Alexander␣Schwendener␣Farrington,␣Luis␣
    Ricardo␣Gomez␣Velasquez,␣Julio␣Emanuel␣Lopez␣Ozaeta.␣Noel␣
    Francisco␣Prado␣Bucaro,␣Diana
Sofia␣Alvarado␣Mota,␣Katherine␣Arletth␣Rac␣Pocon,␣Lourdes␣Maria
    ␣Ruiz␣Vasquez,␣Juan␣Pablo␣Mora␣Argueta,␣Kevin␣David␣
    Hernandez␣Guarcas.␣Jose␣Mario␣Mendez␣Rodriguez,␣Byron␣
    Estuardo␣Barrientos␣Perez,␣Luis␣Pablo␣Gustavo␣Carranza␣
    Vasquez,␣Miguel␣Alberto␣Chacon␣Ortiz␣and,␣Carlos␣Julio␣
    Valdes␣Oajaca.␣The␣idea␣of␣manufacturing
a␣CHIP␣was␣born␣at␣the␣beginning␣of␣two␣thousand␣nineteen,␣when
    ␣graduating␣students␣obtained␣the␣support␣from␣Euro-practice
    ␣to␣finance␣a␣large␣part␣of␣El␣Gran␣Jaguar␣manufacturing␣
    process.␣Also,␣IMEC␣provided␣access␣to␣manufacturing␣
    libraries␣implemented␣by␣the␣Taiwan␣Semiconductor␣
    Manufacturing␣Company␣in␣a␣process
of␣sixty␣five␣nanometers.␣Work␣is␣also␣being␣done␣to␣automate␣
    the␣entire␣process␣achieved␣throughout␣the␣research␣line,␣
    from␣the␣installation␣and␣updating␣of␣Synopsys␣applications␣
    to␣the␣final␣simulation␣of␣the␣circuit.␣El␣Gran␣Jaguar␣is␣an
    ␣ambitious␣project␣that␣will␣open␣up␣vast␣opportunities␣for␣
    the␣research␣and
development␣of␣new␣technologies␣that␣will␣benefit␣society␣
    worldwide,␣with␣a␣particular␣focus␣on␣Guatemalan␣society."
```

```
print(len(texto))
print(len(texto2))

archivo = open("nanochip.v", "w")
archivo.writelines("//␣Las␣librerias␣de␣65␣nm␣poseen␣escala␣de␣
   tiempo,␣para␣simular␣todos␣los␣modulos␣deben␣tenerla\n")
archivo.writelines("`timescale␣␣1ns/1ps\n\n")

archivo.writelines("module␣nanochip(q_out,␣clk,␣reset_nanochip,
   ␣select,␣reset_ring_osc,␣out_ring_osc);\n\n")

archivo.writelines("//␣14␣pins␣de␣senales,␣1␣VDD␣y␣1␣VSS␣=␣16\n
   ")
archivo.writelines("input␣clk,␣reset_nanochip,␣reset_ring_osc;\
   n")
archivo.writelines("input␣[1:0]␣select;\n")
archivo.writelines("output␣out_ring_osc;\n")
archivo.writelines("output␣[7:0]␣q_out;\n\n")

archivo.writelines("reg␣[7:0]␣q;\n\n")

archivo.writelines("//␣Contador␣que␣llega␣hasta␣4096␣caracteres
   \n")
archivo.writelines("reg␣[11:0]␣counter;\n\n")

archivo.writelines("//␣Wires␣para␣el␣ring␣oscillator␣(agregar␣
   delay␣a␣wires␣para␣la␣simulacion␣e.g.␣#1)\n")
archivo.writelines("wire␣#1␣W_1;\n")
archivo.writelines("wire␣#1␣W_2;\n")
archivo.writelines("wire␣#1␣W_3;\n")
archivo.writelines("wire␣#1␣W_4;\n")
archivo.writelines("wire␣#1␣W_5;\n")
archivo.writelines("wire␣#1␣W_6;\n")
archivo.writelines("wire␣#1␣W_7;\n")
archivo.writelines("wire␣#1␣W_8;\n")
archivo.writelines("wire␣#1␣W_9;\n")
archivo.writelines("wire␣#1␣W_10;\n")
archivo.writelines("wire␣#1␣W_11;\n")
archivo.writelines("wire␣#1␣W_12;\n")
archivo.writelines("wire␣#1␣W_13;\n\n")

archivo.writelines("//␣Ring␣oscillator␣de␣13␣NOTs\n")
archivo.writelines("assign␣W_1␣=␣reset_ring_osc␣|␣out_ring_osc
   ;\n")
archivo.writelines("assign␣W_2␣=␣~W_1;\n")
archivo.writelines("assign␣W_3␣=␣~W_2;\n")
archivo.writelines("assign␣W_4␣=␣~W_3;\n")
```

```
archivo.writelines("assign␣W_5␣=␣~W_4;\n")
archivo.writelines("assign␣W_6␣=␣~W_5;\n")
archivo.writelines("assign␣W_7␣=␣~W_6;\n")
archivo.writelines("assign␣W_8␣=␣~W_7;\n")
archivo.writelines("assign␣W_9␣=␣~W_8;\n")
archivo.writelines("assign␣W_10␣=␣~W_9;\n")
archivo.writelines("assign␣W_11␣=␣~W_10;\n")
archivo.writelines("assign␣W_12␣=␣~W_11;\n")
archivo.writelines("assign␣W_13␣=␣~W_12;\n")
archivo.writelines("assign␣out_ring_osc␣=␣~W_13;\n\n")

archivo.writelines("//␣Nanochip\n")
archivo.writelines("always␣@␣(posedge␣reset_nanochip␣or␣posedge
   ␣clk)\n")
archivo.writelines("if␣(reset_nanochip)\n")
archivo.writelines("counter<=12'b000000000000;\n")
archivo.writelines("else␣if(select==2'b00␣||␣select==2'b11)␣
   begin\n")
archivo.writelines("if␣(counter␣<"+str(len(texto)-1)+ ")\n")
archivo.writelines("counter␣<=␣counter␣+␣1;\n")
archivo.writelines("else\n")
archivo.writelines("counter␣<=␣0;\n")
archivo.writelines("end\n")
archivo.writelines("else␣if(select==2'b01␣||␣select==2'b10)␣
   begin\n")
archivo.writelines("if␣(counter␣<"+str(len(texto2)-1)+ ")\n")
archivo.writelines("counter␣<=␣counter␣+␣1;\n")
archivo.writelines("else\n")
archivo.writelines("counter␣<=␣0;\n")
archivo.writelines("end\n")


archivo.writelines("always␣@␣(posedge␣clk)\n")
archivo.writelines("//␣Texto␣en␣espanol\n")
archivo.writelines("if(select==2'b00␣||␣select==2'b11)\n")
archivo.writelines("begin\n")
archivo.writelines("if(counter␣==␣'d0)\n")
archivo.writelines("begin\n")
#archivo.writelines("q<=8'b00100000;\n")
archivo.writelines("q<=8'b")
binario=bin(ord(texto[0]))
for x in range(10-len(binario)):
        archivo.writelines("0")
for j in range(len(binario)-2):
        archivo.writelines(str(binario[j+2]))
archivo.writelines(";\n")
archivo.writelines("end\n")
```

```
for i in range(len(texto)-1):
        archivo.writelines("else␣if(counter␣==␣'d"+str(i+1)+")\
           n")
        archivo.writelines("begin\n")
        archivo.writelines("q<=8'b")
        #print(ord(texto[i]))
        #binascii.hexlify(texto[i])ord(character)
        #binario = bin(int(binascii.hexlify(texto[i]),16))
        binario=bin(ord(texto[i+1]))
        #print len(binario)
        #print binario
        for x in range(10-len(binario)):
                archivo.writelines("0")
        for j in range(len(binario)-2):
                archivo.writelines(str(binario[j+2]))
        archivo.writelines(";\n")
        archivo.writelines("end\n")
        #print(bin(int(binascii.hexlify(texto[i]),16)))
archivo.writelines("end\n\n")

archivo.writelines("//␣Texto␣en␣ingles\n")
archivo.writelines("else␣if(select==2'b01␣||␣select==2'b10)\n")
archivo.writelines("begin\n")
archivo.writelines("if(counter␣==␣'d0)\n")
archivo.writelines("begin\n")
#archivo.writelines("q<=8'b00100000;\n")
archivo.writelines("q<=8'b")
binario=bin(ord(texto2[0]))
for x in range(10-len(binario)):
        archivo.writelines("0")
for j in range(len(binario)-2):
        archivo.writelines(str(binario[j+2]))
archivo.writelines(";\n")
archivo.writelines("end\n")

for i in range(len(texto2)-1):
        archivo.writelines("else␣if(counter␣==␣'d"+str(i+1)+")\
           n")
        archivo.writelines("begin\n")
        archivo.writelines("q<=8'b")
        #binario = bin(int(binascii.hexlify(texto2[i]),16))
        #print len(binario)
        binario=bin(ord(texto2[i+1]))
        for x in range(10-len(binario)):
                archivo.writelines("0")
        for j in range(len(binario)-2):
                archivo.writelines(str(binario[j+2]))
        archivo.writelines(";\n")
```

```
        archivo.writelines("end\n")
        #print(bin(int(binascii.hexlify(texto2[i]),16)))
archivo.writelines("end\n\n")
archivo.writelines("assign␣q_out␣=␣q;\n")
archivo.writelines("endmodule\n")


archivo.close()
```

18.2. Nanochip "El Gran Jaguar"

18.2.1. Código del circuito *core* sintetizado

Cuadro 99. Archivo nanochip_mapped.v en el directorio outputs_no_pads/

```
////////////////////////////////////////////////////////////////
// Created by: Synopsys DC Ultra(TM) in wire load mode
// Version   : W-2024.09-SP4
// Date      : Sat Nov  8 23:04:02 2025
////////////////////////////////////////////////////////////////


module nanochip ( q_out, clk, reset_nanochip, select,
   reset_ring_osc,
        out_ring_osc );
  output [7:0] q_out;
  input [1:0] select;
  input clk, reset_nanochip, reset_ring_osc;
  output out_ring_osc;
  wire   W_1, W_2, W_3, W_4, W_5, W_6, W_7, W_8, W_9, W_10,
     W_11, W_12, W_13,
         N4767, N4768, N4769, N4770, N4771, N4772, N4773, N4774
            , N4775, N4776,
         N4777, N4778, n4811, n4812, n4813, n4814, n4815, n4816
            , n4817, n4818,
         n4819, n4820, n4821, n4822, n4823, n4824, n4825, n4826
            , n4827, n4828,
         n4829, n4830, n4831, n4832, n4833, n4834, n4835, n4836
            , n4837, n4838,
         n4839, n4840, n4841, n4842, n4843, n4844, n4845, n4846
            , n4847, n4848,
         n4849, n4850, n4851, n4852, n4853, n4854, n4855, n4856
            , n4857, n4858,
         n4859, n4860, n4861, n4862, n4863, n4864, n4865, n4866
            , n4867, n4868,
```

```
        n4869, n4870, n4871, n4872, n4873, n4874, n4875, n4876
            , n4877, n4878,
        n4879, n4880, n4881, n4882, n4883, n4884, n4885, n4886
            , n4887, n4888,
        n4889, n4890, n4891, n4892, n4893, n4894, n4895, n4896
            , n4897, n4898,
        n4899, n4900, n4901, n4902, n4903, n4904, n4905, n4906
            , n4907, n4908,
        n4909, n4910, n4911, n4912, n4913, n4914, n4915, n4916
            , n4917, n4918,
        n4919, n4920, n4921, n4922, n4923, n4924, n4925, n4926
            , n4927, n4928,
        n4929, n4930, n4931, n4932, n4933, n4934, n4935, n4936
            , n4937, n4938,
        n4939, n4940, n4941, n4942, n4943, n4944, n4945, n4946
            , n4947, n4948,
        n4949, n4950, n4951, n4952, n4953, n4954, n4955, n4956
            , n4957, n4958,
        n4959, n4960, n4961, n4962, n4963, n4964, n4965, n4966
            , n4967, n4968,
        n4969, n4970, n4971, n4972, n4973, n4974, n4975, n4976
            , n4977, n4978,
        n4979, n4980, n4981, n4982, n4983, n4984, n4985, n4986
            , n4987, n4988,
        n4989, n4990, n4991, n4992, n4993, n4994, n4995, n4996
            , n4997, n4998,
        n4999, n5000, n5001, n5002, n5003, n5004, n5005, n5006
            , n5007, n5008,
        n5009, n5010, n5011, n5012, n5013, n5014, n5015, n5016
            , n5017, n5018,
        n5019, n5020, n5021, n5022, n5023, n5024, n5025, n5026
            , n5027, n5028,
        n5029, n5030, n5031, n5032, n5033, n5034, n5035, n5036
            , n5037, n5038,
        n5039, n5040, n5041, n5042, n5043, n5044, n5045, n5046
            , n5047, n5048,
        n5049, n5050, n5051, n5052, n5053, n5054, n5055, n5056
            , n5057, n5058,
        n5059, n5060, n5061, n5062, n5063, n5064, n5065, n5066
            , n5067, n5068,
        n5069, n5070, n5071, n5072, n5073, n5074, n5075, n5076
            , n5077, n5078,
        n5079, n5080, n5081, n5082, n5083, n5084, n5085, n5086
            , n5087, n5088,
        n5089, n5090, n5091, n5092, n5093, n5094, n5095, n5096
            , n5097, n5098,
        n5099, n5100, n5101, n5102, n5103, n5104, n5105, n5106
            , n5107, n5108,
```

```
        n5109, n5110, n5111, n5112, n5113, n5114, n5115, n5116
            , n5117, n5118,
        n5119, n5120, n5121, n5122, n5123, n5124, n5125, n5126
            , n5127, n5128,
        n5129, n5130, n5131, n5132, n5133, n5134, n5135, n5136
            , n5137, n5138,
        n5139, n5140, n5141, n5142, n5143, n5144, n5145, n5146
            , n5147, n5148,
        n5149, n5150, n5151, n5152, n5153, n5154, n5155, n5156
            , n5157, n5158,
        n5159, n5160, n5161, n5162, n5163, n5164, n5165, n5166
            , n5167, n5168,
        n5169, n5170, n5171, n5172, n5173, n5174, n5175, n5176
            , n5177, n5178,
        n5179, n5180, n5181, n5182, n5183, n5184, n5185, n5186
            , n5187, n5188,
        n5189, n5190, n5191, n5192, n5193, n5194, n5195, n5196
            , n5197, n5198,
        n5199, n5200, n5201, n5202, n5203, n5204, n5205, n5206
            , n5207, n5208,
        n5209, n5210, n5211, n5212, n5213, n5214, n5215, n5216
            , n5217, n5218,
        n5219, n5220, n5221, n5222, n5223, n5224, n5225, n5226
            , n5227, n5228,
        n5229, n5230, n5231, n5232, n5233, n5234, n5235, n5236
            , n5237, n5238,
        n5239, n5240, n5241, n5242, n5243, n5244, n5245, n5246
            , n5247, n5248,
        n5249, n5250, n5251, n5252, n5253, n5254, n5255, n5256
            , n5257, n5258,
        n5259, n5260, n5261, n5262, n5263, n5264, n5265, n5266
            , n5267, n5268,
        n5269, n5270, n5271, n5272, n5273, n5274, n5275, n5276
            , n5277, n5278,
        n5279, n5280, n5281, n5282, n5283, n5284, n5285, n5286
            , n5287, n5288,
        n5289, n5290, n5291, n5292, n5293, n5294, n5295, n5296
            , n5297, n5298,
        n5299, n5300, n5301, n5302, n5303, n5304, n5305, n5306
            , n5307, n5308,
        n5309, n5310, n5311, n5312, n5313, n5314, n5315, n5316
            , n5317, n5318,
        n5319, n5320, n5321, n5322, n5323, n5324, n5325, n5326
            , n5327, n5328,
        n5329, n5330, n5331, n5332, n5333, n5334, n5335, n5336
            , n5337, n5338,
        n5339, n5340, n5341, n5342, n5343, n5344, n5345, n5346
            , n5347, n5348,
```

```
        n5349, n5350, n5351, n5352, n5353, n5354, n5355, n5356
            , n5357, n5358,
        n5359, n5360, n5361, n5362, n5363, n5364, n5365, n5366
            , n5367, n5368,
        n5369, n5370, n5371, n5372, n5373, n5374, n5375, n5376
            , n5377, n5378,
        n5379, n5380, n5381, n5382, n5383, n5384, n5385, n5386
            , n5387, n5388,
        n5389, n5390, n5391, n5392, n5393, n5394, n5395, n5396
            , n5397, n5398,
        n5399, n5400, n5401, n5402, n5403, n5404, n5405, n5406
            , n5407, n5408,
        n5409, n5410, n5411, n5412, n5413, n5414, n5415, n5416
            , n5417, n5418,
        n5419, n5420, n5421, n5422, n5423, n5424, n5425, n5426
            , n5427, n5428,
        n5429, n5430, n5431, n5432, n5433, n5434, n5435, n5436
            , n5437, n5438,
        n5439, n5440, n5441, n5442, n5443, n5444, n5445, n5446
            , n5447, n5448,
        n5449, n5450, n5451, n5452, n5453, n5454, n5455, n5456
            , n5457, n5458,
        n5459, n5460, n5461, n5462, n5463, n5464, n5465, n5466
            , n5467, n5468,
        n5469, n5470, n5471, n5472, n5473, n5474, n5475, n5476
            , n5477, n5478,
        n5479, n5480, n5481, n5482, n5483, n5484, n5485, n5486
            , n5487, n5488,
        n5489, n5490, n5491, n5492, n5493, n5494, n5495, n5496
            , n5497, n5498,
        n5499, n5500, n5501, n5502, n5503, n5504, n5505, n5506
            , n5507, n5508,
        n5509, n5510, n5511, n5512, n5513, n5514, n5515, n5516
            , n5517, n5518,
        n5519, n5520, n5521, n5522, n5523, n5524, n5525, n5526
            , n5527, n5528,
        n5529, n5530, n5531, n5532, n5533, n5534, n5535, n5536
            , n5537, n5538,
        n5539, n5540, n5541, n5542, n5543, n5544, n5545, n5546
            , n5547, n5548,
        n5549, n5550, n5551, n5552, n5553, n5554, n5555, n5556
            , n5557, n5558,
        n5559, n5560, n5561, n5562, n5563, n5564, n5565, n5566
            , n5567, n5568,
        n5569, n5570, n5571, n5572, n5573, n5574, n5575, n5576
            , n5577, n5578,
        n5579, n5580, n5581, n5582, n5583, n5584, n5585, n5586
            , n5587, n5588,
```

```
        n5589, n5590, n5591, n5592, n5593, n5594, n5595, n5596
            , n5597, n5598,
        n5599, n5600, n5601, n5602, n5603, n5604, n5605, n5606
            , n5607, n5608,
        n5609, n5610, n5611, n5612, n5613, n5614, n5615, n5616
            , n5617, n5618,
        n5619, n5620, n5621, n5622, n5623, n5624, n5625, n5626
            , n5627, n5628,
        n5629, n5630, n5631, n5632, n5633, n5634, n5635, n5636
            , n5637, n5638,
        n5639, n5640, n5641, n5642, n5643, n5644, n5645, n5646
            , n5647, n5648,
        n5649, n5650, n5651, n5652, n5653, n5654, n5655, n5656
            , n5657, n5658,
        n5659, n5660, n5661, n5662, n5663, n5664, n5665, n5666
            , n5667, n5668,
        n5669, n5670, n5671, n5672, n5673, n5674, n5675, n5676
            , n5677, n5678,
        n5679, n5680, n5681, n5682, n5683, n5684, n5685, n5686
            , n5687, n5688,
        n5689, n5690, n5691, n5692, n5693, n5694, n5695, n5696
            , n5697, n5698,
        n5699, n5700, n5701, n5702, n5703, n5704, n5705, n5706
            , n5707, n5708,
        n5709, n5710, n5711, n5712, n5713, n5714, n5715, n5716
            , n5717, n5718,
        n5719, n5720, n5721, n5722, n5723, n5724, n5725, n5726
            , n5727, n5728,
        n5729, n5730, n5731, n5732, n5733, n5734, n5735, n5736
            , n5737, n5738,
        n5739, n5740, n5741, n5742, n5743, n5744, n5745, n5746
            , n5747, n5748,
        n5749, n5750, n5751, n5752, n5753, n5754, n5755, n5756
            , n5757, n5758,
        n5759, n5760, n5761, n5762, n5763, n5764, n5765, n5766
            , n5767, n5768,
        n5769, n5770, n5771, n5772, n5773, n5774, n5775, n5776
            , n5777, n5778,
        n5779, n5780, n5781, n5782, n5783, n5784, n5785, n5786
            , n5787, n5788,
        n5789, n5790, n5791, n5792, n5793, n5794, n5795, n5796
            , n5797, n5798,
        n5799, n5800, n5801, n5802, n5803, n5804, n5805, n5806
            , n5807, n5808,
        n5809, n5810, n5811, n5812, n5813, n5814, n5815, n5816
            , n5817, n5818,
        n5819, n5820, n5821, n5822, n5823, n5824, n5825, n5826
            , n5827, n5828,
```

```
        n5829, n5830, n5831, n5832, n5833, n5834, n5835, n5836
            , n5837, n5838,
        n5839, n5840, n5841, n5842, n5843, n5844, n5845, n5846
            , n5847, n5848,
        n5849, n5850, n5851, n5852, n5853, n5854, n5855, n5856
            , n5857, n5858,
        n5859, n5860, n5861, n5862, n5863, n5864, n5865, n5866
            , n5867, n5868,
        n5869, n5870, n5871, n5872, n5873, n5874, n5875, n5876
            , n5877, n5878,
        n5879, n5880, n5881, n5882, n5883, n5884, n5885, n5886
            , n5887, n5888,
        n5889, n5890, n5891, n5892, n5893, n5894, n5895, n5896
            , n5897, n5898,
        n5899, n5900, n5901, n5902, n5903, n5904, n5905, n5906
            , n5907, n5908,
        n5909, n5910, n5911, n5912, n5913, n5914, n5915, n5916
            , n5917, n5918,
        n5919, n5920, n5921, n5922, n5923, n5924, n5925, n5926
            , n5927, n5928,
        n5929, n5930, n5931, n5932, n5933, n5934, n5935, n5936
            , n5937, n5938,
        n5939, n5940, n5941, n5942, n5943, n5944, n5945, n5946
            , n5947, n5948,
        n5949, n5950, n5951, n5952, n5953, n5954, n5955, n5956
            , n5957, n5958,
        n5959, n5960, n5961, n5962, n5963, n5964, n5965, n5966
            , n5967, n5968,
        n5969, n5970, n5971, n5972, n5973, n5974, n5975, n5976
            , n5977, n5978,
        n5979, n5980, n5981, n5982, n5983, n5984, n5985, n5986
            , n5987, n5988,
        n5989, n5990, n5991, n5992, n5993, n5994, n5995, n5996
            , n5997, n5998,
        n5999, n6000, n6001, n6002, n6003, n6004, n6005, n6006
            , n6007, n6008,
        n6009, n6010, n6011, n6012, n6013, n6014, n6015, n6016
            , n6017, n6018,
        n6019, n6020, n6021, n6022, n6023, n6024, n6025, n6026
            , n6027, n6028,
        n6029, n6030, n6031, n6032, n6033, n6034, n6035, n6036
            , n6037, n6038,
        n6039, n6040, n6041, n6042, n6043, n6044, n6045, n6046
            , n6047, n6048,
        n6049, n6050, n6051, n6052, n6053, n6054, n6055, n6056
            , n6057, n6058,
        n6059, n6060, n6061, n6062, n6063, n6064, n6065, n6066
            , n6067, n6068,
```

```
        n6069, n6070, n6071, n6072, n6073, n6074, n6075, n6076
            , n6077, n6078,
        n6079, n6080, n6081, n6082, n6083, n6084, n6085, n6086
            , n6087, n6088,
        n6089, n6090, n6091, n6092, n6093, n6094, n6095, n6096
            , n6097, n6098,
        n6099, n6100, n6101, n6102, n6103, n6104, n6105, n6106
            , n6107, n6108,
        n6109, n6110, n6111, n6112, n6113, n6114, n6115, n6116
            , n6117, n6118,
        n6119, n6120, n6121, n6122, n6123, n6124, n6125, n6126
            , n6127, n6128,
        n6129, n6130, n6131, n6132, n6133, n6134, n6135, n6136
            , n6137, n6138,
        n6139, n6140, n6141, n6142, n6143, n6144, n6145, n6146
            , n6147, n6148,
        n6149, n6150, n6151, n6152, n6153, n6154, n6155, n6156
            , n6157, n6158,
        n6159, n6160, n6161, n6162, n6163, n6164, n6165, n6166
            , n6167, n6168,
        n6169, n6170, n6171, n6172, n6173, n6174, n6175, n6176
            , n6177, n6178,
        n6179, n6180, n6181, n6182, n6183, n6184, n6185, n6186
            , n6187, n6188,
        n6189, n6190, n6191, n6192, n6193, n6194, n6195, n6196
            , n6197, n6198,
        n6199, n6200, n6201, n6202, n6203, n6204, n6205, n6206
            , n6207, n6208,
        n6209, n6210, n6211, n6212, n6213, n6214, n6215, n6216
            , n6217, n6218,
        n6219, n6220, n6221, n6222, n6223, n6224, n6225, n6226
            , n6227, n6228,
        n6229, n6230, n6231, n6232, n6233, n6234, n6235, n6236
            , n6237, n6238,
        n6239, n6240, n6241, n6242, n6243, n6244, n6245, n6246
            , n6247, n6248,
        n6249, n6250, n6251, n6252, n6253, n6254, n6255, n6256
            , n6257, n6258,
        n6259, n6260, n6261, n6262, n6263, n6264, n6265, n6266
            , n6267, n6268,
        n6269, n6270, n6271, n6272, n6273, n6274, n6275, n6276
            , n6277, n6278,
        n6279, n6280, n6281, n6282, n6283, n6284, n6285, n6286
            , n6287, n6288,
        n6289, n6290, n6291, n6292, n6293, n6294, n6295, n6296
            , n6297, n6298,
        n6299, n6300, n6301, n6302, n6303, n6304, n6305, n6306
            , n6307, n6308,
```

```
        n6309, n6310, n6311, n6312, n6313, n6314, n6315, n6316
            , n6317, n6318,
        n6319, n6320, n6321, n6322, n6323, n6324, n6325, n6326
            , n6327, n6328,
        n6329, n6330, n6331, n6332, n6333, n6334, n6335, n6336
            , n6337, n6338,
        n6339, n6340, n6341, n6342, n6343, n6344, n6345, n6346
            , n6347, n6348,
        n6349, n6350, n6351, n6352, n6353, n6354, n6355, n6356
            , n6357, n6358,
        n6359, n6360, n6361, n6362, n6363, n6364, n6365, n6366
            , n6367, n6368,
        n6369, n6370, n6371, n6372, n6373, n6374, n6375, n6376
            , n6377, n6378,
        n6379, n6380, n6381, n6382, n6383, n6384, n6385, n6386
            , n6387, n6388,
        n6389, n6390, n6391, n6392, n6393, n6394, n6395, n6396
            , n6397, n6398,
        n6399, n6400, n6401, n6402, n6403, n6404, n6405, n6406
            , n6407, n6408,
        n6409, n6410, n6411, n6412, n6413, n6414, n6415, n6416
            , n6417, n6418,
        n6419, n6420, n6421, n6422, n6423, n6424, n6425, n6426
            , n6427, n6428,
        n6429, n6430, n6431, n6432, n6433, n6434, n6435, n6436
            , n6437, n6438,
        n6439, n6440, n6441, n6442, n6443, n6444, n6445, n6446
            , n6447, n6448,
        n6449, n6450, n6451, n6452, n6453, n6454, n6455, n6456
            , n6457, n6458,
        n6459, n6460, n6461, n6462, n6463, n6464, n6465, n6466
            , n6467, n6468,
        n6469, n6470, n6471, n6472, n6473, n6474, n6475, n6476
            , n6477, n6478,
        n6479, n6480, n6481, n6482, n6483, n6484, n6485, n6486
            , n6487, n6488,
        n6489, n6490, n6491, n6492, n6493, n6494, n6495, n6496
            , n6497, n6498,
        n6499, n6500, n6501, n6502, n6503, n6504, n6505, n6506
            , n6507, n6508,
        n6509, n6510, n6511, n6512, n6513, n6514, n6515, n6516
            , n6517, n6518,
        n6519, n6520, n6521, n6522, n6523, n6524, n6525, n6526
            , n6527, n6528,
        n6529, n6530, n6531, n6532, n6533, n6534, n6535, n6536
            , n6537, n6538,
        n6539, n6540, n6541, n6542, n6543, n6544, n6545, n6546
            , n6547, n6548,
```

```
        n6549, n6550, n6551, n6552, n6553, n6554, n6555, n6556
            , n6557, n6558,
        n6559, n6560, n6561, n6562, n6563, n6564, n6565, n6566
            , n6567, n6568,
        n6569, n6570, n6571, n6572, n6573, n6574, n6575, n6576
            , n6577, n6578,
        n6579, n6580, n6581, n6582, n6583, n6584, n6585, n6586
            , n6587, n6588,
        n6589, n6590, n6591, n6592, n6593, n6594, n6595, n6596
            , n6597, n6598,
        n6599, n6600, n6601, n6602, n6603, n6604, n6605, n6606
            , n6607, n6608,
        n6609, n6610, n6611, n6612, n6613, n6614, n6615, n6616
            , n6617, n6618,
        n6619, n6620, n6621, n6622, n6623, n6624, n6625, n6626
            , n6627, n6628,
        n6629, n6630, n6631, n6632, n6633, n6634, n6635, n6636
            , n6637, n6638,
        n6639, n6640, n6641, n6642, n6643, n6644, n6645, n6646
            , n6647, n6648,
        n6649, n6650, n6651, n6652, n6653, n6654, n6655, n6656
            , n6657, n6658,
        n6659, n6660, n6661, n6662, n6663, n6664, n6665, n6666
            , n6667, n6668,
        n6669, n6670, n6671, n6672, n6673, n6674, n6675, n6676
            , n6677, n6678,
        n6679, n6680, n6681, n6682, n6683, n6684, n6685, n6686
            , n6687, n6688,
        n6689, n6690, n6691, n6692, n6693, n6694, n6695, n6696
            , n6697, n6698,
        n6699, n6700, n6701, n6702, n6703, n6704, n6705, n6706
            , n6707, n6708,
        n6709, n6710, n6711, n6712, n6713, n6714, n6715, n6716
            , n6717, n6718,
        n6719, n6720, n6721, n6722, n6723, n6724, n6725, n6726
            , n6727, n6728,
        n6729, n6730, n6731, n6732, n6733, n6734, n6735, n6736
            , n6737, n6738,
        n6739, n6740, n6741, n6742, n6743, n6744, n6745, n6746
            , n6747, n6748,
        n6749, n6750, n6751, n6752, n6753, n6754, n6755, n6756
            , n6757, n6758,
        n6759, n6760, n6761, n6762, n6763, n6764, n6765, n6766
            , n6767, n6768,
        n6769, n6770, n6771, n6772, n6773, n6774, n6775, n6776
            , n6777, n6778,
        n6779, n6780, n6781, n6782, n6783, n6784, n6785, n6786
            , n6787, n6788,
```

```
        n6789, n6790, n6791, n6792, n6793, n6794, n6795, n6796
            , n6797, n6798,
        n6799, n6800, n6801, n6802, n6803, n6804, n6805, n6806
            , n6807, n6808,
        n6809, n6810, n6811, n6812, n6813, n6814, n6815, n6816
            , n6817, n6818,
        n6819, n6820, n6821, n6822, n6823, n6824, n6825, n6826
            , n6827, n6828,
        n6829, n6830, n6831, n6832, n6833, n6834, n6835, n6836
            , n6837, n6838,
        n6839, n6840, n6841, n6842, n6843, n6844, n6845, n6846
            , n6847, n6848,
        n6849, n6850, n6851, n6852, n6853, n6854, n6855, n6856
            , n6857, n6858,
        n6859, n6860, n6861, n6862, n6863, n6864, n6865, n6866
            , n6867, n6868,
        n6869, n6870, n6871, n6872, n6873, n6874, n6875, n6876
            , n6877, n6878,
        n6879, n6880, n6881, n6882, n6883, n6884, n6885, n6886
            , n6887, n6888,
        n6889, n6890, n6891, n6892, n6893, n6894, n6895, n6896
            , n6897, n6898,
        n6899, n6900, n6901, n6902, n6903, n6904, n6905, n6906
            , n6907, n6908,
        n6909, n6910, n6911, n6912, n6913, n6914, n6915, n6916
            , n6917, n6918,
        n6919, n6920, n6921, n6922, n6923, n6924, n6925, n6926
            , n6927, n6928,
        n6929, n6930, n6931, n6932, n6933, n6934, n6935, n6936
            , n6937, n6938,
        n6939, n6940, n6941, n6942, n6943, n6944, n6945, n6946
            , n6947, n6948,
        n6949, n6950, n6951, n6952, n6953, n6954, n6955, n6956
            , n6957, n6958,
        n6959, n6960, n6961, n6962, n6963, n6964, n6965, n6966
            , n6967, n6968,
        n6969, n6970, n6971, n6972, n6973, n6974, n6975, n6976
            , n6977, n6978,
        n6979, n6980, n6981, n6982, n6983, n6984, n6985, n6986
            , n6987, n6988,
        n6989, n6990, n6991, n6992, n6993, n6994, n6995, n6996
            , n6997, n6998,
        n6999, n7000, n7001, n7002, n7003, n7004, n7005, n7006
            , n7007, n7008,
        n7009, n7010, n7011, n7012, n7013, n7014, n7015, n7016
            , n7017, n7018,
        n7019, n7020, n7021, n7022, n7023, n7024, n7025, n7026
            , n7027, n7028,
```

```
        n7029, n7030, n7031, n7032, n7033, n7034, n7035, n7036
            , n7037, n7038,
        n7039, n7040, n7041, n7042, n7043, n7044, n7045, n7046
            , n7047, n7048,
        n7049, n7050, n7051, n7052, n7053, n7054, n7055, n7056
            , n7057, n7058,
        n7059, n7060, n7061, n7062, n7063, n7064, n7065, n7066
            , n7067, n7068,
        n7069, n7070, n7071, n7072, n7073, n7074, n7075, n7076
            , n7077, n7078,
        n7079, n7080, n7081, n7082, n7083, n7084, n7085, n7086
            , n7087, n7088,
        n7089, n7090, n7091, n7092, n7093, n7094, n7095, n7096
            , n7097, n7098,
        n7099, n7100, n7101, n7102, n7103, n7104, n7105, n7106
            , n7107, n7108,
        n7109, n7110, n7111, n7112, n7113, n7114, n7115, n7116
            , n7117, n7118,
        n7119, n7120, n7121, n7122, n7123, n7124, n7125, n7126
            , n7127, n7128,
        n7129, n7130, n7131, n7132, n7133, n7134, n7135, n7136
            , n7137, n7138,
        n7139, n7140, n7141, n7142, n7143, n7144, n7145, n7146
            , n7147, n7148,
        n7149, n7150, n7151, n7152, n7153, n7154, n7155, n7156
            , n7157, n7158,
        n7159, n7160, n7161, n7162, n7163, n7164, n7165, n7166
            , n7167, n7168,
        n7169, n7170, n7171, n7172, n7173, n7174, n7175, n7176
            , n7177, n7178,
        n7179, n7180, n7181, n7182, n7183, n7184, n7185, n7186
            , n7187, n7188,
        n7189, n7190, n7191, n7192, n7193, n7194, n7195, n7196
            , n7197, n7198,
        n7199, n7200, n7201, n7202, n7203, n7204, n7205, n7206
            , n7207, n7208,
        n7209, n7210, n7211, n7212, n7213, n7214, n7215, n7216
            , n7217, n7218,
        n7219, n7220, n7221, n7222, n7223, n7224, n7225, n7226
            , n7227, n7228,
        n7229, n7230, n7231, n7232, n7233, n7234, n7235, n7236
            , n7237, n7238,
        n7239, n7240, n7241, n7242, n7243, n7244, n7245, n7246
            , n7247, n7248,
        n7249, n7250, n7251, n7252, n7253, n7254, n7255, n7256
            , n7257, n7258,
        n7259, n7260, n7261, n7262, n7263, n7264, n7265, n7266
            , n7267, n7268,
```

```
        n7269, n7270, n7271, n7272, n7273, n7274, n7275, n7276
            , n7277, n7278,
        n7279, n7280, n7281, n7282, n7283, n7284, n7285, n7286
            , n7287, n7288,
        n7289, n7290, n7291, n7292, n7293, n7294, n7295, n7296
            , n7297, n7298,
        n7299, n7300, n7301, n7302, n7303, n7304, n7305, n7306
            , n7307, n7308,
        n7309, n7310, n7311, n7312, n7313, n7314, n7315, n7316
            , n7317, n7318,
        n7319, n7320, n7321, n7322, n7323, n7324, n7325, n7326
            , n7327, n7328,
        n7329, n7330, n7331, n7332, n7333, n7334, n7335, n7336
            , n7337, n7338,
        n7339, n7340, n7341, n7342, n7343, n7344, n7345, n7346
            , n7347, n7348,
        n7349, n7350, n7351, n7352, n7353, n7354, n7355, n7356
            , n7357, n7358,
        n7359, n7360, n7361, n7362, n7363, n7364, n7365, n7366
            , n7367, n7368,
        n7369, n7370, n7371, n7372, n7373, n7374, n7375, n7376
            , n7377, n7378,
        n7379, n7380, n7381, n7382, n7383, n7384, n7385, n7386
            , n7387, n7388,
        n7389, n7390, n7391, n7392, n7393, n7394, n7395, n7396
            , n7397, n7398,
        n7399, n7400, n7401, n7402, n7403, n7404, n7405, n7406
            , n7407, n7408,
        n7409, n7410, n7411, n7412, n7413, n7414, n7415, n7416
            , n7417, n7418,
        n7419, n7420, n7421, n7422, n7423, n7424, n7425, n7426
            , n7427, n7428,
        n7429, n7430, n7431, n7432, n7433, n7434, n7435, n7436
            , n7437, n7438,
        n7439, n7440, n7441, n7442, n7443, n7444, n7445, n7446
            , n7447, n7448,
        n7449, n7450, n7451, n7452, n7453, n7454, n7455, n7456
            , n7457, n7458,
        n7459, n7460, n7461, n7462, n7463, n7464, n7465, n7466
            , n7467, n7468,
        n7469, n7470, n7471, n7472, n7473, n7474, n7475, n7476
            , n7477, n7478,
        n7479, n7480, n7481, n7482, n7483, n7484, n7485, n7486
            , n7487, n7488,
        n7489, n7490, n7491, n7492, n7493, n7494, n7495, n7496
            , n7497, n7498,
        n7499, n7500, n7501, n7502, n7503, n7504, n7505, n7506
            , n7507, n7508,
```

```
        n7509, n7510, n7511, n7512, n7513, n7514, n7515, n7516
            , n7517, n7518,
        n7519, n7520, n7521, n7522, n7523, n7524, n7525, n7526
            , n7527, n7528,
        n7529, n7530, n7531, n7532, n7533, n7534, n7535, n7536
            , n7537, n7538,
        n7539, n7540, n7541, n7542, n7543, n7544, n7545, n7546
            , n7547, n7548,
        n7549, n7550, n7551, n7552, n7553, n7554, n7555, n7556
            , n7557, n7558,
        n7559, n7560, n7561, n7562, n7563, n7564, n7565, n7566
            , n7567, n7568,
        n7569, n7570, n7571, n7572, n7573, n7574, n7575, n7576
            , n7577, n7578,
        n7579, n7580, n7581, n7582, n7583, n7584, n7585, n7586
            , n7587, n7588,
        n7589, n7590, n7591, n7592, n7593, n7594, n7595, n7596
            , n7597, n7598,
        n7599, n7600, n7601, n7602, n7603, n7604, n7605, n7606
            , n7607, n7608,
        n7609, n7610, n7611, n7612, n7613, n7614, n7615, n7616
            , n7617, n7618,
        n7619, n7620, n7621, n7622, n7623, n7624, n7625, n7626
            , n7627, n7628,
        n7629, n7630, n7631, n7632, n7633, n7634, n7635, n7636
            , n7637, n7638,
        n7639, n7640, n7641, n7642, n7643, n7644, n7645, n7646
            , n7647, n7648,
        n7649, n7650, n7651, n7652, n7653, n7654, n7655, n7656
            , n7657, n7658,
        n7659, n7660, n7661, n7662, n7663, n7664, n7665, n7666
            , n7667, n7668,
        n7669, n7670, n7671, n7672, n7673, n7674, n7675, n7676
            , n7677, n7678,
        n7679, n7680, n7681, n7682, n7683, n7684, n7685, n7686
            , n7687, n7688,
        n7689, n7690, n7691, n7692, n7693, n7694, n7695, n7696
            , n7697, n7698,
        n7699, n7700, n7701, n7702, n7703, n7704, n7705, n7706
            , n7707, n7708,
        n7709, n7710, n7711, n7712, n7713, n7714, n7715, n7716
            , n7717, n7718,
        n7719, n7720, n7721, n7722, n7723, n7724, n7725, n7726
            , n7727, n7728,
        n7729, n7730, n7731, n7732, n7733, n7734, n7735, n7736
            , n7737, n7738,
        n7739, n7740, n7741, n7742, n7743, n7744, n7745, n7746
            , n7747, n7748,
```

```
        n7749, n7750, n7751, n7752, n7753, n7754, n7755, n7756
            , n7757, n7758,
        n7759, n7760, n7761, n7762, n7763, n7764, n7765, n7766
            , n7767, n7768,
        n7769, n7770, n7771, n7772, n7773, n7774, n7775, n7776
            , n7777, n7778,
        n7779, n7780, n7781, n7782, n7783, n7784, n7785, n7786
            , n7787, n7788,
        n7789, n7790, n7791, n7792, n7793, n7794, n7795, n7796
            , n7797, n7798,
        n7799, n7800, n7801, n7802, n7803, n7804, n7805, n7806
            , n7807, n7808,
        n7809, n7810, n7811, n7812, n7813, n7814, n7815, n7816
            , n7817, n7818,
        n7819, n7820, n7821, n7822, n7823, n7824, n7825, n7826
            , n7827, n7828,
        n7829, n7830, n7831, n7832, n7833, n7834, n7835, n7836
            , n7837, n7838,
        n7839, n7840, n7841, n7842, n7843, n7844, n7845, n7846
            , n7847, n7848,
        n7849, n7850, n7851, n7852, n7853, n7854, n7855, n7856
            , n7857, n7858,
        n7859, n7860, n7861, n7862, n7863, n7864, n7865, n7866
            , n7867, n7868,
        n7869, n7870, n7871, n7872, n7873, n7874, n7875, n7876
            , n7877, n7878,
        n7879, n7880, n7881, n7882, n7883, n7884, n7885, n7886
            , n7887, n7888,
        n7889, n7890, n7891, n7892, n7893, n7894, n7895, n7896
            , n7897, n7898,
        n7899, n7900, n7901, n7902, n7903, n7904, n7905, n7906
            , n7907, n7908,
        n7909, n7910, n7911, n7912, n7913, n7914, n7915, n7916
            , n7917, n7918,
        n7919, n7920, n7921, n7922, n7923, n7924, n7925, n7926
            , n7927, n7928,
        n7929, n7930, n7931, n7932, n7933, n7934, n7935, n7936
            , n7937, n7938,
        n7939, n7940, n7941, n7942, n7943, n7944, n7945, n7946
            , n7947, n7948,
        n7949, n7950, n7951, n7952, n7953, n7954, n7955, n7956
            , n7957, n7958,
        n7959, n7960, n7961, n7962, n7963, n7964, n7965, n7966
            , n7967, n7968,
        n7969, n7970, n7971, n7972, n7973, n7974, n7975, n7976
            , n7977, n7978,
        n7979, n7980, n7981, n7982, n7983, n7984, n7985, n7986
            , n7987, n7988,
```

```
        n7989, n7990, n7991, n7992, n7993, n7994, n7995, n7996
            , n7997, n7998,
        n7999, n8000, n8001, n8002, n8003, n8004, n8005, n8006
            , n8007, n8008,
        n8009, n8010, n8011, n8012, n8013, n8014, n8015, n8016
            , n8017, n8018,
        n8019, n8020, n8021, n8022, n8023, n8024, n8025, n8026
            , n8027, n8028,
        n8029, n8030, n8031, n8032, n8033, n8034, n8035, n8036
            , n8037, n8038,
        n8039, n8040, n8041, n8042, n8043, n8044, n8045, n8046
            , n8047, n8048,
        n8049, n8050, n8051, n8052, n8053, n8054, n8055, n8056
            , n8057, n8058,
        n8059, n8060, n8061, n8062, n8063, n8064, n8065, n8066
            , n8067, n8068,
        n8069, n8070, n8071, n8072, n8073, n8074, n8075, n8076
            , n8077, n8078,
        n8079, n8080, n8081, n8082, n8083, n8084, n8085, n8086
            , n8087, n8088,
        n8089, n8090, n8091, n8092, n8093, n8094, n8095, n8096
            , n8097, n8098,
        n8099, n8100, n8101, n8102, n8103, n8104, n8105, n8106
            , n8107, n8108,
        n8109, n8110, n8111, n8112, n8113, n8114, n8115, n8116
            , n8117, n8118,
        n8119, n8120, n8121, n8122, n8123, n8124, n8125, n8126
            , n8127, n8128,
        n8129, n8130, n8131, n8132, n8133, n8134, n8135, n8136
            , n8137, n8138,
        n8139, n8140, n8141, n8142, n8143, n8144, n8145, n8146
            , n8147, n8148,
        n8149, n8150, n8151, n8152, n8153, n8154, n8155, n8156
            , n8157, n8158,
        n8159, n8160, n8161, n8162, n8163, n8164, n8165, n8166
            , n8167, n8168,
        n8169, n8170, n8171, n8172, n8173, n8174, n8175, n8176
            , n8177, n8178,
        n8179, n8180, n8181, n8182, n8183, n8184, n8185, n8186
            , n8187, n8188,
        n8189, n8190, n8191, n8192, n8193, n8194, n8195, n8196
            , n8197, n8198,
        n8199, n8200, n8201, n8202, n8203, n8204, n8205, n8206
            , n8207, n8208,
        n8209, n8210, n8211, n8212, n8213, n8214, n8215, n8216
            , n8217, n8218,
        n8219, n8220, n8221, n8222, n8223, n8224, n8225, n8226
            , n8227, n8228,
```

```
        n8229, n8230, n8231, n8232, n8233, n8234, n8235, n8236
            , n8237, n8238,
        n8239, n8240, n8241, n8242, n8243, n8244, n8245, n8246
            , n8247, n8248,
        n8249, n8250, n8251, n8252, n8253, n8254, n8255, n8256
            , n8257, n8258,
        n8259, n8260, n8261, n8262, n8263, n8264, n8265, n8266
            , n8267, n8268,
        n8269, n8270, n8271, n8272, n8273, n8274, n8275, n8276
            , n8277, n8278,
        n8279, n8280, n8281, n8282, n8283, n8284, n8285, n8286
            , n8287, n8288,
        n8289, n8290, n8291, n8292, n8293, n8294, n8295, n8296
            , n8297, n8298,
        n8299, n8300, n8301, n8302, n8303, n8304, n8305, n8306
            , n8307, n8308,
        n8309, n8310, n8311, n8312, n8313, n8314, n8315, n8316
            , n8317, n8318,
        n8319, n8320, n8321, n8322, n8323, n8324, n8325, n8326
            , n8327, n8328,
        n8329, n8330, n8331, n8332, n8333, n8334, n8335, n8336
            , n8337, n8338,
        n8339, n8340, n8341, n8342, n8343, n8344, n8345, n8346
            , n8347, n8348,
        n8349, n8350, n8351, n8352, n8353, n8354, n8355, n8356
            , n8357, n8358,
        n8359, n8360, n8361, n8362, n8363, n8364, n8365, n8366
            , n8367, n8368,
        n8369, n8370, n8371, n8372, n8373, n8374, n8375, n8376
            , n8377, n8378,
        n8379, n8380, n8381, n8382, n8383, n8384, n8385, n8386
            , n8387, n8388,
        n8389, n8390, n8391, n8392, n8393, n8394, n8395, n8396
            , n8397, n8398,
        n8399, n8400, n8401, n8402, n8403, n8404, n8405, n8406
            , n8407, n8408,
        n8409, n8410, n8411, n8412, n8413, n8414, n8415, n8416
            , n8417, n8418,
        n8419, n8420, n8421, n8422, n8423, n8424, n8425, n8426
            , n8427, n8428,
        n8429, n8430, n8431, n8432, n8433, n8434, n8435, n8436
            , n8437, n8438,
        n8439, n8440, n8441, n8442, n8443, n8444, n8445, n8446
            , n8447, n8448,
        n8449, n8450, n8451, n8452, n8453, n8454, n8455, n8456
            , n8457, n8458,
        n8459, n8460, n8461, n8462, n8463, n8464, n8465, n8466
            , n8467, n8468,
```

```
        n8469, n8470, n8471, n8472, n8473, n8474, n8475, n8476
           , n8477, n8478,
        n8479, n8480, n8481, n8482, n8483, n8484, n8485, n8486
           , n8487, n8488,
        n8489, n8490, n8491, n8492, n8493, n8494, n8495, n8496
           , n8497, n8498,
        n8499, n8500, n8501, n8502, n8503, n8504, n8505, n8506
           , n8507, n8508,
        n8509, n8510, n8511, n8512, n8513, n8514, n8515, n8516
           , n8517, n8518,
        n8519, n8520, n8521, n8522, n8523, n8524, n8525, n8526
           , n8527, n8528,
        n8529, n8530, n8531, n8532, n8533, n8534, n8535, n8536
           , n8537, n8538,
        n8539, n8540, n8541, n8542, n8543, n8544, n8545, n8546
           , n8547, n8548,
        n8549, n8550, n8551, n8552, n8553, n8554, n8555, n8556
           , n8557, n8558,
        n8559, n8560, n8561, n8562, n8563, n8564, n8565, n8566
           , n8567, n8568,
        n8569, n8570, n8571, n8572, n8573, n8574, n8575, n8576
           , n8577, n8578,
        n8579, n8580, n8581, n8582, n8583, n8584, n8585, n8586
           , n8587, n8588,
        n8589, n8590, n8591, n8592, n8593, n8594, n8595, n8596
           , n8597, n8598,
        n8599, n8600, n8601, n8602, n8603, n8604, n8605, n8606
           , n8607, n8608,
        n8609, n8610, n8611, n8612, n8613, n8614, n8615, n8616
           , n8617, n8618,
        n8619, n8620, n8621, n8622, n8623, n8624, n8625, n8626
           , n8627, n8628,
        n8629, n8630, n8631, n8632, n8633, n8634, n8635, n8636
           , n8637, n8638,
        n8639, n8640, n8641, n8642, n8643, n8644, n8645, n8646
           , n8647, n8648,
        n8649, n8650, n8651, n8652, n8653, n8654, n8655, n8656
           , n8657, n8658,
        n8659, n8660, n8661, n8662, n8663, n8664, n8665, n8666
           , n8667, n8668,
        n8669, n8670, n8671, n8672, n8673, n8674, n8675, n8676
           , n8677, n8678,
        n8679, n8680, n8681, n8682, n8683, n8684, n8685, n8686
           , n8687, n8688,
        n8689, n8690, n8691, n8692, n8693, n8694, n8695, n8696
           , n8697, n8698,
        n8699, n8700, n8701, n8702, n8703, n8704, n8705, n8706
           , n8707, n8708,
```

```
        n8709, n8710, n8711, n8712, n8713, n8714, n8715, n8716
           , n8717, n8718,
        n8719, n8720, n8721, n8722, n8723, n8724, n8725, n8726
           , n8727, n8728,
        n8729, n8730, n8731, n8732, n8733, n8734, n8735, n8736
           , n8737, n8738,
        n8739, n8740, n8741, n8742, n8743, n8744, n8745, n8746
           , n8747, n8748,
        n8749, n8750, n8751, n8752, n8753, n8754, n8755, n8756
           , n8757, n8758,
        n8759, n8760, n8761, n8762, n8763, n8764, n8765, n8766
           , n8767, n8768,
        n8769, n8770, n8771, n8772, n8773, n8774, n8775, n8776
           , n8777, n8778,
        n8779, n8780, n8781, n8782, n8783, n8784, n8785, n8786
           , n8787, n8788,
        n8789, n8790, n8791, n8792, n8793, n8794, n8795, n8796
           , n8797, n8798,
        n8799, n8800, n8801, n8802, n8803, n8804, n8805, n8806
           , n8807, n8808,
        n8809, n8810, n8811, n8812, n8813, n8814, n8815, n8816
           , n8817, n8818,
        n8819, n8820, n8821, n8822, n8823, n8824, n8825, n8826
           , n8827, n8828,
        n8829, n8830, n8831, n8832, n8833, n8834, n8835, n8836
           , n8837, n8838,
        n8839, n8840, n8841, n8842, n8843, n8844, n8845, n8846
           , n8847, n8848,
        n8849, n8850, n8851, n8852, n8853, n8854, n8855, n8856
           , n8857, n8858,
        n8859, n8860, n8861, n8862, n8863, n8864, n8865, n8866
           , n8867, n8868,
        n8869, n8870, n8871, n8872, n8873, n8874, n8875, n8876
           , n8877, n8878,
        n8879, n8880, n8881, n8882, n8883, n8884, n8885, n8886
           , n8887, n8888,
        n8889, n8890, n8891, n8892, n8893, n8894, n8895, n8896
           , n8897, n8898,
        n8899, n8900, n8901, n8902, n8903, n8904, n8905, n8906
           , n8907, n8908,
        n8909, n8910, n8911, n8912, n8913, n8914, n8915, n8916
           , n8917, n8918,
        n8919, n8920, n8921, n8922, n8923, n8924, n8925, n8926
           , n8927, n8928,
        n8929, n8930, n8931, n8932, n8933, n8934, n8935, n8936
           , n8937, n8938,
        n8939, n8940, n8941, n8942, n8943, n8944, n8945, n8946
           , n8947, n8948,
```

```
        n8949, n8950, n8951, n8952, n8953, n8954, n8955, n8956
            , n8957, n8958,
        n8959, n8960, n8961, n8962, n8963, n8964, n8965, n8966
            , n8967, n8968,
        n8969, n8970, n8971, n8972, n8973, n8974, n8975, n8976
            , n8977, n8978,
        n8979, n8980, n8981, n8982, n8983, n8984, n8985, n8986
            , n8987, n8988,
        n8989, n8990, n8991, n8992, n8993, n8994, n8995, n8996
            , n8997, n8998,
        n8999, n9000, n9001, n9002, n9003, n9004, n9005, n9006
            , n9007, n9008,
        n9009, n9010, n9011, n9012, n9013, n9014, n9015, n9016
            , n9017, n9018,
        n9019, n9020, n9021, n9022, n9023, n9024, n9025, n9026
            , n9027, n9028,
        n9029, n9030, n9031, n9032, n9033, n9034, n9035, n9036
            , n9037, n9038,
        n9039, n9040, n9041, n9042, n9043, n9044, n9045, n9046
            , n9047, n9048,
        n9049, n9050, n9051, n9052, n9053, n9054, n9055, n9056
            , n9057, n9058,
        n9059, n9060, n9061, n9062, n9063, n9064, n9065, n9066
            , n9067, n9068,
        n9069, n9070, n9071, n9072, n9073, n9074, n9075, n9076
            , n9077, n9078,
        n9079, n9080, n9081, n9082, n9083, n9084, n9085, n9086
            , n9087, n9088,
        n9089, n9090, n9091, n9092, n9093, n9094, n9095, n9096
            , n9097, n9098,
        n9099, n9100, n9101, n9102, n9103, n9104, n9105, n9106
            , n9107, n9108,
        n9109, n9110, n9111, n9112, n9113, n9114, n9115, n9116
            , n9117, n9118,
        n9119, n9120, n9121, n9122, n9123, n9124, n9125, n9126
            , n9127, n9128,
        n9129, n9130, n9131, n9132, n9133, n9134, n9135, n9136
            , n9137, n9138,
        n9139, n9140, n9141, n9142, n9143, n9144, n9145, n9146
            , n9147, n9148,
        n9149, n9150, n9151, n9152, n9153, n9154, n9155, n9156
            , n9157, n9158,
        n9159, n9160, n9161, n9162, n9163, n9164, n9165, n9166
            , n9167, n9168,
        n9169, n9170, n9171, n9172, n9173, n9174, n9175, n9176
            , n9177, n9178,
        n9179, n9180, n9181, n9182, n9183, n9184, n9185, n9186
            , n9187, n9188,
```

```
        n9189, n9190, n9191, n9192, n9193, n9194, n9195, n9196
           , n9197, n9198,
        n9199, n9200, n9201, n9202, n9203, n9204, n9205, n9206
           , n9207, n9208,
        n9209, n9210, n9211, n9212, n9213, n9214, n9215, n9216
           , n9217, n9218,
        n9219, n9220, n9221, n9222, n9223, n9224, n9225, n9226
           , n9227, n9228,
        n9229, n9230, n9231, n9232, n9233, n9234, n9235, n9236
           , n9237, n9238,
        n9239, n9240, n9241, n9242, n9243, n9244, n9245, n9246
           , n9247, n9248,
        n9249, n9250, n9251, n9252, n9253, n9254, n9255, n9256
           , n9257, n9258,
        n9259, n9260, n9261, n9262, n9263, n9264, n9265, n9266
           , n9267, n9268,
        n9269, n9270, n9271, n9272, n9273, n9274, n9275, n9276
           , n9277, n9278,
        n9279, n9280, n9281, n9282, n9283, n9284, n9285, n9286
           , n9287, n9288,
        n9289, n9290, n9291, n9292, n9293, n9294, n9295, n9296
           , n9297, n9298,
        n9299, n9300, n9301, n9302, n9303, n9304, n9305, n9306
           , n9307, n9308,
        n9309, n9310, n9311, n9312, n9313, n9314, n9315, n9316
           , n9317, n9318,
        n9319, n9320, n9321, n9322, n9323, n9324, n9325, n9326
           , n9327, n9328,
        n9329, n9330, n9331, n9332, n9333, n9334, n9335, n9336
           , n9337, n9338,
        n9339, n9340, n9341, n9342, n9343, n9344, n9345, n9346
           , n9347, n9348,
        n9349, n9350, n9351, n9352;
wire    [11:0] counter;

OR2D1 C91623 ( .A1(reset_ring_osc), .A2(out_ring_osc), .Z(W_1
   ) );
DFCNQD1 \counter_reg[11]  ( .D(N4778), .CP(clk), .CDN(n4811),
    .Q(counter[11]) );
DFCNQD1 \counter_reg[10]  ( .D(N4777), .CP(clk), .CDN(n4811),
    .Q(counter[10]) );
DFCNQD1 \counter_reg[9]  ( .D(N4776), .CP(clk), .CDN(n4811),
   .Q(counter[9])
       );
DFCNQD1 \counter_reg[8]  ( .D(N4775), .CP(clk), .CDN(n4811),
   .Q(counter[8])
       );
```

```
DFCNQD1 \counter_reg[7]  ( .D(N4774), .CP(clk), .CDN(n4811),
   .Q(counter[7])
       );
DFCNQD1 \counter_reg[6]  ( .D(N4773), .CP(clk), .CDN(n4811),
   .Q(counter[6])
       );
DFCNQD1 \counter_reg[5]  ( .D(N4772), .CP(clk), .CDN(n4811),
   .Q(counter[5])
       );
DFCNQD1 \counter_reg[4]  ( .D(N4771), .CP(clk), .CDN(n4811),
   .Q(counter[4])
       );
DFCNQD1 \counter_reg[3]  ( .D(N4770), .CP(clk), .CDN(n4811),
   .Q(counter[3])
       );
DFCNQD1 \counter_reg[2]  ( .D(N4769), .CP(clk), .CDN(n4811),
   .Q(counter[2])
       );
DFCNQD1 \counter_reg[1]  ( .D(N4768), .CP(clk), .CDN(n4811),
   .Q(counter[1])
       );
DFCNQD1 \counter_reg[0]  ( .D(N4767), .CP(clk), .CDN(n4811),
   .Q(counter[0])
       );
DFQD1 \q_reg[6]  ( .D(n4818), .CP(clk), .Q(q_out[6]) );
DFQD1 \q_reg[5]  ( .D(n4817), .CP(clk), .Q(q_out[5]) );
DFQD1 \q_reg[4]  ( .D(n4816), .CP(clk), .Q(q_out[4]) );
DFQD1 \q_reg[3]  ( .D(n4815), .CP(clk), .Q(q_out[3]) );
DFQD1 \q_reg[2]  ( .D(n4814), .CP(clk), .Q(q_out[2]) );
DFQD1 \q_reg[1]  ( .D(n4813), .CP(clk), .Q(q_out[1]) );
DFQD1 \q_reg[0]  ( .D(n4812), .CP(clk), .Q(q_out[0]) );
INVD1 I_4722 ( .I(W_1), .ZN(W_2) );
INVD1 I_4723 ( .I(W_2), .ZN(W_3) );
INVD1 I_4724 ( .I(W_3), .ZN(W_4) );
INVD1 I_4725 ( .I(W_4), .ZN(W_5) );
INVD1 I_4726 ( .I(W_5), .ZN(W_6) );
INVD1 I_4727 ( .I(W_6), .ZN(W_7) );
INVD1 I_4728 ( .I(W_7), .ZN(W_8) );
INVD1 I_4729 ( .I(W_8), .ZN(W_9) );
INVD1 I_4730 ( .I(W_9), .ZN(W_10) );
INVD1 I_4731 ( .I(W_10), .ZN(W_11) );
INVD1 I_4732 ( .I(W_11), .ZN(W_12) );
INVD1 I_4733 ( .I(W_12), .ZN(W_13) );
INVD0 I_4734 ( .I(W_13), .ZN(out_ring_osc) );
INVD0 U4834 ( .I(n5576), .ZN(n4819) );
NR2XD0 U4835 ( .A1(n7514), .A2(counter[6]), .ZN(n9307) );
CKND2D1 U4836 ( .A1(n7514), .A2(counter[4]), .ZN(n8923) );
INVD0 U4837 ( .I(counter[5]), .ZN(n7514) );
```

```
NR2XD0 U4838 ( .A1(n9352), .A2(counter[5]), .ZN(n9312) );
NR2XD0 U4839 ( .A1(n9352), .A2(n7514), .ZN(n9141) );
CKND2D0 U4840 ( .A1(n9348), .A2(counter[5]), .ZN(n8674) );
TIEL U4841 ( .ZN(q_out[7]) );
INVD0 U4842 ( .I(q_out[0]), .ZN(n5358) );
INVD0 U4843 ( .I(counter[7]), .ZN(n9153) );
INVD0 U4844 ( .I(counter[9]), .ZN(n9075) );
INVD0 U4845 ( .I(counter[10]), .ZN(n9342) );
CKND2D0 U4846 ( .A1(n9075), .A2(n9342), .ZN(n4889) );
INVD0 U4847 ( .I(n4889), .ZN(n7437) );
INVD0 U4848 ( .I(counter[3]), .ZN(n9345) );
NR2D0 U4849 ( .A1(counter[3]), .A2(counter[2]), .ZN(n6639) );
INVD0 U4850 ( .I(n6639), .ZN(n7743) );
NR2XD0 U4851 ( .A1(counter[4]), .A2(counter[5]), .ZN(n8582) )
   ;
CKND1 U4852 ( .I(n8582), .ZN(n8935) );
INVD0 U4853 ( .I(counter[6]), .ZN(n9352) );
OAI21D0 U4854 ( .A1(n7743), .A2(n8935), .B(counter[6]), .ZN(
   n6995) );
INVD0 U4855 ( .I(counter[8]), .ZN(n9338) );
AOI32D0 U4856 ( .A1(n9153), .A2(n7437), .A3(n6995), .B1(n9338
   ), .B2(n7437),
      .ZN(n4820) );
INVD0 U4857 ( .I(counter[11]), .ZN(n9329) );
XNR2D0 U4858 ( .A1(select[0]), .A2(select[1]), .ZN(n9328) );
IOA21D0 U4859 ( .A1(n4820), .A2(counter[11]), .B(n9328), .ZN(
   n7000) );
INVD0 U4860 ( .I(counter[4]), .ZN(n9348) );
CKND2D0 U4861 ( .A1(counter[3]), .A2(counter[2]), .ZN(n7282)
   );
NR2D0 U4862 ( .A1(n9348), .A2(n7282), .ZN(n6997) );
NR2D0 U4863 ( .A1(counter[0]), .A2(counter[1]), .ZN(n9132) );
INVD0 U4864 ( .I(n9132), .ZN(n9189) );
NR2D0 U4865 ( .A1(counter[7]), .A2(counter[5]), .ZN(n7529) );
INVD0 U4866 ( .I(n7529), .ZN(n7556) );
AOI211D0 U4867 ( .A1(n6997), .A2(n9189), .B(counter[6]), .C(
   n7556), .ZN(
      n4821) );
OAI21D0 U4868 ( .A1(n4821), .A2(n9338), .B(n7437), .ZN(n4822)
    );
AOI21D0 U4869 ( .A1(counter[11]), .A2(n4822), .B(n9328), .ZN(
   n8949) );
INVD0 U4870 ( .I(n8949), .ZN(n6992) );
CKND2D0 U4871 ( .A1(n7000), .A2(n6992), .ZN(n7277) );
CKND2D0 U4872 ( .A1(n9338), .A2(counter[7]), .ZN(n5030) );
NR2D0 U4873 ( .A1(n9342), .A2(counter[9]), .ZN(n4854) );
INVD0 U4874 ( .I(n4854), .ZN(n4983) );
NR2D0 U4875 ( .A1(n5030), .A2(n4983), .ZN(n8037) );
```

```
  NR2D0 U4876 ( .A1(n9348), .A2(counter[3]), .ZN(n8998) );
  CKND2D0 U4877 ( .A1(counter[1]), .A2(n8998), .ZN(n7672) );
  INVD0 U4878 ( .I(counter[2]), .ZN(n8983) );
  NR2D0 U4879 ( .A1(n8983), .A2(counter[3]), .ZN(n5766) );
  INVD0 U4880 ( .I(counter[0]), .ZN(n9143) );
  NR2D0 U4881 ( .A1(n9143), .A2(counter[11]), .ZN(n7785) );
  CKND2D0 U4882 ( .A1(n5766), .A2(n7785), .ZN(n6536) );
  NR2D0 U4883 ( .A1(n7672), .A2(n6536), .ZN(n7911) );
  INVD0 U4884 ( .I(n7911), .ZN(n6548) );
  NR2D0 U4885 ( .A1(counter[11]), .A2(counter[0]), .ZN(n5202) )
     ;
  CKND2D0 U4886 ( .A1(counter[2]), .A2(n5202), .ZN(n4823) );
  INVD0 U4887 ( .I(n4823), .ZN(n8216) );
  INVD0 U4888 ( .I(counter[1]), .ZN(n6205) );
  NR2D0 U4889 ( .A1(n9345), .A2(n6205), .ZN(n7002) );
  INVD0 U4890 ( .I(n7002), .ZN(n7169) );
  NR2D0 U4891 ( .A1(n9348), .A2(n7169), .ZN(n7786) );
  CKND2D0 U4892 ( .A1(n8216), .A2(n7786), .ZN(n7729) );
  CKND2D0 U4893 ( .A1(n6548), .A2(n7729), .ZN(n7799) );
  INVD0 U4894 ( .I(n5202), .ZN(n6512) );
  NR2D0 U4895 ( .A1(n6205), .A2(n6512), .ZN(n4992) );
  CKND2D0 U4896 ( .A1(n9345), .A2(n9348), .ZN(n6559) );
  INVD0 U4897 ( .I(n6559), .ZN(n8461) );
  CKND2D0 U4898 ( .A1(n4992), .A2(n8461), .ZN(n6539) );
  NR2D0 U4899 ( .A1(counter[2]), .A2(n6539), .ZN(n7634) );
  INVD0 U4900 ( .I(n7634), .ZN(n7641) );
  CKND2D0 U4901 ( .A1(counter[3]), .A2(counter[4]), .ZN(n8472)
     );
  NR2D0 U4902 ( .A1(n9143), .A2(n8472), .ZN(n8971) );
  CKND2D0 U4903 ( .A1(n6205), .A2(counter[2]), .ZN(n8957) );
  INVD0 U4904 ( .I(n8957), .ZN(n8053) );
  CKND2D0 U4905 ( .A1(n9329), .A2(n8053), .ZN(n8278) );
  INVD0 U4906 ( .I(n8278), .ZN(n5452) );
  CKND2D0 U4907 ( .A1(n8971), .A2(n5452), .ZN(n7656) );
  CKND2D0 U4908 ( .A1(n7641), .A2(n7656), .ZN(n7836) );
  CKND2D0 U4909 ( .A1(n8983), .A2(n7785), .ZN(n5869) );
  INVD0 U4910 ( .I(n7786), .ZN(n4989) );
  NR2D0 U4911 ( .A1(n5869), .A2(n4989), .ZN(n6616) );
  NR2D0 U4912 ( .A1(n7836), .A2(n6616), .ZN(n7875) );
  INVD0 U4913 ( .I(n7875), .ZN(n5052) );
  CKND2D0 U4914 ( .A1(n9143), .A2(counter[4]), .ZN(n7280) );
  INVD0 U4915 ( .I(n7280), .ZN(n9025) );
  CKND2D0 U4916 ( .A1(n9025), .A2(n5452), .ZN(n6640) );
  NR2D0 U4917 ( .A1(n6640), .A2(counter[3]), .ZN(n7585) );
  CKND2D0 U4918 ( .A1(n8983), .A2(counter[3]), .ZN(n6592) );
  INVD0 U4919 ( .I(n4992), .ZN(n5861) );
  NR2D0 U4920 ( .A1(n6592), .A2(n5861), .ZN(n6275) );
  CKND2D0 U4921 ( .A1(n6275), .A2(n9348), .ZN(n6523) );
```

```
  INVD0 U4922 ( .I(n6523), .ZN(n6662) );
  NR2D0 U4923 ( .A1(n7585), .A2(n6662), .ZN(n7868) );
  NR2D0 U4924 ( .A1(n7169), .A2(n4823), .ZN(n7870) );
  INVD0 U4925 ( .I(n7870), .ZN(n7618) );
  NR2D0 U4926 ( .A1(n7618), .A2(counter[4]), .ZN(n7766) );
  INVD0 U4927 ( .I(n7766), .ZN(n7825) );
  CKND2D0 U4928 ( .A1(counter[4]), .A2(counter[0]), .ZN(n8970)
     );
  INVD0 U4929 ( .I(n8970), .ZN(n9294) );
  CKND2D0 U4930 ( .A1(n9345), .A2(n9294), .ZN(n8954) );
  INVD0 U4931 ( .I(n8954), .ZN(n9014) );
  CKND2D0 U4932 ( .A1(n8983), .A2(n6205), .ZN(n8953) );
  INVD0 U4933 ( .I(n8953), .ZN(n8156) );
  CKND2D0 U4934 ( .A1(n9329), .A2(n8156), .ZN(n8725) );
  INVD0 U4935 ( .I(n8725), .ZN(n5385) );
  CKND2D0 U4936 ( .A1(n9014), .A2(n5385), .ZN(n7642) );
  INVD0 U4937 ( .I(n7642), .ZN(n5050) );
  ND3D0 U4938 ( .A1(n7868), .A2(n7825), .A3(n7642), .ZN(n5040)
     );
  CKND2D0 U4939 ( .A1(n6205), .A2(n7785), .ZN(n8814) );
  NR2D0 U4940 ( .A1(n8814), .A2(n6559), .ZN(n7569) );
  NR2D0 U4941 ( .A1(n8983), .A2(n6205), .ZN(n8951) );
  INVD0 U4942 ( .I(n8951), .ZN(n8055) );
  NR2D0 U4943 ( .A1(n9345), .A2(n8055), .ZN(n7414) );
  CKND2D0 U4944 ( .A1(n7414), .A2(n7785), .ZN(n7668) );
  NR2D0 U4945 ( .A1(counter[4]), .A2(n7668), .ZN(n7578) );
  INVD0 U4946 ( .I(n7578), .ZN(n7969) );
  NR2D0 U4947 ( .A1(n9345), .A2(counter[4]), .ZN(n4915) );
  INVD0 U4948 ( .I(n8814), .ZN(n5102) );
  CKND2D0 U4949 ( .A1(n4915), .A2(n5102), .ZN(n7636) );
  NR2D0 U4950 ( .A1(n7636), .A2(counter[2]), .ZN(n7609) );
  CKND2D0 U4951 ( .A1(n9345), .A2(counter[1]), .ZN(n6340) );
  NR2D0 U4952 ( .A1(n6340), .A2(counter[2]), .ZN(n7350) );
  INVD0 U4953 ( .I(n7350), .ZN(n7294) );
  CKND2D0 U4954 ( .A1(n9348), .A2(n7785), .ZN(n7744) );
  NR2D0 U4955 ( .A1(n7294), .A2(n7744), .ZN(n6696) );
  NR2D0 U4956 ( .A1(n7609), .A2(n6696), .ZN(n6677) );
  CKND2D0 U4957 ( .A1(n7350), .A2(n5202), .ZN(n5677) );
  NR2D0 U4958 ( .A1(n9348), .A2(n5677), .ZN(n6694) );
  INVD0 U4959 ( .I(n8971), .ZN(n9178) );
  NR2D0 U4960 ( .A1(n9178), .A2(n8725), .ZN(n6562) );
  NR2D0 U4961 ( .A1(n6694), .A2(n6562), .ZN(n5056) );
  ND3D0 U4962 ( .A1(n7969), .A2(n6677), .A3(n5056), .ZN(n5022)
     );
  INVD0 U4963 ( .I(n4915), .ZN(n9029) );
  CKND2D0 U4964 ( .A1(n5452), .A2(n9143), .ZN(n8028) );
  NR2D0 U4965 ( .A1(n9029), .A2(n8028), .ZN(n7596) );
```

```
OR3D0 U4966 ( .A1(n7569), .A2(n5022), .A3(n7596), .Z(n4975) )
   ;
NR4D0 U4967 ( .A1(n7799), .A2(n5052), .A3(n5040), .A4(n4975),
    .ZN(n4829) );
INVD0 U4968 ( .I(n9141), .ZN(n9300) );
CKND2D0 U4969 ( .A1(counter[3]), .A2(n9025), .ZN(n8982) );
INVD0 U4970 ( .I(n8982), .ZN(n8964) );
CKND2D0 U4971 ( .A1(n8964), .A2(n5385), .ZN(n7842) );
CKND2D0 U4972 ( .A1(counter[0]), .A2(counter[1]), .ZN(n7288)
   );
NR2D0 U4973 ( .A1(n7282), .A2(n7288), .ZN(n9343) );
INVD0 U4974 ( .I(n9343), .ZN(n9347) );
NR2D0 U4975 ( .A1(n9348), .A2(n9347), .ZN(n9346) );
CKND2D0 U4976 ( .A1(n9329), .A2(n9346), .ZN(n7678) );
ND3D0 U4977 ( .A1(n6539), .A2(n7842), .A3(n7678), .ZN(n4984)
   );
NR2XD0 U4978 ( .A1(counter[6]), .A2(counter[5]), .ZN(n9226) )
   ;
INVD0 U4979 ( .I(n7636), .ZN(n4971) );
CKND2D0 U4980 ( .A1(counter[2]), .A2(n4971), .ZN(n7915) );
INVD0 U4981 ( .I(n7915), .ZN(n7640) );
NR2D0 U4982 ( .A1(n7911), .A2(n7640), .ZN(n5020) );
INVD0 U4983 ( .I(n7729), .ZN(n7579) );
CKND2D0 U4984 ( .A1(n8983), .A2(counter[1]), .ZN(n8950) );
INVD0 U4985 ( .I(n8950), .ZN(n8046) );
CKND2D0 U4986 ( .A1(n9329), .A2(n8046), .ZN(n8924) );
INVD0 U4987 ( .I(n8924), .ZN(n6347) );
CKND2D0 U4988 ( .A1(n6347), .A2(n9014), .ZN(n6529) );
INVD0 U4989 ( .I(n6529), .ZN(n7854) );
NR2D0 U4990 ( .A1(n7579), .A2(n7854), .ZN(n7781) );
NR2D0 U4991 ( .A1(n8953), .A2(counter[3]), .ZN(n7413) );
NR2D0 U4992 ( .A1(n6512), .A2(counter[4]), .ZN(n5858) );
CKND2D0 U4993 ( .A1(n7413), .A2(n5858), .ZN(n7795) );
INVD0 U4994 ( .I(n7795), .ZN(n5073) );
NR2D0 U4995 ( .A1(n6592), .A2(n7744), .ZN(n6576) );
NR2D0 U4996 ( .A1(n5073), .A2(n6576), .ZN(n5058) );
INVD0 U4997 ( .I(n5058), .ZN(n5080) );
INVD0 U4998 ( .I(n5056), .ZN(n5063) );
NR2D0 U4999 ( .A1(n5080), .A2(n5063), .ZN(n4830) );
ND4D0 U5000 ( .A1(n5020), .A2(n7781), .A3(n4830), .A4(n6640),
    .ZN(n4826) );
NR2D0 U5001 ( .A1(n7854), .A2(n6662), .ZN(n6520) );
INVD0 U5002 ( .I(n6520), .ZN(n4904) );
NR2D0 U5003 ( .A1(n9345), .A2(n8950), .ZN(n7064) );
CKND2D0 U5004 ( .A1(counter[4]), .A2(n5202), .ZN(n6618) );
INVD0 U5005 ( .I(n6618), .ZN(n5862) );
CKND2D0 U5006 ( .A1(n7064), .A2(n5862), .ZN(n7959) );
NR2D0 U5007 ( .A1(n8055), .A2(n7744), .ZN(n7714) );
```

```
CKND2D0 U5008 ( .A1(n9345), .A2(n7714), .ZN(n5018) );
CKND2D0 U5009 ( .A1(n7959), .A2(n5018), .ZN(n7580) );
NR2D0 U5010 ( .A1(n8953), .A2(n7744), .ZN(n7855) );
CKND2D0 U5011 ( .A1(n9345), .A2(n7855), .ZN(n6527) );
IND2D0 U5012 ( .A1(n7580), .B1(n6527), .ZN(n7821) );
NR2D0 U5013 ( .A1(n8957), .A2(counter[3]), .ZN(n7314) );
INVD0 U5014 ( .I(n7314), .ZN(n7295) );
NR2D0 U5015 ( .A1(n7295), .A2(n7744), .ZN(n7899) );
NR2D0 U5016 ( .A1(n6512), .A2(counter[2]), .ZN(n5870) );
CKND2D0 U5017 ( .A1(n4915), .A2(n5870), .ZN(n4965) );
NR2D0 U5018 ( .A1(n4965), .A2(counter[1]), .ZN(n6589) );
CKND2D0 U5019 ( .A1(n9014), .A2(n5452), .ZN(n7877) );
INVD0 U5020 ( .I(n7877), .ZN(n7804) );
NR2D0 U5021 ( .A1(n6589), .A2(n7804), .ZN(n6676) );
INVD0 U5022 ( .I(n6676), .ZN(n6612) );
NR2D0 U5023 ( .A1(n7899), .A2(n6612), .ZN(n7600) );
ND3D0 U5024 ( .A1(counter[2]), .A2(n8998), .A3(n4992), .ZN(
   n7850) );
CKND2D0 U5025 ( .A1(n7600), .A2(n7850), .ZN(n4877) );
NR2D0 U5026 ( .A1(n8953), .A2(counter[0]), .ZN(n8130) );
CKND2D0 U5027 ( .A1(n8998), .A2(n8130), .ZN(n8966) );
INVD0 U5028 ( .I(n8966), .ZN(n9124) );
CKND2D0 U5029 ( .A1(n9329), .A2(n9124), .ZN(n7774) );
CKND2D0 U5030 ( .A1(n7729), .A2(n7774), .ZN(n6530) );
INVD0 U5031 ( .I(n6530), .ZN(n7581) );
CKND2D0 U5032 ( .A1(n7581), .A2(n6548), .ZN(n4911) );
NR4D0 U5033 ( .A1(n4904), .A2(n7821), .A3(n4877), .A4(n4911),
    .ZN(n4824) );
INVD0 U5034 ( .I(n9307), .ZN(n9272) );
AOI21D0 U5035 ( .A1(n4830), .A2(n4824), .B(n9272), .ZN(n4825)
    );
AOI221D0 U5036 ( .A1(n4984), .A2(n9226), .B1(n4826), .B2(
   n9226), .C(n4825),
      .ZN(n4828) );
CKND2D0 U5037 ( .A1(n5766), .A2(n4992), .ZN(n5601) );
NR2D0 U5038 ( .A1(n7585), .A2(n7596), .ZN(n6537) );
ND3D0 U5039 ( .A1(n5601), .A2(n7842), .A3(n6537), .ZN(n5051)
   );
NR2D0 U5040 ( .A1(n5051), .A2(n6696), .ZN(n5085) );
INVD0 U5041 ( .I(n5085), .ZN(n5007) );
NR2D0 U5042 ( .A1(n6694), .A2(n7579), .ZN(n6615) );
CKND2D0 U5043 ( .A1(n6615), .A2(n5020), .ZN(n7691) );
NR2D0 U5044 ( .A1(n9345), .A2(n8957), .ZN(n7987) );
CKND2D0 U5045 ( .A1(n7987), .A2(n5202), .ZN(n6557) );
NR2D0 U5046 ( .A1(n9348), .A2(n6557), .ZN(n7635) );
INVD0 U5047 ( .I(n7635), .ZN(n5006) );
ND3D0 U5048 ( .A1(n7600), .A2(n5006), .A3(n7795), .ZN(n5033)
   );
```

```
  OAI31D0 U5049 ( .A1(n5007), .A2(n7691), .A3(n5033), .B(n9312)
     , .ZN(n4827) );
  OAI211D0 U5050 ( .A1(n4829), .A2(n9300), .B(n4828), .C(n4827)
     , .ZN(n5100) );
  NR2D0 U5051 ( .A1(n8983), .A2(n6539), .ZN(n6519) );
  NR2D0 U5052 ( .A1(n6519), .A2(n6562), .ZN(n7582) );
  INVD0 U5053 ( .I(n7064), .ZN(n7285) );
  NR2D0 U5054 ( .A1(n7285), .A2(n7744), .ZN(n7767) );
  INVD0 U5055 ( .I(n7656), .ZN(n6568) );
  NR2D0 U5056 ( .A1(n7767), .A2(n6568), .ZN(n7796) );
  CKND2D0 U5057 ( .A1(n7959), .A2(n7642), .ZN(n4976) );
  INVD0 U5058 ( .I(n4976), .ZN(n4962) );
  ND3D0 U5059 ( .A1(n7796), .A2(n4962), .A3(n7915), .ZN(n4994)
     );
  NR2D0 U5060 ( .A1(n4994), .A2(n4911), .ZN(n4960) );
  IND4D0 U5061 ( .A1(n4877), .B1(n7582), .B2(n6677), .B3(n4960)
     , .ZN(n4837) );
  NR2D0 U5062 ( .A1(n7640), .A2(n6662), .ZN(n5023) );
  NR2D0 U5063 ( .A1(n7899), .A2(n7635), .ZN(n7938) );
  INVD0 U5064 ( .I(n7938), .ZN(n7676) );
  INVD0 U5065 ( .I(n7678), .ZN(n7586) );
  NR2D0 U5066 ( .A1(n7676), .A2(n7586), .ZN(n6693) );
  INVD0 U5067 ( .I(n6592), .ZN(n6248) );
  CKND2D0 U5068 ( .A1(n6248), .A2(n5862), .ZN(n7577) );
  INVD0 U5069 ( .I(n7577), .ZN(n7608) );
  NR2D0 U5070 ( .A1(n7608), .A2(n6696), .ZN(n6575) );
  INVD0 U5071 ( .I(n6575), .ZN(n7769) );
  NR2D0 U5072 ( .A1(n7769), .A2(n7596), .ZN(n4977) );
  AN4D0 U5073 ( .A1(n5023), .A2(n6693), .A3(n4830), .A4(n4977),
      .Z(n4831) );
  INVD0 U5074 ( .I(n6519), .ZN(n6648) );
  CKND2D0 U5075 ( .A1(n8998), .A2(n5102), .ZN(n5054) );
  CKND1 U5076 ( .I(n9226), .ZN(n9318) );
  AOI31D0 U5077 ( .A1(n4831), .A2(n6648), .A3(n5054), .B(n9318)
     , .ZN(n4836) );
  INVD0 U5078 ( .I(n6527), .ZN(n7890) );
  NR3D0 U5079 ( .A1(n7585), .A2(n6616), .A3(n7890), .ZN(n4974)
     );
  CKND2D0 U5080 ( .A1(n5018), .A2(n7825), .ZN(n6550) );
  NR2D0 U5081 ( .A1(counter[4]), .A2(counter[0]), .ZN(n8968) );
  INVD0 U5082 ( .I(n8968), .ZN(n9193) );
  NR2D0 U5083 ( .A1(n9193), .A2(n8278), .ZN(n4862) );
  NR2D0 U5084 ( .A1(n6550), .A2(n4862), .ZN(n4959) );
  CKND2D0 U5085 ( .A1(n6648), .A2(n7642), .ZN(n7689) );
  INVD0 U5086 ( .I(n7689), .ZN(n7873) );
  NR2D0 U5087 ( .A1(n7911), .A2(n6562), .ZN(n5031) );
  ND3D0 U5088 ( .A1(n7873), .A2(n5031), .A3(n7668), .ZN(n5059)
     );
```

```
  IINR4D0 U5089 ( .A1(n4974), .A2(n4959), .B1(n5059), .B2(n4877
     ), .ZN(n4834)
       );
  INVD0 U5090 ( .I(n9312), .ZN(n9274) );
  NR2D0 U5091 ( .A1(n7578), .A2(n6568), .ZN(n4988) );
  CKND2D0 U5092 ( .A1(n4988), .A2(n6529), .ZN(n5081) );
  INVD0 U5093 ( .I(n7774), .ZN(n6632) );
  NR2D0 U5094 ( .A1(n7635), .A2(n6632), .ZN(n4949) );
  CKND2D0 U5095 ( .A1(n7641), .A2(n7877), .ZN(n6695) );
  INVD0 U5096 ( .I(n6695), .ZN(n7925) );
  INVD0 U5097 ( .I(n6589), .ZN(n7780) );
  CKND2D0 U5098 ( .A1(n7925), .A2(n7780), .ZN(n7779) );
  NR2D0 U5099 ( .A1(n7899), .A2(n7779), .ZN(n4902) );
  ND4D0 U5100 ( .A1(n4949), .A2(n4977), .A3(n4902), .A4(n7636),
      .ZN(n4832) );
  NR2D0 U5101 ( .A1(n5081), .A2(n4832), .ZN(n4833) );
  OAI22D0 U5102 ( .A1(n4834), .A2(n9274), .B1(n4833), .B2(n9300
     ), .ZN(n4835)
       );
  AOI211D0 U5103 ( .A1(n9307), .A2(n4837), .B(n4836), .C(n4835)
     , .ZN(n4860) );
  CKND2D0 U5104 ( .A1(n9342), .A2(counter[9]), .ZN(n5047) );
  CKND2D0 U5105 ( .A1(n9153), .A2(n9338), .ZN(n4957) );
  NR2D0 U5106 ( .A1(n5047), .A2(n4957), .ZN(n7939) );
  INVD0 U5107 ( .I(n7939), .ZN(n8848) );
  INVD0 U5108 ( .I(n7850), .ZN(n7819) );
  NR2D0 U5109 ( .A1(n7635), .A2(n7819), .ZN(n6561) );
  IND2D0 U5110 ( .A1(n5040), .B1(n6561), .ZN(n5000) );
  NR4D0 U5111 ( .A1(n7890), .A2(n6616), .A3(n7580), .A4(n5000),
      .ZN(n4838) );
  INVD0 U5112 ( .I(n4862), .ZN(n6116) );
  AOI31D0 U5113 ( .A1(n7581), .A2(n4838), .A3(n6116), .B(n9274)
     , .ZN(n4846) );
  CKND2D0 U5114 ( .A1(n7774), .A2(n7656), .ZN(n7675) );
  INVD0 U5115 ( .I(n7675), .ZN(n7687) );
  INVD0 U5116 ( .I(n5023), .ZN(n7997) );
  CKND2D0 U5117 ( .A1(n6527), .A2(n7825), .ZN(n6583) );
  NR4D0 U5118 ( .A1(n7997), .A2(n6612), .A3(n5059), .A4(n6583),
      .ZN(n4839) );
  INVD0 U5119 ( .I(n6696), .ZN(n7721) );
  AOI31D0 U5120 ( .A1(n7687), .A2(n4839), .A3(n7721), .B(n9300)
     , .ZN(n4845) );
  NR3D0 U5121 ( .A1(n6589), .A2(n7596), .A3(n6632), .ZN(n7979)
     );
  INVD0 U5122 ( .I(n6677), .ZN(n5074) );
  NR2D0 U5123 ( .A1(n5074), .A2(n5063), .ZN(n4840) );
  ND3D0 U5124 ( .A1(n4988), .A2(n4840), .A3(n7795), .ZN(n5049)
     );
```

```
  NR2D0 U5125 ( .A1(n7586), .A2(n7890), .ZN(n7628) );
  CKND2D0 U5126 ( .A1(n8046), .A2(n5858), .ZN(n8157) );
  INVD0 U5127 ( .I(n7767), .ZN(n7772) );
  ND4D0 U5128 ( .A1(n7581), .A2(n7628), .A3(n8157), .A4(n7772),
      .ZN(n4841) );
  OAI31D0 U5129 ( .A1(n5049), .A2(n5051), .A3(n4841), .B(n9307)
     , .ZN(n4843) );
  INVD0 U5130 ( .I(n4984), .ZN(n4849) );
  CKND2D0 U5131 ( .A1(n4849), .A2(n4974), .ZN(n4958) );
  NR4D0 U5132 ( .A1(n7714), .A2(n6562), .A3(n6576), .A4(n4958),
      .ZN(n4842) );
  AOI32D0 U5133 ( .A1(n7979), .A2(n4843), .A3(n4842), .B1(n9318
     ), .B2(n4843),
        .ZN(n4844) );
  CKND2D0 U5134 ( .A1(counter[7]), .A2(counter[8]), .ZN(n9304)
     );
  NR2D0 U5135 ( .A1(n4889), .A2(n9304), .ZN(n7933) );
  OAI31D0 U5136 ( .A1(n4846), .A2(n4845), .A3(n4844), .B(n7933)
     , .ZN(n4859) );
  NR3D0 U5137 ( .A1(n6550), .A2(n4994), .A3(n5007), .ZN(n4847)
     );
  INVD0 U5138 ( .I(n8472), .ZN(n9060) );
  INVD0 U5139 ( .I(n5869), .ZN(n5917) );
  CKND2D0 U5140 ( .A1(n9060), .A2(n5917), .ZN(n7849) );
  AOI31D0 U5141 ( .A1(n4847), .A2(n7877), .A3(n7849), .B(n9272)
     , .ZN(n4857) );
  INVD0 U5142 ( .I(n6616), .ZN(n7923) );
  ND4D0 U5143 ( .A1(n5020), .A2(n7795), .A3(n5018), .A4(n7923),
      .ZN(n5082) );
  NR2D0 U5144 ( .A1(n7578), .A2(n7804), .ZN(n5078) );
  INVD0 U5145 ( .I(n5078), .ZN(n6528) );
  NR2D0 U5146 ( .A1(n5082), .A2(n6528), .ZN(n4966) );
  INR4D0 U5147 ( .A1(n4949), .B1(n7609), .B2(n4904), .B3(n4976)
     , .ZN(n4848) );
  AOI31D0 U5148 ( .A1(n4849), .A2(n4966), .A3(n4848), .B(n9274)
     , .ZN(n4856) );
  NR2D0 U5149 ( .A1(n6589), .A2(n6696), .ZN(n7758) );
  INVD0 U5150 ( .I(n7842), .ZN(n7814) );
  NR2D0 U5151 ( .A1(n7578), .A2(n7814), .ZN(n7872) );
  ND4D0 U5152 ( .A1(n7758), .A2(n7872), .A3(n5020), .A4(n7628),
      .ZN(n4850) );
  INR4D0 U5153 ( .A1(n5054), .B1(n5052), .B2(n5080), .B3(n4850)
     , .ZN(n4853) );
  INVD0 U5154 ( .I(n4965), .ZN(n4972) );
  CKND2D0 U5155 ( .A1(n4988), .A2(n7850), .ZN(n7941) );
  NR2D0 U5156 ( .A1(n7596), .A2(n7854), .ZN(n6666) );
  CKND2D0 U5157 ( .A1(n7314), .A2(n5202), .ZN(n6935) );
  NR2D0 U5158 ( .A1(counter[4]), .A2(n6935), .ZN(n7898) );
```

```
NR2D0 U5159 ( .A1(n7635), .A2(n7898), .ZN(n6697) );
IND4D0 U5160 ( .A1(n7941), .B1(n4962), .B2(n6666), .B3(n6697)
   , .ZN(n4851) );
NR4D0 U5161 ( .A1(n4972), .A2(n6562), .A3(n7569), .A4(n4851),
    .ZN(n4852) );
OAI22D0 U5162 ( .A1(n4853), .A2(n9318), .B1(n4852), .B2(n9300
   ), .ZN(n4855)
      );
INVD0 U5163 ( .I(n4957), .ZN(n8485) );
CKND2D0 U5164 ( .A1(n4854), .A2(n8485), .ZN(n6795) );
INVD0 U5165 ( .I(n6795), .ZN(n8740) );
OAI31D0 U5166 ( .A1(n4857), .A2(n4856), .A3(n4855), .B(n8740)
   , .ZN(n4858) );
OAI211D0 U5167 ( .A1(n4860), .A2(n8848), .B(n4859), .C(n4858)
   , .ZN(n5099) );
NR2D0 U5168 ( .A1(n6519), .A2(n7854), .ZN(n7827) );
CKND2D0 U5169 ( .A1(n9143), .A2(n4915), .ZN(n8463) );
NR2D0 U5170 ( .A1(n9329), .A2(n8953), .ZN(n6812) );
INVD0 U5171 ( .I(n6812), .ZN(n7717) );
NR2D0 U5172 ( .A1(n8463), .A2(n7717), .ZN(n6579) );
CKND2D0 U5173 ( .A1(n9143), .A2(counter[11]), .ZN(n5299) );
INVD0 U5174 ( .I(n5299), .ZN(n5932) );
CKND2D0 U5175 ( .A1(counter[4]), .A2(n5932), .ZN(n6656) );
NR2D0 U5176 ( .A1(n7285), .A2(n6656), .ZN(n7722) );
NR2D0 U5177 ( .A1(n9329), .A2(n9193), .ZN(n7709) );
CKND2D0 U5178 ( .A1(n7064), .A2(n7709), .ZN(n7715) );
NR2D0 U5179 ( .A1(n8983), .A2(n5299), .ZN(n5893) );
CKND2D0 U5180 ( .A1(n5893), .A2(n7786), .ZN(n7755) );
CKND2D0 U5181 ( .A1(n7715), .A2(n7755), .ZN(n4874) );
NR2D0 U5182 ( .A1(n9329), .A2(n9143), .ZN(n5794) );
CKND2D0 U5183 ( .A1(n8983), .A2(n5794), .ZN(n7673) );
INVD0 U5184 ( .I(n7673), .ZN(n8045) );
OAI211D0 U5185 ( .A1(n8216), .A2(n8045), .B(n9060), .C(n6205)
   , .ZN(n4880) );
OAI211D0 U5186 ( .A1(n6559), .A2(n7717), .B(n7628), .C(n4880)
   , .ZN(n4861) );
NR4D0 U5187 ( .A1(n6579), .A2(n7722), .A3(n4874), .A4(n4861),
    .ZN(n4894) );
INVD0 U5188 ( .I(n7709), .ZN(n5895) );
NR2D0 U5189 ( .A1(n7294), .A2(n5895), .ZN(n6646) );
AOI211D0 U5190 ( .A1(n9124), .A2(counter[11]), .B(n4862), .C(
   n6646), .ZN(
      n4898) );
INVD0 U5191 ( .I(n5794), .ZN(n8038) );
NR2D0 U5192 ( .A1(n8038), .A2(counter[1]), .ZN(n6361) );
INVD0 U5193 ( .I(n6361), .ZN(n6578) );
NR2D0 U5194 ( .A1(n9029), .A2(n6578), .ZN(n6654) );
```

```
IAO21D0 U5195 ( .A1(n5893), .A2(n8045), .B(n7672), .ZN(n4918)
    );
CKND2D0 U5196 ( .A1(n9345), .A2(n6205), .ZN(n5121) );
CKND2D0 U5197 ( .A1(n7169), .A2(n5121), .ZN(n6192) );
AOI22D0 U5198 ( .A1(counter[4]), .A2(n5121), .B1(n6192), .B2(
   n9348), .ZN(
      n4924) );
NR2D0 U5199 ( .A1(n8983), .A2(n8038), .ZN(n8547) );
CKND2D0 U5200 ( .A1(n8547), .A2(n7169), .ZN(n4863) );
OAI211D0 U5201 ( .A1(n4924), .A2(n4863), .B(n7772), .C(n7642)
   , .ZN(n4872) );
NR4D0 U5202 ( .A1(n7804), .A2(n6654), .A3(n4918), .A4(n4872),
    .ZN(n4864) );
ND4D0 U5203 ( .A1(n7827), .A2(n4894), .A3(n4898), .A4(n4864),
    .ZN(n4888) );
NR2D0 U5204 ( .A1(n8970), .A2(n8278), .ZN(n8011) );
CKND2D0 U5205 ( .A1(n5766), .A2(n6361), .ZN(n8611) );
NR2D0 U5206 ( .A1(n9348), .A2(n8611), .ZN(n7748) );
NR2D0 U5207 ( .A1(n8814), .A2(n6592), .ZN(n8672) );
INVD0 U5208 ( .I(n8672), .ZN(n8730) );
CKND2D0 U5209 ( .A1(counter[11]), .A2(n8046), .ZN(n8532) );
CKND2D0 U5210 ( .A1(n9348), .A2(counter[0]), .ZN(n9092) );
INVD0 U5211 ( .I(n9092), .ZN(n8978) );
CKND2D0 U5212 ( .A1(counter[3]), .A2(n8978), .ZN(n8962) );
INVD0 U5213 ( .I(n8962), .ZN(n9129) );
NR2D0 U5214 ( .A1(n9014), .A2(n9129), .ZN(n9011) );
AOI32D0 U5215 ( .A1(n8730), .A2(n7755), .A3(n8532), .B1(n9011
   ), .B2(n7755),
      .ZN(n4870) );
INR2D0 U5216 ( .A1(n6192), .B1(counter[4]), .ZN(n4868) );
INVD0 U5217 ( .I(n5018), .ZN(n6620) );
NR2D0 U5218 ( .A1(n9329), .A2(n9092), .ZN(n6573) );
INVD0 U5219 ( .I(n6573), .ZN(n7972) );
NR2D0 U5220 ( .A1(n7294), .A2(n7972), .ZN(n7739) );
NR4D0 U5221 ( .A1(n7899), .A2(n5073), .A3(n6646), .A4(n7739),
    .ZN(n4867) );
CKND2D0 U5222 ( .A1(counter[11]), .A2(n9294), .ZN(n8054) );
INVD0 U5223 ( .I(n8054), .ZN(n8052) );
CKND2D0 U5224 ( .A1(n8053), .A2(n8052), .ZN(n4865) );
NR2D0 U5225 ( .A1(n9345), .A2(n4865), .ZN(n7741) );
NR2D0 U5226 ( .A1(n6639), .A2(n8998), .ZN(n5041) );
CKND2D0 U5227 ( .A1(n5932), .A2(n6205), .ZN(n5377) );
CKND2D0 U5228 ( .A1(n6654), .A2(n8983), .ZN(n7728) );
OAI211D0 U5229 ( .A1(n5041), .A2(n5377), .B(n7728), .C(n7678)
   , .ZN(n4930) );
NR4D0 U5230 ( .A1(n7898), .A2(n7741), .A3(n5063), .A4(n4930),
    .ZN(n4866) );
CKND2D0 U5231 ( .A1(n4867), .A2(n4866), .ZN(n4917) );
```

```
AO211D0 U5232 ( .A1(n5893), .A2(n4868), .B(n6620), .C(n4917),
    .Z(n4883) );
INVD0 U5233 ( .I(n9346), .ZN(n7948) );
NR2D0 U5234 ( .A1(n9329), .A2(n7948), .ZN(n6597) );
NR2D0 U5235 ( .A1(counter[0]), .A2(n8983), .ZN(n8183) );
CKND2D0 U5236 ( .A1(n8183), .A2(n9348), .ZN(n8013) );
NR2D0 U5237 ( .A1(n9329), .A2(n6205), .ZN(n6825) );
INVD0 U5238 ( .I(n6825), .ZN(n8768) );
OR2D0 U5239 ( .A1(n8013), .A2(n8768), .Z(n8039) );
NR2D0 U5240 ( .A1(n8039), .A2(counter[3]), .ZN(n7964) );
INVD0 U5241 ( .I(n7964), .ZN(n6581) );
NR2D0 U5242 ( .A1(n7672), .A2(n5299), .ZN(n6665) );
CKND2D0 U5243 ( .A1(n6665), .A2(n8983), .ZN(n4897) );
CKND2D0 U5244 ( .A1(n6581), .A2(n4897), .ZN(n4891) );
NR2D0 U5245 ( .A1(n9092), .A2(counter[3]), .ZN(n8997) );
INVD0 U5246 ( .I(n8997), .ZN(n9311) );
NR2D0 U5247 ( .A1(n9329), .A2(n8957), .ZN(n6878) );
INVD0 U5248 ( .I(n6878), .ZN(n7756) );
OAI22D0 U5249 ( .A1(n8982), .A2(n7717), .B1(n9311), .B2(n7756
   ), .ZN(n7746)
      );
INVD0 U5250 ( .I(n7596), .ZN(n7912) );
NR2D0 U5251 ( .A1(n7673), .A2(n4989), .ZN(n4879) );
INVD0 U5252 ( .I(n4879), .ZN(n6580) );
ND4D0 U5253 ( .A1(n7729), .A2(n6648), .A3(n7912), .A4(n6580),
    .ZN(n4869) );
OR4D0 U5254 ( .A1(n6597), .A2(n4891), .A3(n7746), .A4(n4869),
    .Z(n4925) );
INVD0 U5255 ( .I(n7959), .ZN(n7659) );
OR4D0 U5256 ( .A1(n4870), .A2(n4883), .A3(n4925), .A4(n7659),
    .Z(n4871) );
NR4D0 U5257 ( .A1(n8011), .A2(n7748), .A3(n5040), .A4(n4871),
    .ZN(n4886) );
CKND2D0 U5258 ( .A1(counter[2]), .A2(n6825), .ZN(n8108) );
INVD0 U5259 ( .I(n8108), .ZN(n6647) );
AOI211D0 U5260 ( .A1(n6647), .A2(n4915), .B(n7739), .C(n4879)
   , .ZN(n4932) );
CKND2D0 U5261 ( .A1(n7729), .A2(n7656), .ZN(n7647) );
NR2D0 U5262 ( .A1(n7647), .A2(n4872), .ZN(n4873) );
OAI211D0 U5263 ( .A1(n9029), .A2(n5377), .B(n4932), .C(n4873)
   , .ZN(n4903) );
NR2D0 U5264 ( .A1(n7578), .A2(n4874), .ZN(n4920) );
NR2D0 U5265 ( .A1(n7743), .A2(n6578), .ZN(n7749) );
NR4D0 U5266 ( .A1(n7749), .A2(n7634), .A3(n7722), .A4(n7748),
    .ZN(n4913) );
NR4D0 U5267 ( .A1(n6620), .A2(n6616), .A3(n6597), .A4(n7997),
    .ZN(n4875) );
```

```
ND4D0 U5268 ( .A1(n5056), .A2(n4920), .A3(n4913), .A4(n4875),
    .ZN(n4876) );
OAI31D0 U5269 ( .A1(n4903), .A2(n4877), .A3(n4876), .B(n9307)
   , .ZN(n4885) );
INVD0 U5270 ( .I(n6340), .ZN(n6947) );
CKND2D0 U5271 ( .A1(n6947), .A2(n8547), .ZN(n6488) );
NR2D0 U5272 ( .A1(n6488), .A2(counter[4]), .ZN(n7733) );
INVD0 U5273 ( .I(n7733), .ZN(n4878) );
NR2D0 U5274 ( .A1(n8957), .A2(n8962), .ZN(n8992) );
CKND2D0 U5275 ( .A1(counter[11]), .A2(n8992), .ZN(n4912) );
OAI211D0 U5276 ( .A1(n5299), .A2(n7672), .B(n4878), .C(n4912)
   , .ZN(n7712) );
CKND2D0 U5277 ( .A1(n7842), .A2(n7772), .ZN(n5004) );
NR4D0 U5278 ( .A1(n7911), .A2(n4879), .A3(n5004), .A4(n5052),
    .ZN(n4881) );
CKND2D0 U5279 ( .A1(n7987), .A2(n7709), .ZN(n7958) );
ND4D0 U5280 ( .A1(n5078), .A2(n4881), .A3(n7958), .A4(n4880),
    .ZN(n4882) );
OAI31D0 U5281 ( .A1(n4883), .A2(n7712), .A3(n4882), .B(n9226)
   , .ZN(n4884) );
OAI211D0 U5282 ( .A1(n4886), .A2(n9300), .B(n4885), .C(n4884)
   , .ZN(n4887) );
AOI221D0 U5283 ( .A1(n5049), .A2(n9312), .B1(n4888), .B2(
   n9312), .C(n4887),
      .ZN(n5097) );
INVD0 U5284 ( .I(n5030), .ZN(n9251) );
CKND2D0 U5285 ( .A1(n7437), .A2(n9251), .ZN(n6604) );
NR2D0 U5286 ( .A1(n4889), .A2(counter[7]), .ZN(n4940) );
INVD0 U5287 ( .I(n4940), .ZN(n4890) );
NR2D0 U5288 ( .A1(n4890), .A2(n9338), .ZN(n7694) );
CKND2D0 U5289 ( .A1(n4902), .A2(n7969), .ZN(n4946) );
NR2D0 U5290 ( .A1(n4994), .A2(n4946), .ZN(n4948) );
NR2D0 U5291 ( .A1(n8982), .A2(n7717), .ZN(n4892) );
NR4D0 U5292 ( .A1(n7911), .A2(n4892), .A3(n7748), .A4(n4891),
    .ZN(n4893) );
ND4D0 U5293 ( .A1(n7868), .A2(n4948), .A3(n4894), .A4(n4893),
    .ZN(n4895) );
INR4D0 U5294 ( .A1(n4932), .B1(n6694), .B2(n7609), .B3(n4895)
   , .ZN(n4910) );
CKND2D0 U5295 ( .A1(n6205), .A2(n9060), .ZN(n6595) );
INVD0 U5296 ( .I(n6595), .ZN(n4900) );
NR2D0 U5297 ( .A1(n7659), .A2(n7890), .ZN(n7878) );
INVD0 U5298 ( .I(n5121), .ZN(n8765) );
CKND2D0 U5299 ( .A1(n8765), .A2(n8183), .ZN(n7950) );
NR2D0 U5300 ( .A1(n9348), .A2(n7950), .ZN(n4896) );
CKND2D0 U5301 ( .A1(counter[11]), .A2(n4896), .ZN(n4922) );
ND4D0 U5302 ( .A1(n7878), .A2(n4898), .A3(n4897), .A4(n4922),
    .ZN(n4899) );
```

```
AOI211D0 U5303 ( .A1(counter[11]), .A2(n4900), .B(n7749), .C(
   n4899), .ZN(
      n4901) );
IIND4D0 U5304 ( .A1(n4903), .A2(n5022), .B1(n4902), .B2(n4901
   ), .ZN(n4907)
       );
INVD0 U5305 ( .I(n7758), .ZN(n6560) );
CKND2D0 U5306 ( .A1(n7912), .A2(n7850), .ZN(n7998) );
NR2D0 U5307 ( .A1(n6560), .A2(n7998), .ZN(n7657) );
NR2D0 U5308 ( .A1(n6519), .A2(n6616), .ZN(n6684) );
INVD0 U5309 ( .I(n6684), .ZN(n7574) );
NR2D0 U5310 ( .A1(n7574), .A2(n4904), .ZN(n7794) );
NR3D0 U5311 ( .A1(n5080), .A2(n6812), .A3(n6646), .ZN(n4905)
   );
ND4D0 U5312 ( .A1(n7687), .A2(n7657), .A3(n7794), .A4(n4905),
    .ZN(n4906) );
AOI22D0 U5313 ( .A1(n9307), .A2(n4907), .B1(n9312), .B2(n4906
   ), .ZN(n4909)
       );
CKND2D0 U5314 ( .A1(n6347), .A2(n9294), .ZN(n8314) );
IND3D0 U5315 ( .A1(n4911), .B1(n8314), .B2(n6648), .ZN(n5002)
    );
OAI31D0 U5316 ( .A1(n5049), .A2(n7998), .A3(n5002), .B(n9141)
   , .ZN(n4908) );
OAI211D0 U5317 ( .A1(n4910), .A2(n9318), .B(n4909), .C(n4908)
   , .ZN(n4956) );
NR2D0 U5318 ( .A1(n7855), .A2(n4911), .ZN(n5034) );
INVD0 U5319 ( .I(n6488), .ZN(n5324) );
INVD0 U5320 ( .I(n4912), .ZN(n7684) );
INVD0 U5321 ( .I(n7585), .ZN(n6631) );
CKND2D0 U5322 ( .A1(n5601), .A2(n6631), .ZN(n6685) );
NR4D0 U5323 ( .A1(n5324), .A2(n7684), .A3(n6597), .A4(n6685),
    .ZN(n4934) );
NR2D0 U5324 ( .A1(n7672), .A2(n7673), .ZN(n7983) );
INVD0 U5325 ( .I(n4913), .ZN(n4914) );
AOI211D0 U5326 ( .A1(n8216), .A2(n4915), .B(n7983), .C(n4914)
   , .ZN(n4916) );
IND4D0 U5327 ( .A1(n4917), .B1(n5034), .B2(n4934), .B3(n4916)
   , .ZN(n4929) );
INVD0 U5328 ( .I(n4918), .ZN(n4919) );
OAI211D0 U5329 ( .A1(n8038), .A2(n6595), .B(n7958), .C(n4919)
   , .ZN(n4921) );
NR2D0 U5330 ( .A1(n9193), .A2(counter[3]), .ZN(n9172) );
INVD0 U5331 ( .I(n9172), .ZN(n9110) );
NR2D0 U5332 ( .A1(n8957), .A2(n9110), .ZN(n9299) );
INVD0 U5333 ( .I(n6656), .ZN(n7986) );
AOI22D0 U5334 ( .A1(counter[11]), .A2(n9299), .B1(n7987), .B2
   (n7986), .ZN(
```

```
      n7731) );
IND4D0 U5335 ( .A1(n4921), .B1(n4920), .B2(n7731), .B3(n6520)
   , .ZN(n4939) );
INVD0 U5336 ( .I(n4922), .ZN(n4923) );
AOI211D0 U5337 ( .A1(n4924), .A2(n8045), .B(n5004), .C(n4923)
   , .ZN(n4927) );
CKND2D0 U5338 ( .A1(n7758), .A2(n7795), .ZN(n7901) );
NR4D0 U5339 ( .A1(n6694), .A2(n5050), .A3(n7901), .A4(n4925),
    .ZN(n4926) );
IND4D0 U5340 ( .A1(n4939), .B1(n6561), .B2(n4927), .B3(n4926)
   , .ZN(n4928) );
AOI22D0 U5341 ( .A1(n9312), .A2(n4929), .B1(n9141), .B2(n4928
   ), .ZN(n4943)
      );
AOI211D0 U5342 ( .A1(n8978), .A2(n7413), .B(n7733), .C(n4930)
   , .ZN(n4931) );
ND4D0 U5343 ( .A1(n7600), .A2(n4962), .A3(n4932), .A4(n4931),
    .ZN(n4933) );
OAI31D0 U5344 ( .A1(n4939), .A2(n5082), .A3(n4933), .B(n9307)
   , .ZN(n4942) );
INVD0 U5345 ( .I(n4934), .ZN(n4938) );
NR2D0 U5346 ( .A1(n7804), .A2(n7821), .ZN(n5076) );
NR2D0 U5347 ( .A1(n6205), .A2(counter[0]), .ZN(n6191) );
INVD0 U5348 ( .I(n6191), .ZN(n5327) );
NR2D0 U5349 ( .A1(n9329), .A2(n5327), .ZN(n8110) );
INVD0 U5350 ( .I(n8110), .ZN(n6368) );
OR2D0 U5351 ( .A1(n7282), .A2(n6368), .Z(n6814) );
NR2D0 U5352 ( .A1(counter[4]), .A2(n6814), .ZN(n7683) );
AOI21D0 U5353 ( .A1(n9348), .A2(n8547), .B(n7683), .ZN(n4936)
    );
NR4D0 U5354 ( .A1(n7586), .A2(n7640), .A3(n6579), .A4(n6646),
    .ZN(n4935) );
ND4D0 U5355 ( .A1(n7758), .A2(n5076), .A3(n4936), .A4(n4935),
    .ZN(n4937) );
OAI31D0 U5356 ( .A1(n4939), .A2(n4938), .A3(n4937), .B(n9226)
   , .ZN(n4941) );
CKND2D0 U5357 ( .A1(n9338), .A2(n4940), .ZN(n5523) );
AOI31D0 U5358 ( .A1(n4943), .A2(n4942), .A3(n4941), .B(n5523)
   , .ZN(n4955) );
NR2D0 U5359 ( .A1(n7819), .A2(n7767), .ZN(n7990) );
CKND2D0 U5360 ( .A1(n7990), .A2(n4949), .ZN(n5035) );
NR2D0 U5361 ( .A1(n7609), .A2(n7898), .ZN(n7989) );
NR2D0 U5362 ( .A1(n6696), .A2(n7659), .ZN(n5057) );
IIND4D0 U5363 ( .A1(n5035), .A2(n5082), .B1(n7989), .B2(n5057
   ), .ZN(n4945)
      );
IIND4D0 U5364 ( .A1(n5051), .A2(n4946), .B1(n7796), .B2(n5034
   ), .ZN(n4944)
```

```
        );
AOI22D0 U5365 ( .A1(n9226), .A2(n4945), .B1(n9307), .B2(n4944
   ), .ZN(n4953)
        );
CKND2D0 U5366 ( .A1(n6615), .A2(n4949), .ZN(n5069) );
OR4D0 U5367 ( .A1(n5069), .A2(n4946), .A3(n7814), .A4(n6662),
    .Z(n4947) );
OAI31D0 U5368 ( .A1(n6550), .A2(n7998), .A3(n4947), .B(n9312)
   , .ZN(n4952) );
ND4D0 U5369 ( .A1(n4949), .A2(n5031), .A3(n4948), .A4(n7678),
    .ZN(n4950) );
OAI21D0 U5370 ( .A1(n7998), .A2(n4950), .B(n9141), .ZN(n4951)
    );
CKND2D0 U5371 ( .A1(counter[8]), .A2(n9153), .ZN(n5046) );
NR2D0 U5372 ( .A1(n4983), .A2(n5046), .ZN(n8895) );
INVD0 U5373 ( .I(n8895), .ZN(n7593) );
AOI31D0 U5374 ( .A1(n4953), .A2(n4952), .A3(n4951), .B(n7593)
   , .ZN(n4954) );
AOI211D0 U5375 ( .A1(n7694), .A2(n4956), .B(n4955), .C(n4954)
   , .ZN(n5096) );
NR2D0 U5376 ( .A1(n9075), .A2(n9342), .ZN(n7402) );
INVD0 U5377 ( .I(n7402), .ZN(n9335) );
NR2D0 U5378 ( .A1(n9335), .A2(n4957), .ZN(n8105) );
NR4D0 U5379 ( .A1(n4976), .A2(n4958), .A3(n5022), .A4(n5035),
    .ZN(n4970) );
NR2D0 U5380 ( .A1(n6340), .A2(n5869), .ZN(n6678) );
INVD0 U5381 ( .I(n6678), .ZN(n6628) );
ND4D0 U5382 ( .A1(n4960), .A2(n4959), .A3(n6628), .A4(n6648),
    .ZN(n4964) );
CKND2D0 U5383 ( .A1(n6529), .A2(n7774), .ZN(n6610) );
NR2D0 U5384 ( .A1(n7899), .A2(n6610), .ZN(n6683) );
INVD0 U5385 ( .I(n6562), .ZN(n7671) );
CKND2D0 U5386 ( .A1(n7671), .A2(n6523), .ZN(n7724) );
NR4D0 U5387 ( .A1(n7724), .A2(n7901), .A3(n5052), .A4(n5051),
    .ZN(n4961) );
ND4D0 U5388 ( .A1(n5020), .A2(n4962), .A3(n6683), .A4(n4961),
    .ZN(n4963) );
AOI22D0 U5389 ( .A1(n9312), .A2(n4964), .B1(n9141), .B2(n4963
   ), .ZN(n4969)
        );
NR2D0 U5390 ( .A1(n7898), .A2(n5004), .ZN(n7669) );
ND4D0 U5391 ( .A1(n7669), .A2(n7781), .A3(n4966), .A4(n4965),
    .ZN(n4967) );
OAI21D0 U5392 ( .A1(n6583), .A2(n4967), .B(n9226), .ZN(n4968)
    );
OAI211D0 U5393 ( .A1(n4970), .A2(n9272), .B(n4969), .C(n4968)
   , .ZN(n5094) );
INVD0 U5394 ( .I(n7898), .ZN(n6609) );
```

```
  CKND2D0 U5395 ( .A1(n6527), .A2(n6609), .ZN(n7770) );
  NR2D0 U5396 ( .A1(n4971), .A2(n7770), .ZN(n7699) );
  NR2D0 U5397 ( .A1(n4972), .A2(n7659), .ZN(n4986) );
  IND4D0 U5398 ( .A1(n5002), .B1(n7699), .B2(n4986), .B3(n7850)
     , .ZN(n4982) );
  CKND2D0 U5399 ( .A1(n7678), .A2(n7825), .ZN(n6617) );
  CKND2D0 U5400 ( .A1(n6116), .A2(n6529), .ZN(n6625) );
  INVD0 U5401 ( .I(n7609), .ZN(n7844) );
  CKND2D0 U5402 ( .A1(n7844), .A2(n7842), .ZN(n6687) );
  NR4D0 U5403 ( .A1(n7647), .A2(n6617), .A3(n6625), .A4(n6687),
      .ZN(n4973) );
  AOI31D0 U5404 ( .A1(n4974), .A2(n7600), .A3(n4973), .B(n9272)
     , .ZN(n4981) );
  NR4D0 U5405 ( .A1(n7814), .A2(n4976), .A3(n4975), .A4(n5002),
      .ZN(n4979) );
  INVD0 U5406 ( .I(n6615), .ZN(n6556) );
  CKND2D0 U5407 ( .A1(n7868), .A2(n4977), .ZN(n5016) );
  NR4D0 U5408 ( .A1(n7767), .A2(n6556), .A3(n5082), .A4(n5016),
      .ZN(n4978) );
  OAI22D0 U5409 ( .A1(n4979), .A2(n9274), .B1(n4978), .B2(n9318
     ), .ZN(n4980)
        );
  AOI211D0 U5410 ( .A1(n9141), .A2(n4982), .B(n4981), .C(n4980)
     , .ZN(n5015) );
  NR2D0 U5411 ( .A1(n9304), .A2(n4983), .ZN(n8715) );
  INVD0 U5412 ( .I(n8715), .ZN(n6393) );
  NR2D0 U5413 ( .A1(n7799), .A2(n6616), .ZN(n7616) );
  CKND2D0 U5414 ( .A1(n7616), .A2(n7774), .ZN(n7818) );
  NR2D0 U5415 ( .A1(n4984), .A2(n7899), .ZN(n5077) );
  INVD0 U5416 ( .I(n5077), .ZN(n5036) );
  NR4D0 U5417 ( .A1(n7818), .A2(n5063), .A3(n6583), .A4(n5036),
      .ZN(n4985) );
  AOI31D0 U5418 ( .A1(n4986), .A2(n4985), .A3(n6609), .B(n9272)
     , .ZN(n4999) );
  NR4D0 U5419 ( .A1(n7585), .A2(n7569), .A3(n6617), .A4(n4994),
      .ZN(n4987) );
  AOI31D0 U5420 ( .A1(n5031), .A2(n7758), .A3(n4987), .B(n9274)
     , .ZN(n4998) );
  CKND2D0 U5421 ( .A1(n7844), .A2(n5018), .ZN(n4991) );
  OAI211D0 U5422 ( .A1(n6512), .A2(n4989), .B(n4988), .C(n5077)
     , .ZN(n4990) );
  NR4D0 U5423 ( .A1(n6560), .A2(n5000), .A3(n4991), .A4(n4990),
      .ZN(n4996) );
  AO211D0 U5424 ( .A1(n4992), .A2(n8998), .B(n5073), .C(n6662),
      .Z(n4993) );
  NR4D0 U5425 ( .A1(n4994), .A2(n6583), .A3(n5036), .A4(n4993),
      .ZN(n4995) );
```

```
OAI22D0 U5426 ( .A1(n4996), .A2(n9300), .B1(n4995), .B2(n9318
   ), .ZN(n4997)
      );
NR2D0 U5427 ( .A1(n9335), .A2(n9304), .ZN(n8656) );
OAI31D0 U5428 ( .A1(n4999), .A2(n4998), .A3(n4997), .B(n8656)
   , .ZN(n5014) );
INVD0 U5429 ( .I(n7855), .ZN(n8394) );
CKND2D0 U5430 ( .A1(n8394), .A2(n5018), .ZN(n7801) );
INR4D0 U5431 ( .A1(n5031), .B1(n8011), .B2(n7801), .B3(n5000)
   , .ZN(n5001) );
AOI21D0 U5432 ( .A1(n7669), .A2(n5001), .B(n9318), .ZN(n5012)
    );
NR2D0 U5433 ( .A1(n7282), .A2(n7744), .ZN(n7784) );
NR4D0 U5434 ( .A1(n7659), .A2(n7784), .A3(n6550), .A4(n5002),
    .ZN(n5003) );
AOI31D0 U5435 ( .A1(n7669), .A2(n7600), .A3(n5003), .B(n9300)
   , .ZN(n5011) );
NR2D0 U5436 ( .A1(n5004), .A2(n7821), .ZN(n5071) );
ND4D0 U5437 ( .A1(n5071), .A2(n5677), .A3(n6631), .A4(n7849),
    .ZN(n5005) );
OAI31D0 U5438 ( .A1(n5033), .A2(n5081), .A3(n5005), .B(n9307)
   , .ZN(n5009) );
NR2D0 U5439 ( .A1(n6694), .A2(n7766), .ZN(n7606) );
CKND2D0 U5440 ( .A1(n5006), .A2(n7606), .ZN(n7621) );
INVD0 U5441 ( .I(n7621), .ZN(n7924) );
CKND2D0 U5442 ( .A1(n7924), .A2(n7774), .ZN(n7605) );
NR4D0 U5443 ( .A1(n7605), .A2(n6617), .A3(n5080), .A4(n5007),
    .ZN(n5008) );
AOI32D0 U5444 ( .A1(n5076), .A2(n5009), .A3(n5008), .B1(n9274
   ), .B2(n5009),
      .ZN(n5010) );
NR2D0 U5445 ( .A1(n9335), .A2(n5046), .ZN(n8910) );
OAI31D0 U5446 ( .A1(n5012), .A2(n5011), .A3(n5010), .B(n8910)
   , .ZN(n5013) );
OAI211D0 U5447 ( .A1(n5015), .A2(n6393), .B(n5014), .C(n5013)
   , .ZN(n5093) );
NR2D0 U5448 ( .A1(n6617), .A2(n5081), .ZN(n5072) );
IND4D0 U5449 ( .A1(n5016), .B1(n7582), .B2(n5072), .B3(n5018)
   , .ZN(n5029) );
CKND2D0 U5450 ( .A1(n6539), .A2(n7842), .ZN(n5017) );
CKND2D0 U5451 ( .A1(n5023), .A2(n6561), .ZN(n5048) );
NR4D0 U5452 ( .A1(n7579), .A2(n5017), .A3(n5081), .A4(n5048),
    .ZN(n5019) );
AOI31D0 U5453 ( .A1(n5019), .A2(n7844), .A3(n5018), .B(n9272)
   , .ZN(n5028) );
CKND2D0 U5454 ( .A1(n7780), .A2(n7656), .ZN(n7662) );
NR2D0 U5455 ( .A1(n6620), .A2(n7586), .ZN(n7876) );
INVD0 U5456 ( .I(n7876), .ZN(n7853) );
```

```
NR2D0 U5457 ( .A1(n6616), .A2(n7766), .ZN(n7917) );
NR2D0 U5458 ( .A1(n6512), .A2(counter[1]), .ZN(n7815) );
CKND2D0 U5459 ( .A1(n6639), .A2(n7815), .ZN(n7867) );
ND4D0 U5460 ( .A1(n7917), .A2(n7669), .A3(n5020), .A4(n7867),
    .ZN(n5021) );
NR4D0 U5461 ( .A1(n7662), .A2(n7853), .A3(n5022), .A4(n5021),
    .ZN(n5026) );
NR2D0 U5462 ( .A1(n6340), .A2(n7744), .ZN(n7813) );
ND4D0 U5463 ( .A1(n5023), .A2(n7989), .A3(n7781), .A4(n6631),
    .ZN(n5024) );
NR4D0 U5464 ( .A1(n7634), .A2(n7813), .A3(n5059), .A4(n5024),
    .ZN(n5025) );
OAI22D0 U5465 ( .A1(n5026), .A2(n9274), .B1(n5025), .B2(n9318
   ), .ZN(n5027)
      );
AOI211D0 U5466 ( .A1(n9141), .A2(n5029), .B(n5028), .C(n5027)
   , .ZN(n5091) );
NR2D0 U5467 ( .A1(n5030), .A2(n5047), .ZN(n8408) );
INVD0 U5468 ( .I(n8408), .ZN(n8945) );
NR2D0 U5469 ( .A1(n5047), .A2(n9304), .ZN(n8323) );
CKND2D0 U5470 ( .A1(n7774), .A2(n6609), .ZN(n6555) );
CKND2D0 U5471 ( .A1(n6529), .A2(n7772), .ZN(n7812) );
INVD0 U5472 ( .I(n7812), .ZN(n6538) );
ND4D0 U5473 ( .A1(n7917), .A2(n6538), .A3(n5031), .A4(n7642),
    .ZN(n5032) );
NR4D0 U5474 ( .A1(n6555), .A2(n7853), .A3(n5033), .A4(n5032),
    .ZN(n5045) );
NR2D0 U5475 ( .A1(n8957), .A2(n7744), .ZN(n6686) );
NR2D0 U5476 ( .A1(n7819), .A2(n6686), .ZN(n6526) );
CKND2D0 U5477 ( .A1(counter[3]), .A2(n8156), .ZN(n7235) );
INVD0 U5478 ( .I(n7235), .ZN(n7296) );
CKND2D0 U5479 ( .A1(n7296), .A2(n5202), .ZN(n5617) );
ND4D0 U5480 ( .A1(n6526), .A2(n5034), .A3(n7849), .A4(n5617),
    .ZN(n5039) );
NR4D0 U5481 ( .A1(n6620), .A2(n5063), .A3(n5036), .A4(n5035),
    .ZN(n5037) );
AOI31D0 U5482 ( .A1(n5037), .A2(n7912), .A3(n7915), .B(n9272)
   , .ZN(n5038) );
AOI221D0 U5483 ( .A1(n5040), .A2(n9141), .B1(n5039), .B2(
   n9141), .C(n5038),
      .ZN(n5044) );
AO211D0 U5484 ( .A1(n7815), .A2(n5041), .B(n6528), .C(n7586),
    .Z(n5042) );
OAI31D0 U5485 ( .A1(n6694), .A2(n7821), .A3(n5042), .B(n9226)
   , .ZN(n5043) );
OAI211D0 U5486 ( .A1(n5045), .A2(n9274), .B(n5044), .C(n5043)
   , .ZN(n5068) );
NR2D0 U5487 ( .A1(n5047), .A2(n5046), .ZN(n8803) );
```

```
NR4D0 U5488 ( .A1(n5050), .A2(n7821), .A3(n5049), .A4(n5048),
    .ZN(n5066) );
NR2D0 U5489 ( .A1(n9029), .A2(n8725), .ZN(n5053) );
NR4D0 U5490 ( .A1(n7766), .A2(n5053), .A3(n5052), .A4(n5051),
    .ZN(n5055) );
ND4D0 U5491 ( .A1(n5056), .A2(n5055), .A3(n7795), .A4(n5054),
    .ZN(n5061) );
IND4D0 U5492 ( .A1(n5059), .B1(n5058), .B2(n5057), .B3(n8028)
   , .ZN(n5060) );
AOI22D0 U5493 ( .A1(n9226), .A2(n5061), .B1(n9312), .B2(n5060
   ), .ZN(n5065)
      );
ND4D0 U5494 ( .A1(n5076), .A2(n5072), .A3(n6523), .A4(n6609),
    .ZN(n5062) );
OAI21D0 U5495 ( .A1(n5063), .A2(n5062), .B(n9141), .ZN(n5064)
    );
OAI211D0 U5496 ( .A1(n5066), .A2(n9272), .B(n5065), .C(n5064)
   , .ZN(n5067) );
AOI22D0 U5497 ( .A1(n8323), .A2(n5068), .B1(n8803), .B2(n5067
   ), .ZN(n5090)
      );
INR4D0 U5498 ( .A1(n6526), .B1(n6589), .B2(n6616), .B3(n5069)
   , .ZN(n5070) );
AOI31D0 U5499 ( .A1(n5072), .A2(n5071), .A3(n5070), .B(n9300)
   , .ZN(n5088) );
NR2D0 U5500 ( .A1(n5073), .A2(n7854), .ZN(n7680) );
CKND2D0 U5501 ( .A1(n7680), .A2(n7825), .ZN(n7633) );
INVD0 U5502 ( .I(n6537), .ZN(n6619) );
NR4D0 U5503 ( .A1(n6556), .A2(n5074), .A3(n7633), .A4(n6619),
    .ZN(n5075) );
AOI31D0 U5504 ( .A1(n5076), .A2(n7875), .A3(n5075), .B(n9274)
   , .ZN(n5087) );
ND4D0 U5505 ( .A1(n6615), .A2(n5078), .A3(n5077), .A4(n6609),
    .ZN(n5079) );
OAI31D0 U5506 ( .A1(n6616), .A2(n5080), .A3(n5079), .B(n9307)
   , .ZN(n5084) );
INVD0 U5507 ( .I(n7899), .ZN(n7826) );
CKND2D0 U5508 ( .A1(n7826), .A2(n6523), .ZN(n7652) );
NR4D0 U5509 ( .A1(n6632), .A2(n7652), .A3(n5082), .A4(n5081),
    .ZN(n5083) );
AOI32D0 U5510 ( .A1(n5085), .A2(n5084), .A3(n5083), .B1(n9318
   ), .B2(n5084),
     .ZN(n5086) );
CKND2D0 U5511 ( .A1(n7402), .A2(n9251), .ZN(n6986) );
INVD0 U5512 ( .I(n6986), .ZN(n8939) );
OAI31D0 U5513 ( .A1(n5088), .A2(n5087), .A3(n5086), .B(n8939)
   , .ZN(n5089) );
```

```
OAI211D0 U5514 ( .A1(n5091), .A2(n8945), .B(n5090), .C(n5089)
   , .ZN(n5092) );
AOI211D0 U5515 ( .A1(n8105), .A2(n5094), .B(n5093), .C(n5092)
   , .ZN(n5095) );
OAI211D0 U5516 ( .A1(n5097), .A2(n6604), .B(n5096), .C(n5095)
   , .ZN(n5098) );
AOI211D0 U5517 ( .A1(n8037), .A2(n5100), .B(n5099), .C(n5098)
   , .ZN(n5357) );
NR2D0 U5518 ( .A1(n9352), .A2(n5677), .ZN(n8876) );
INVD0 U5519 ( .I(n7785), .ZN(n6049) );
NR2D0 U5520 ( .A1(n6205), .A2(n6049), .ZN(n5913) );
CKND2D0 U5521 ( .A1(n9345), .A2(counter[6]), .ZN(n8416) );
NR2D0 U5522 ( .A1(n8983), .A2(n8416), .ZN(n5767) );
CKND2D0 U5523 ( .A1(n5913), .A2(n5767), .ZN(n8920) );
INVD0 U5524 ( .I(n8920), .ZN(n8782) );
NR2D0 U5525 ( .A1(n8876), .A2(n8782), .ZN(n5218) );
INVD0 U5526 ( .I(n5677), .ZN(n7587) );
CKND2D0 U5527 ( .A1(n9352), .A2(n7587), .ZN(n6847) );
INVD0 U5528 ( .I(n6847), .ZN(n8733) );
CKND2D0 U5529 ( .A1(n9352), .A2(n8053), .ZN(n7322) );
INVD0 U5530 ( .I(n7322), .ZN(n7436) );
CKND2D0 U5531 ( .A1(n7436), .A2(n7785), .ZN(n6726) );
INVD0 U5532 ( .I(n6726), .ZN(n6113) );
NR2D0 U5533 ( .A1(n8733), .A2(n6113), .ZN(n5216) );
CKND2D0 U5534 ( .A1(n5218), .A2(n5216), .ZN(n5106) );
CKND1 U5535 ( .I(n8674), .ZN(n8899) );
NR2D0 U5536 ( .A1(n9143), .A2(n8416), .ZN(n7310) );
CKND2D0 U5537 ( .A1(n7310), .A2(n5452), .ZN(n6724) );
INVD0 U5538 ( .I(n6724), .ZN(n8753) );
CKND2D0 U5539 ( .A1(counter[6]), .A2(counter[3]), .ZN(n8441)
   );
INVD0 U5540 ( .I(n8441), .ZN(n8309) );
CKND2D0 U5541 ( .A1(n9143), .A2(n8309), .ZN(n5107) );
NR2D0 U5542 ( .A1(n8924), .A2(n5107), .ZN(n8860) );
NR2D0 U5543 ( .A1(n8753), .A2(n8860), .ZN(n5126) );
CKND2D0 U5544 ( .A1(n9352), .A2(n8951), .ZN(n7328) );
INVD0 U5545 ( .I(n7328), .ZN(n7369) );
CKND2D0 U5546 ( .A1(n7369), .A2(n5202), .ZN(n6013) );
NR2D0 U5547 ( .A1(counter[3]), .A2(n6013), .ZN(n5668) );
NR2D0 U5548 ( .A1(n9345), .A2(counter[6]), .ZN(n8316) );
CKND2D0 U5549 ( .A1(counter[2]), .A2(n8316), .ZN(n5801) );
INVD0 U5550 ( .I(n5913), .ZN(n8613) );
NR2D0 U5551 ( .A1(n5801), .A2(n8613), .ZN(n8735) );
INVD0 U5552 ( .I(n8316), .ZN(n8612) );
NR2D0 U5553 ( .A1(counter[1]), .A2(n8612), .ZN(n5101) );
CKND2D0 U5554 ( .A1(n5917), .A2(n5101), .ZN(n8601) );
INVD0 U5555 ( .I(n8601), .ZN(n8771) );
```

```
  NR3D0 U5556 ( .A1(n5668), .A2(n8735), .A3(n8771), .ZN(n5179)
     );
  CKND2D0 U5557 ( .A1(counter[6]), .A2(n7870), .ZN(n8514) );
  CKND2D0 U5558 ( .A1(n9352), .A2(n5202), .ZN(n8811) );
  INVD0 U5559 ( .I(n8811), .ZN(n5415) );
  CKND2D0 U5560 ( .A1(n7064), .A2(n5415), .ZN(n6926) );
  CKND2D0 U5561 ( .A1(n8514), .A2(n6926), .ZN(n5589) );
  INVD0 U5562 ( .I(n5589), .ZN(n5598) );
  CKND2D0 U5563 ( .A1(counter[6]), .A2(n7785), .ZN(n5856) );
  INVD0 U5564 ( .I(n5856), .ZN(n5646) );
  CKND2D0 U5565 ( .A1(n6248), .A2(n5646), .ZN(n8791) );
  ND4D0 U5566 ( .A1(n5126), .A2(n5179), .A3(n5598), .A4(n8791),
      .ZN(n5105) );
  CKND2D0 U5567 ( .A1(n7414), .A2(n5415), .ZN(n8521) );
  INVD0 U5568 ( .I(n8521), .ZN(n8509) );
  INVD0 U5569 ( .I(n5801), .ZN(n8761) );
  CKND2D0 U5570 ( .A1(n5102), .A2(n8761), .ZN(n8702) );
  INVD0 U5571 ( .I(n8702), .ZN(n8596) );
  NR2D0 U5572 ( .A1(n9352), .A2(n5601), .ZN(n8748) );
  INVD0 U5573 ( .I(n8748), .ZN(n8724) );
  CKND2D0 U5574 ( .A1(n8724), .A2(n7867), .ZN(n5282) );
  NR3D0 U5575 ( .A1(n8509), .A2(n8596), .A3(n5282), .ZN(n5215)
     );
  CKND2D0 U5576 ( .A1(counter[0]), .A2(n8309), .ZN(n5157) );
  NR2D0 U5577 ( .A1(n8278), .A2(n5157), .ZN(n8551) );
  NR2D0 U5578 ( .A1(n7235), .A2(n8811), .ZN(n8648) );
  CKND2D0 U5579 ( .A1(n7436), .A2(n5202), .ZN(n5939) );
  INVD0 U5580 ( .I(n5939), .ZN(n5536) );
  NR2D0 U5581 ( .A1(counter[0]), .A2(n8416), .ZN(n6813) );
  CKND2D0 U5582 ( .A1(n5452), .A2(n6813), .ZN(n6894) );
  INVD0 U5583 ( .I(n6894), .ZN(n8620) );
  CKND2D0 U5584 ( .A1(n8316), .A2(n8983), .ZN(n5784) );
  NR2D0 U5585 ( .A1(n8613), .A2(n5784), .ZN(n8831) );
  NR2D0 U5586 ( .A1(n8620), .A2(n8831), .ZN(n8794) );
  INVD0 U5587 ( .I(n8860), .ZN(n8512) );
  CKND2D0 U5588 ( .A1(n9352), .A2(n9345), .ZN(n8426) );
  NR2D0 U5589 ( .A1(n9143), .A2(n8426), .ZN(n5379) );
  CKND2D0 U5590 ( .A1(n5385), .A2(n5379), .ZN(n8556) );
  ND3D0 U5591 ( .A1(n8794), .A2(n8512), .A3(n8556), .ZN(n5193)
     );
  NR4D0 U5592 ( .A1(n8551), .A2(n8648), .A3(n5536), .A4(n5193),
      .ZN(n5103) );
  AOI31D0 U5593 ( .A1(n5218), .A2(n5215), .A3(n5103), .B(n8935)
     , .ZN(n5104) );
  AOI221D0 U5594 ( .A1(n5106), .A2(n8899), .B1(n5105), .B2(
     n8899), .C(n5104),
        .ZN(n5111) );
  CKND2D0 U5595 ( .A1(n9352), .A2(n6678), .ZN(n5586) );
```

```
  INVD0 U5596 ( .I(n5586), .ZN(n8808) );
  NR2D0 U5597 ( .A1(n8808), .A2(n8620), .ZN(n5626) );
  INVD0 U5598 ( .I(n7310), .ZN(n8533) );
  NR2D0 U5599 ( .A1(n8533), .A2(n8725), .ZN(n6732) );
  INVD0 U5600 ( .I(n6732), .ZN(n6927) );
  INVD0 U5601 ( .I(n5107), .ZN(n6942) );
  CKND2D0 U5602 ( .A1(n5452), .A2(n6942), .ZN(n8836) );
  CKND2D0 U5603 ( .A1(n6927), .A2(n8836), .ZN(n5205) );
  INVD0 U5604 ( .I(n5205), .ZN(n8855) );
  ND3D0 U5605 ( .A1(n5626), .A2(n8855), .A3(n8601), .ZN(n5283)
     );
  INVD0 U5606 ( .I(n8831), .ZN(n8789) );
  NR2D0 U5607 ( .A1(n9352), .A2(n6512), .ZN(n5837) );
  CKND2D0 U5608 ( .A1(n8156), .A2(n5837), .ZN(n6776) );
  CKND2D0 U5609 ( .A1(n8789), .A2(n6776), .ZN(n5545) );
  NR2D0 U5610 ( .A1(n5545), .A2(n8648), .ZN(n5275) );
  INVD0 U5611 ( .I(n5275), .ZN(n5135) );
  CKND2D0 U5612 ( .A1(n7414), .A2(n5646), .ZN(n8721) );
  INVD0 U5613 ( .I(n8876), .ZN(n6717) );
  CKND2D0 U5614 ( .A1(n8721), .A2(n6717), .ZN(n8776) );
  NR2D0 U5615 ( .A1(n8924), .A2(n5157), .ZN(n5724) );
  NR2D0 U5616 ( .A1(n8776), .A2(n5724), .ZN(n5274) );
  CKND2D0 U5617 ( .A1(counter[6]), .A2(n8672), .ZN(n5585) );
  CKND2D0 U5618 ( .A1(n9143), .A2(n8316), .ZN(n5292) );
  NR2D0 U5619 ( .A1(n8278), .A2(n5292), .ZN(n5360) );
  INVD0 U5620 ( .I(n5360), .ZN(n8883) );
  CKND2D0 U5621 ( .A1(n5585), .A2(n8883), .ZN(n8530) );
  INR2D0 U5622 ( .A1(n5274), .B1(n8530), .ZN(n5289) );
  NR2D0 U5623 ( .A1(n8924), .A2(n8533), .ZN(n8503) );
  INVD0 U5624 ( .I(n8503), .ZN(n8813) );
  CKND2D0 U5625 ( .A1(n8512), .A2(n8813), .ZN(n5509) );
  INVD0 U5626 ( .I(n5509), .ZN(n6889) );
  INVD0 U5627 ( .I(n8735), .ZN(n8691) );
  CKND2D0 U5628 ( .A1(n5452), .A2(n5379), .ZN(n5576) );
  ND4D0 U5629 ( .A1(n5289), .A2(n6889), .A3(n8691), .A4(n5576),
      .ZN(n5108) );
  CKND1 U5630 ( .I(n8923), .ZN(n8697) );
  OAI31D0 U5631 ( .A1(n5283), .A2(n5135), .A3(n5108), .B(n8697)
     , .ZN(n5110) );
  INVD0 U5632 ( .I(n5585), .ZN(n5765) );
  AOI21D0 U5633 ( .A1(n8216), .A2(n6192), .B(n5765), .ZN(n5129)
      );
  ND3D0 U5634 ( .A1(n5179), .A2(n5129), .A3(n8883), .ZN(n5142)
     );
  NR2D0 U5635 ( .A1(n8860), .A2(n6732), .ZN(n8723) );
  INVD0 U5636 ( .I(n8723), .ZN(n8650) );
  NR2D0 U5637 ( .A1(n8551), .A2(n4819), .ZN(n5476) );
  INVD0 U5638 ( .I(n5476), .ZN(n6915) );
```

```
  NR2D0 U5639 ( .A1(n9348), .A2(n7514), .ZN(n8609) );
  CKND1 U5640 ( .I(n8609), .ZN(n8919) );
  INVD0 U5641 ( .I(n8919), .ZN(n8822) );
  OAI31D0 U5642 ( .A1(n5142), .A2(n8650), .A3(n6915), .B(n8822)
     , .ZN(n5109) );
  AOI31D0 U5643 ( .A1(n5111), .A2(n5110), .A3(n5109), .B(n6393)
     , .ZN(n5355) );
  NR2D0 U5644 ( .A1(n8426), .A2(n8028), .ZN(n8755) );
  NR2D0 U5645 ( .A1(n8755), .A2(n8503), .ZN(n5145) );
  CKND2D0 U5646 ( .A1(n8721), .A2(n8556), .ZN(n8531) );
  INVD0 U5647 ( .I(n5126), .ZN(n8572) );
  NR2D0 U5648 ( .A1(n8531), .A2(n8572), .ZN(n5234) );
  CKND2D0 U5649 ( .A1(n5145), .A2(n5234), .ZN(n5178) );
  CKND2D0 U5650 ( .A1(n5275), .A2(n5216), .ZN(n5320) );
  IND4D0 U5651 ( .A1(n5320), .B1(n5586), .B2(n8521), .B3(n6557)
     , .ZN(n5114) );
  INVD0 U5652 ( .I(n7815), .ZN(n6558) );
  INVD0 U5653 ( .I(n8416), .ZN(n8396) );
  CKND2D0 U5654 ( .A1(n8396), .A2(n8983), .ZN(n5770) );
  NR2D0 U5655 ( .A1(n6558), .A2(n5770), .ZN(n5650) );
  INVD0 U5656 ( .I(n5650), .ZN(n6857) );
  CKND2D0 U5657 ( .A1(n6857), .A2(n8691), .ZN(n8631) );
  OAI211D0 U5658 ( .A1(n6592), .A2(n8811), .B(n5145), .C(n5585)
     , .ZN(n5197) );
  NR3D0 U5659 ( .A1(n8733), .A2(n8631), .A3(n5197), .ZN(n5122)
     );
  CKND2D0 U5660 ( .A1(n8556), .A2(n8920), .ZN(n6904) );
  NR2D0 U5661 ( .A1(n6904), .A2(n6732), .ZN(n5259) );
  INVD0 U5662 ( .I(n5668), .ZN(n6737) );
  AN3D0 U5663 ( .A1(n5259), .A2(n6737), .A3(n8789), .Z(n5112) )
     ;
  AOI31D0 U5664 ( .A1(n5122), .A2(n5274), .A3(n5112), .B(n8919)
     , .ZN(n5113) );
  AOI221D0 U5665 ( .A1(n5178), .A2(n8899), .B1(n5114), .B2(
     n8899), .C(n5113),
        .ZN(n5120) );
  INVD0 U5666 ( .I(n8556), .ZN(n6957) );
  NR2D0 U5667 ( .A1(n5545), .A2(n6957), .ZN(n5302) );
  INVD0 U5668 ( .I(n5302), .ZN(n5116) );
  INVD0 U5669 ( .I(n8551), .ZN(n8716) );
  CKND2D0 U5670 ( .A1(n8716), .A2(n6717), .ZN(n5336) );
  INVD0 U5671 ( .I(n5336), .ZN(n5294) );
  INVD0 U5672 ( .I(n8836), .ZN(n8778) );
  NR2D0 U5673 ( .A1(n5668), .A2(n8778), .ZN(n6740) );
  CKND2D0 U5674 ( .A1(n5294), .A2(n6740), .ZN(n5304) );
  NR2D0 U5675 ( .A1(n8503), .A2(n8596), .ZN(n6930) );
  ND4D0 U5676 ( .A1(n5598), .A2(n6930), .A3(n6847), .A4(n8724),
      .ZN(n5115) );
```

```
OAI31D0 U5677 ( .A1(n5116), .A2(n5304), .A3(n5115), .B(n8697)
   , .ZN(n5119) );
INVD0 U5678 ( .I(n8514), .ZN(n8505) );
NR2D0 U5679 ( .A1(n5869), .A2(n8612), .ZN(n5649) );
INVD0 U5680 ( .I(n5197), .ZN(n5269) );
ND4D0 U5681 ( .A1(n8855), .A2(n5215), .A3(n5269), .A4(n5586),
    .ZN(n5117) );
INVD0 U5682 ( .I(n8935), .ZN(n8685) );
OAI31D0 U5683 ( .A1(n8505), .A2(n5649), .A3(n5117), .B(n8685)
   , .ZN(n5118) );
INVD0 U5684 ( .I(n7933), .ZN(n8662) );
AOI31D0 U5685 ( .A1(n5120), .A2(n5119), .A3(n5118), .B(n8662)
   , .ZN(n5354) );
CKND2D0 U5686 ( .A1(n6248), .A2(n5837), .ZN(n5757) );
CKND2D0 U5687 ( .A1(n5576), .A2(n5757), .ZN(n5439) );
NR2D0 U5688 ( .A1(n5121), .A2(n8811), .ZN(n5243) );
CKND2D0 U5689 ( .A1(n5243), .A2(n8983), .ZN(n8882) );
INVD0 U5690 ( .I(n8882), .ZN(n8508) );
NR2D0 U5691 ( .A1(n8508), .A2(n8505), .ZN(n5123) );
IND4D0 U5692 ( .A1(n5439), .B1(n8855), .B2(n5123), .B3(n5122)
   , .ZN(n5132) );
NR2D0 U5693 ( .A1(n8551), .A2(n8782), .ZN(n6823) );
CKND2D0 U5694 ( .A1(n5626), .A2(n6926), .ZN(n8670) );
INVD0 U5695 ( .I(n5724), .ZN(n5699) );
CKND2D0 U5696 ( .A1(n8521), .A2(n5699), .ZN(n5328) );
NR2D0 U5697 ( .A1(n8055), .A2(counter[3]), .ZN(n7283) );
NR2D0 U5698 ( .A1(n6049), .A2(counter[6]), .ZN(n8902) );
CKND2D0 U5699 ( .A1(n7283), .A2(n8902), .ZN(n8617) );
INVD0 U5700 ( .I(n8617), .ZN(n8878) );
NR2D0 U5701 ( .A1(n5328), .A2(n8878), .ZN(n5278) );
INVD0 U5702 ( .I(n5278), .ZN(n5171) );
NR4D0 U5703 ( .A1(n8508), .A2(n8670), .A3(n5135), .A4(n5171),
    .ZN(n5124) );
AOI31D0 U5704 ( .A1(n6823), .A2(n5124), .A3(n8721), .B(n8674)
   , .ZN(n5131) );
CKND2D0 U5705 ( .A1(n5586), .A2(n6926), .ZN(n6802) );
IND3D0 U5706 ( .A1(n6802), .B1(n5129), .B2(n8836), .ZN(n5177)
    );
INVD0 U5707 ( .I(n8721), .ZN(n8877) );
NR2D0 U5708 ( .A1(n8877), .A2(n8831), .ZN(n5433) );
NR2D0 U5709 ( .A1(counter[1]), .A2(n6536), .ZN(n6392) );
INVD0 U5710 ( .I(n6392), .ZN(n6521) );
CKND2D0 U5711 ( .A1(n6942), .A2(n5385), .ZN(n8844) );
CKND2D0 U5712 ( .A1(n6521), .A2(n8844), .ZN(n5342) );
INVD0 U5713 ( .I(n5342), .ZN(n5293) );
NR2D0 U5714 ( .A1(n8878), .A2(n8648), .ZN(n5577) );
ND4D0 U5715 ( .A1(n8723), .A2(n5433), .A3(n5293), .A4(n5577),
    .ZN(n5125) );
```

```
OAI21D0 U5716 ( .A1(n5177), .A2(n5125), .B(n8685), .ZN(n5128)
    );
NR2D0 U5717 ( .A1(n8508), .A2(n8771), .ZN(n5808) );
CKND2D0 U5718 ( .A1(n7369), .A2(n7785), .ZN(n5321) );
CKAN2D0 U5719 ( .A1(n5808), .A2(n5321), .Z(n5295) );
NR2D0 U5720 ( .A1(n5650), .A2(n8596), .ZN(n5261) );
AN4D0 U5721 ( .A1(n5699), .A2(n5126), .A3(n5295), .A4(n5261),
    .Z(n5127) );
AOI32D0 U5722 ( .A1(n5129), .A2(n5128), .A3(n5127), .B1(n8919
   ), .B2(n5128),
      .ZN(n5130) );
AOI211D0 U5723 ( .A1(n8697), .A2(n5132), .B(n5131), .C(n5130)
   , .ZN(n5144) );
IND4D0 U5724 ( .A1(n5282), .B1(n5677), .B2(n8836), .B3(n5757)
   , .ZN(n5141) );
NR2D0 U5725 ( .A1(n5724), .A2(n8782), .ZN(n5241) );
ND4D0 U5726 ( .A1(n8794), .A2(n5241), .A3(n5234), .A4(n6927),
    .ZN(n5148) );
INVD0 U5727 ( .I(n8648), .ZN(n8519) );
IND4D0 U5728 ( .A1(n5243), .B1(n5216), .B2(n8716), .B3(n8519)
   , .ZN(n5133) );
NR4D0 U5729 ( .A1(n8631), .A2(n6802), .A3(n5148), .A4(n5133),
    .ZN(n5139) );
ND3D0 U5730 ( .A1(n8716), .A2(n8882), .A3(n5321), .ZN(n5290)
   );
NR2D0 U5731 ( .A1(n8860), .A2(n8782), .ZN(n6763) );
INVD0 U5732 ( .I(n6763), .ZN(n6827) );
NR2D0 U5733 ( .A1(n5668), .A2(n6827), .ZN(n5556) );
NR2D0 U5734 ( .A1(n8924), .A2(n8426), .ZN(n6731) );
INVD0 U5735 ( .I(n6731), .ZN(n5276) );
ND4D0 U5736 ( .A1(n5556), .A2(n8813), .A3(n8791), .A4(n5276),
    .ZN(n5134) );
OAI31D0 U5737 ( .A1(n5290), .A2(n5135), .A3(n5134), .B(n8899)
   , .ZN(n5138) );
CKND2D0 U5738 ( .A1(n8811), .A2(n5856), .ZN(n5654) );
INVD0 U5739 ( .I(n5654), .ZN(n5639) );
CKND2D0 U5740 ( .A1(n8514), .A2(n5321), .ZN(n5153) );
NR2D0 U5741 ( .A1(n8648), .A2(n5153), .ZN(n5260) );
OAI211D0 U5742 ( .A1(n5639), .A2(n8055), .B(n5261), .C(n5260)
   , .ZN(n5136) );
OAI31D0 U5743 ( .A1(n5283), .A2(n8530), .A3(n5136), .B(n8822)
   , .ZN(n5137) );
OAI211D0 U5744 ( .A1(n5139), .A2(n8935), .B(n5138), .C(n5137)
   , .ZN(n5140) );
AOI221D0 U5745 ( .A1(n5142), .A2(n8697), .B1(n5141), .B2(
   n8697), .C(n5140),
      .ZN(n5143) );
INVD0 U5746 ( .I(n8910), .ZN(n8230) );
```

```
OAI22D0 U5747 ( .A1(n5144), .A2(n8945), .B1(n5143), .B2(n8230
   ), .ZN(n5353)
      );
NR2D0 U5748 ( .A1(n8530), .A2(n6904), .ZN(n5156) );
NR2D0 U5749 ( .A1(n8509), .A2(n6732), .ZN(n5414) );
INVD0 U5750 ( .I(n5414), .ZN(n8507) );
NR2D0 U5751 ( .A1(n8878), .A2(n8748), .ZN(n6727) );
INVD0 U5752 ( .I(n6727), .ZN(n6862) );
NR3D0 U5753 ( .A1(n8551), .A2(n8530), .A3(n6862), .ZN(n5266)
   );
INR2D0 U5754 ( .A1(n5145), .B1(n6802), .ZN(n5210) );
IND4D0 U5755 ( .A1(n5545), .B1(n5241), .B2(n5266), .B3(n5210)
   , .ZN(n5152) );
INVD0 U5756 ( .I(n5260), .ZN(n5162) );
NR2D0 U5757 ( .A1(n8808), .A2(n5205), .ZN(n5146) );
INVD0 U5758 ( .I(n7987), .ZN(n7321) );
CKND2D0 U5759 ( .A1(n7321), .A2(n7294), .ZN(n7708) );
CKND2D0 U5760 ( .A1(n5415), .A2(n7708), .ZN(n8548) );
ND4D0 U5761 ( .A1(n5218), .A2(n5146), .A3(n8548), .A4(n8601),
    .ZN(n5147) );
INR4D0 U5762 ( .A1(n5433), .B1(n8572), .B2(n5162), .B3(n5147)
   , .ZN(n5150) );
INVD0 U5763 ( .I(n6926), .ZN(n6970) );
NR4D0 U5764 ( .A1(n6970), .A2(n6113), .A3(n5148), .A4(n5162),
    .ZN(n5149) );
OAI22D0 U5765 ( .A1(n5150), .A2(n8919), .B1(n5149), .B2(n8674
   ), .ZN(n5151)
      );
AOI221D0 U5766 ( .A1(n8507), .A2(n8697), .B1(n5152), .B2(
   n8697), .C(n5151),
      .ZN(n5155) );
CKND2D0 U5767 ( .A1(n8521), .A2(n8724), .ZN(n8701) );
NR4D0 U5768 ( .A1(n8701), .A2(n5509), .A3(n5153), .A4(n5320),
    .ZN(n5154) );
AOI32D0 U5769 ( .A1(n5156), .A2(n5155), .A3(n5154), .B1(n8935
   ), .B2(n5155),
      .ZN(n5191) );
NR2D0 U5770 ( .A1(n5157), .A2(n8725), .ZN(n6925) );
NR2D0 U5771 ( .A1(n8782), .A2(n6925), .ZN(n6752) );
ND4D0 U5772 ( .A1(n5216), .A2(n6740), .A3(n6752), .A4(n5617),
    .ZN(n5160) );
NR2D0 U5773 ( .A1(n5668), .A2(n8505), .ZN(n8699) );
CKND2D0 U5774 ( .A1(n8699), .A2(n8836), .ZN(n5472) );
INVD0 U5775 ( .I(n5472), .ZN(n5265) );
AN4D0 U5776 ( .A1(n5586), .A2(n6776), .A3(n5295), .A4(n5265),
    .Z(n5158) );
AOI31D0 U5777 ( .A1(n5216), .A2(n5218), .A3(n5158), .B(n8937)
   , .ZN(n5159) );
```

```
AOI221D0 U5778 ( .A1(n5193), .A2(n8697), .B1(n5160), .B2(
   n8697), .C(n5159),
      .ZN(n5166) );
NR2D0 U5779 ( .A1(n8755), .A2(n8753), .ZN(n8839) );
CKND2D0 U5780 ( .A1(n8839), .A2(n8691), .ZN(n5201) );
NR2D0 U5781 ( .A1(n5328), .A2(n8771), .ZN(n5258) );
CKAN2D0 U5782 ( .A1(n8548), .A2(n5258), .Z(n5209) );
ND4D0 U5783 ( .A1(n5275), .A2(n5209), .A3(n6926), .A4(n8920),
    .ZN(n5161) );
OAI31D0 U5784 ( .A1(n5201), .A2(n8776), .A3(n5161), .B(n8822)
   , .ZN(n5165) );
NR2D0 U5785 ( .A1(n6970), .A2(n8596), .ZN(n5286) );
INVD0 U5786 ( .I(n5286), .ZN(n5268) );
NR2D0 U5787 ( .A1(n5243), .A2(n5162), .ZN(n5306) );
CKND2D0 U5788 ( .A1(n8053), .A2(n5837), .ZN(n6047) );
ND4D0 U5789 ( .A1(n5274), .A2(n5306), .A3(n6047), .A4(n8548),
    .ZN(n5163) );
OAI31D0 U5790 ( .A1(n5268), .A2(n8650), .A3(n5163), .B(n8582)
   , .ZN(n5164) );
INVD0 U5791 ( .I(n8656), .ZN(n6221) );
AOI31D0 U5792 ( .A1(n5166), .A2(n5165), .A3(n5164), .B(n6221)
   , .ZN(n5190) );
NR2D0 U5793 ( .A1(n5724), .A2(n8748), .ZN(n5708) );
NR4D0 U5794 ( .A1(n5243), .A2(n8507), .A3(n5545), .A4(n5268),
    .ZN(n5167) );
ND4D0 U5795 ( .A1(n5234), .A2(n5294), .A3(n5708), .A4(n5167),
    .ZN(n5176) );
NR2D0 U5796 ( .A1(n8876), .A2(n8530), .ZN(n5720) );
CKND2D0 U5797 ( .A1(n8716), .A2(n6737), .ZN(n8516) );
INVD0 U5798 ( .I(n8516), .ZN(n8624) );
NR2D0 U5799 ( .A1(n9352), .A2(n8953), .ZN(n7284) );
INVD0 U5800 ( .I(n7284), .ZN(n7444) );
NR2D0 U5801 ( .A1(n8416), .A2(n6558), .ZN(n5584) );
AOI221D0 U5802 ( .A1(n5917), .A2(n7444), .B1(n5584), .B2(
   n7444), .C(n5201),
      .ZN(n5168) );
AOI31D0 U5803 ( .A1(n5720), .A2(n8624), .A3(n5168), .B(n8919)
   , .ZN(n5175) );
NR2D0 U5804 ( .A1(n8426), .A2(n5869), .ZN(n8731) );
NR2D0 U5805 ( .A1(n5765), .A2(n8731), .ZN(n8718) );
CKND2D0 U5806 ( .A1(n8721), .A2(n8601), .ZN(n5647) );
NR2D0 U5807 ( .A1(n8748), .A2(n5647), .ZN(n5170) );
ND4D0 U5808 ( .A1(n8855), .A2(n5293), .A3(n5170), .A4(n5306),
    .ZN(n5169) );
OAI31D0 U5809 ( .A1(n6904), .A2(n8530), .A3(n5169), .B(n8899)
   , .ZN(n5173) );
INVD0 U5810 ( .I(n5170), .ZN(n5751) );
```

```
NR4D0 U5811 ( .A1(n6678), .A2(n5668), .A3(n5171), .A4(n5751),
    .ZN(n5172) );
AOI32D0 U5812 ( .A1(n8718), .A2(n5173), .A3(n5172), .B1(n8935
   ), .B2(n5173),
      .ZN(n5174) );
AOI211D0 U5813 ( .A1(n8697), .A2(n5176), .B(n5175), .C(n5174)
   , .ZN(n5188) );
CKND2D0 U5814 ( .A1(n7321), .A2(n7285), .ZN(n7170) );
OAI21D0 U5815 ( .A1(n7283), .A2(n7170), .B(n8902), .ZN(n5337)
    );
IIND4D0 U5816 ( .A1(n6904), .A2(n5177), .B1(n8844), .B2(n5337
   ), .ZN(n5186)
       );
INR4D0 U5817 ( .A1(n5179), .B1(n8733), .B2(n8748), .B3(n5178)
   , .ZN(n5180) );
AOI31D0 U5818 ( .A1(n5180), .A2(n8855), .A3(n5337), .B(n8674)
   , .ZN(n5185) );
ND4D0 U5819 ( .A1(n8855), .A2(n5260), .A3(n6628), .A4(n8844),
    .ZN(n5181) );
NR4D0 U5820 ( .A1(n5668), .A2(n8748), .A3(n5193), .A4(n5181),
    .ZN(n5183) );
ND3D0 U5821 ( .A1(n5209), .A2(n8617), .A3(n6857), .ZN(n5203)
   );
NR2D0 U5822 ( .A1(n7314), .A2(n7296), .ZN(n6513) );
OAI21D0 U5823 ( .A1(n6513), .A2(n5856), .B(n8882), .ZN(n5588)
    );
NR4D0 U5824 ( .A1(n8778), .A2(n5193), .A3(n5203), .A4(n5588),
    .ZN(n5182) );
OAI22D0 U5825 ( .A1(n5183), .A2(n8919), .B1(n5182), .B2(n8923
   ), .ZN(n5184)
       );
AOI211D0 U5826 ( .A1(n8582), .A2(n5186), .B(n5185), .C(n5184)
   , .ZN(n5187) );
INVD0 U5827 ( .I(n8105), .ZN(n8903) );
OAI22D0 U5828 ( .A1(n5188), .A2(n6986), .B1(n5187), .B2(n8903
   ), .ZN(n5189)
       );
AOI211D0 U5829 ( .A1(n7939), .A2(n5191), .B(n5190), .C(n5189)
   , .ZN(n5351) );
AN4D0 U5830 ( .A1(n6717), .A2(n5321), .A3(n5210), .A4(n5265),
    .Z(n5192) );
AOI31D0 U5831 ( .A1(n5234), .A2(n5275), .A3(n5192), .B(n8919)
   , .ZN(n5231) );
NR2D0 U5832 ( .A1(n9352), .A2(n8950), .ZN(n7326) );
CKND2D0 U5833 ( .A1(n7326), .A2(n7785), .ZN(n8819) );
INVD0 U5834 ( .I(n8819), .ZN(n8610) );
NR2D0 U5835 ( .A1(n8811), .A2(n6592), .ZN(n5195) );
CKND2D0 U5836 ( .A1(n6717), .A2(n8624), .ZN(n5221) );
```

```
INVD0 U5837 ( .I(n5221), .ZN(n5206) );
IND4D0 U5838 ( .A1(n5193), .B1(n5206), .B2(n8882), .B3(n8548)
   , .ZN(n5194) );
NR4D0 U5839 ( .A1(n8610), .A2(n5195), .A3(n5751), .A4(n5194),
    .ZN(n5200) );
CKND2D0 U5840 ( .A1(n5259), .A2(n5266), .ZN(n5262) );
OAI31D0 U5841 ( .A1(n7870), .A2(n8670), .A3(n5262), .B(n8685)
   , .ZN(n5199) );
CKND2D0 U5842 ( .A1(n7785), .A2(n8761), .ZN(n8666) );
ND4D0 U5843 ( .A1(n6823), .A2(n5293), .A3(n5586), .A4(n8666),
    .ZN(n5196) );
OAI31D0 U5844 ( .A1(n5751), .A2(n5197), .A3(n5196), .B(n8697)
   , .ZN(n5198) );
OAI211D0 U5845 ( .A1(n5200), .A2(n8674), .B(n5199), .C(n5198)
   , .ZN(n5230) );
NR2D0 U5846 ( .A1(n5328), .A2(n5201), .ZN(n5301) );
CKND2D0 U5847 ( .A1(n9352), .A2(n8156), .ZN(n7319) );
INVD0 U5848 ( .I(n7319), .ZN(n7332) );
CKND2D0 U5849 ( .A1(n7332), .A2(n5202), .ZN(n6165) );
ND4D0 U5850 ( .A1(n5261), .A2(n5266), .A3(n5301), .A4(n6165),
    .ZN(n5214) );
NR2D0 U5851 ( .A1(n5203), .A2(n5221), .ZN(n5279) );
CKND2D0 U5852 ( .A1(n8794), .A2(n8512), .ZN(n5204) );
NR2D0 U5853 ( .A1(n5204), .A2(n5589), .ZN(n5277) );
ND4D0 U5854 ( .A1(n5279), .A2(n5277), .A3(n8920), .A4(n5576),
    .ZN(n5208) );
INR4D0 U5855 ( .A1(n5617), .B1(n6925), .B2(n5205), .B3(n5282)
   , .ZN(n5217) );
ND4D0 U5856 ( .A1(n5209), .A2(n5206), .A3(n5217), .A4(n6894),
    .ZN(n5207) );
AOI22D0 U5857 ( .A1(n8822), .A2(n5208), .B1(n8697), .B2(n5207
   ), .ZN(n5212)
       );
AN4D0 U5858 ( .A1(n8514), .A2(n5757), .A3(n5210), .A4(n5209),
    .Z(n5211) );
AOI32D0 U5859 ( .A1(n5259), .A2(n5212), .A3(n5211), .B1(n8935
   ), .B2(n5212),
      .ZN(n5213) );
AOI221D0 U5860 ( .A1(n8876), .A2(n8899), .B1(n5214), .B2(
   n8899), .C(n5213),
      .ZN(n5228) );
CKND2D0 U5861 ( .A1(n5586), .A2(n8512), .ZN(n8770) );
ND4D0 U5862 ( .A1(n8699), .A2(n5215), .A3(n5289), .A4(n5577),
    .ZN(n5226) );
IINR4D0 U5863 ( .A1(n5217), .A2(n5216), .B1(n6392), .B2(n5647
   ), .ZN(n5224)
       );
CKND2D0 U5864 ( .A1(n8617), .A2(n8836), .ZN(n5578) );
```

```
NR2D0 U5865 ( .A1(n8808), .A2(n8672), .ZN(n8891) );
INVD0 U5866 ( .I(n8844), .ZN(n8632) );
NR2D0 U5867 ( .A1(n8632), .A2(n6732), .ZN(n5492) );
ND4D0 U5868 ( .A1(n8891), .A2(n5218), .A3(n5234), .A4(n5492),
    .ZN(n5219) );
OAI31D0 U5869 ( .A1(n5578), .A2(n5268), .A3(n5219), .B(n8822)
   , .ZN(n5223) );
NR2D0 U5870 ( .A1(n8508), .A2(n8748), .ZN(n8704) );
NR2D0 U5871 ( .A1(n8860), .A2(n8831), .ZN(n6741) );
ND4D0 U5872 ( .A1(n8704), .A2(n6741), .A3(n8844), .A4(n8702),
    .ZN(n5220) );
OAI31D0 U5873 ( .A1(n5221), .A2(n5283), .A3(n5220), .B(n8697)
   , .ZN(n5222) );
OAI211D0 U5874 ( .A1(n5224), .A2(n8935), .B(n5223), .C(n5222)
   , .ZN(n5225) );
AOI221D0 U5875 ( .A1(n8770), .A2(n8899), .B1(n5226), .B2(
   n8899), .C(n5225),
      .ZN(n5227) );
INVD0 U5876 ( .I(n8323), .ZN(n8687) );
OAI22D0 U5877 ( .A1(n5228), .A2(n6795), .B1(n5227), .B2(n8687
   ), .ZN(n5229)
      );
AOI221D0 U5878 ( .A1(n5231), .A2(n8037), .B1(n5230), .B2(
   n8037), .C(n5229),
      .ZN(n5350) );
INVD0 U5879 ( .I(n5523), .ZN(n8918) );
NR2D0 U5880 ( .A1(counter[6]), .A2(n5299), .ZN(n5236) );
CKND2D0 U5881 ( .A1(n7295), .A2(n7285), .ZN(n7360) );
NR2D0 U5882 ( .A1(n6205), .A2(n8038), .ZN(n5762) );
CKND2D0 U5883 ( .A1(n8761), .A2(n5762), .ZN(n5506) );
INVD0 U5884 ( .I(n5506), .ZN(n6819) );
AOI211D0 U5885 ( .A1(n5236), .A2(n7360), .B(n6819), .C(n8505)
   , .ZN(n5300) );
NR2D0 U5886 ( .A1(n9143), .A2(n8612), .ZN(n5237) );
CKND2D0 U5887 ( .A1(n5237), .A2(n6878), .ZN(n6876) );
INVD0 U5888 ( .I(n6876), .ZN(n5499) );
CKND2D0 U5889 ( .A1(n8396), .A2(n5917), .ZN(n6764) );
NR2D0 U5890 ( .A1(n9352), .A2(n5299), .ZN(n5504) );
INVD0 U5891 ( .I(n5504), .ZN(n5956) );
AOI32D0 U5892 ( .A1(n6513), .A2(n6764), .A3(n7294), .B1(n5956
   ), .B2(n6764),
      .ZN(n5232) );
NR4D0 U5893 ( .A1(n5668), .A2(n8782), .A3(n5499), .A4(n5232),
    .ZN(n5233) );
NR2D0 U5894 ( .A1(n9352), .A2(n8611), .ZN(n6818) );
INVD0 U5895 ( .I(n6818), .ZN(n8570) );
CKND2D0 U5896 ( .A1(counter[6]), .A2(n5794), .ZN(n5332) );
NR2D0 U5897 ( .A1(n7169), .A2(n5332), .ZN(n6837) );
```

```
CKND2D0 U5898 ( .A1(counter[2]), .A2(n6837), .ZN(n5378) );
ND4D0 U5899 ( .A1(n5300), .A2(n5233), .A3(n8570), .A4(n5378),
    .ZN(n5325) );
IND2D0 U5900 ( .A1(n5236), .B1(n5332), .ZN(n5772) );
NR2D0 U5901 ( .A1(n5837), .A2(n5772), .ZN(n5235) );
NR2D0 U5902 ( .A1(n5801), .A2(n5377), .ZN(n8772) );
INVD0 U5903 ( .I(n8772), .ZN(n8571) );
OAI211D0 U5904 ( .A1(n5235), .A2(n7235), .B(n5234), .C(n8571)
   , .ZN(n5251) );
CKND2D0 U5905 ( .A1(n7350), .A2(n5236), .ZN(n6952) );
CKND2D0 U5906 ( .A1(n7414), .A2(n5504), .ZN(n8537) );
CKND2D0 U5907 ( .A1(n5379), .A2(n6812), .ZN(n5621) );
ND4D0 U5908 ( .A1(n8521), .A2(n6952), .A3(n8537), .A4(n5621),
    .ZN(n5240) );
INVD0 U5909 ( .I(n8426), .ZN(n8311) );
CKND2D0 U5910 ( .A1(n8311), .A2(n8983), .ZN(n8769) );
NR2D0 U5911 ( .A1(n8769), .A2(n5377), .ZN(n8545) );
INVD0 U5912 ( .I(n5237), .ZN(n5246) );
NR2D0 U5913 ( .A1(n5246), .A2(n7717), .ZN(n8538) );
NR2D0 U5914 ( .A1(n8545), .A2(n8538), .ZN(n5291) );
CKND2D0 U5915 ( .A1(n7326), .A2(n5794), .ZN(n5238) );
ND4D0 U5916 ( .A1(n5291), .A2(n5476), .A3(n5577), .A4(n5238),
    .ZN(n5239) );
NR4D0 U5917 ( .A1(n5325), .A2(n5251), .A3(n5240), .A4(n5239),
    .ZN(n5256) );
INVD0 U5918 ( .I(n5767), .ZN(n5326) );
NR2D0 U5919 ( .A1(n5326), .A2(n6368), .ZN(n6969) );
NR2D0 U5920 ( .A1(n5292), .A2(n8108), .ZN(n5491) );
NR2D0 U5921 ( .A1(n6969), .A2(n5491), .ZN(n5248) );
NR2D0 U5922 ( .A1(n8950), .A2(n5956), .ZN(n5935) );
CKND2D0 U5923 ( .A1(n6724), .A2(n8702), .ZN(n8598) );
NR2D0 U5924 ( .A1(n8038), .A2(counter[6]), .ZN(n8764) );
NR2D0 U5925 ( .A1(n7321), .A2(n5956), .ZN(n5795) );
CKND2D0 U5926 ( .A1(n9143), .A2(n8311), .ZN(n6824) );
NR2D0 U5927 ( .A1(n8108), .A2(n6824), .ZN(n5372) );
NR4D0 U5928 ( .A1(n5795), .A2(n5372), .A3(n6818), .A4(n8771),
    .ZN(n5245) );
ND4D0 U5929 ( .A1(n5241), .A2(n5245), .A3(n6847), .A4(n5378),
    .ZN(n5242) );
AOI211D0 U5930 ( .A1(n6639), .A2(n8764), .B(n5243), .C(n5242)
   , .ZN(n5340) );
CKND2D0 U5931 ( .A1(n5265), .A2(n5340), .ZN(n5334) );
INR4D0 U5932 ( .A1(n5291), .B1(n5935), .B2(n8598), .B3(n5334)
   , .ZN(n5244) );
INVD0 U5933 ( .I(n8611), .ZN(n8539) );
CKND2D0 U5934 ( .A1(n8539), .A2(n9352), .ZN(n8575) );
ND4D0 U5935 ( .A1(n5720), .A2(n5248), .A3(n5244), .A4(n8575),
    .ZN(n5250) );
```

```
AOI21D0 U5936 ( .A1(n7360), .A2(n5504), .B(n6819), .ZN(n5339)
    );
OAI211D0 U5937 ( .A1(n8416), .A2(n7673), .B(n5266), .C(n5245)
   , .ZN(n5253) );
NR2D0 U5938 ( .A1(n5246), .A2(n8532), .ZN(n8762) );
INR2D0 U5939 ( .A1(n6952), .B1(n8762), .ZN(n5298) );
CKND2D0 U5940 ( .A1(counter[2]), .A2(n8309), .ZN(n5595) );
CKND2D0 U5941 ( .A1(n5784), .A2(n5595), .ZN(n5817) );
CKND2D0 U5942 ( .A1(n6361), .A2(n5817), .ZN(n8633) );
CKND2D0 U5943 ( .A1(n5298), .A2(n8633), .ZN(n6964) );
INR4D0 U5944 ( .A1(n5339), .B1(n8670), .B2(n5253), .B3(n6964)
   , .ZN(n5247) );
ND4D0 U5945 ( .A1(n5274), .A2(n5248), .A3(n5247), .A4(n8556),
    .ZN(n5249) );
AOI22D0 U5946 ( .A1(n8822), .A2(n5250), .B1(n8899), .B2(n5249
   ), .ZN(n5255)
      );
CKND2D0 U5947 ( .A1(n6813), .A2(n6812), .ZN(n8543) );
ND4D0 U5948 ( .A1(n5298), .A2(n6927), .A3(n8543), .A4(n6814),
    .ZN(n5318) );
IND4D0 U5949 ( .A1(n5251), .B1(n7950), .B2(n8691), .B3(n6926)
   , .ZN(n5252) );
OAI31D0 U5950 ( .A1(n5253), .A2(n5318), .A3(n5252), .B(n8685)
   , .ZN(n5254) );
OAI211D0 U5951 ( .A1(n5256), .A2(n8923), .B(n5255), .C(n5254)
   , .ZN(n5317) );
NR4D0 U5952 ( .A1(n5668), .A2(n5282), .A3(n5439), .A4(n5336),
    .ZN(n5257) );
ND4D0 U5953 ( .A1(n5260), .A2(n5259), .A3(n5258), .A4(n5257),
    .ZN(n5264) );
NR2D0 U5954 ( .A1(n8648), .A2(n8503), .ZN(n6874) );
IND4D0 U5955 ( .A1(n5262), .B1(n5261), .B2(n5433), .B3(n6874)
   , .ZN(n5263) );
AOI22D0 U5956 ( .A1(n8822), .A2(n5264), .B1(n8697), .B2(n5263
   ), .ZN(n5273)
      );
ND4D0 U5957 ( .A1(n5266), .A2(n5265), .A3(n8813), .A4(n8601),
    .ZN(n5267) );
OAI31D0 U5958 ( .A1(n6731), .A2(n5268), .A3(n5267), .B(n8899)
   , .ZN(n5272) );
NR3D0 U5959 ( .A1(n8509), .A2(n8632), .A3(n6915), .ZN(n6765)
   );
ND3D0 U5960 ( .A1(n5269), .A2(n5708), .A3(n6765), .ZN(n5270)
   );
OAI31D0 U5961 ( .A1(n5360), .A2(n8531), .A3(n5270), .B(n8685)
   , .ZN(n5271) );
AOI31D0 U5962 ( .A1(n5273), .A2(n5272), .A3(n5271), .B(n7593)
   , .ZN(n5316) );
```

```
CKND2D0 U5963 ( .A1(n6724), .A2(n6847), .ZN(n8867) );
INVD0 U5964 ( .I(n8867), .ZN(n8511) );
INVD0 U5965 ( .I(n8755), .ZN(n6786) );
CKND2D0 U5966 ( .A1(n8511), .A2(n6786), .ZN(n8529) );
CKND2D0 U5967 ( .A1(n7002), .A2(n5415), .ZN(n6762) );
ND4D0 U5968 ( .A1(n5275), .A2(n5274), .A3(n8556), .A4(n6762),
    .ZN(n5288) );
ND4D0 U5969 ( .A1(n5278), .A2(n6930), .A3(n5277), .A4(n5276),
    .ZN(n5281) );
NR2D0 U5970 ( .A1(n8808), .A2(n6275), .ZN(n5841) );
AOI31D0 U5971 ( .A1(n5279), .A2(n5293), .A3(n5841), .B(n8923)
   , .ZN(n5280) );
AOI221D0 U5972 ( .A1(n5282), .A2(n8609), .B1(n5281), .B2(
   n8822), .C(n5280),
      .ZN(n5285) );
NR4D0 U5973 ( .A1(n8572), .A2(n5283), .A3(n5290), .A4(n6904),
    .ZN(n5284) );
AOI32D0 U5974 ( .A1(n5286), .A2(n5285), .A3(n5284), .B1(n8935
   ), .B2(n5285),
      .ZN(n5287) );
AOI221D0 U5975 ( .A1(n8529), .A2(n8899), .B1(n5288), .B2(
   n8899), .C(n5287),
      .ZN(n5314) );
INVD0 U5976 ( .I(n8803), .ZN(n8169) );
IND4D0 U5977 ( .A1(n5290), .B1(n8794), .B2(n5289), .B3(n5492)
   , .ZN(n5312) );
INVD0 U5978 ( .I(n5372), .ZN(n5802) );
CKND2D0 U5979 ( .A1(n5291), .A2(n5802), .ZN(n8779) );
NR2D0 U5980 ( .A1(n5292), .A2(n7717), .ZN(n8763) );
INVD0 U5981 ( .I(n8763), .ZN(n8626) );
ND4D0 U5982 ( .A1(n5293), .A2(n5298), .A3(n8548), .A4(n8626),
    .ZN(n5297) );
CKND2D0 U5983 ( .A1(n5766), .A2(n8764), .ZN(n5620) );
ND4D0 U5984 ( .A1(n5295), .A2(n5294), .A3(n8721), .A4(n5620),
    .ZN(n5296) );
NR4D0 U5985 ( .A1(n6970), .A2(n8779), .A3(n5297), .A4(n5296),
    .ZN(n5310) );
INVD0 U5986 ( .I(n8764), .ZN(n5397) );
NR2D0 U5987 ( .A1(n7294), .A2(n5397), .ZN(n8566) );
INR2D0 U5988 ( .A1(n5620), .B1(n8566), .ZN(n5776) );
OAI211D0 U5989 ( .A1(n5801), .A2(n5299), .B(n5298), .C(n5776)
   , .ZN(n5305) );
ND4D0 U5990 ( .A1(n5302), .A2(n5301), .A3(n5300), .A4(n8512),
    .ZN(n5303) );
OAI31D0 U5991 ( .A1(n5305), .A2(n5304), .A3(n5303), .B(n8685)
   , .ZN(n5309) );
NR2D0 U5992 ( .A1(n6592), .A2(n8613), .ZN(n8747) );
NR2D0 U5993 ( .A1(n8753), .A2(n8632), .ZN(n6738) );
```

```
ND4D0 U5994 ( .A1(n5306), .A2(n6738), .A3(n5677), .A4(n6726),
    .ZN(n5307) );
OAI31D0 U5995 ( .A1(n5360), .A2(n8747), .A3(n5307), .B(n8899)
   , .ZN(n5308) );
OAI211D0 U5996 ( .A1(n5310), .A2(n8923), .B(n5309), .C(n5308)
   , .ZN(n5311) );
AOI221D0 U5997 ( .A1(n8529), .A2(n8609), .B1(n5312), .B2(
   n8822), .C(n5311),
      .ZN(n5313) );
INVD0 U5998 ( .I(n7694), .ZN(n8797) );
OAI22D0 U5999 ( .A1(n5314), .A2(n8169), .B1(n5313), .B2(n8797
   ), .ZN(n5315)
       );
AOI211D0 U6000 ( .A1(n8918), .A2(n5317), .B(n5316), .C(n5315)
   , .ZN(n5349) );
AOI211D0 U6001 ( .A1(n7064), .A2(n5772), .B(n8505), .C(n8508)
   , .ZN(n5323) );
CKND2D0 U6002 ( .A1(n5802), .A2(n5621), .ZN(n8573) );
NR2D0 U6003 ( .A1(n8533), .A2(n8108), .ZN(n8565) );
NR3D0 U6004 ( .A1(n8565), .A2(n8701), .A3(n5318), .ZN(n5333)
   );
IND4D0 U6005 ( .A1(n5795), .B1(n5333), .B2(n5339), .B3(n6521)
   , .ZN(n5319) );
INR4D0 U6006 ( .A1(n6823), .B1(n8573), .B2(n5320), .B3(n5319)
   , .ZN(n5322) );
AOI31D0 U6007 ( .A1(n5323), .A2(n5322), .A3(n5321), .B(n8919)
   , .ZN(n5347) );
NR4D0 U6008 ( .A1(n8672), .A2(n5324), .A3(n5795), .A4(n8566),
    .ZN(n5331) );
INVD0 U6009 ( .I(n5325), .ZN(n5330) );
NR2D0 U6010 ( .A1(n5327), .A2(n5326), .ZN(n6956) );
CKND2D0 U6011 ( .A1(n8691), .A2(n6894), .ZN(n8828) );
INR4D0 U6012 ( .A1(n8548), .B1(n6956), .B2(n8828), .B3(n5328)
   , .ZN(n5329) );
AOI31D0 U6013 ( .A1(n5331), .A2(n5330), .A3(n5329), .B(n8935)
   , .ZN(n5346) );
CKND2D0 U6014 ( .A1(n8046), .A2(n8902), .ZN(n6715) );
INVD0 U6015 ( .I(n6715), .ZN(n6783) );
NR2D0 U6016 ( .A1(n7235), .A2(n5332), .ZN(n8536) );
NR2D0 U6017 ( .A1(n8536), .A2(n6819), .ZN(n6832) );
NR2D0 U6018 ( .A1(n6578), .A2(n5595), .ZN(n6820) );
INVD0 U6019 ( .I(n6820), .ZN(n8544) );
IND4D0 U6020 ( .A1(n5334), .B1(n5333), .B2(n6832), .B3(n8544)
   , .ZN(n5335) );
NR4D0 U6021 ( .A1(n5650), .A2(n6783), .A3(n5336), .A4(n5335),
    .ZN(n5344) );
NR2D0 U6022 ( .A1(n7743), .A2(n5956), .ZN(n5822) );
```

```
NR4D0 U6023 ( .A1(n8503), .A2(n8545), .A3(n5822), .A4(n8536),
    .ZN(n5338) );
ND4D0 U6024 ( .A1(n5340), .A2(n5339), .A3(n5338), .A4(n5337),
    .ZN(n5341) );
AOI211D0 U6025 ( .A1(n8764), .A2(n7170), .B(n5342), .C(n5341)
   , .ZN(n5343) );
OAI22D0 U6026 ( .A1(n5344), .A2(n8923), .B1(n5343), .B2(n8937
   ), .ZN(n5345)
       );
INVD0 U6027 ( .I(n6604), .ZN(n8639) );
OAI31D0 U6028 ( .A1(n5347), .A2(n5346), .A3(n5345), .B(n8639)
   , .ZN(n5348) );
ND4D0 U6029 ( .A1(n5351), .A2(n5350), .A3(n5349), .A4(n5348),
    .ZN(n5352) );
NR4D0 U6030 ( .A1(n5355), .A2(n5354), .A3(n5353), .A4(n5352),
    .ZN(n5356) );
OAI222D0 U6031 ( .A1(n5358), .A2(n7277), .B1(n7000), .B2(
   n5357), .C1(n6992),
      .C2(n5356), .ZN(n4812) );
INVD0 U6032 ( .I(q_out[1]), .ZN(n5855) );
NR2D0 U6033 ( .A1(n6678), .A2(n8877), .ZN(n6753) );
INVD0 U6034 ( .I(n8699), .ZN(n8664) );
NR2D0 U6035 ( .A1(n8620), .A2(n6925), .ZN(n6867) );
INVD0 U6036 ( .I(n6867), .ZN(n6793) );
NR2D0 U6037 ( .A1(n6793), .A2(n6957), .ZN(n5464) );
INVD0 U6038 ( .I(n5464), .ZN(n5535) );
NR2D0 U6039 ( .A1(n8664), .A2(n5535), .ZN(n5425) );
ND4D0 U6040 ( .A1(n5708), .A2(n6753), .A3(n5425), .A4(n6762),
    .ZN(n5365) );
NR2D0 U6041 ( .A1(n8596), .A2(n6925), .ZN(n6870) );
NR2D0 U6042 ( .A1(n8753), .A2(n5360), .ZN(n8678) );
INVD0 U6043 ( .I(n8678), .ZN(n8668) );
NR2D0 U6044 ( .A1(n8878), .A2(n8782), .ZN(n6774) );
INVD0 U6045 ( .I(n6774), .ZN(n6804) );
NR3D0 U6046 ( .A1(n8668), .A2(n6915), .A3(n6804), .ZN(n5540)
   );
CKND2D0 U6047 ( .A1(n6347), .A2(n8316), .ZN(n8698) );
AOI31D0 U6048 ( .A1(n6870), .A2(n5540), .A3(n8698), .B(n8923)
   , .ZN(n5364) );
NR2D0 U6049 ( .A1(n5724), .A2(n8735), .ZN(n6840) );
INVD0 U6050 ( .I(n6840), .ZN(n6855) );
NR2D0 U6051 ( .A1(n8748), .A2(n6855), .ZN(n6777) );
NR2D0 U6052 ( .A1(n8551), .A2(n8503), .ZN(n6971) );
NR2D0 U6053 ( .A1(n9345), .A2(n8811), .ZN(n5665) );
CKND2D0 U6054 ( .A1(n5665), .A2(n8957), .ZN(n5557) );
ND4D0 U6055 ( .A1(n6738), .A2(n6971), .A3(n5586), .A4(n5557),
    .ZN(n5359) );
```

```
OAI31D0 U6056 ( .A1(n6925), .A2(n8505), .A3(n5359), .B(n8685)
   , .ZN(n5362) );
NR2D0 U6057 ( .A1(n6732), .A2(n5360), .ZN(n8518) );
INVD0 U6058 ( .I(n8518), .ZN(n6917) );
CKND2D0 U6059 ( .A1(n6717), .A2(n8601), .ZN(n8590) );
INVD0 U6060 ( .I(n8590), .ZN(n6868) );
NR2D0 U6061 ( .A1(n8808), .A2(n8753), .ZN(n5696) );
CKND2D0 U6062 ( .A1(n6868), .A2(n5696), .ZN(n6720) );
NR2D0 U6063 ( .A1(n6915), .A2(n6720), .ZN(n5553) );
INVD0 U6064 ( .I(n5553), .ZN(n5482) );
NR4D0 U6065 ( .A1(n8620), .A2(n6957), .A3(n6917), .A4(n5482),
    .ZN(n5361) );
AOI32D0 U6066 ( .A1(n6777), .A2(n5362), .A3(n5361), .B1(n8919
   ), .B2(n5362),
      .ZN(n5363) );
AOI211D0 U6067 ( .A1(n8899), .A2(n5365), .B(n5364), .C(n5363)
   , .ZN(n5413) );
INVD0 U6068 ( .I(n5491), .ZN(n5758) );
CKND2D0 U6069 ( .A1(n6717), .A2(n8556), .ZN(n8744) );
INVD0 U6070 ( .I(n8744), .ZN(n8703) );
ND3D0 U6071 ( .A1(n8626), .A2(n5758), .A3(n8703), .ZN(n5404)
   );
CKND2D0 U6072 ( .A1(n6786), .A2(n8721), .ZN(n6903) );
INVD0 U6073 ( .I(n6903), .ZN(n6851) );
NR2D0 U6074 ( .A1(n8808), .A2(n8778), .ZN(n6858) );
NR4D0 U6075 ( .A1(n5795), .A2(n8782), .A3(n8536), .A4(n8545),
    .ZN(n6967) );
IND4D0 U6076 ( .A1(n5404), .B1(n6851), .B2(n6858), .B3(n6967)
   , .ZN(n5368) );
INVD0 U6077 ( .I(n8543), .ZN(n6836) );
OAI211D0 U6078 ( .A1(n8416), .A2(n8532), .B(n8521), .C(n8544)
   , .ZN(n5366) );
NR2D0 U6079 ( .A1(n6836), .A2(n5366), .ZN(n5502) );
CKND2D0 U6080 ( .A1(n6837), .A2(n8983), .ZN(n8561) );
ND4D0 U6081 ( .A1(n5502), .A2(n8561), .A3(n6876), .A4(n8514),
    .ZN(n5367) );
NR2D0 U6082 ( .A1(n5368), .A2(n5367), .ZN(n5384) );
NR2D0 U6083 ( .A1(n5724), .A2(n4819), .ZN(n6890) );
CKND2D0 U6084 ( .A1(n8702), .A2(n6890), .ZN(n5462) );
OAI211D0 U6085 ( .A1(n7717), .A2(n8533), .B(n5378), .C(n8691)
   , .ZN(n5513) );
CKND2D0 U6086 ( .A1(n8518), .A2(n8699), .ZN(n5370) );
ND4D0 U6087 ( .A1(n8844), .A2(n8626), .A3(n6814), .A4(n6717),
    .ZN(n5369) );
NR4D0 U6088 ( .A1(n6964), .A2(n5513), .A3(n5370), .A4(n5369),
    .ZN(n5371) );
CKND2D0 U6089 ( .A1(n8532), .A2(n7756), .ZN(n6941) );
CKND2D0 U6090 ( .A1(n7310), .A2(n6941), .ZN(n6838) );
```

```
IND4D0 U6091 ( .A1(n5462), .B1(n5371), .B2(n5586), .B3(n6838)
   , .ZN(n5376) );
INVD0 U6092 ( .I(n8698), .ZN(n8504) );
NR2D0 U6093 ( .A1(n5724), .A2(n8504), .ZN(n8513) );
CKND2D0 U6094 ( .A1(counter[3]), .A2(n5935), .ZN(n6960) );
INVD0 U6095 ( .I(n6960), .ZN(n8542) );
NR2D0 U6096 ( .A1(n8596), .A2(n6731), .ZN(n6816) );
CKND2D0 U6097 ( .A1(n6816), .A2(n8724), .ZN(n5543) );
NR4D0 U6098 ( .A1(n5372), .A2(n8538), .A3(n6819), .A4(n5543),
    .ZN(n5405) );
ND4D0 U6099 ( .A1(n5502), .A2(n5405), .A3(n5378), .A4(n8601),
    .ZN(n5373) );
AOI211D0 U6100 ( .A1(n5379), .A2(n6941), .B(n8542), .C(n5373)
   , .ZN(n5374) );
IND4D0 U6101 ( .A1(n6969), .B1(n8513), .B2(n5374), .B3(n8844)
   , .ZN(n5375) );
AOI22D0 U6102 ( .A1(n8822), .A2(n5376), .B1(n8899), .B2(n5375
   ), .ZN(n5383)
      );
NR2D0 U6103 ( .A1(counter[2]), .A2(n8441), .ZN(n5810) );
IND2D0 U6104 ( .A1(n5377), .B1(n5810), .ZN(n8555) );
ND3D0 U6105 ( .A1(n8626), .A2(n5378), .A3(n8555), .ZN(n6965)
   );
OR2D0 U6106 ( .A1(n5379), .A2(n6942), .Z(n5500) );
NR2D0 U6107 ( .A1(n8778), .A2(n8876), .ZN(n6716) );
CKND2D0 U6108 ( .A1(n6716), .A2(n6776), .ZN(n6792) );
AOI211D0 U6109 ( .A1(n6825), .A2(n5500), .B(n6818), .C(n6792)
   , .ZN(n5380) );
CKND2D0 U6110 ( .A1(n5932), .A2(n8983), .ZN(n8068) );
NR2D0 U6111 ( .A1(n8612), .A2(n8068), .ZN(n5809) );
CKND2D0 U6112 ( .A1(counter[1]), .A2(n5809), .ZN(n8557) );
ND4D0 U6113 ( .A1(n5380), .A2(n8716), .A3(n8557), .A4(n8543),
    .ZN(n5381) );
OAI31D0 U6114 ( .A1(n6965), .A2(n6862), .A3(n5381), .B(n8697)
   , .ZN(n5382) );
OAI211D0 U6115 ( .A1(n5384), .A2(n8935), .B(n5383), .C(n5382)
   , .ZN(n5393) );
CKND2D0 U6116 ( .A1(n8813), .A2(n8836), .ZN(n6887) );
NR2D0 U6117 ( .A1(n8878), .A2(n6887), .ZN(n6908) );
CKND2D0 U6118 ( .A1(n6908), .A2(n6776), .ZN(n6923) );
NR2D0 U6119 ( .A1(n8735), .A2(n8508), .ZN(n8517) );
CKND2D0 U6120 ( .A1(n8517), .A2(n8699), .ZN(n6932) );
NR4D0 U6121 ( .A1(n5536), .A2(n6915), .A3(n6923), .A4(n6932),
    .ZN(n5391) );
NR2D0 U6122 ( .A1(n5668), .A2(n8877), .ZN(n6954) );
CKND2D0 U6123 ( .A1(n6954), .A2(n8844), .ZN(n6913) );
NR2D0 U6124 ( .A1(n8551), .A2(n8831), .ZN(n5686) );
```

```
ND4D0 U6125 ( .A1(n6774), .A2(n8517), .A3(n5686), .A4(n5557),
    .ZN(n5387) );
NR2D0 U6126 ( .A1(n8509), .A2(n8771), .ZN(n5483) );
CKND2D0 U6127 ( .A1(n5766), .A2(n8902), .ZN(n8854) );
CKND2D0 U6128 ( .A1(n5385), .A2(n5500), .ZN(n6835) );
AOI31D0 U6129 ( .A1(n5483), .A2(n8854), .A3(n6835), .B(n8923)
   , .ZN(n5386) );
AOI221D0 U6130 ( .A1(n6913), .A2(n8609), .B1(n5387), .B2(
   n8822), .C(n5386),
      .ZN(n5390) );
CKND2D0 U6131 ( .A1(n6786), .A2(n5699), .ZN(n6893) );
INVD0 U6132 ( .I(n6893), .ZN(n6775) );
CKND2D0 U6133 ( .A1(n6775), .A2(n6927), .ZN(n6914) );
CKND2D0 U6134 ( .A1(n8813), .A2(n8724), .ZN(n6794) );
ND4D0 U6135 ( .A1(n6741), .A2(n5483), .A3(n8721), .A4(n6776),
    .ZN(n5388) );
OAI31D0 U6136 ( .A1(n6914), .A2(n6794), .A3(n5388), .B(n8899)
   , .ZN(n5389) );
OAI211D0 U6137 ( .A1(n5391), .A2(n8935), .B(n5390), .C(n5389)
   , .ZN(n5392) );
AOI22D0 U6138 ( .A1(n8639), .A2(n5393), .B1(n8656), .B2(n5392
   ), .ZN(n5412)
       );
CKND2D0 U6139 ( .A1(n6737), .A2(n6857), .ZN(n6896) );
INVD0 U6140 ( .I(n6896), .ZN(n6886) );
NR2D0 U6141 ( .A1(counter[6]), .A2(n6488), .ZN(n5756) );
NR2D0 U6142 ( .A1(n8596), .A2(n8782), .ZN(n6869) );
CKND2D0 U6143 ( .A1(n6869), .A2(n8882), .ZN(n8629) );
INVD0 U6144 ( .I(n6890), .ZN(n5394) );
NR2D0 U6145 ( .A1(n8629), .A2(n5394), .ZN(n6880) );
NR2D0 U6146 ( .A1(n5649), .A2(n6793), .ZN(n6856) );
ND4D0 U6147 ( .A1(n6716), .A2(n6880), .A3(n6856), .A4(n6847),
    .ZN(n5395) );
NR4D0 U6148 ( .A1(n8509), .A2(n5756), .A3(n8545), .A4(n5395),
    .ZN(n5396) );
AOI31D0 U6149 ( .A1(n6886), .A2(n6727), .A3(n5396), .B(n8919)
   , .ZN(n5410) );
CKAN2D0 U6150 ( .A1(n8517), .A2(n8512), .Z(n6725) );
NR2D0 U6151 ( .A1(n6340), .A2(n5397), .ZN(n5399) );
CKND2D0 U6152 ( .A1(n8703), .A2(n6774), .ZN(n5485) );
CKND2D0 U6153 ( .A1(n7332), .A2(n5932), .ZN(n5902) );
IND4D0 U6154 ( .A1(n6914), .B1(n6894), .B2(n5802), .B3(n5902)
   , .ZN(n5398) );
NR4D0 U6155 ( .A1(n6819), .A2(n5399), .A3(n5485), .A4(n5398),
    .ZN(n5400) );
AOI31D0 U6156 ( .A1(n6858), .A2(n6725), .A3(n5400), .B(n8923)
   , .ZN(n5409) );
NR2D0 U6157 ( .A1(n8768), .A2(n6824), .ZN(n5403) );
```

```
INVD0 U6158 ( .I(n8629), .ZN(n6743) );
CKND2D0 U6159 ( .A1(n8691), .A2(n6743), .ZN(n5447) );
INVD0 U6160 ( .I(n5447), .ZN(n5420) );
ND3D0 U6161 ( .A1(n5414), .A2(n6954), .A3(n5420), .ZN(n6859)
   );
NR2D0 U6162 ( .A1(n8648), .A2(n4819), .ZN(n8506) );
INVD0 U6163 ( .I(n8506), .ZN(n5555) );
NR2D0 U6164 ( .A1(n5555), .A2(n6925), .ZN(n6789) );
NR2D0 U6165 ( .A1(n8538), .A2(n6819), .ZN(n5401) );
ND4D0 U6166 ( .A1(n6789), .A2(n5401), .A3(n5758), .A4(n6715),
    .ZN(n5402) );
NR4D0 U6167 ( .A1(n5403), .A2(n6859), .A3(n8668), .A4(n5402),
    .ZN(n5407) );
INVD0 U6168 ( .I(n8899), .ZN(n8937) );
CKND2D0 U6169 ( .A1(n8396), .A2(n8045), .ZN(n5505) );
CKND2D0 U6170 ( .A1(n6867), .A2(n8882), .ZN(n6826) );
IINR4D0 U6171 ( .A1(n5505), .A2(n5405), .B1(n6826), .B2(n5404
   ), .ZN(n5406)
      );
OAI22D0 U6172 ( .A1(n5407), .A2(n8937), .B1(n5406), .B2(n8935
   ), .ZN(n5408)
      );
OAI31D0 U6173 ( .A1(n5410), .A2(n5409), .A3(n5408), .B(n7694)
   , .ZN(n5411) );
OAI211D0 U6174 ( .A1(n5413), .A2(n7593), .B(n5412), .C(n5411)
   , .ZN(n5575) );
CKND2D0 U6175 ( .A1(n8721), .A2(n5576), .ZN(n5517) );
INVD0 U6176 ( .I(n5517), .ZN(n6962) );
NR2D0 U6177 ( .A1(n8551), .A2(n8828), .ZN(n6799) );
ND4D0 U6178 ( .A1(n5414), .A2(n6962), .A3(n6799), .A4(n6908),
    .ZN(n5419) );
CKND2D0 U6179 ( .A1(n6786), .A2(n8601), .ZN(n6951) );
NR2D0 U6180 ( .A1(n6862), .A2(n6951), .ZN(n6849) );
CKND2D0 U6181 ( .A1(n5415), .A2(n7170), .ZN(n8576) );
ND4D0 U6182 ( .A1(n5626), .A2(n6849), .A3(n6835), .A4(n8576),
    .ZN(n5418) );
CKND2D0 U6183 ( .A1(n8518), .A2(n6908), .ZN(n5440) );
NR4D0 U6184 ( .A1(n8782), .A2(n8735), .A3(n6903), .A4(n5440),
    .ZN(n5416) );
ND3D0 U6185 ( .A1(n5416), .A2(n8724), .A3(n6847), .ZN(n5417)
   );
AOI222D0 U6186 ( .A1(n5419), .A2(n8685), .B1(n5418), .B2(
   n8899), .C1(n5417),
      .C2(n8697), .ZN(n5424) );
CKND2D0 U6187 ( .A1(n8882), .A2(n6847), .ZN(n6805) );
NR2D0 U6188 ( .A1(n6793), .A2(n6805), .ZN(n5552) );
ND4D0 U6189 ( .A1(n6738), .A2(n6849), .A3(n5552), .A4(n7618),
    .ZN(n5422) );
```

```
  NR2D0 U6190 ( .A1(n8551), .A2(n8632), .ZN(n6788) );
  ND4D0 U6191 ( .A1(n6788), .A2(n5420), .A3(n6717), .A4(n8514),
      .ZN(n5421) );
  AOI22D0 U6192 ( .A1(n8899), .A2(n5422), .B1(n8697), .B2(n5421
     ), .ZN(n5423)
        );
  OAI22D0 U6193 ( .A1(n5424), .A2(n6986), .B1(n5423), .B2(n8903
     ), .ZN(n5574)
        );
  CKND2D0 U6194 ( .A1(n8882), .A2(n5576), .ZN(n6910) );
  INVD0 U6195 ( .I(n6910), .ZN(n6761) );
  ND4D0 U6196 ( .A1(n6763), .A2(n6858), .A3(n6761), .A4(n5425),
      .ZN(n5432) );
  CKND2D0 U6197 ( .A1(n8156), .A2(n8902), .ZN(n8520) );
  INVD0 U6198 ( .I(n8520), .ZN(n8869) );
  INVD0 U6199 ( .I(n6753), .ZN(n5531) );
  NR2D0 U6200 ( .A1(n8869), .A2(n5531), .ZN(n6773) );
  NR2D0 U6201 ( .A1(n6805), .A2(n6914), .ZN(n5496) );
  AOI31D0 U6202 ( .A1(n6773), .A2(n6823), .A3(n5496), .B(n8935)
     , .ZN(n5431) );
  NR2D0 U6203 ( .A1(n8551), .A2(n8620), .ZN(n8676) );
  CKND2D0 U6204 ( .A1(n6786), .A2(n8836), .ZN(n8553) );
  INVD0 U6205 ( .I(n8553), .ZN(n5426) );
  NR2D0 U6206 ( .A1(n8860), .A2(n7870), .ZN(n6742) );
  ND4D0 U6207 ( .A1(n5426), .A2(n8518), .A3(n6742), .A4(n6868),
      .ZN(n5427) );
  OAI31D0 U6208 ( .A1(n6805), .A2(n6793), .A3(n5427), .B(n8697)
     , .ZN(n5429) );
  CKND2D0 U6209 ( .A1(n8617), .A2(n6776), .ZN(n5622) );
  CKND2D0 U6210 ( .A1(n6724), .A2(n8521), .ZN(n5719) );
  NR4D0 U6211 ( .A1(n8733), .A2(n5462), .A3(n5622), .A4(n5719),
      .ZN(n5428) );
  AOI32D0 U6212 ( .A1(n8676), .A2(n5429), .A3(n5428), .B1(n8674
     ), .B2(n5429),
        .ZN(n5430) );
  AOI211D0 U6213 ( .A1(n8822), .A2(n5432), .B(n5431), .C(n5430)
     , .ZN(n5460) );
  CKND2D0 U6214 ( .A1(n6775), .A2(n6717), .ZN(n6933) );
  INVD0 U6215 ( .I(n6925), .ZN(n5507) );
  ND4D0 U6216 ( .A1(n6743), .A2(n5433), .A3(n8617), .A4(n5507),
      .ZN(n5434) );
  NR4D0 U6217 ( .A1(n8668), .A2(n6915), .A3(n6933), .A4(n5434),
      .ZN(n5438) );
  INVD0 U6218 ( .I(n6776), .ZN(n6863) );
  NR2D0 U6219 ( .A1(n7870), .A2(n6863), .ZN(n5461) );
  NR3D0 U6220 ( .A1(n8878), .A2(n6732), .A3(n6970), .ZN(n5554)
     );
```

```
IINR4D0 U6221 ( .A1(n5461), .A2(n5554), .B1(n8629), .B2(n6794
   ), .ZN(n5437)
      );
NR2D0 U6222 ( .A1(n8831), .A2(n8505), .ZN(n5542) );
ND4D0 U6223 ( .A1(n6851), .A2(n5542), .A3(n8617), .A4(n6717),
    .ZN(n5435) );
NR2D0 U6224 ( .A1(n8598), .A2(n5435), .ZN(n5436) );
OAI222D0 U6225 ( .A1(n8919), .A2(n5438), .B1(n8923), .B2(
   n5437), .C1(n8935),
      .C2(n5436), .ZN(n5446) );
NR4D0 U6226 ( .A1(n8505), .A2(n8735), .A3(n5439), .A4(n6720),
    .ZN(n5444) );
OAI31D0 U6227 ( .A1(n5517), .A2(n6802), .A3(n5440), .B(n8697)
   , .ZN(n5443) );
ND4D0 U6228 ( .A1(n6752), .A2(n8703), .A3(n6908), .A4(n8519),
    .ZN(n5441) );
OAI31D0 U6229 ( .A1(n8753), .A2(n7870), .A3(n5441), .B(n8899)
   , .ZN(n5442) );
OAI211D0 U6230 ( .A1(n5444), .A2(n8919), .B(n5443), .C(n5442)
   , .ZN(n5445) );
AOI22D0 U6231 ( .A1(n8408), .A2(n5446), .B1(n7933), .B2(n5445
   ), .ZN(n5459)
      );
IND4D0 U6232 ( .A1(n5649), .B1(n8676), .B2(n8518), .B3(n6908)
   , .ZN(n5534) );
NR4D0 U6233 ( .A1(n8753), .A2(n6863), .A3(n6805), .A4(n5534),
    .ZN(n5451) );
CKND2D0 U6234 ( .A1(n6741), .A2(n6927), .ZN(n6909) );
ND4D0 U6235 ( .A1(n6788), .A2(n6777), .A3(n6726), .A4(n8601),
    .ZN(n5449) );
NR2D0 U6236 ( .A1(n8055), .A2(counter[11]), .ZN(n8010) );
INVD0 U6237 ( .I(n8010), .ZN(n5937) );
OAI211D0 U6238 ( .A1(n8441), .A2(n5937), .B(n5699), .C(n8576)
   , .ZN(n5481) );
OA31D0 U6239 ( .A1(n6792), .A2(n5447), .A3(n5481), .B(n8939),
    .Z(n5448) );
AOI221D0 U6240 ( .A1(n6909), .A2(n8715), .B1(n5449), .B2(
   n8715), .C(n5448),
      .ZN(n5450) );
NR2D0 U6241 ( .A1(n9352), .A2(n8055), .ZN(n7396) );
INVD0 U6242 ( .I(n7396), .ZN(n7309) );
NR2D0 U6243 ( .A1(n7309), .A2(n6512), .ZN(n8515) );
INVD0 U6244 ( .I(n8515), .ZN(n5680) );
AOI32D0 U6245 ( .A1(n5451), .A2(n5450), .A3(n5680), .B1(n8903
   ), .B2(n5450),
      .ZN(n5457) );
CKND2D0 U6246 ( .A1(n8716), .A2(n5699), .ZN(n5774) );
INVD0 U6247 ( .I(n5774), .ZN(n5682) );
```

```
CKND2D0 U6248 ( .A1(n8519), .A2(n6857), .ZN(n6934) );
CKND2D0 U6249 ( .A1(n8311), .A2(n5452), .ZN(n8788) );
OAI211D0 U6250 ( .A1(n6512), .A2(n5801), .B(n5464), .C(n8788)
   , .ZN(n5453) );
OAI31D0 U6251 ( .A1(n8808), .A2(n6934), .A3(n5453), .B(n8697)
   , .ZN(n5455) );
ND3D0 U6252 ( .A1(n6725), .A2(n6868), .A3(n8699), .ZN(n6801)
   );
NR4D0 U6253 ( .A1(n6793), .A2(n6801), .A3(n6934), .A4(n5543),
    .ZN(n5454) );
AOI32D0 U6254 ( .A1(n5682), .A2(n5455), .A3(n5454), .B1(n8674
   ), .B2(n5455),
      .ZN(n5456) );
AOI22D0 U6255 ( .A1(n8822), .A2(n5457), .B1(n8715), .B2(n5456
   ), .ZN(n5458)
       );
OAI211D0 U6256 ( .A1(n5460), .A2(n6795), .B(n5459), .C(n5458)
   , .ZN(n5573) );
NR2D0 U6257 ( .A1(n8648), .A2(n8748), .ZN(n6778) );
ND4D0 U6258 ( .A1(n6778), .A2(n5461), .A3(n5586), .A4(n5699),
    .ZN(n5471) );
INR4D0 U6259 ( .A1(n5557), .B1(n6732), .B2(n6783), .B3(n5462)
   , .ZN(n5463) );
AOI31D0 U6260 ( .A1(n5464), .A2(n6908), .A3(n5463), .B(n8935)
   , .ZN(n5470) );
NR2D0 U6261 ( .A1(n8648), .A2(n8668), .ZN(n6885) );
OAI211D0 U6262 ( .A1(n5937), .A2(n8441), .B(n6885), .C(n6857)
   , .ZN(n5465) );
NR4D0 U6263 ( .A1(n8733), .A2(n8828), .A3(n6933), .A4(n5465),
    .ZN(n5468) );
ND3D0 U6264 ( .A1(n5708), .A2(n6786), .A3(n5557), .ZN(n5529)
   );
NR2D0 U6265 ( .A1(n8620), .A2(n8782), .ZN(n6888) );
ND4D0 U6266 ( .A1(n6888), .A2(n8813), .A3(n6776), .A4(n8691),
    .ZN(n5466) );
NR4D0 U6267 ( .A1(n6917), .A2(n8664), .A3(n5529), .A4(n5466),
    .ZN(n5467) );
OAI22D0 U6268 ( .A1(n5468), .A2(n8937), .B1(n5467), .B2(n8923
   ), .ZN(n5469)
       );
AOI211D0 U6269 ( .A1(n8822), .A2(n5471), .B(n5470), .C(n5469)
   , .ZN(n5571) );
INVD0 U6270 ( .I(n8037), .ZN(n8708) );
NR4D0 U6271 ( .A1(n8648), .A2(n5472), .A3(n8828), .A4(n5485),
    .ZN(n5480) );
NR4D0 U6272 ( .A1(n8648), .A2(n8778), .A3(n8748), .A4(n8664),
    .ZN(n6751) );
```

```
ND4D0 U6273 ( .A1(n6751), .A2(n6851), .A3(n6761), .A4(n6724),
    .ZN(n5475) );
NR2D0 U6274 ( .A1(n8860), .A2(n6783), .ZN(n6871) );
NR4D0 U6275 ( .A1(n8878), .A2(n6925), .A3(n6893), .A4(n8664),
    .ZN(n5473) );
ND4D0 U6276 ( .A1(n8703), .A2(n6871), .A3(n5473), .A4(n8576),
    .ZN(n5474) );
AOI22D0 U6277 ( .A1(n8685), .A2(n5475), .B1(n8899), .B2(n5474
   ), .ZN(n5479)
      );
INVD0 U6278 ( .I(n6858), .ZN(n6767) );
NR2D0 U6279 ( .A1(n8504), .A2(n6914), .ZN(n6929) );
ND4D0 U6280 ( .A1(n5476), .A2(n6929), .A3(n5680), .A4(n8519),
    .ZN(n5477) );
OAI31D0 U6281 ( .A1(n6826), .A2(n6767), .A3(n5477), .B(n8697)
   , .ZN(n5478) );
OAI211D0 U6282 ( .A1(n5480), .A2(n8919), .B(n5479), .C(n5478)
   , .ZN(n5528) );
INR4D0 U6283 ( .A1(n6741), .B1(n6923), .B2(n5482), .B3(n5481)
   , .ZN(n5488) );
CKND2D0 U6284 ( .A1(n6775), .A2(n8699), .ZN(n5486) );
ND4D0 U6285 ( .A1(n6741), .A2(n6738), .A3(n5483), .A4(n8813),
    .ZN(n5484) );
NR2D0 U6286 ( .A1(n8935), .A2(n6393), .ZN(n8742) );
OAI31D0 U6287 ( .A1(n5486), .A2(n5485), .A3(n5484), .B(n8742)
   , .ZN(n5487) );
OAI31D0 U6288 ( .A1(n5488), .A2(n8937), .A3(n8945), .B(n5487)
   , .ZN(n5527) );
ND3D0 U6289 ( .A1(n6752), .A2(n8556), .A3(n6926), .ZN(n6718)
   );
IND4D0 U6290 ( .A1(n6718), .B1(n5696), .B2(n6971), .B3(n6840)
   , .ZN(n5490) );
NR2D0 U6291 ( .A1(n8551), .A2(n6957), .ZN(n5508) );
NR2D0 U6292 ( .A1(n5665), .A2(n5531), .ZN(n6937) );
AOI31D0 U6293 ( .A1(n5508), .A2(n6937), .A3(n6894), .B(n8903)
   , .ZN(n5489) );
AOI21D0 U6294 ( .A1(n7933), .A2(n5490), .B(n5489), .ZN(n5525)
    );
NR4D0 U6295 ( .A1(n8648), .A2(n5491), .A3(n8565), .A4(n6836),
    .ZN(n5495) );
CKND2D0 U6296 ( .A1(n8765), .A2(n5893), .ZN(n8535) );
NR2D0 U6297 ( .A1(counter[6]), .A2(n8535), .ZN(n8759) );
INVD0 U6298 ( .I(n8759), .ZN(n5514) );
ND3D0 U6299 ( .A1(n6889), .A2(n5621), .A3(n5514), .ZN(n6841)
   );
AOI21D0 U6300 ( .A1(n6813), .A2(n6941), .B(n8772), .ZN(n6953)
    );
```

```
OAI211D0 U6301 ( .A1(n8426), .A2(n5937), .B(n5492), .C(n6953)
   , .ZN(n5493) );
NR4D0 U6302 ( .A1(n6964), .A2(n6841), .A3(n6933), .A4(n5493),
    .ZN(n5494) );
ND4D0 U6303 ( .A1(n5495), .A2(n5494), .A3(n6960), .A4(n6847),
    .ZN(n5522) );
CKND2D0 U6304 ( .A1(n6724), .A2(n5506), .ZN(n5498) );
INVD0 U6305 ( .I(n8545), .ZN(n6944) );
ND4D0 U6306 ( .A1(n6727), .A2(n5496), .A3(n8555), .A4(n6944),
    .ZN(n5497) );
NR4D0 U6307 ( .A1(n8877), .A2(n5499), .A3(n5498), .A4(n5497),
    .ZN(n5503) );
OAI22D0 U6308 ( .A1(n8503), .A2(n6647), .B1(n7310), .B2(n5500
   ), .ZN(n5501)
      );
AOI31D0 U6309 ( .A1(n5503), .A2(n5502), .A3(n5501), .B(n8923)
   , .ZN(n5521) );
AOI211D0 U6310 ( .A1(n5504), .A2(n5766), .B(n8762), .C(n8545)
   , .ZN(n6828) );
ND4D0 U6311 ( .A1(n8521), .A2(n6876), .A3(n5505), .A4(n8555),
    .ZN(n5512) );
ND4D0 U6312 ( .A1(n5508), .A2(n6724), .A3(n5507), .A4(n5506),
    .ZN(n6945) );
NR2D0 U6313 ( .A1(n8596), .A2(n5509), .ZN(n6850) );
CKND2D0 U6314 ( .A1(n8311), .A2(n8010), .ZN(n5510) );
ND4D0 U6315 ( .A1(n6850), .A2(n6828), .A3(n8575), .A4(n5510),
    .ZN(n5511) );
OAI31D0 U6316 ( .A1(n5512), .A2(n6945), .A3(n5511), .B(n8685)
   , .ZN(n5519) );
INVD0 U6317 ( .I(n5513), .ZN(n6966) );
AOI31D0 U6318 ( .A1(counter[11]), .A2(n6942), .A3(n8953), .B(
   n5535), .ZN(
      n5515) );
IND4D0 U6319 ( .A1(n6805), .B1(n6966), .B2(n5515), .B3(n5514)
   , .ZN(n5516) );
NR4D0 U6320 ( .A1(n6836), .A2(n5665), .A3(n5517), .A4(n5516),
    .ZN(n5518) );
AOI32D0 U6321 ( .A1(n6828), .A2(n5519), .A3(n5518), .B1(n8919
   ), .B2(n5519),
      .ZN(n5520) );
AOI211D0 U6322 ( .A1(n8899), .A2(n5522), .B(n5521), .C(n5520)
   , .ZN(n5524) );
OAI22D0 U6323 ( .A1(n5525), .A2(n8935), .B1(n5524), .B2(n5523
   ), .ZN(n5526)
      );
AOI211D0 U6324 ( .A1(n8323), .A2(n5528), .B(n5527), .C(n5526)
   , .ZN(n5570) );
NR2D0 U6325 ( .A1(n6925), .A2(n8733), .ZN(n5539) );
```

```
  CKND2D0 U6326 ( .A1(n6738), .A2(n6737), .ZN(n5530) );
  INR4D0 U6327 ( .A1(n5539), .B1(n8877), .B2(n5530), .B3(n5529)
     , .ZN(n5533) );
  ND4D0 U6328 ( .A1(n6716), .A2(n6737), .A3(n6786), .A4(n5557),
      .ZN(n6803) );
  IINR4D0 U6329 ( .A1(n5542), .A2(n6870), .B1(n5531), .B2(n6803
     ), .ZN(n5532)
        );
  OAI22D0 U6330 ( .A1(n5533), .A2(n8935), .B1(n5532), .B2(n8923
     ), .ZN(n5568)
        );
  INVD0 U6331 ( .I(n6720), .ZN(n6931) );
  OAI31D0 U6332 ( .A1(n8632), .A2(n6855), .A3(n5534), .B(n8609)
     , .ZN(n5538) );
  NR4D0 U6333 ( .A1(n8596), .A2(n5536), .A3(n6913), .A4(n5535),
      .ZN(n5537) );
  AOI32D0 U6334 ( .A1(n6931), .A2(n5538), .A3(n5537), .B1(n8674
     ), .B2(n5538),
        .ZN(n5567) );
  ND4D0 U6335 ( .A1(n6840), .A2(n6773), .A3(n5540), .A4(n5539),
      .ZN(n5550) );
  NR4D0 U6336 ( .A1(n8876), .A2(n5665), .A3(n6794), .A4(n5622),
      .ZN(n5541) );
  AOI31D0 U6337 ( .A1(n6954), .A2(n6725), .A3(n5541), .B(n8919)
     , .ZN(n5549) );
  IND4D0 U6338 ( .A1(n5543), .B1(n8839), .B2(n6840), .B3(n5542)
     , .ZN(n5544) );
  OAI31D0 U6339 ( .A1(n6863), .A2(n6910), .A3(n5544), .B(n8899)
     , .ZN(n5547) );
  CKND2D0 U6340 ( .A1(n8512), .A2(n8519), .ZN(n6895) );
  NR4D0 U6341 ( .A1(n8509), .A2(n5545), .A3(n6804), .A4(n6895),
      .ZN(n5546) );
  AOI32D0 U6342 ( .A1(n8678), .A2(n5547), .A3(n5546), .B1(n8923
     ), .B2(n5547),
        .ZN(n5548) );
  AOI211D0 U6343 ( .A1(n8582), .A2(n5550), .B(n5549), .C(n5548)
     , .ZN(n5565) );
  INR2D0 U6344 ( .A1(n6888), .B1(n6863), .ZN(n5551) );
  ND4D0 U6345 ( .A1(n6753), .A2(n6885), .A3(n5686), .A4(n5551),
      .ZN(n5563) );
  AOI31D0 U6346 ( .A1(n5554), .A2(n5553), .A3(n5552), .B(n8923)
     , .ZN(n5562) );
  NR2D0 U6347 ( .A1(n8620), .A2(n5555), .ZN(n6749) );
  INR4D0 U6348 ( .A1(n6749), .B1(n6887), .B2(n6913), .B3(n6909)
     , .ZN(n5560) );
  AOI31D0 U6349 ( .A1(n6870), .A2(n5557), .A3(n5556), .B(n6970)
     , .ZN(n5558) );
```

```
NR4D0 U6350 ( .A1(n6917), .A2(n6903), .A3(n5774), .A4(n5558),
    .ZN(n5559) );
OAI22D0 U6351 ( .A1(n5560), .A2(n8935), .B1(n5559), .B2(n8919
   ), .ZN(n5561)
      );
AOI211D0 U6352 ( .A1(n8899), .A2(n5563), .B(n5562), .C(n5561)
   , .ZN(n5564) );
OAI22D0 U6353 ( .A1(n5565), .A2(n8169), .B1(n5564), .B2(n8230
   ), .ZN(n5566)
      );
AOI221D0 U6354 ( .A1(n5568), .A2(n7939), .B1(n5567), .B2(
   n7939), .C(n5566),
      .ZN(n5569) );
OAI211D0 U6355 ( .A1(n5571), .A2(n8708), .B(n5570), .C(n5569)
   , .ZN(n5572) );
NR4D0 U6356 ( .A1(n5575), .A2(n5574), .A3(n5573), .A4(n5572),
    .ZN(n5854) );
CKND2D0 U6357 ( .A1(n5576), .A2(n8813), .ZN(n8780) );
NR2D0 U6358 ( .A1(n8780), .A2(n8596), .ZN(n5755) );
CKND2D0 U6359 ( .A1(n5841), .A2(n5755), .ZN(n5700) );
NR2D0 U6360 ( .A1(n8505), .A2(n8553), .ZN(n5666) );
IND4D0 U6361 ( .A1(n5700), .B1(n5577), .B2(n5666), .B3(n8730)
   , .ZN(n5580) );
CKND2D0 U6362 ( .A1(n6847), .A2(n5699), .ZN(n5729) );
NR3D0 U6363 ( .A1(n8755), .A2(n8748), .A3(n6957), .ZN(n5803)
   );
NR2D0 U6364 ( .A1(n8612), .A2(n6558), .ZN(n5742) );
NR2D0 U6365 ( .A1(n5742), .A2(n8632), .ZN(n5606) );
NR2D0 U6366 ( .A1(n8831), .A2(n5578), .ZN(n5747) );
IND4D0 U6367 ( .A1(n5729), .B1(n5803), .B2(n5606), .B3(n5747)
   , .ZN(n5579) );
AOI22D0 U6368 ( .A1(n8582), .A2(n5580), .B1(n8697), .B2(n5579
   ), .ZN(n5583)
      );
NR2D0 U6369 ( .A1(n8877), .A2(n8782), .ZN(n8833) );
INVD0 U6370 ( .I(n8833), .ZN(n5832) );
NR2D0 U6371 ( .A1(n4819), .A2(n5832), .ZN(n8692) );
NR2D0 U6372 ( .A1(n8416), .A2(n8814), .ZN(n5752) );
INVD0 U6373 ( .I(n5752), .ZN(n5638) );
AO31D0 U6374 ( .A1(n8704), .A2(n8692), .A3(n5638), .B(n8937),
    .Z(n5582) );
CKND2D0 U6375 ( .A1(n8517), .A2(n8844), .ZN(n8653) );
INVD0 U6376 ( .I(n8891), .ZN(n8550) );
NR2D0 U6377 ( .A1(n8653), .A2(n8550), .ZN(n8600) );
NR2D0 U6378 ( .A1(n5650), .A2(n5742), .ZN(n8522) );
AO31D0 U6379 ( .A1(n8600), .A2(n8522), .A3(n5747), .B(n8919),
    .Z(n5581) );
```

```
AOI31D0 U6380 ( .A1(n5583), .A2(n5582), .A3(n5581), .B(n6221)
   , .ZN(n5852) );
NR2D0 U6381 ( .A1(n5765), .A2(n8860), .ZN(n8884) );
NR2D0 U6382 ( .A1(n8731), .A2(n5584), .ZN(n5828) );
ND4D0 U6383 ( .A1(n8884), .A2(n6874), .A3(n5666), .A4(n5828),
    .ZN(n5594) );
CKND2D0 U6384 ( .A1(n8522), .A2(n8844), .ZN(n5831) );
NR2D0 U6385 ( .A1(n5719), .A2(n5831), .ZN(n5678) );
CKND2D0 U6386 ( .A1(n5586), .A2(n5585), .ZN(n8682) );
INVD0 U6387 ( .I(n8682), .ZN(n5687) );
CKND2D0 U6388 ( .A1(n5680), .A2(n8512), .ZN(n5662) );
CKND2D0 U6389 ( .A1(n8882), .A2(n6894), .ZN(n5711) );
NR2D0 U6390 ( .A1(n5711), .A2(n4819), .ZN(n5637) );
INVD0 U6391 ( .I(n5637), .ZN(n5596) );
NR4D0 U6392 ( .A1(n8877), .A2(n6732), .A3(n5662), .A4(n5596),
    .ZN(n5587) );
AOI31D0 U6393 ( .A1(n5678), .A2(n5687), .A3(n5587), .B(n8937)
   , .ZN(n5593) );
NR3D0 U6394 ( .A1(n8755), .A2(n5765), .A3(n8631), .ZN(n5818)
   );
NR2D0 U6395 ( .A1(n8632), .A2(n8733), .ZN(n8815) );
CKND2D0 U6396 ( .A1(n8815), .A2(n5841), .ZN(n5709) );
OAI31D0 U6397 ( .A1(n8831), .A2(n5588), .A3(n5709), .B(n8609)
   , .ZN(n5591) );
ND3D0 U6398 ( .A1(n8519), .A2(n5699), .A3(n8920), .ZN(n5619)
   );
INR4D0 U6399 ( .A1(n5808), .B1(n8780), .B2(n5589), .B3(n5619)
   , .ZN(n5590) );
AOI32D0 U6400 ( .A1(n5818), .A2(n5591), .A3(n5590), .B1(n8923
   ), .B2(n5591),
      .ZN(n5592) );
AOI211D0 U6401 ( .A1(n8582), .A2(n5594), .B(n5593), .C(n5592)
   , .ZN(n5636) );
NR2D0 U6402 ( .A1(n6049), .A2(n5595), .ZN(n5653) );
OR3D0 U6403 ( .A1(n5719), .A2(n8503), .A3(n8771), .Z(n5721) )
   ;
NR2D0 U6404 ( .A1(n8596), .A2(n8831), .ZN(n5834) );
IND2D0 U6405 ( .A1(n5662), .B1(n5834), .ZN(n5836) );
NR4D0 U6406 ( .A1(n6732), .A2(n5653), .A3(n5721), .A4(n5836),
    .ZN(n5605) );
CKND2D0 U6407 ( .A1(n8882), .A2(n6926), .ZN(n5786) );
CKND2D0 U6408 ( .A1(n8624), .A2(n6775), .ZN(n5835) );
IND4D0 U6409 ( .A1(n5835), .B1(n8723), .B2(n5606), .B3(n8920)
   , .ZN(n5600) );
ND3D0 U6410 ( .A1(n6737), .A2(n8666), .A3(n5638), .ZN(n5726)
   );
NR4D0 U6411 ( .A1(n5742), .A2(n5724), .A3(n5726), .A4(n5596),
    .ZN(n5597) );
```

```
AOI21D0 U6412 ( .A1(n5598), .A2(n5597), .B(n8923), .ZN(n5599)
    );
AOI221D0 U6413 ( .A1(n5786), .A2(n8582), .B1(n5600), .B2(
   n8685), .C(n5599),
      .ZN(n5604) );
CKND2D0 U6414 ( .A1(n5576), .A2(n5626), .ZN(n5624) );
INVD0 U6415 ( .I(n5624), .ZN(n5694) );
INVD0 U6416 ( .I(n5601), .ZN(n6263) );
NR2D0 U6417 ( .A1(n5650), .A2(n6263), .ZN(n5685) );
ND4D0 U6418 ( .A1(n8517), .A2(n5694), .A3(n5685), .A4(n8521),
    .ZN(n5602) );
OAI31D0 U6419 ( .A1(n5619), .A2(n6909), .A3(n5602), .B(n8899)
   , .ZN(n5603) );
OAI211D0 U6420 ( .A1(n5605), .A2(n8919), .B(n5604), .C(n5603)
   , .ZN(n5615) );
ND3D0 U6421 ( .A1(n6786), .A2(n8512), .A3(n6847), .ZN(n5727)
   );
NR2D0 U6422 ( .A1(n8828), .A2(n5727), .ZN(n5743) );
ND4D0 U6423 ( .A1(n8833), .A2(n5743), .A3(n5747), .A4(n6724),
    .ZN(n5611) );
INVD0 U6424 ( .I(n5836), .ZN(n5607) );
AOI31D0 U6425 ( .A1(n5607), .A2(n5606), .A3(n6013), .B(n8923)
   , .ZN(n5610) );
CKND2D0 U6426 ( .A1(n8813), .A2(n6926), .ZN(n8887) );
ND3D0 U6427 ( .A1(n5666), .A2(n5685), .A3(n8702), .ZN(n5710)
   );
NR4D0 U6428 ( .A1(n8753), .A2(n7587), .A3(n8887), .A4(n5710),
    .ZN(n5608) );
AOI21D0 U6429 ( .A1(n5608), .A2(n8920), .B(n8919), .ZN(n5609)
    );
AOI211D0 U6430 ( .A1(n8899), .A2(n5611), .B(n5610), .C(n5609)
   , .ZN(n5613) );
NR4D0 U6431 ( .A1(n8877), .A2(n8620), .A3(n5649), .A4(n5662),
    .ZN(n5612) );
AOI32D0 U6432 ( .A1(n5818), .A2(n5613), .A3(n5612), .B1(n8935
   ), .B2(n5613),
      .ZN(n5614) );
AOI22D0 U6433 ( .A1(n7939), .A2(n5615), .B1(n8895), .B2(n5614
   ), .ZN(n5635)
       );
CKND2D0 U6434 ( .A1(n8617), .A2(n8702), .ZN(n8643) );
CKND2D0 U6435 ( .A1(n7284), .A2(n7785), .ZN(n6848) );
CKND2D0 U6436 ( .A1(n8691), .A2(n6848), .ZN(n5781) );
CKND2D0 U6437 ( .A1(n8794), .A2(n5682), .ZN(n5616) );
NR4D0 U6438 ( .A1(n8643), .A2(n8590), .A3(n5781), .A4(n5616),
    .ZN(n5618) );
AOI31D0 U6439 ( .A1(n5618), .A2(n5680), .A3(n5617), .B(n8919)
   , .ZN(n5633) );
```

```
INR4D0 U6440 ( .A1(n6874), .B1(n8596), .B2(n8831), .B3(n5619)
   , .ZN(n5712) );
AOI31D0 U6441 ( .A1(n8891), .A2(n5712), .A3(n6847), .B(n8674)
   , .ZN(n5632) );
NR2D0 U6442 ( .A1(n8769), .A2(n8768), .ZN(n5625) );
IND4D0 U6443 ( .A1(n5622), .B1(n5708), .B2(n5621), .B3(n5620)
   , .ZN(n5623) );
NR4D0 U6444 ( .A1(n5809), .A2(n5625), .A3(n5624), .A4(n5623),
    .ZN(n5630) );
INVD0 U6445 ( .I(n5755), .ZN(n5689) );
ND3D0 U6446 ( .A1(n5626), .A2(n8730), .A3(n8514), .ZN(n5640)
   );
NR2D0 U6447 ( .A1(n5742), .A2(n8748), .ZN(n6739) );
CKND2D0 U6448 ( .A1(n5776), .A2(n6739), .ZN(n5628) );
ND4D0 U6449 ( .A1(n8716), .A2(n8836), .A3(n8571), .A4(n5802),
    .ZN(n5627) );
NR4D0 U6450 ( .A1(n5689), .A2(n5640), .A3(n5628), .A4(n5627),
    .ZN(n5629) );
OAI22D0 U6451 ( .A1(n5630), .A2(n8935), .B1(n5629), .B2(n8923
   ), .ZN(n5631)
      );
OAI31D0 U6452 ( .A1(n5633), .A2(n5632), .A3(n5631), .B(n7694)
   , .ZN(n5634) );
OAI211D0 U6453 ( .A1(n5636), .A2(n6986), .B(n5635), .C(n5634)
   , .ZN(n5851) );
CKND2D0 U6454 ( .A1(n6857), .A2(n8920), .ZN(n8754) );
INVD0 U6455 ( .I(n8754), .ZN(n8856) );
ND4D0 U6456 ( .A1(n8856), .A2(n5747), .A3(n5637), .A4(n8512),
    .ZN(n5645) );
CKND2D0 U6457 ( .A1(n8692), .A2(n8883), .ZN(n5838) );
NR2D0 U6458 ( .A1(n8682), .A2(n5838), .ZN(n5706) );
AOI31D0 U6459 ( .A1(n5706), .A2(n5682), .A3(n5638), .B(n8935)
   , .ZN(n5644) );
INVD0 U6460 ( .I(n8854), .ZN(n5728) );
CKND2D0 U6461 ( .A1(n8514), .A2(n8883), .ZN(n8852) );
NR4D0 U6462 ( .A1(n6957), .A2(n5728), .A3(n8852), .A4(n5721),
    .ZN(n5642) );
NR2D0 U6463 ( .A1(n5639), .A2(n7285), .ZN(n8868) );
NR4D0 U6464 ( .A1(n8782), .A2(n8868), .A3(n5752), .A4(n5640),
    .ZN(n5641) );
OAI22D0 U6465 ( .A1(n5642), .A2(n8923), .B1(n5641), .B2(n8937
   ), .ZN(n5643)
      );
AOI211D0 U6466 ( .A1(n8609), .A2(n5645), .B(n5644), .C(n5643)
   , .ZN(n5676) );
CKND2D0 U6467 ( .A1(n8053), .A2(n5646), .ZN(n5857) );
INVD0 U6468 ( .I(n5857), .ZN(n5741) );
```

```
NR3D0 U6469 ( .A1(n8878), .A2(n5741), .A3(n5647), .ZN(n5648)
   );
AOI31D0 U6470 ( .A1(n6874), .A2(n8815), .A3(n5648), .B(n8919)
   , .ZN(n5659) );
INR2D0 U6471 ( .A1(n5666), .B1(n8744), .ZN(n5693) );
NR4D0 U6472 ( .A1(n5650), .A2(n5649), .A3(n8572), .A4(n5838),
    .ZN(n5651) );
AOI31D0 U6473 ( .A1(n8624), .A2(n5693), .A3(n5651), .B(n8923)
   , .ZN(n5658) );
ND4D0 U6474 ( .A1(n5808), .A2(n5685), .A3(n5576), .A4(n8920),
    .ZN(n5652) );
NR2D0 U6475 ( .A1(n8878), .A2(n6957), .ZN(n5713) );
INVD0 U6476 ( .I(n5713), .ZN(n5723) );
OAI31D0 U6477 ( .A1(n5653), .A2(n5652), .A3(n5723), .B(n8685)
   , .ZN(n5656) );
AOI22D0 U6478 ( .A1(counter[1]), .A2(n8441), .B1(n8612), .B2(
   n6205), .ZN(
      n8546) );
AOI211D0 U6479 ( .A1(n8546), .A2(n5654), .B(n5689), .C(n5727)
   , .ZN(n5655) );
AOI32D0 U6480 ( .A1(n5747), .A2(n5656), .A3(n5655), .B1(n8674
   ), .B2(n5656),
      .ZN(n5657) );
OAI31D0 U6481 ( .A1(n5659), .A2(n5658), .A3(n5657), .B(n8323)
   , .ZN(n5675) );
CKND2D0 U6482 ( .A1(n8521), .A2(n8789), .ZN(n8888) );
INVD0 U6483 ( .I(n8888), .ZN(n8809) );
CKND2D0 U6484 ( .A1(n8416), .A2(n6592), .ZN(n5660) );
AOI211D0 U6485 ( .A1(n7815), .A2(n5660), .B(n8744), .C(n5781)
   , .ZN(n5661) );
AOI31D0 U6486 ( .A1(n8833), .A2(n8809), .A3(n5661), .B(n8923)
   , .ZN(n5673) );
NR4D0 U6487 ( .A1(n8509), .A2(n6957), .A3(n5831), .A4(n5662),
    .ZN(n5663) );
AOI31D0 U6488 ( .A1(n5755), .A2(n6740), .A3(n5663), .B(n8919)
   , .ZN(n5672) );
CKND2D0 U6489 ( .A1(n8721), .A2(n8724), .ZN(n5664) );
AOI211D0 U6490 ( .A1(n5665), .A2(n8950), .B(n8780), .C(n5664)
   , .ZN(n5670) );
CKND2D0 U6491 ( .A1(n6927), .A2(n5666), .ZN(n5722) );
NR4D0 U6492 ( .A1(n5765), .A2(n8877), .A3(n8748), .A4(n8887),
    .ZN(n5732) );
IND4D0 U6493 ( .A1(n5722), .B1(n8794), .B2(n5682), .B3(n5732)
   , .ZN(n5667) );
NR4D0 U6494 ( .A1(n8596), .A2(n5668), .A3(n5742), .A4(n5667),
    .ZN(n5669) );
OAI22D0 U6495 ( .A1(n5670), .A2(n8935), .B1(n5669), .B2(n8937
   ), .ZN(n5671)
```

```
        );
OAI31D0 U6496 ( .A1(n5673), .A2(n5672), .A3(n5671), .B(n8408)
   , .ZN(n5674) );
OAI211D0 U6497 ( .A1(n5676), .A2(n8662), .B(n5675), .C(n5674)
   , .ZN(n5850) );
NR4D0 U6498 ( .A1(n8551), .A2(n8515), .A3(n8868), .A4(n5831),
    .ZN(n5692) );
NR2D0 U6499 ( .A1(n4819), .A2(n8782), .ZN(n5679) );
ND4D0 U6500 ( .A1(n5679), .A2(n5678), .A3(n5677), .A4(n8556),
    .ZN(n5684) );
CKND2D0 U6501 ( .A1(n5680), .A2(n8920), .ZN(n5796) );
NR4D0 U6502 ( .A1(n8509), .A2(n8632), .A3(n8650), .A4(n5796),
    .ZN(n5681) );
AOI31D0 U6503 ( .A1(n5682), .A2(n5808), .A3(n5681), .B(n8937)
   , .ZN(n5683) );
AOI221D0 U6504 ( .A1(n5722), .A2(n8685), .B1(n5684), .B2(
   n8685), .C(n5683),
      .ZN(n5691) );
ND4D0 U6505 ( .A1(n5687), .A2(n6738), .A3(n5686), .A4(n5685),
    .ZN(n5688) );
OAI31D0 U6506 ( .A1(n5786), .A2(n5689), .A3(n5688), .B(n8697)
   , .ZN(n5690) );
OAI211D0 U6507 ( .A1(n5692), .A2(n8919), .B(n5691), .C(n5690)
   , .ZN(n5740) );
ND4D0 U6508 ( .A1(n8815), .A2(n5694), .A3(n5693), .A4(n6926),
    .ZN(n5698) );
INR2D0 U6509 ( .A1(n6013), .B1(n8778), .ZN(n6750) );
NR2D0 U6510 ( .A1(n8515), .A2(n8648), .ZN(n5695) );
INVD0 U6511 ( .I(n5781), .ZN(n5829) );
ND4D0 U6512 ( .A1(n5696), .A2(n6750), .A3(n5695), .A4(n5829),
    .ZN(n5697) );
AOI22D0 U6513 ( .A1(n8685), .A2(n5698), .B1(n8899), .B2(n5697
   ), .ZN(n5705)
      );
ND4D0 U6514 ( .A1(n6857), .A2(n5699), .A3(n5857), .A4(n7668),
    .ZN(n5701) );
OAI31D0 U6515 ( .A1(n5701), .A2(n8744), .A3(n5700), .B(n8697)
   , .ZN(n5704) );
IND4D0 U6516 ( .A1(n5727), .B1(n6738), .B2(n5713), .B3(n6737)
   , .ZN(n5702) );
OAI31D0 U6517 ( .A1(n8877), .A2(n5774), .A3(n5702), .B(n8609)
   , .ZN(n5703) );
AOI31D0 U6518 ( .A1(n5705), .A2(n5704), .A3(n5703), .B(n6795)
   , .ZN(n5739) );
ND4D0 U6519 ( .A1(n6750), .A2(n5706), .A3(n5803), .A4(n6927),
    .ZN(n5718) );
CKND2D0 U6520 ( .A1(n8666), .A2(n6848), .ZN(n5744) );
NR2D0 U6521 ( .A1(n5831), .A2(n5744), .ZN(n5707) );
```

```
AOI31D0 U6522 ( .A1(n5708), .A2(n5808), .A3(n5707), .B(n8923)
   , .ZN(n5717) );
NR4D0 U6523 ( .A1(n8648), .A2(n5710), .A3(n5709), .A4(n5711),
    .ZN(n5715) );
NR2D0 U6524 ( .A1(n5711), .A2(n8771), .ZN(n5799) );
AN4D0 U6525 ( .A1(n5713), .A2(n5829), .A3(n5712), .A4(n5799),
    .Z(n5714) );
OAI22D0 U6526 ( .A1(n5715), .A2(n8935), .B1(n5714), .B2(n8919
   ), .ZN(n5716)
      );
AOI211D0 U6527 ( .A1(n8899), .A2(n5718), .B(n5717), .C(n5716)
   , .ZN(n5737) );
NR2D0 U6528 ( .A1(n5774), .A2(n5719), .ZN(n5830) );
IND4D0 U6529 ( .A1(n5786), .B1(n8699), .B2(n5720), .B3(n5830)
   , .ZN(n5735) );
NR4D0 U6530 ( .A1(n5724), .A2(n5723), .A3(n5722), .A4(n5721),
    .ZN(n5725) );
AOI31D0 U6531 ( .A1(n5841), .A2(n8833), .A3(n5725), .B(n8674)
   , .ZN(n5734) );
CKND2D0 U6532 ( .A1(n5576), .A2(n8601), .ZN(n8649) );
OAI31D0 U6533 ( .A1(n5727), .A2(n8649), .A3(n5726), .B(n8697)
   , .ZN(n5731) );
NR2D0 U6534 ( .A1(n8778), .A2(n5728), .ZN(n8705) );
INR4D0 U6535 ( .A1(n8705), .B1(n8828), .B2(n5831), .B3(n5729)
   , .ZN(n5730) );
AOI32D0 U6536 ( .A1(n5732), .A2(n5731), .A3(n5730), .B1(n8935
   ), .B2(n5731),
      .ZN(n5733) );
AOI211D0 U6537 ( .A1(n8609), .A2(n5735), .B(n5734), .C(n5733)
   , .ZN(n5736) );
OAI22D0 U6538 ( .A1(n5737), .A2(n8169), .B1(n5736), .B2(n6393
   ), .ZN(n5738)
      );
AOI211D0 U6539 ( .A1(n8910), .A2(n5740), .B(n5739), .C(n5738)
   , .ZN(n5848) );
NR3D0 U6540 ( .A1(n5742), .A2(n5741), .A3(n8888), .ZN(n5779)
   );
IND4D0 U6541 ( .A1(n8590), .B1(n6740), .B2(n5743), .B3(n5779)
   , .ZN(n5750) );
CKND2D0 U6542 ( .A1(n8721), .A2(n8856), .ZN(n8684) );
NR2D0 U6543 ( .A1(n8553), .A2(n8684), .ZN(n8752) );
AOI31D0 U6544 ( .A1(n5808), .A2(n8752), .A3(n5829), .B(n8919)
   , .ZN(n5749) );
CKND2D0 U6545 ( .A1(n6738), .A2(n6717), .ZN(n5745) );
NR4D0 U6546 ( .A1(n8672), .A2(n8620), .A3(n5745), .A4(n5744),
    .ZN(n5746) );
AOI31D0 U6547 ( .A1(n5747), .A2(n8692), .A3(n5746), .B(n8923)
   , .ZN(n5748) );
```

```
AOI211D0 U6548 ( .A1(n8582), .A2(n5750), .B(n5749), .C(n5748)
   , .ZN(n5754) );
NR4D0 U6549 ( .A1(n8508), .A2(n5752), .A3(n5751), .A4(n6767),
    .ZN(n5753) );
AOI32D0 U6550 ( .A1(n5755), .A2(n5754), .A3(n5753), .B1(n8674
   ), .B2(n5754),
      .ZN(n5792) );
INR2D0 U6551 ( .A1(n8557), .B1(n5756), .ZN(n8767) );
CKND2D0 U6552 ( .A1(n8767), .A2(n8555), .ZN(n8591) );
INVD0 U6553 ( .I(n5757), .ZN(n5806) );
CKND2D0 U6554 ( .A1(n6952), .A2(n5758), .ZN(n5814) );
NR2D0 U6555 ( .A1(n5806), .A2(n5814), .ZN(n5760) );
NR2D0 U6556 ( .A1(n8110), .A2(n6361), .ZN(n5816) );
NR2D0 U6557 ( .A1(n5816), .A2(n5770), .ZN(n8564) );
AOI211D0 U6558 ( .A1(n5762), .A2(n5817), .B(n8564), .C(n6818)
   , .ZN(n5759) );
ND4D0 U6559 ( .A1(n5760), .A2(n5759), .A3(n6717), .A4(n8535),
    .ZN(n5761) );
INR4D0 U6560 ( .A1(n5830), .B1(n8591), .B2(n8780), .B3(n5761)
   , .ZN(n5790) );
INVD0 U6561 ( .I(n5762), .ZN(n8534) );
NR3D0 U6562 ( .A1(n8396), .A2(n6248), .A3(n8534), .ZN(n5782)
   );
CKND2D0 U6563 ( .A1(n8521), .A2(n6952), .ZN(n5764) );
CKND2D0 U6564 ( .A1(n8836), .A2(n8571), .ZN(n5763) );
NR4D0 U6565 ( .A1(n5765), .A2(n5782), .A3(n5764), .A4(n5763),
    .ZN(n5769) );
INVD0 U6566 ( .I(n5766), .ZN(n7421) );
NR2D0 U6567 ( .A1(n7421), .A2(n5956), .ZN(n5771) );
OAI22D0 U6568 ( .A1(n5767), .A2(n5810), .B1(n6361), .B2(n5771
   ), .ZN(n5768)
      );
ND4D0 U6569 ( .A1(n5799), .A2(n8703), .A3(n5769), .A4(n5768),
    .ZN(n5778) );
INVD0 U6570 ( .I(n8684), .ZN(n8652) );
ND3D0 U6571 ( .A1(n8652), .A2(n8555), .A3(n5802), .ZN(n5780)
   );
CKND2D0 U6572 ( .A1(n5801), .A2(n5770), .ZN(n5793) );
NR2D0 U6573 ( .A1(n6820), .A2(n5771), .ZN(n5815) );
OAI31D0 U6574 ( .A1(n9329), .A2(n7282), .A3(n5772), .B(n5815)
   , .ZN(n5773) );
AOI211D0 U6575 ( .A1(n5913), .A2(n5793), .B(n5774), .C(n5773)
   , .ZN(n5775) );
IND4D0 U6576 ( .A1(n5780), .B1(n5776), .B2(n6739), .B3(n5775)
   , .ZN(n5777) );
AOI22D0 U6577 ( .A1(n8899), .A2(n5778), .B1(n8697), .B2(n5777
   ), .ZN(n5789)
      );
```

```
INVD0 U6578 ( .I(n5779), .ZN(n5787) );
NR4D0 U6579 ( .A1(n6731), .A2(n5782), .A3(n5781), .A4(n5780),
    .ZN(n5783) );
OAI211D0 U6580 ( .A1(n8038), .A2(n5784), .B(n5783), .C(n8561)
   , .ZN(n5785) );
OAI31D0 U6581 ( .A1(n5787), .A2(n5786), .A3(n5785), .B(n8609)
   , .ZN(n5788) );
OAI211D0 U6582 ( .A1(n5790), .A2(n8935), .B(n5789), .C(n5788)
   , .ZN(n5791) );
AOI22D0 U6583 ( .A1(n8037), .A2(n5792), .B1(n8639), .B2(n5791
   ), .ZN(n5847)
       );
AOI211D0 U6584 ( .A1(n5794), .A2(n5793), .B(n7587), .C(n8545)
   , .ZN(n5800) );
NR2D0 U6585 ( .A1(n6592), .A2(n6578), .ZN(n5797) );
NR3D0 U6586 ( .A1(n5795), .A2(n6969), .A3(n5814), .ZN(n8541)
   );
INVD0 U6587 ( .I(n8541), .ZN(n8562) );
NR4D0 U6588 ( .A1(n5797), .A2(n8562), .A3(n5796), .A4(n5835),
    .ZN(n5798) );
AOI31D0 U6589 ( .A1(n5800), .A2(n5799), .A3(n5798), .B(n8923)
   , .ZN(n5827) );
NR2D0 U6590 ( .A1(n5801), .A2(n6512), .ZN(n5805) );
ND4D0 U6591 ( .A1(n8767), .A2(n5803), .A3(n8570), .A4(n5802),
    .ZN(n5804) );
NR4D0 U6592 ( .A1(n8868), .A2(n5806), .A3(n5805), .A4(n5804),
    .ZN(n5807) );
AOI31D0 U6593 ( .A1(n5807), .A2(n6960), .A3(n8633), .B(n8919)
   , .ZN(n5826) );
CKND2D0 U6594 ( .A1(n8624), .A2(n5808), .ZN(n5813) );
CKND2D0 U6595 ( .A1(n8038), .A2(n6368), .ZN(n8760) );
AOI211D0 U6596 ( .A1(n5810), .A2(n8760), .B(n8648), .C(n5809)
   , .ZN(n5811) );
ND4D0 U6597 ( .A1(n5841), .A2(n5811), .A3(n6847), .A4(n8543),
    .ZN(n5812) );
INR4D0 U6598 ( .A1(n5815), .B1(n5814), .B2(n5813), .B3(n5812)
   , .ZN(n5824) );
OAI31D0 U6599 ( .A1(counter[6]), .A2(n5816), .A3(n8983), .B(
   n8920), .ZN(
      n5821) );
AOI21D0 U6600 ( .A1(counter[6]), .A2(n6947), .B(n5817), .ZN(
   n5819) );
OAI31D0 U6601 ( .A1(n6191), .A2(n5819), .A3(n9329), .B(n5818)
   , .ZN(n5820) );
NR4D0 U6602 ( .A1(n8778), .A2(n5822), .A3(n5821), .A4(n5820),
    .ZN(n5823) );
OAI22D0 U6603 ( .A1(n5824), .A2(n8937), .B1(n5823), .B2(n8935
   ), .ZN(n5825)
```

```
        );
OAI31D0 U6604 ( .A1(n5827), .A2(n5826), .A3(n5825), .B(n8918)
   , .ZN(n5846) );
AOI31D0 U6605 ( .A1(n5830), .A2(n5829), .A3(n5828), .B(n8923)
   , .ZN(n5844) );
NR4D0 U6606 ( .A1(n6957), .A2(n6275), .A3(n5832), .A4(n5831),
    .ZN(n5833) );
AOI31D0 U6607 ( .A1(n5834), .A2(n5833), .A3(n6737), .B(n8919)
   , .ZN(n5843) );
OAI31D0 U6608 ( .A1(n5836), .A2(n8653), .A3(n5835), .B(n8685)
   , .ZN(n5840) );
CKAN2D0 U6609 ( .A1(n5837), .A2(n6639), .Z(n8554) );
NR4D0 U6610 ( .A1(n8554), .A2(n8551), .A3(n8643), .A4(n5838),
    .ZN(n5839) );
AOI32D0 U6611 ( .A1(n5841), .A2(n5840), .A3(n5839), .B1(n8674
   ), .B2(n5840),
      .ZN(n5842) );
OAI31D0 U6612 ( .A1(n5844), .A2(n5843), .A3(n5842), .B(n8105)
   , .ZN(n5845) );
ND4D0 U6613 ( .A1(n5848), .A2(n5847), .A3(n5846), .A4(n5845),
    .ZN(n5849) );
NR4D0 U6614 ( .A1(n5852), .A2(n5851), .A3(n5850), .A4(n5849),
    .ZN(n5853) );
OAI222D0 U6615 ( .A1(n5855), .A2(n7277), .B1(n7000), .B2(
   n5854), .C1(n6992),
      .C2(n5853), .ZN(n4813) );
INVD0 U6616 ( .I(q_out[3]), .ZN(n6511) );
CKND2D0 U6617 ( .A1(n9345), .A2(counter[5]), .ZN(n6151) );
INVD0 U6618 ( .I(n6151), .ZN(n6346) );
INVD0 U6619 ( .I(n6047), .ZN(n8523) );
CKND2D0 U6620 ( .A1(n9348), .A2(n8523), .ZN(n5982) );
CKND2D0 U6621 ( .A1(n9352), .A2(n7714), .ZN(n6155) );
CKND2D0 U6622 ( .A1(n5982), .A2(n6155), .ZN(n6147) );
NR3D0 U6623 ( .A1(n9348), .A2(n8055), .A3(n5856), .ZN(n6019)
   );
INVD0 U6624 ( .I(n6019), .ZN(n6028) );
INVD0 U6625 ( .I(n7326), .ZN(n7301) );
NR2D0 U6626 ( .A1(n7301), .A2(n7744), .ZN(n5945) );
INVD0 U6627 ( .I(n5945), .ZN(n6117) );
INVD0 U6628 ( .I(n7744), .ZN(n6638) );
CKND2D0 U6629 ( .A1(n7332), .A2(n6638), .ZN(n6074) );
ND3D0 U6630 ( .A1(n6028), .A2(n6117), .A3(n6074), .ZN(n6139)
   );
NR2D0 U6631 ( .A1(n9348), .A2(n6047), .ZN(n5905) );
NR2D0 U6632 ( .A1(n7328), .A2(n6618), .ZN(n6030) );
NR2D0 U6633 ( .A1(n5905), .A2(n6030), .ZN(n5925) );
INVD0 U6634 ( .I(n5925), .ZN(n5983) );
CKND2D0 U6635 ( .A1(n9352), .A2(counter[4]), .ZN(n6494) );
```

```
  INVD0 U6636 ( .I(n6494), .ZN(n7542) );
  ND3D0 U6637 ( .A1(counter[2]), .A2(n7542), .A3(n5913), .ZN(
     n6166) );
  CKND2D0 U6638 ( .A1(counter[6]), .A2(n8053), .ZN(n7336) );
  INVD0 U6639 ( .I(n7336), .ZN(n7327) );
  CKND2D0 U6640 ( .A1(n7327), .A2(n6638), .ZN(n6129) );
  CKND2D0 U6641 ( .A1(n6166), .A2(n6129), .ZN(n6120) );
  NR3D0 U6642 ( .A1(n6139), .A2(n5983), .A3(n6120), .ZN(n5967)
     );
  CKND2D0 U6643 ( .A1(n7284), .A2(n5862), .ZN(n6009) );
  INVD0 U6644 ( .I(n6009), .ZN(n5871) );
  CKND2D0 U6645 ( .A1(n7284), .A2(n6638), .ZN(n6066) );
  NR2D0 U6646 ( .A1(n8970), .A2(n8725), .ZN(n7597) );
  CKND2D0 U6647 ( .A1(n9352), .A2(n7597), .ZN(n5969) );
  CKND2D0 U6648 ( .A1(n6066), .A2(n5969), .ZN(n6111) );
  NR2D0 U6649 ( .A1(n9348), .A2(n5857), .ZN(n5963) );
  INVD0 U6650 ( .I(n5963), .ZN(n5994) );
  IND2D0 U6651 ( .A1(n6111), .B1(n5994), .ZN(n6097) );
  NR2D0 U6652 ( .A1(n5871), .A2(n6097), .ZN(n6057) );
  NR2D0 U6653 ( .A1(n6494), .A2(counter[2]), .ZN(n7289) );
  INVD0 U6654 ( .I(n7289), .ZN(n5898) );
  NR2D0 U6655 ( .A1(n5898), .A2(n5861), .ZN(n5873) );
  INVD0 U6656 ( .I(n5873), .ZN(n5914) );
  IND4D0 U6657 ( .A1(n6147), .B1(n5967), .B2(n6057), .B3(n5914)
     , .ZN(n5868) );
  CKND2D0 U6658 ( .A1(n7396), .A2(n5858), .ZN(n5997) );
  CKND2D0 U6659 ( .A1(n5997), .A2(n6165), .ZN(n6064) );
  INVD0 U6660 ( .I(n6064), .ZN(n5860) );
  INVD0 U6661 ( .I(n5858), .ZN(n6515) );
  NR2D0 U6662 ( .A1(n7444), .A2(n6515), .ZN(n5976) );
  NR2D0 U6663 ( .A1(n9348), .A2(n6726), .ZN(n5921) );
  NR2D0 U6664 ( .A1(n5976), .A2(n5921), .ZN(n6029) );
  INVD0 U6665 ( .I(n6029), .ZN(n6132) );
  NR2D0 U6666 ( .A1(n5873), .A2(n6132), .ZN(n6077) );
  CKND2D0 U6667 ( .A1(n7436), .A2(n6638), .ZN(n5998) );
  INVD0 U6668 ( .I(n5998), .ZN(n6089) );
  NR2D0 U6669 ( .A1(n9352), .A2(n9092), .ZN(n7412) );
  CKND2D0 U6670 ( .A1(n7412), .A2(n8010), .ZN(n5901) );
  INVD0 U6671 ( .I(n5901), .ZN(n6060) );
  NR4D0 U6672 ( .A1(n6089), .A2(n5905), .A3(n6060), .A4(n6139),
      .ZN(n5859) );
  NR2D0 U6673 ( .A1(counter[3]), .A2(counter[5]), .ZN(n6081) );
  INVD0 U6674 ( .I(n6081), .ZN(n6178) );
  AOI31D0 U6675 ( .A1(n5860), .A2(n6077), .A3(n5859), .B(n6178)
     , .ZN(n5867) );
  INVD0 U6676 ( .I(n5997), .ZN(n6107) );
  NR2D0 U6677 ( .A1(n6107), .A2(n5945), .ZN(n6124) );
  CKAN2D0 U6678 ( .A1(n6124), .A2(n5901), .Z(n5977) );
```

```
  CKND2D0 U6679 ( .A1(n7289), .A2(n7815), .ZN(n6025) );
  INVD0 U6680 ( .I(n6025), .ZN(n6142) );
  NR2D0 U6681 ( .A1(counter[6]), .A2(counter[4]), .ZN(n6406) );
  CKND2D0 U6682 ( .A1(counter[0]), .A2(n6406), .ZN(n7388) );
  INVD0 U6683 ( .I(n7388), .ZN(n5890) );
  CKND2D0 U6684 ( .A1(n6347), .A2(n5890), .ZN(n6031) );
  INVD0 U6685 ( .I(n6031), .ZN(n6167) );
  NR2D0 U6686 ( .A1(n6142), .A2(n6167), .ZN(n6052) );
  CKND2D0 U6687 ( .A1(n6052), .A2(n5997), .ZN(n6133) );
  INVD0 U6688 ( .I(n6406), .ZN(n7297) );
  NR2D0 U6689 ( .A1(n7297), .A2(n5861), .ZN(n6095) );
  CKND2D0 U6690 ( .A1(counter[2]), .A2(n6095), .ZN(n6114) );
  CKND2D0 U6691 ( .A1(n6114), .A2(n5914), .ZN(n5924) );
  CKND2D0 U6692 ( .A1(n6406), .A2(n5870), .ZN(n6056) );
  NR2D0 U6693 ( .A1(n6205), .A2(n6056), .ZN(n6071) );
  CKND2D0 U6694 ( .A1(n7289), .A2(n5913), .ZN(n5961) );
  INVD0 U6695 ( .I(n5961), .ZN(n6026) );
  NR2D0 U6696 ( .A1(n5963), .A2(n6026), .ZN(n5955) );
  CKND2D0 U6697 ( .A1(n7326), .A2(n5862), .ZN(n5923) );
  NR2D0 U6698 ( .A1(n7309), .A2(n6618), .ZN(n5960) );
  INVD0 U6699 ( .I(n5960), .ZN(n6035) );
  IND4D0 U6700 ( .A1(n6071), .B1(n5955), .B2(n5923), .B3(n6035)
     , .ZN(n5863) );
  CKND2D0 U6701 ( .A1(counter[3]), .A2(counter[5]), .ZN(n6370)
     );
  INVD0 U6702 ( .I(n6370), .ZN(n6458) );
  OAI31D0 U6703 ( .A1(n6133), .A2(n5924), .A3(n5863), .B(n6458)
     , .ZN(n5865) );
  NR2D0 U6704 ( .A1(n5939), .A2(counter[4]), .ZN(n6163) );
  NR2D0 U6705 ( .A1(n6163), .A2(n5905), .ZN(n5891) );
  INVD0 U6706 ( .I(n5891), .ZN(n5872) );
  NR4D0 U6707 ( .A1(n5963), .A2(n6071), .A3(n6863), .A4(n5872),
      .ZN(n5864) );
  NR2D0 U6708 ( .A1(n9345), .A2(counter[5]), .ZN(n7538) );
  INVD0 U6709 ( .I(n7538), .ZN(n6369) );
  AOI32D0 U6710 ( .A1(n5977), .A2(n5865), .A3(n5864), .B1(n6369
     ), .B2(n5865),
        .ZN(n5866) );
  AOI211D0 U6711 ( .A1(n6346), .A2(n5868), .B(n5867), .C(n5866)
     , .ZN(n5882) );
  NR2D0 U6712 ( .A1(n6640), .A2(counter[6]), .ZN(n5892) );
  INVD0 U6713 ( .I(n5892), .ZN(n6075) );
  ND3D0 U6714 ( .A1(n6075), .A2(n5982), .A3(n6009), .ZN(n6094)
     );
  CKND2D0 U6715 ( .A1(counter[6]), .A2(counter[4]), .ZN(n7293)
     );
  OAI21D0 U6716 ( .A1(n5869), .A2(n7293), .B(n6114), .ZN(n6027)
      );
```

```
CKND2D0 U6717 ( .A1(n7542), .A2(n5870), .ZN(n6157) );
IND4D0 U6718 ( .A1(n6027), .B1(n5891), .B2(n6157), .B3(n5901)
   , .ZN(n5880) );
CKND2D0 U6719 ( .A1(counter[6]), .A2(n8968), .ZN(n7422) );
NR2D0 U6720 ( .A1(n8924), .A2(n7422), .ZN(n5975) );
NR2D0 U6721 ( .A1(n6019), .A2(n5975), .ZN(n5995) );
CKND2D0 U6722 ( .A1(n5995), .A2(n6155), .ZN(n6010) );
NR4D0 U6723 ( .A1(n6095), .A2(n5871), .A3(n6010), .A4(n5983),
    .ZN(n5878) );
INVD0 U6724 ( .I(n6056), .ZN(n6106) );
CKND2D0 U6725 ( .A1(n6106), .A2(n6205), .ZN(n6011) );
ND3D0 U6726 ( .A1(n6117), .A2(n5923), .A3(n6011), .ZN(n5920)
   );
OAI21D0 U6727 ( .A1(n8814), .A2(n7293), .B(n5914), .ZN(n6105)
    );
INVD0 U6728 ( .I(n6105), .ZN(n5985) );
INVD0 U6729 ( .I(n6010), .ZN(n5916) );
ND4D0 U6730 ( .A1(n5985), .A2(n5916), .A3(n6129), .A4(n5961),
    .ZN(n5876) );
NR4D0 U6731 ( .A1(n5873), .A2(n8011), .A3(n6139), .A4(n5872),
    .ZN(n5874) );
AOI31D0 U6732 ( .A1(n5874), .A2(n6075), .A3(n6155), .B(n6178)
   , .ZN(n5875) );
AOI221D0 U6733 ( .A1(n5920), .A2(n6458), .B1(n5876), .B2(
   n6458), .C(n5875),
      .ZN(n5877) );
AOI32D0 U6734 ( .A1(n5878), .A2(n5877), .A3(n6117), .B1(n6369
   ), .B2(n5877),
      .ZN(n5879) );
AOI221D0 U6735 ( .A1(n6094), .A2(n6346), .B1(n5880), .B2(
   n6346), .C(n5879),
      .ZN(n5881) );
OAI22D0 U6736 ( .A1(n5882), .A2(n8903), .B1(n5881), .B2(n6393
   ), .ZN(n6190)
       );
INVD0 U6737 ( .I(n5920), .ZN(n5964) );
NR2D0 U6738 ( .A1(n5983), .A2(n5975), .ZN(n5979) );
CKND2D0 U6739 ( .A1(n5964), .A2(n5979), .ZN(n6112) );
INVD0 U6740 ( .I(n5969), .ZN(n6023) );
OR4D0 U6741 ( .A1(n6094), .A2(n6112), .A3(n6107), .A4(n6023),
    .Z(n5889) );
INR2D0 U6742 ( .A1(n6166), .B1(n6089), .ZN(n6118) );
NR2D0 U6743 ( .A1(n6019), .A2(n5905), .ZN(n6067) );
OA211D0 U6744 ( .A1(n6515), .A2(n8951), .B(n5985), .C(n6067),
    .Z(n5883) );
AOI31D0 U6745 ( .A1(n6052), .A2(n6118), .A3(n5883), .B(n6178)
   , .ZN(n5888) );
INVD0 U6746 ( .I(n6640), .ZN(n7771) );
```

```
CKND2D0 U6747 ( .A1(n6129), .A2(n8314), .ZN(n5884) );
NR4D0 U6748 ( .A1(n7771), .A2(n6095), .A3(n7597), .A4(n5884),
    .ZN(n5886) );
INVD0 U6749 ( .I(n5884), .ZN(n6172) );
CKND2D0 U6750 ( .A1(n6172), .A2(n6114), .ZN(n6083) );
CKND2D0 U6751 ( .A1(n6011), .A2(n6031), .ZN(n5996) );
OAI31D0 U6752 ( .A1(n5960), .A2(n6083), .A3(n5996), .B(n6458)
   , .ZN(n5885) );
AOI32D0 U6753 ( .A1(n5886), .A2(n5885), .A3(n6074), .B1(n6369
   ), .B2(n5885),
      .ZN(n5887) );
AOI211D0 U6754 ( .A1(n6346), .A2(n5889), .B(n5888), .C(n5887)
   , .ZN(n5912) );
CKND2D0 U6755 ( .A1(n6028), .A2(n5923), .ZN(n6122) );
AO211D0 U6756 ( .A1(n6825), .A2(n5890), .B(n6089), .C(n6122),
    .Z(n5904) );
INVD0 U6757 ( .I(n6066), .ZN(n6169) );
NR3D0 U6758 ( .A1(n6169), .A2(n5921), .A3(n5960), .ZN(n6125)
   );
OAI211D0 U6759 ( .A1(n7388), .A2(n7717), .B(n6125), .C(n5891)
   , .ZN(n5896) );
CKND2D0 U6760 ( .A1(counter[4]), .A2(n8610), .ZN(n6130) );
INVD0 U6761 ( .I(n7714), .ZN(n6090) );
CKND2D0 U6762 ( .A1(n6130), .A2(n6090), .ZN(n6121) );
INVD0 U6763 ( .I(n6121), .ZN(n6014) );
AOI211D0 U6764 ( .A1(n5893), .A2(n7542), .B(n5892), .C(n6107)
   , .ZN(n5894) );
IIND4D0 U6765 ( .A1(n5904), .A2(n5896), .B1(n6014), .B2(n5894
   ), .ZN(n5910)
      );
NR2D0 U6766 ( .A1(n8970), .A2(n7319), .ZN(n7348) );
NR2D0 U6767 ( .A1(n7322), .A2(n5895), .ZN(n6140) );
NR4D0 U6768 ( .A1(n7348), .A2(n6140), .A3(n5896), .A4(n5924),
    .ZN(n5897) );
AOI31D0 U6769 ( .A1(n6118), .A2(n5897), .A3(n5982), .B(n6151)
   , .ZN(n5909) );
AOI221D0 U6770 ( .A1(n7301), .A2(n5902), .B1(n8038), .B2(
   n5902), .C(
      counter[4]), .ZN(n5944) );
NR2D0 U6771 ( .A1(n9025), .A2(n8978), .ZN(n7313) );
INVD0 U6772 ( .I(n7313), .ZN(n8991) );
AOI221D0 U6773 ( .A1(counter[11]), .A2(n7313), .B1(n9329), .
   B2(n8991), .C(
      n7328), .ZN(n5900) );
NR2D0 U6774 ( .A1(n6122), .A2(n6106), .ZN(n6051) );
OA211D0 U6775 ( .A1(n6656), .A2(n7319), .B(n6066), .C(n6051),
    .Z(n6159) );
INVD0 U6776 ( .I(n5905), .ZN(n5954) );
```

```
OAI211D0 U6777 ( .A1(n8534), .A2(n5898), .B(n6159), .C(n5954)
   , .ZN(n5899) );
NR4D0 U6778 ( .A1(n5944), .A2(n5900), .A3(n6133), .A4(n5899),
    .ZN(n5907) );
CKND2D0 U6779 ( .A1(n6031), .A2(n5901), .ZN(n5962) );
NR2D0 U6780 ( .A1(n6094), .A2(n5962), .ZN(n5970) );
IIND4D0 U6781 ( .A1(n5921), .A2(n6140), .B1(n5970), .B2(n5902
   ), .ZN(n5903)
       );
INR4D0 U6782 ( .A1(n6130), .B1(n5905), .B2(n5904), .B3(n5903)
   , .ZN(n5906) );
OAI22D0 U6783 ( .A1(n5907), .A2(n6178), .B1(n5906), .B2(n6369
   ), .ZN(n5908)
       );
AOI211D0 U6784 ( .A1(n6458), .A2(n5910), .B(n5909), .C(n5908)
   , .ZN(n5911) );
OAI22D0 U6785 ( .A1(n5912), .A2(n6795), .B1(n5911), .B2(n8797
   ), .ZN(n6189)
       );
AOI211D0 U6786 ( .A1(n7542), .A2(n5913), .B(n8869), .C(n6060)
   , .ZN(n5915) );
AOI31D0 U6787 ( .A1(n5916), .A2(n5915), .A3(n5914), .B(n6178)
   , .ZN(n5929) );
INVD0 U6788 ( .I(n6155), .ZN(n6072) );
CKND2D0 U6789 ( .A1(n6075), .A2(n6166), .ZN(n6024) );
NR4D0 U6790 ( .A1(n6072), .A2(n6106), .A3(n6113), .A4(n6024),
    .ZN(n5919) );
INVD0 U6791 ( .I(n7293), .ZN(n6483) );
CKND2D0 U6792 ( .A1(n6483), .A2(n5917), .ZN(n5918) );
INVD0 U6793 ( .I(n5975), .ZN(n6012) );
AOI31D0 U6794 ( .A1(n5919), .A2(n5918), .A3(n6012), .B(n6369)
   , .ZN(n5928) );
NR3D0 U6795 ( .A1(n6163), .A2(n5921), .A3(n5920), .ZN(n6150)
   );
INVD0 U6796 ( .I(n6120), .ZN(n5922) );
AOI31D0 U6797 ( .A1(n6014), .A2(n6150), .A3(n5922), .B(n6151)
   , .ZN(n5927) );
CKND2D0 U6798 ( .A1(n6074), .A2(n5923), .ZN(n6065) );
CKND2D0 U6799 ( .A1(n6075), .A2(n5994), .ZN(n6038) );
INVD0 U6800 ( .I(n5924), .ZN(n5978) );
CKND2D0 U6801 ( .A1(n5925), .A2(n5978), .ZN(n6119) );
OA31D0 U6802 ( .A1(n6065), .A2(n6038), .A3(n6119), .B(n6458),
    .Z(n5926) );
NR4D0 U6803 ( .A1(n5929), .A2(n5928), .A3(n5927), .A4(n5926),
    .ZN(n6008) );
NR2D0 U6804 ( .A1(n7313), .A2(n9329), .ZN(n6567) );
NR2D0 U6805 ( .A1(n8983), .A2(n9189), .ZN(n9216) );
INVD0 U6806 ( .I(n9216), .ZN(n8277) );
```

```
NR2D0 U6807 ( .A1(n7288), .A2(counter[2]), .ZN(n8141) );
INVD0 U6808 ( .I(n8141), .ZN(n8050) );
CKND2D0 U6809 ( .A1(n8277), .A2(n8050), .ZN(n8193) );
AOI22D0 U6810 ( .A1(n8951), .A2(n8052), .B1(n6567), .B2(n8193
   ), .ZN(n5930)
      );
NR2D0 U6811 ( .A1(n9348), .A2(n6848), .ZN(n6037) );
AOI31D0 U6812 ( .A1(n8053), .A2(counter[6]), .A3(n8052), .B(
   n6037), .ZN(
      n5942) );
OAI211D0 U6813 ( .A1(counter[6]), .A2(n5930), .B(n5942), .C(
   n5982), .ZN(
      n5931) );
NR3D0 U6814 ( .A1(n5976), .A2(n6095), .A3(n5931), .ZN(n6143)
   );
NR2D0 U6815 ( .A1(n7319), .A2(n6656), .ZN(n5934) );
CKND2D0 U6816 ( .A1(counter[1]), .A2(n8183), .ZN(n8143) );
NR2D0 U6817 ( .A1(n8143), .A2(n7293), .ZN(n7334) );
NR2D0 U6818 ( .A1(n7297), .A2(n8532), .ZN(n6170) );
OAI21D0 U6819 ( .A1(n7334), .A2(n6170), .B(n5932), .ZN(n6152)
    );
OAI211D0 U6820 ( .A1(n7322), .A2(n8054), .B(n6130), .C(n6152)
   , .ZN(n5933) );
NR4D0 U6821 ( .A1(n5934), .A2(n6107), .A3(n6024), .A4(n5933),
    .ZN(n6145) );
NR2D0 U6822 ( .A1(n7309), .A2(n8038), .ZN(n6160) );
INVD0 U6823 ( .I(n5976), .ZN(n5941) );
CKND2D0 U6824 ( .A1(n5955), .A2(n5941), .ZN(n6032) );
AOI221D0 U6825 ( .A1(n5935), .A2(n6567), .B1(n6160), .B2(
   n6567), .C(n6032),
      .ZN(n5936) );
CKND2D0 U6826 ( .A1(counter[0]), .A2(n8156), .ZN(n8016) );
INVD0 U6827 ( .I(n8016), .ZN(n8042) );
CKND2D0 U6828 ( .A1(n6406), .A2(n8042), .ZN(n7299) );
ND4D0 U6829 ( .A1(n6145), .A2(n6051), .A3(n5936), .A4(n7299),
    .ZN(n5953) );
INVD0 U6830 ( .I(n7412), .ZN(n7366) );
AOI21D0 U6831 ( .A1(n5937), .A2(n7756), .B(n7366), .ZN(n5948)
    );
OAI211D0 U6832 ( .A1(n8038), .A2(n7309), .B(n5979), .C(n6028)
   , .ZN(n5938) );
AOI211D0 U6833 ( .A1(n7289), .A2(n8760), .B(n5948), .C(n5938)
   , .ZN(n5940) );
AOI31D0 U6834 ( .A1(n5940), .A2(n5939), .A3(n8039), .B(n6369)
   , .ZN(n5952) );
ND3D0 U6835 ( .A1(n5941), .A2(n5969), .A3(n5998), .ZN(n6048)
   );
CKND2D0 U6836 ( .A1(n6035), .A2(n6009), .ZN(n6070) );
```

```
INVD0 U6837 ( .I(n5942), .ZN(n5943) );
AOI211D0 U6838 ( .A1(n7326), .A2(n7709), .B(n5944), .C(n5943)
   , .ZN(n6164) );
AOI211D0 U6839 ( .A1(n6812), .A2(n7412), .B(n6142), .C(n5945)
   , .ZN(n5946) );
OAI211D0 U6840 ( .A1(n6494), .A2(n8534), .B(n6164), .C(n5946)
   , .ZN(n5947) );
NR4D0 U6841 ( .A1(n6048), .A2(n6070), .A3(n5948), .A4(n5947),
    .ZN(n5950) );
OAI21D0 U6842 ( .A1(n7369), .A2(n7284), .B(n7986), .ZN(n5949)
    );
AOI31D0 U6843 ( .A1(n5950), .A2(n6011), .A3(n5949), .B(n6178)
   , .ZN(n5951) );
AOI211D0 U6844 ( .A1(n6458), .A2(n5953), .B(n5952), .C(n5951)
   , .ZN(n5959) );
CKND2D0 U6845 ( .A1(n9348), .A2(counter[6]), .ZN(n7489) );
CKND2D0 U6846 ( .A1(n8046), .A2(n7986), .ZN(n6594) );
OAI211D0 U6847 ( .A1(n8068), .A2(n7489), .B(n5954), .C(n6594)
   , .ZN(n6148) );
CKND2D0 U6848 ( .A1(n5955), .A2(n6125), .ZN(n6104) );
NR2D0 U6849 ( .A1(n7436), .A2(n7284), .ZN(n7300) );
OAI22D0 U6850 ( .A1(n7300), .A2(n7972), .B1(n8957), .B2(n5956
   ), .ZN(n5957)
      );
NR4D0 U6851 ( .A1(n6060), .A2(n6148), .A3(n6104), .A4(n5957),
    .ZN(n5958) );
AOI32D0 U6852 ( .A1(n6143), .A2(n5959), .A3(n5958), .B1(n6151
   ), .B2(n5959),
     .ZN(n5993) );
NR2D0 U6853 ( .A1(n5960), .A2(n6089), .ZN(n6096) );
ND4D0 U6854 ( .A1(n6096), .A2(n5979), .A3(n6077), .A4(n5961),
    .ZN(n5966) );
NR2D0 U6855 ( .A1(n5963), .A2(n5962), .ZN(n6080) );
ND4D0 U6856 ( .A1(n5964), .A2(n6080), .A3(n6157), .A4(n6776),
    .ZN(n5965) );
AOI22D0 U6857 ( .A1(n7538), .A2(n5966), .B1(n6346), .B2(n5965
   ), .ZN(n5974)
      );
INVD0 U6858 ( .I(n5967), .ZN(n5968) );
OAI31D0 U6859 ( .A1(n6783), .A2(n6070), .A3(n5968), .B(n6081)
   , .ZN(n5973) );
ND3D0 U6860 ( .A1(n5969), .A2(n6129), .A3(n6114), .ZN(n6149)
   );
INVD0 U6861 ( .I(n6163), .ZN(n6036) );
IND4D0 U6862 ( .A1(n6149), .B1(n5970), .B2(n6036), .B3(n8157)
   , .ZN(n5971) );
OAI21D0 U6863 ( .A1(n6032), .A2(n5971), .B(n6458), .ZN(n5972)
    );
```

```
AOI31D0 U6864 ( .A1(n5974), .A2(n5973), .A3(n5972), .B(n8945)
   , .ZN(n5992) );
NR2D0 U6865 ( .A1(n6037), .A2(n5975), .ZN(n5999) );
CKND2D0 U6866 ( .A1(n8046), .A2(n6638), .ZN(n8214) );
ND4D0 U6867 ( .A1(n6029), .A2(n6051), .A3(n5999), .A4(n8214),
    .ZN(n5981) );
INR2D0 U6868 ( .A1(n6118), .B1(n5976), .ZN(n6046) );
ND4D0 U6869 ( .A1(n6046), .A2(n5979), .A3(n5978), .A4(n5977),
    .ZN(n5980) );
AOI22D0 U6870 ( .A1(n6458), .A2(n5981), .B1(n6346), .B2(n5980
   ), .ZN(n5990)
      );
CKND2D0 U6871 ( .A1(n5982), .A2(n6074), .ZN(n6098) );
NR4D0 U6872 ( .A1(n6060), .A2(n5983), .A3(n6098), .A4(n6070),
    .ZN(n5984) );
ND4D0 U6873 ( .A1(n5985), .A2(n5984), .A3(n6066), .A4(n6726),
    .ZN(n5986) );
OAI21D0 U6874 ( .A1(n6133), .A2(n5986), .B(n7538), .ZN(n5989)
    );
NR2D0 U6875 ( .A1(n6163), .A2(n6030), .ZN(n6091) );
ND4D0 U6876 ( .A1(n6029), .A2(n6091), .A3(n5999), .A4(n6011),
    .ZN(n5987) );
OAI21D0 U6877 ( .A1(n6094), .A2(n5987), .B(n6081), .ZN(n5988)
    );
AOI31D0 U6878 ( .A1(n5990), .A2(n5989), .A3(n5988), .B(n8687)
   , .ZN(n5991) );
AOI211D0 U6879 ( .A1(n8639), .A2(n5993), .B(n5992), .C(n5991)
   , .ZN(n6007) );
AOI31D0 U6880 ( .A1(n5995), .A2(n6014), .A3(n5994), .B(n6369)
   , .ZN(n6005) );
INVD0 U6881 ( .I(n5996), .ZN(n6068) );
AOI31D0 U6882 ( .A1(n6118), .A2(n6068), .A3(n6074), .B(n6151)
   , .ZN(n6004) );
AN4D0 U6883 ( .A1(n6091), .A2(n5997), .A3(n6028), .A4(n6009),
    .Z(n6002) );
ND4D0 U6884 ( .A1(n5999), .A2(n6013), .A3(n6025), .A4(n5998),
    .ZN(n6000) );
OAI31D0 U6885 ( .A1(n6139), .A2(n6147), .A3(n6000), .B(n6081)
   , .ZN(n6001) );
AOI32D0 U6886 ( .A1(n6002), .A2(n6001), .A3(n6129), .B1(n6370
   ), .B2(n6001),
      .ZN(n6003) );
OAI31D0 U6887 ( .A1(n6005), .A2(n6004), .A3(n6003), .B(n8939)
   , .ZN(n6006) );
OAI211D0 U6888 ( .A1(n6008), .A2(n8662), .B(n6007), .C(n6006)
   , .ZN(n6188) );
CKND2D0 U6889 ( .A1(n6031), .A2(n6009), .ZN(n6039) );
```

```
NR4D0 U6890 ( .A1(n6107), .A2(n6039), .A3(n6048), .A4(n6010),
    .ZN(n6022) );
INVD0 U6891 ( .I(n6125), .ZN(n6017) );
INR2D0 U6892 ( .A1(n6011), .B1(n6139), .ZN(n6045) );
ND4D0 U6893 ( .A1(n6045), .A2(n6013), .A3(n8314), .A4(n6012),
    .ZN(n6016) );
AOI31D0 U6894 ( .A1(n6096), .A2(n6045), .A3(n6014), .B(n6178)
   , .ZN(n6015) );
AOI221D0 U6895 ( .A1(n6017), .A2(n6458), .B1(n6016), .B2(
   n6458), .C(n6015),
      .ZN(n6021) );
ND4D0 U6896 ( .A1(n6172), .A2(n6046), .A3(n6036), .A4(n6035),
    .ZN(n6018) );
OAI31D0 U6897 ( .A1(n6142), .A2(n6019), .A3(n6018), .B(n6346)
   , .ZN(n6020) );
OAI211D0 U6898 ( .A1(n6022), .A2(n6369), .B(n6021), .C(n6020)
   , .ZN(n6044) );
NR3D0 U6899 ( .A1(n6024), .A2(n6105), .A3(n6023), .ZN(n6069)
   );
INVD0 U6900 ( .I(n6065), .ZN(n6058) );
AN4D0 U6901 ( .A1(n6069), .A2(n6025), .A3(n6029), .A4(n6058),
    .Z(n6042) );
NR2D0 U6902 ( .A1(n6026), .A2(n6030), .ZN(n6110) );
NR3D0 U6903 ( .A1(n6039), .A2(n6105), .A3(n6027), .ZN(n6115)
   );
ND4D0 U6904 ( .A1(n6029), .A2(n6110), .A3(n6115), .A4(n6028),
    .ZN(n6034) );
INVD0 U6905 ( .I(n6098), .ZN(n6158) );
NR2D0 U6906 ( .A1(n6037), .A2(n6030), .ZN(n6073) );
IND4D0 U6907 ( .A1(n6032), .B1(n6158), .B2(n6073), .B3(n6031)
   , .ZN(n6033) );
AOI22D0 U6908 ( .A1(n6458), .A2(n6034), .B1(n6346), .B2(n6033
   ), .ZN(n6041)
      );
IND4D0 U6909 ( .A1(n6037), .B1(n6124), .B2(n6036), .B3(n6035)
   , .ZN(n6055) );
OAI31D0 U6910 ( .A1(n6039), .A2(n6055), .A3(n6038), .B(n6081)
   , .ZN(n6040) );
OAI211D0 U6911 ( .A1(n6042), .A2(n6369), .B(n6041), .C(n6040)
   , .ZN(n6043) );
AOI22D0 U6912 ( .A1(n8656), .A2(n6044), .B1(n8910), .B2(n6043
   ), .ZN(n6186)
      );
AN4D0 U6913 ( .A1(n6047), .A2(n6155), .A3(n6046), .A4(n6045),
    .Z(n6063) );
IND4D0 U6914 ( .A1(n6048), .B1(n6110), .B2(n6058), .B3(n6052)
   , .ZN(n6054) );
```

```
OA31D0 U6915 ( .A1(n8983), .A2(n6494), .A3(n6049), .B(n6057),
    .Z(n6050) );
AOI31D0 U6916 ( .A1(n6052), .A2(n6051), .A3(n6050), .B(n6178)
   , .ZN(n6053) );
AOI221D0 U6917 ( .A1(n6055), .A2(n6346), .B1(n6054), .B2(
   n6346), .C(n6053),
      .ZN(n6062) );
ND4D0 U6918 ( .A1(n6073), .A2(n6058), .A3(n6057), .A4(n6056),
    .ZN(n6059) );
OAI31D0 U6919 ( .A1(n6089), .A2(n6060), .A3(n6059), .B(n7538)
   , .ZN(n6061) );
OAI211D0 U6920 ( .A1(n6063), .A2(n6370), .B(n6062), .C(n6061)
   , .ZN(n6088) );
NR4D0 U6921 ( .A1(n8523), .A2(n6065), .A3(n6064), .A4(n6097),
    .ZN(n6086) );
ND4D0 U6922 ( .A1(n6069), .A2(n6068), .A3(n6067), .A4(n6066),
    .ZN(n6079) );
INR4D0 U6923 ( .A1(n6073), .B1(n6072), .B2(n6071), .B3(n6070)
   , .ZN(n6076) );
ND4D0 U6924 ( .A1(n6077), .A2(n6076), .A3(n6075), .A4(n6074),
    .ZN(n6078) );
AOI22D0 U6925 ( .A1(n6458), .A2(n6079), .B1(n7538), .B2(n6078
   ), .ZN(n6085)
      );
INVD0 U6926 ( .I(n6080), .ZN(n6082) );
OAI31D0 U6927 ( .A1(n6083), .A2(n6098), .A3(n6082), .B(n6081)
   , .ZN(n6084) );
OAI211D0 U6928 ( .A1(n6086), .A2(n6151), .B(n6085), .C(n6084)
   , .ZN(n6087) );
AOI22D0 U6929 ( .A1(n8803), .A2(n6088), .B1(n8895), .B2(n6087
   ), .ZN(n6185)
      );
NR2D0 U6930 ( .A1(n6142), .A2(n6089), .ZN(n6092) );
ND4D0 U6931 ( .A1(n6092), .A2(n6091), .A3(n8819), .A4(n6090),
    .ZN(n6103) );
ND4D0 U6932 ( .A1(n6124), .A2(n6166), .A3(n6155), .A4(n8394),
    .ZN(n6093) );
INR4D0 U6933 ( .A1(n6096), .B1(n6095), .B2(n6094), .B3(n6093)
   , .ZN(n6101) );
NR2D0 U6934 ( .A1(n6494), .A2(n8613), .ZN(n6099) );
NR4D0 U6935 ( .A1(n8610), .A2(n6099), .A3(n6098), .A4(n6097),
    .ZN(n6100) );
OAI22D0 U6936 ( .A1(n6101), .A2(n6369), .B1(n6100), .B2(n6178
   ), .ZN(n6102)
      );
AOI221D0 U6937 ( .A1(n6104), .A2(n6458), .B1(n6103), .B2(
   n6458), .C(n6102),
      .ZN(n6109) );
```

```
NR4D0 U6938 ( .A1(n6107), .A2(n6106), .A3(n6149), .A4(n6105),
    .ZN(n6108) );
AOI32D0 U6939 ( .A1(n6110), .A2(n6109), .A3(n6108), .B1(n6151
   ), .B2(n6109),
      .ZN(n6138) );
INR4D0 U6940 ( .A1(n6114), .B1(n6113), .B2(n6112), .B3(n6111)
   , .ZN(n6136) );
INVD0 U6941 ( .I(n6115), .ZN(n6128) );
ND4D0 U6942 ( .A1(n6118), .A2(n6125), .A3(n6117), .A4(n6116),
    .ZN(n6127) );
NR4D0 U6943 ( .A1(n6122), .A2(n6121), .A3(n6120), .A4(n6119),
    .ZN(n6123) );
AOI31D0 U6944 ( .A1(n6125), .A2(n6124), .A3(n6123), .B(n6369)
   , .ZN(n6126) );
AOI221D0 U6945 ( .A1(n6128), .A2(n6458), .B1(n6127), .B2(
   n6458), .C(n6126),
      .ZN(n6135) );
ND4D0 U6946 ( .A1(n6640), .A2(n6130), .A3(n6129), .A4(n8157),
    .ZN(n6131) );
OAI31D0 U6947 ( .A1(n6133), .A2(n6132), .A3(n6131), .B(n6346)
   , .ZN(n6134) );
OAI211D0 U6948 ( .A1(n6136), .A2(n6178), .B(n6135), .C(n6134)
   , .ZN(n6137) );
AOI22D0 U6949 ( .A1(n8037), .A2(n6138), .B1(n7939), .B2(n6137
   ), .ZN(n6184)
       );
AOI211D0 U6950 ( .A1(n6483), .A2(n8045), .B(n6140), .C(n6139)
   , .ZN(n6146) );
NR2D0 U6951 ( .A1(n7366), .A2(n7756), .ZN(n6141) );
NR2D0 U6952 ( .A1(n7328), .A2(n7972), .ZN(n6176) );
INR4D0 U6953 ( .A1(n6143), .B1(n6142), .B2(n6141), .B3(n6176)
   , .ZN(n6144) );
AOI31D0 U6954 ( .A1(n6146), .A2(n6145), .A3(n6144), .B(n6370)
   , .ZN(n6182) );
AOI22D0 U6955 ( .A1(n6406), .A2(n6812), .B1(n7284), .B2(n8052
   ), .ZN(n6154)
       );
INR4D0 U6956 ( .A1(n6150), .B1(n6149), .B2(n6148), .B3(n6147)
   , .ZN(n6153) );
AOI31D0 U6957 ( .A1(n6154), .A2(n6153), .A3(n6152), .B(n6151)
   , .ZN(n6181) );
CKND2D0 U6958 ( .A1(n9352), .A2(n8046), .ZN(n7492) );
INVD0 U6959 ( .I(n7492), .ZN(n7359) );
OAI21D0 U6960 ( .A1(n7359), .A2(n7327), .B(n7709), .ZN(n6156)
    );
OAI211D0 U6961 ( .A1(n7300), .A2(n8054), .B(n6156), .C(n6155)
   , .ZN(n6162) );
```

```
IND4D0 U6962 ( .A1(n6160), .B1(n6159), .B2(n6158), .B3(n6157)
   , .ZN(n6161) );
INR4D0 U6963 ( .A1(n6164), .B1(n6163), .B2(n6162), .B3(n6161)
   , .ZN(n6179) );
NR2D0 U6964 ( .A1(n7388), .A2(n7717), .ZN(n6175) );
CKND2D0 U6965 ( .A1(n8046), .A2(n8052), .ZN(n7725) );
ND4D0 U6966 ( .A1(n6166), .A2(n8039), .A3(n6165), .A4(n7725),
    .ZN(n6174) );
NR2D0 U6967 ( .A1(n7489), .A2(n8068), .ZN(n6168) );
NR4D0 U6968 ( .A1(n6170), .A2(n6169), .A3(n6168), .A4(n6167),
    .ZN(n6171) );
OAI211D0 U6969 ( .A1(n7300), .A2(n6656), .B(n6172), .C(n6171)
   , .ZN(n6173) );
NR4D0 U6970 ( .A1(n6176), .A2(n6175), .A3(n6174), .A4(n6173),
    .ZN(n6177) );
OAI22D0 U6971 ( .A1(n6179), .A2(n6178), .B1(n6177), .B2(n6369
   ), .ZN(n6180)
      );
OAI31D0 U6972 ( .A1(n6182), .A2(n6181), .A3(n6180), .B(n8918)
   , .ZN(n6183) );
ND4D0 U6973 ( .A1(n6186), .A2(n6185), .A3(n6184), .A4(n6183),
    .ZN(n6187) );
NR4D0 U6974 ( .A1(n6190), .A2(n6189), .A3(n6188), .A4(n6187),
    .ZN(n6510) );
NR2D0 U6975 ( .A1(counter[11]), .A2(counter[5]), .ZN(n8009) )
   ;
CKND2D0 U6976 ( .A1(n8009), .A2(n9345), .ZN(n7029) );
NR2D0 U6977 ( .A1(n7029), .A2(n8143), .ZN(n6320) );
NR2D0 U6978 ( .A1(n6521), .A2(counter[5]), .ZN(n6319) );
NR2D0 U6979 ( .A1(n6320), .A2(n6319), .ZN(n6380) );
INVD0 U6980 ( .I(n6380), .ZN(n6294) );
NR2D0 U6981 ( .A1(n7514), .A2(counter[11]), .ZN(n6321) );
CKND2D0 U6982 ( .A1(n9345), .A2(n6321), .ZN(n7001) );
INVD0 U6983 ( .I(n7001), .ZN(n6216) );
INVD0 U6984 ( .I(n8009), .ZN(n8276) );
NR2D0 U6985 ( .A1(n9345), .A2(n8276), .ZN(n7006) );
NR2D0 U6986 ( .A1(n9143), .A2(n8957), .ZN(n8126) );
CKND2D0 U6987 ( .A1(n7006), .A2(n8126), .ZN(n6471) );
INVD0 U6988 ( .I(n6471), .ZN(n6242) );
NR2D0 U6989 ( .A1(n7001), .A2(n8050), .ZN(n6492) );
INVD0 U6990 ( .I(n6492), .ZN(n6202) );
INVD0 U6991 ( .I(n7006), .ZN(n7003) );
INVD0 U6992 ( .I(n8130), .ZN(n8186) );
NR2D0 U6993 ( .A1(n7003), .A2(n8186), .ZN(n6425) );
INVD0 U6994 ( .I(n6425), .ZN(n6467) );
CKND2D0 U6995 ( .A1(n6202), .A2(n6467), .ZN(n6269) );
AOI211D0 U6996 ( .A1(n8042), .A2(n6216), .B(n6242), .C(n6269)
   , .ZN(n6398) );
```

```
  NR2D0 U6997 ( .A1(n7514), .A2(n6557), .ZN(n6424) );
  NR2D0 U6998 ( .A1(n7003), .A2(n8277), .ZN(n6300) );
  NR2D0 U6999 ( .A1(n7514), .A2(n6521), .ZN(n6387) );
  OR2D0 U7000 ( .A1(n6300), .A2(n6387), .Z(n6311) );
  NR2D0 U7001 ( .A1(n6424), .A2(n6311), .ZN(n6381) );
  INVD0 U7002 ( .I(n6321), .ZN(n8014) );
  NR2D0 U7003 ( .A1(n8014), .A2(n9347), .ZN(n6304) );
  INVD0 U7004 ( .I(n6304), .ZN(n6359) );
  CKND2D0 U7005 ( .A1(n8983), .A2(n6191), .ZN(n8175) );
  NR2D0 U7006 ( .A1(n8276), .A2(n8175), .ZN(n8378) );
  INVD0 U7007 ( .I(n8378), .ZN(n8263) );
  ND4D0 U7008 ( .A1(n6398), .A2(n6381), .A3(n6359), .A4(n8263),
      .ZN(n6201) );
  CKND2D0 U7009 ( .A1(n6216), .A2(n8183), .ZN(n6234) );
  NR2D0 U7010 ( .A1(counter[1]), .A2(n6234), .ZN(n6345) );
  NR2D0 U7011 ( .A1(n7029), .A2(n8050), .ZN(n6193) );
  CKND2D0 U7012 ( .A1(n6458), .A2(n9329), .ZN(n7028) );
  NR2D0 U7013 ( .A1(n7028), .A2(n8143), .ZN(n6382) );
  NR2D0 U7014 ( .A1(n9143), .A2(n8055), .ZN(n9215) );
  INVD0 U7015 ( .I(n9215), .ZN(n9344) );
  NR2D0 U7016 ( .A1(n7001), .A2(n9344), .ZN(n6447) );
  NR2D0 U7017 ( .A1(n6382), .A2(n6447), .ZN(n6260) );
  INVD0 U7018 ( .I(n8143), .ZN(n8066) );
  CKND2D0 U7019 ( .A1(n7006), .A2(n8066), .ZN(n6414) );
  INVD0 U7020 ( .I(n6414), .ZN(n6478) );
  NR2D0 U7021 ( .A1(n6492), .A2(n6478), .ZN(n6330) );
  CKND2D0 U7022 ( .A1(n6260), .A2(n6330), .ZN(n6295) );
  NR4D0 U7023 ( .A1(n6392), .A2(n6345), .A3(n6193), .A4(n6295),
      .ZN(n6199) );
  INVD0 U7024 ( .I(n8175), .ZN(n8078) );
  NR2D0 U7025 ( .A1(n8014), .A2(n9344), .ZN(n8420) );
  AOI211D0 U7026 ( .A1(n8078), .A2(n6321), .B(n8420), .C(n6425)
     , .ZN(n6440) );
  NR2D0 U7027 ( .A1(n6440), .A2(n6192), .ZN(n6209) );
  CKND2D0 U7028 ( .A1(n7006), .A2(n8078), .ZN(n6418) );
  NR2D0 U7029 ( .A1(n7029), .A2(n8016), .ZN(n6327) );
  INVD0 U7030 ( .I(n6327), .ZN(n6312) );
  CKND2D0 U7031 ( .A1(n6418), .A2(n6312), .ZN(n6292) );
  INVD0 U7032 ( .I(n6292), .ZN(n6303) );
  INVD0 U7033 ( .I(n6193), .ZN(n6413) );
  ND4D0 U7034 ( .A1(n6303), .A2(n6381), .A3(n6471), .A4(n6413),
      .ZN(n6197) );
  NR2D0 U7035 ( .A1(n7029), .A2(n8186), .ZN(n6220) );
  INVD0 U7036 ( .I(n6220), .ZN(n6229) );
  NR2D0 U7037 ( .A1(n8276), .A2(n9347), .ZN(n6318) );
  NR2D0 U7038 ( .A1(n6193), .A2(n6318), .ZN(n6302) );
  INR2D0 U7039 ( .A1(n6302), .B1(n6424), .ZN(n6399) );
```

```
ND3D0 U7040 ( .A1(n6229), .A2(n6380), .A3(n6399), .ZN(n6385)
   );
INVD0 U7041 ( .I(n6385), .ZN(n6195) );
CKND2D0 U7042 ( .A1(n8009), .A2(n7314), .ZN(n7059) );
NR2D0 U7043 ( .A1(counter[0]), .A2(n7059), .ZN(n6215) );
NR2D0 U7044 ( .A1(n8014), .A2(n8016), .ZN(n8280) );
CKND2D0 U7045 ( .A1(n7006), .A2(n8141), .ZN(n6217) );
INVD0 U7046 ( .I(n6217), .ZN(n6391) );
NR4D0 U7047 ( .A1(n6215), .A2(n8280), .A3(n6304), .A4(n6391),
    .ZN(n6194) );
AOI31D0 U7048 ( .A1(n6195), .A2(n6303), .A3(n6194), .B(n7297)
   , .ZN(n6196) );
AOI221D0 U7049 ( .A1(n6209), .A2(n7542), .B1(n6197), .B2(
   n7542), .C(n6196),
      .ZN(n6198) );
AOI32D0 U7050 ( .A1(n6199), .A2(n6198), .A3(n6312), .B1(n7489
   ), .B2(n6198),
      .ZN(n6200) );
AOI221D0 U7051 ( .A1(n6294), .A2(n6483), .B1(n6201), .B2(
   n6483), .C(n6200),
      .ZN(n6228) );
NR2D0 U7052 ( .A1(n6447), .A2(n6242), .ZN(n6428) );
OAI211D0 U7053 ( .A1(n7235), .A2(n8014), .B(n6399), .C(n6202)
   , .ZN(n6203) );
INR3D0 U7054 ( .A1(n6428), .B1(n6345), .B2(n6203), .ZN(n6286)
    );
INVD0 U7055 ( .I(n7413), .ZN(n7367) );
NR2D0 U7056 ( .A1(n8014), .A2(n7367), .ZN(n7011) );
NR2D0 U7057 ( .A1(n7028), .A2(n8050), .ZN(n6258) );
NR2D0 U7058 ( .A1(n6220), .A2(n6258), .ZN(n6463) );
CKND2D0 U7059 ( .A1(n6414), .A2(n6463), .ZN(n6283) );
INVD0 U7060 ( .I(n6283), .ZN(n6289) );
CKND2D0 U7061 ( .A1(n6289), .A2(n6359), .ZN(n6328) );
NR2D0 U7062 ( .A1(n7028), .A2(n8186), .ZN(n6461) );
IND4D0 U7063 ( .A1(n6461), .B1(n6260), .B2(n6303), .B3(n6471)
   , .ZN(n6204) );
NR4D0 U7064 ( .A1(n7011), .A2(n6328), .A3(n6269), .A4(n6204),
    .ZN(n6207) );
CKND2D0 U7065 ( .A1(n6216), .A2(n8130), .ZN(n6313) );
NR2D0 U7066 ( .A1(n6205), .A2(n6234), .ZN(n6397) );
NR3D0 U7067 ( .A1(n6397), .A2(n6319), .A3(n6424), .ZN(n6476)
   );
INVD0 U7068 ( .I(n6418), .ZN(n6444) );
NR2D0 U7069 ( .A1(n6311), .A2(n6444), .ZN(n6438) );
AN4D0 U7070 ( .A1(n6467), .A2(n6313), .A3(n6476), .A4(n6438),
    .Z(n6206) );
OAI222D0 U7071 ( .A1(n7293), .A2(n6286), .B1(n7489), .B2(
   n6207), .C1(n7297),
```

```
      .C2(n6206), .ZN(n6214) );
INVD0 U7072 ( .I(n6387), .ZN(n6358) );
NR2D0 U7073 ( .A1(n7028), .A2(n8175), .ZN(n6333) );
INVD0 U7074 ( .I(n6333), .ZN(n6487) );
CKND2D0 U7075 ( .A1(n6358), .A2(n6487), .ZN(n6233) );
NR2D0 U7076 ( .A1(n8078), .A2(n9215), .ZN(n8056) );
INVD0 U7077 ( .I(n6215), .ZN(n6259) );
CKND2D0 U7078 ( .A1(n7006), .A2(n8042), .ZN(n6407) );
CKND2D0 U7079 ( .A1(n6259), .A2(n6407), .ZN(n6310) );
NR2D0 U7080 ( .A1(n6382), .A2(n6310), .ZN(n6323) );
INVD0 U7081 ( .I(n8126), .ZN(n8142) );
NR2D0 U7082 ( .A1(n7028), .A2(n8142), .ZN(n6354) );
NR2D0 U7083 ( .A1(n6354), .A2(n6320), .ZN(n6231) );
OAI211D0 U7084 ( .A1(n8056), .A2(n7003), .B(n6323), .C(n6231)
   , .ZN(n6208) );
NR4D0 U7085 ( .A1(n6492), .A2(n6397), .A3(n6233), .A4(n6208),
    .ZN(n6212) );
INVD0 U7086 ( .I(n6209), .ZN(n6378) );
ND3D0 U7087 ( .A1(n6407), .A2(n6378), .A3(n6229), .ZN(n6232)
   );
NR4D0 U7088 ( .A1(n6392), .A2(n6333), .A3(n6320), .A4(n6232),
    .ZN(n6211) );
NR2D0 U7089 ( .A1(n6215), .A2(n6424), .ZN(n6331) );
NR2D0 U7090 ( .A1(n7029), .A2(n9344), .ZN(n6462) );
NR2D0 U7091 ( .A1(n7587), .A2(n6462), .ZN(n6222) );
NR2D0 U7092 ( .A1(n8276), .A2(n7285), .ZN(n6479) );
IINR4D0 U7093 ( .A1(n6331), .A2(n6222), .B1(n6479), .B2(n8280
   ), .ZN(n6210)
      );
OAI222D0 U7094 ( .A1(n7293), .A2(n6212), .B1(n7489), .B2(
   n6211), .C1(n7297),
      .C2(n6210), .ZN(n6213) );
AOI22D0 U7095 ( .A1(n8656), .A2(n6214), .B1(n8895), .B2(n6213
   ), .ZN(n6227)
      );
NR2D0 U7096 ( .A1(n6215), .A2(n6242), .ZN(n6460) );
INVD0 U7097 ( .I(n6460), .ZN(n6262) );
CKND2D0 U7098 ( .A1(n6216), .A2(n8042), .ZN(n6285) );
IND2D0 U7099 ( .A1(n6354), .B1(n6285), .ZN(n6377) );
NR4D0 U7100 ( .A1(n6318), .A2(n6283), .A3(n6262), .A4(n6377),
    .ZN(n6218) );
CKND2D0 U7101 ( .A1(n6321), .A2(n8078), .ZN(n8021) );
AOI31D0 U7102 ( .A1(n6218), .A2(n8021), .A3(n6217), .B(n7593)
   , .ZN(n6225) );
NR2D0 U7103 ( .A1(n8276), .A2(n7367), .ZN(n7068) );
INVD0 U7104 ( .I(n6313), .ZN(n6271) );
OR2D0 U7105 ( .A1(n6462), .A2(n6271), .Z(n6446) );
NR2D0 U7106 ( .A1(n6446), .A2(n6304), .ZN(n6416) );
```

```
INVD0 U7107 ( .I(n6416), .ZN(n6423) );
NR4D0 U7108 ( .A1(n7068), .A2(n6397), .A3(n6300), .A4(n6423),
    .ZN(n6219) );
AOI31D0 U7109 ( .A1(n6399), .A2(n6219), .A3(n6378), .B(n8848)
   , .ZN(n6224) );
NR2D0 U7110 ( .A1(n8014), .A2(n8186), .ZN(n8355) );
NR2D0 U7111 ( .A1(n6391), .A2(n8355), .ZN(n6403) );
NR3D0 U7112 ( .A1(n6320), .A2(n6220), .A3(n6300), .ZN(n6245)
   );
AOI31D0 U7113 ( .A1(n6403), .A2(n6222), .A3(n6245), .B(n6221)
   , .ZN(n6223) );
OAI31D0 U7114 ( .A1(n6225), .A2(n6224), .A3(n6223), .B(n7542)
   , .ZN(n6226) );
OAI211D0 U7115 ( .A1(n6228), .A2(n6795), .B(n6227), .C(n6226)
   , .ZN(n6508) );
INVD0 U7116 ( .I(n7489), .ZN(n7291) );
NR2D0 U7117 ( .A1(n7028), .A2(n8016), .ZN(n6341) );
NR2D0 U7118 ( .A1(n6447), .A2(n6341), .ZN(n6388) );
CKND2D0 U7119 ( .A1(n6229), .A2(n6388), .ZN(n6309) );
NR2D0 U7120 ( .A1(n6345), .A2(n6377), .ZN(n6324) );
INVD0 U7121 ( .I(n6324), .ZN(n6329) );
OR4D0 U7122 ( .A1(n6309), .A2(n6329), .A3(n6424), .A4(n8378),
    .Z(n6241) );
NR2D0 U7123 ( .A1(n7029), .A2(n8175), .ZN(n6470) );
NR2D0 U7124 ( .A1(n6318), .A2(n6470), .ZN(n6389) );
NR4D0 U7125 ( .A1(n7011), .A2(n6320), .A3(n6345), .A4(n6292),
    .ZN(n6230) );
AOI31D0 U7126 ( .A1(n6389), .A2(n6230), .A3(n6378), .B(n6494)
   , .ZN(n6240) );
NR2D0 U7127 ( .A1(n8276), .A2(n8050), .ZN(n8062) );
CKND2D0 U7128 ( .A1(n6231), .A2(n6330), .ZN(n6412) );
NR4D0 U7129 ( .A1(n8062), .A2(n6292), .A3(n6232), .A4(n6412),
    .ZN(n6238) );
NR2D0 U7130 ( .A1(n7001), .A2(n8175), .ZN(n6270) );
INVD0 U7131 ( .I(n6270), .ZN(n6417) );
IND2D0 U7132 ( .A1(n6341), .B1(n6417), .ZN(n6236) );
CKND2D0 U7133 ( .A1(n6414), .A2(n6323), .ZN(n6442) );
INVD0 U7134 ( .I(n6442), .ZN(n6441) );
NR2D0 U7135 ( .A1(n6233), .A2(n6292), .ZN(n6284) );
IND4D0 U7136 ( .A1(n8062), .B1(n6441), .B2(n6284), .B3(n6234)
   , .ZN(n6235) );
NR2D0 U7137 ( .A1(n6236), .A2(n6235), .ZN(n6237) );
OAI22D0 U7138 ( .A1(n6238), .A2(n7293), .B1(n6237), .B2(n7297
   ), .ZN(n6239)
      );
AOI211D0 U7139 ( .A1(n7291), .A2(n6241), .B(n6240), .C(n6239)
   , .ZN(n6257) );
NR2D0 U7140 ( .A1(n6377), .A2(n6300), .ZN(n6401) );
```

```
CKND2D0 U7141 ( .A1(n6303), .A2(n6401), .ZN(n6445) );
NR2D0 U7142 ( .A1(n6242), .A2(n6461), .ZN(n6243) );
INR4D0 U7143 ( .A1(n6323), .B1(n6263), .B2(n6424), .B3(n6271)
   , .ZN(n6272) );
IND4D0 U7144 ( .A1(n6445), .B1(n6243), .B2(n6272), .B3(n6414)
   , .ZN(n6255) );
CKND2D0 U7145 ( .A1(counter[11]), .A2(n7514), .ZN(n8116) );
AOI221D0 U7146 ( .A1(n6368), .A2(n8050), .B1(n7421), .B2(
   n8050), .C(n8116),
      .ZN(n6348) );
NR2D0 U7147 ( .A1(counter[3]), .A2(n8116), .ZN(n6455) );
CKND2D0 U7148 ( .A1(n8042), .A2(n6455), .ZN(n6244) );
ND4D0 U7149 ( .A1(n6245), .A2(n6388), .A3(n6413), .A4(n6244),
    .ZN(n6246) );
NR4D0 U7150 ( .A1(n6382), .A2(n6304), .A3(n6348), .A4(n6246),
    .ZN(n6247) );
NR2D0 U7151 ( .A1(n9345), .A2(n8116), .ZN(n6362) );
CKND2D0 U7152 ( .A1(n9132), .A2(n6362), .ZN(n6481) );
AOI31D0 U7153 ( .A1(n6247), .A2(n6389), .A3(n6481), .B(n6494)
   , .ZN(n6254) );
OAI31D0 U7154 ( .A1(n6248), .A2(n8056), .A3(n8116), .B(n6259)
   , .ZN(n6250) );
CKAN2D0 U7155 ( .A1(n6388), .A2(n6303), .Z(n6288) );
INVD0 U7156 ( .I(n6362), .ZN(n6456) );
AOI32D0 U7157 ( .A1(n7028), .A2(n6288), .A3(n6456), .B1(n8186
   ), .B2(n6288),
      .ZN(n6249) );
OAI31D0 U7158 ( .A1(n6250), .A2(n6423), .A3(n6249), .B(n6406)
   , .ZN(n6252) );
CKND2D0 U7159 ( .A1(n6413), .A2(n6285), .ZN(n6453) );
NR4D0 U7160 ( .A1(n6397), .A2(n6425), .A3(n6258), .A4(n6453),
    .ZN(n6251) );
AOI32D0 U7161 ( .A1(n6330), .A2(n6252), .A3(n6251), .B1(n7489
   ), .B2(n6252),
      .ZN(n6253) );
AOI211D0 U7162 ( .A1(n6483), .A2(n6255), .B(n6254), .C(n6253)
   , .ZN(n6256) );
OAI22D0 U7163 ( .A1(n6257), .A2(n8169), .B1(n6256), .B2(n8797
   ), .ZN(n6507)
       );
INVD0 U7164 ( .I(n6258), .ZN(n6422) );
CKND2D0 U7165 ( .A1(n8009), .A2(n8126), .ZN(n8189) );
ND4D0 U7166 ( .A1(n6324), .A2(n6422), .A3(n8189), .A4(n6417),
    .ZN(n6443) );
CKAN2D0 U7167 ( .A1(n6403), .A2(n6259), .Z(n6419) );
ND4D0 U7168 ( .A1(n6260), .A2(n6302), .A3(n6419), .A4(n6312),
    .ZN(n6267) );
```

```
NR3D0 U7169 ( .A1(n6333), .A2(n6354), .A3(n6311), .ZN(n6415)
   );
INVD0 U7170 ( .I(n6470), .ZN(n6268) );
ND4D0 U7171 ( .A1(n6416), .A2(n6324), .A3(n6358), .A4(n6268),
    .ZN(n6261) );
OAI31D0 U7172 ( .A1(n6382), .A2(n6425), .A3(n6261), .B(n6406)
   , .ZN(n6265) );
NR4D0 U7173 ( .A1(n6263), .A2(n6327), .A3(n6328), .A4(n6262),
    .ZN(n6264) );
AOI32D0 U7174 ( .A1(n6415), .A2(n6265), .A3(n6264), .B1(n7293
   ), .B2(n6265),
      .ZN(n6266) );
AOI221D0 U7175 ( .A1(n6443), .A2(n7291), .B1(n6267), .B2(
   n7291), .C(n6266),
      .ZN(n6281) );
CKND2D0 U7176 ( .A1(n6331), .A2(n6268), .ZN(n6400) );
NR2D0 U7177 ( .A1(n6382), .A2(n6345), .ZN(n6497) );
NR4D0 U7178 ( .A1(n8420), .A2(n6270), .A3(n6341), .A4(n6269),
    .ZN(n6437) );
ND4D0 U7179 ( .A1(n6497), .A2(n6437), .A3(n6471), .A4(n7867),
    .ZN(n6279) );
INVD0 U7180 ( .I(n6389), .ZN(n6282) );
OR4D0 U7181 ( .A1(n6479), .A2(n6377), .A3(n6282), .A4(n6271),
    .Z(n6274) );
AOI31D0 U7182 ( .A1(n6272), .A2(n6417), .A3(n7668), .B(n7293)
   , .ZN(n6273) );
AOI221D0 U7183 ( .A1(n6295), .A2(n6406), .B1(n6274), .B2(
   n6406), .C(n6273),
      .ZN(n6277) );
CKND2D0 U7184 ( .A1(n6476), .A2(n6557), .ZN(n6287) );
NR4D0 U7185 ( .A1(n6447), .A2(n6275), .A3(n6287), .A4(n6282),
    .ZN(n6276) );
AOI32D0 U7186 ( .A1(n6460), .A2(n6277), .A3(n6276), .B1(n7489
   ), .B2(n6277),
      .ZN(n6278) );
AOI221D0 U7187 ( .A1(n6400), .A2(n7542), .B1(n6279), .B2(
   n7542), .C(n6278),
      .ZN(n6280) );
OAI22D0 U7188 ( .A1(n6281), .A2(n8848), .B1(n6280), .B2(n8708
   ), .ZN(n6506)
       );
CKND2D0 U7189 ( .A1(n6407), .A2(n6471), .ZN(n6305) );
INR4D0 U7190 ( .A1(n6284), .B1(n6283), .B2(n6282), .B3(n6305)
   , .ZN(n6298) );
INVD0 U7191 ( .I(n6319), .ZN(n6439) );
ND4D0 U7192 ( .A1(n6286), .A2(n6439), .A3(n6422), .A4(n6285),
    .ZN(n6291) );
INVD0 U7193 ( .I(n6287), .ZN(n6357) );
```

```
AOI31D0 U7194 ( .A1(n6357), .A2(n6289), .A3(n6288), .B(n7297)
   , .ZN(n6290) );
AOI221D0 U7195 ( .A1(n6292), .A2(n7291), .B1(n6291), .B2(
   n7291), .C(n6290),
      .ZN(n6297) );
NR2D0 U7196 ( .A1(n8276), .A2(n9344), .ZN(n8020) );
NR2D0 U7197 ( .A1(n6391), .A2(n8020), .ZN(n6364) );
OAI211D0 U7198 ( .A1(n8186), .A2(n7028), .B(n6364), .C(n6407)
   , .ZN(n6293) );
OAI31D0 U7199 ( .A1(n6295), .A2(n6294), .A3(n6293), .B(n7542)
   , .ZN(n6296) );
OAI211D0 U7200 ( .A1(n6298), .A2(n7293), .B(n6297), .C(n6296)
   , .ZN(n6339) );
CKND2D0 U7201 ( .A1(n6407), .A2(n6417), .ZN(n6299) );
NR4D0 U7202 ( .A1(n6397), .A2(n8280), .A3(n6300), .A4(n6299),
    .ZN(n6301) );
ND4D0 U7203 ( .A1(n6463), .A2(n6303), .A3(n6302), .A4(n6301),
    .ZN(n6308) );
NR2D0 U7204 ( .A1(n6304), .A2(n6391), .ZN(n6306) );
NR2D0 U7205 ( .A1(n6424), .A2(n6305), .ZN(n6379) );
ND4D0 U7206 ( .A1(n6306), .A2(n6497), .A3(n6379), .A4(n6358),
    .ZN(n6307) );
AOI22D0 U7207 ( .A1(n6406), .A2(n6308), .B1(n6483), .B2(n6307
   ), .ZN(n6317)
      );
OAI31D0 U7208 ( .A1(n6311), .A2(n6310), .A3(n6309), .B(n7291)
   , .ZN(n6316) );
INVD0 U7209 ( .I(n6399), .ZN(n6485) );
ND4D0 U7210 ( .A1(n6388), .A2(n6439), .A3(n6313), .A4(n6312),
    .ZN(n6314) );
OAI31D0 U7211 ( .A1(n6442), .A2(n6485), .A3(n6314), .B(n7542)
   , .ZN(n6315) );
AOI31D0 U7212 ( .A1(n6317), .A2(n6316), .A3(n6315), .B(n8662)
   , .ZN(n6338) );
NR4D0 U7213 ( .A1(n6319), .A2(n6318), .A3(n6341), .A4(n6446),
    .ZN(n6420) );
OAI211D0 U7214 ( .A1(n8014), .A2(n8143), .B(n6398), .C(n6420)
   , .ZN(n6326) );
AOI211D0 U7215 ( .A1(n7296), .A2(n6321), .B(n8747), .C(n6320)
   , .ZN(n6322) );
AOI31D0 U7216 ( .A1(n6324), .A2(n6323), .A3(n6322), .B(n7489)
   , .ZN(n6325) );
AOI221D0 U7217 ( .A1(n6327), .A2(n6483), .B1(n6326), .B2(
   n6483), .C(n6325),
      .ZN(n6336) );
OAI31D0 U7218 ( .A1(n6329), .A2(n6328), .A3(n6400), .B(n6406)
   , .ZN(n6335) );
```

```
ND4D0 U7219 ( .A1(n6331), .A2(n6330), .A3(n6401), .A4(n6364),
    .ZN(n6332) );
OAI31D0 U7220 ( .A1(n6333), .A2(n6425), .A3(n6332), .B(n7542)
   , .ZN(n6334) );
AOI31D0 U7221 ( .A1(n6336), .A2(n6335), .A3(n6334), .B(n8945)
   , .ZN(n6337) );
AOI211D0 U7222 ( .A1(n8323), .A2(n6339), .B(n6338), .C(n6337)
   , .ZN(n6504) );
CKND2D0 U7223 ( .A1(counter[11]), .A2(counter[5]), .ZN(n8041)
    );
NR2D0 U7224 ( .A1(n9345), .A2(n8041), .ZN(n6480) );
OAI21D0 U7225 ( .A1(n7006), .A2(n6480), .B(n8078), .ZN(n6344)
    );
AOI22D0 U7226 ( .A1(n8042), .A2(n6455), .B1(n8126), .B2(n6480
   ), .ZN(n6493)
       );
NR2D0 U7227 ( .A1(n8130), .A2(n8066), .ZN(n8210) );
INVD0 U7228 ( .I(n8210), .ZN(n6342) );
NR2D0 U7229 ( .A1(n6340), .A2(n8068), .ZN(n6572) );
AOI211D0 U7230 ( .A1(n6455), .A2(n6342), .B(n6341), .C(n6572)
   , .ZN(n6343) );
IND4D0 U7231 ( .A1(n6345), .B1(n6344), .B2(n6493), .B3(n6343)
   , .ZN(n6365) );
CKND2D0 U7232 ( .A1(counter[2]), .A2(counter[0]), .ZN(n7304)
   );
NR2D0 U7233 ( .A1(n8041), .A2(counter[3]), .ZN(n6351) );
INVD0 U7234 ( .I(n6351), .ZN(n6489) );
CKND2D0 U7235 ( .A1(n6347), .A2(n6346), .ZN(n6421) );
OAI211D0 U7236 ( .A1(n7304), .A2(n6489), .B(n6439), .C(n6421)
   , .ZN(n6350) );
OA32D0 U7237 ( .A1(n6455), .A2(n6400), .A3(n6348), .B1(n8193)
   , .B2(n6400),
      .Z(n6349) );
INR4D0 U7238 ( .A1(n6407), .B1(n6365), .B2(n6350), .B3(n6349)
   , .ZN(n6376) );
ND4D0 U7239 ( .A1(n6407), .A2(n6422), .A3(n6413), .A4(n6417),
    .ZN(n6411) );
NR2D0 U7240 ( .A1(n6369), .A2(n6368), .ZN(n6352) );
AOI22D0 U7241 ( .A1(n8046), .A2(n6352), .B1(n8141), .B2(n6351
   ), .ZN(n6356)
       );
NR2D0 U7242 ( .A1(n7421), .A2(n6368), .ZN(n6353) );
NR2D0 U7243 ( .A1(n8041), .A2(n8016), .ZN(n8075) );
NR4D0 U7244 ( .A1(n6354), .A2(n6353), .A3(n7749), .A4(n8075),
    .ZN(n6355) );
IND4D0 U7245 ( .A1(n6411), .B1(n6357), .B2(n6356), .B3(n6355)
   , .ZN(n6367) );
NR2D0 U7246 ( .A1(n8116), .A2(n9347), .ZN(n6469) );
```

```
OAI211D0 U7247 ( .A1(n8186), .A2(n6489), .B(n6359), .C(n6358)
   , .ZN(n6360) );
AOI211D0 U7248 ( .A1(n6362), .A2(n6361), .B(n6469), .C(n6360)
   , .ZN(n6363) );
IIND4D0 U7249 ( .A1(n6453), .A2(n6365), .B1(n6364), .B2(n6363
   ), .ZN(n6366)
      );
AOI22D0 U7250 ( .A1(n7542), .A2(n6367), .B1(n6483), .B2(n6366
   ), .ZN(n6375)
      );
OAI22D0 U7251 ( .A1(n8041), .A2(n7950), .B1(n6369), .B2(n6368
   ), .ZN(n6474)
      );
NR2D0 U7252 ( .A1(n6382), .A2(n6425), .ZN(n6372) );
OA22D0 U7253 ( .A1(n6370), .A2(n8108), .B1(n7029), .B2(n7288)
   , .Z(n6371) );
IND4D0 U7254 ( .A1(n6469), .B1(n6372), .B2(n6371), .B3(n6488)
   , .ZN(n6373) );
OAI21D0 U7255 ( .A1(n6474), .A2(n6373), .B(n6406), .ZN(n6374)
    );
OAI211D0 U7256 ( .A1(n6376), .A2(n7489), .B(n6375), .C(n6374)
   , .ZN(n6436) );
NR2D0 U7257 ( .A1(n6387), .A2(n6377), .ZN(n6472) );
ND4D0 U7258 ( .A1(n6380), .A2(n6379), .A3(n6472), .A4(n6378),
    .ZN(n6384) );
IND4D0 U7259 ( .A1(n6382), .B1(n6381), .B2(n6419), .B3(n6422)
   , .ZN(n6383) );
AOI22D0 U7260 ( .A1(n6483), .A2(n6384), .B1(n7291), .B2(n6383
   ), .ZN(n6396)
      );
AOI21D0 U7261 ( .A1(n7367), .A2(n8050), .B(n8014), .ZN(n6386)
    );
OAI31D0 U7262 ( .A1(n6387), .A2(n6386), .A3(n6385), .B(n7542)
   , .ZN(n6395) );
OAI211D0 U7263 ( .A1(n8276), .A2(n8186), .B(n6389), .C(n6388)
   , .ZN(n6390) );
OAI31D0 U7264 ( .A1(n6392), .A2(n6391), .A3(n6390), .B(n6406)
   , .ZN(n6394) );
AOI31D0 U7265 ( .A1(n6396), .A2(n6395), .A3(n6394), .B(n6393)
   , .ZN(n6435) );
CKND2D0 U7266 ( .A1(n8186), .A2(n8142), .ZN(n8384) );
AOI211D0 U7267 ( .A1(n8009), .A2(n8384), .B(n8420), .C(n6397)
   , .ZN(n6409) );
IND4D0 U7268 ( .A1(n6400), .B1(n6463), .B2(n6399), .B3(n6398)
   , .ZN(n6405) );
INR2D0 U7269 ( .A1(n6401), .B1(n6411), .ZN(n6402) );
ND4D0 U7270 ( .A1(n6428), .A2(n6403), .A3(n6402), .A4(n6418),
    .ZN(n6404) );
```

```
AOI22D0 U7271 ( .A1(n6406), .A2(n6405), .B1(n7291), .B2(n6404
   ), .ZN(n6408)
      );
AOI32D0 U7272 ( .A1(n6409), .A2(n6408), .A3(n6407), .B1(n7293
   ), .B2(n6408),
      .ZN(n6410) );
AOI221D0 U7273 ( .A1(n6412), .A2(n7542), .B1(n6411), .B2(
   n7542), .C(n6410),
      .ZN(n6433) );
ND4D0 U7274 ( .A1(n6416), .A2(n6415), .A3(n6414), .A4(n6413),
    .ZN(n6431) );
AOI31D0 U7275 ( .A1(n6419), .A2(n6418), .A3(n6417), .B(n6494)
   , .ZN(n6430) );
AO31D0 U7276 ( .A1(n6422), .A2(n6421), .A3(n6420), .B(n7293),
    .Z(n6427) );
NR4D0 U7277 ( .A1(n6425), .A2(n6424), .A3(n6423), .A4(n6453),
    .ZN(n6426) );
AOI32D0 U7278 ( .A1(n6428), .A2(n6427), .A3(n6426), .B1(n7297
   ), .B2(n6427),
      .ZN(n6429) );
AOI211D0 U7279 ( .A1(n7291), .A2(n6431), .B(n6430), .C(n6429)
   , .ZN(n6432) );
OAI22D0 U7280 ( .A1(n6433), .A2(n8903), .B1(n6432), .B2(n6986
   ), .ZN(n6434)
      );
AOI211D0 U7281 ( .A1(n8639), .A2(n6436), .B(n6435), .C(n6434)
   , .ZN(n6503) );
AOI21D0 U7282 ( .A1(n6438), .A2(n6437), .B(n7297), .ZN(n6452)
    );
AOI31D0 U7283 ( .A1(n6441), .A2(n6440), .A3(n6439), .B(n6494)
   , .ZN(n6451) );
NR4D0 U7284 ( .A1(n8420), .A2(n6444), .A3(n6443), .A4(n6442),
    .ZN(n6449) );
NR4D0 U7285 ( .A1(n6447), .A2(n6470), .A3(n6446), .A4(n6445),
    .ZN(n6448) );
OAI22D0 U7286 ( .A1(n6449), .A2(n7293), .B1(n6448), .B2(n7489
   ), .ZN(n6450)
      );
OAI31D0 U7287 ( .A1(n6452), .A2(n6451), .A3(n6450), .B(n8910)
   , .ZN(n6502) );
AOI211D0 U7288 ( .A1(n8056), .A2(n8277), .B(n8041), .C(n9345)
   , .ZN(n6454) );
AOI211D0 U7289 ( .A1(n8053), .A2(n6455), .B(n6454), .C(n6453)
   , .ZN(n6465) );
NR2D0 U7290 ( .A1(n8078), .A2(n8126), .ZN(n8117) );
INVD0 U7291 ( .I(n7288), .ZN(n8094) );
AOI211D0 U7292 ( .A1(n8186), .A2(n8983), .B(n8094), .C(n6456)
   , .ZN(n6457) );
```

```
AOI211D0 U7293 ( .A1(n8042), .A2(n6458), .B(n8378), .C(n6457)
   , .ZN(n6459) );
OAI211D0 U7294 ( .A1(n8117), .A2(n6489), .B(n6460), .C(n6459)
   , .ZN(n6491) );
INR4D0 U7295 ( .A1(n6463), .B1(n6462), .B2(n6461), .B3(n6491)
   , .ZN(n6464) );
AOI31D0 U7296 ( .A1(n6465), .A2(n6476), .A3(n6464), .B(n7297)
   , .ZN(n6500) );
CKND2D0 U7297 ( .A1(n8078), .A2(n6480), .ZN(n6466) );
CKND2D0 U7298 ( .A1(n6467), .A2(n6466), .ZN(n6468) );
NR4D0 U7299 ( .A1(n6470), .A2(n6469), .A3(n7749), .A4(n6468),
    .ZN(n6477) );
NR2D0 U7300 ( .A1(n8116), .A2(n7304), .ZN(n8069) );
OAI211D0 U7301 ( .A1(n8050), .A2(n8041), .B(n6472), .C(n6471)
   , .ZN(n6473) );
AOI221D0 U7302 ( .A1(n8069), .A2(n7282), .B1(n6474), .B2(
   n7282), .C(n6473),
      .ZN(n6475) );
AOI31D0 U7303 ( .A1(n6477), .A2(n6476), .A3(n6475), .B(n7489)
   , .ZN(n6499) );
INVD0 U7304 ( .I(n6477), .ZN(n6486) );
INVD0 U7305 ( .I(n7283), .ZN(n7290) );
AOI211D0 U7306 ( .A1(n6480), .A2(n8384), .B(n6479), .C(n6478)
   , .ZN(n6482) );
OAI211D0 U7307 ( .A1(n7290), .A2(n7514), .B(n6482), .C(n6481)
   , .ZN(n6484) );
OAI31D0 U7308 ( .A1(n6486), .A2(n6485), .A3(n6484), .B(n6483)
   , .ZN(n6496) );
OAI211D0 U7309 ( .A1(n8210), .A2(n6489), .B(n6488), .C(n6487)
   , .ZN(n6490) );
INR4D0 U7310 ( .A1(n6493), .B1(n6492), .B2(n6491), .B3(n6490)
   , .ZN(n6495) );
AOI32D0 U7311 ( .A1(n6497), .A2(n6496), .A3(n6495), .B1(n6494
   ), .B2(n6496),
      .ZN(n6498) );
OAI31D0 U7312 ( .A1(n6500), .A2(n6499), .A3(n6498), .B(n8918)
   , .ZN(n6501) );
ND4D0 U7313 ( .A1(n6504), .A2(n6503), .A3(n6502), .A4(n6501),
    .ZN(n6505) );
NR4D0 U7314 ( .A1(n6508), .A2(n6507), .A3(n6506), .A4(n6505),
    .ZN(n6509) );
OAI222D0 U7315 ( .A1(n6511), .A2(n7277), .B1(n7000), .B2(
   n6510), .C1(n6992),
      .C2(n6509), .ZN(n4815) );
INVD0 U7316 ( .I(q_out[4]), .ZN(n6994) );
NR4D0 U7317 ( .A1(n7587), .A2(n7804), .A3(n7813), .A4(n7784),
    .ZN(n7664) );
```

```
ND4D0 U7318 ( .A1(n6538), .A2(n7780), .A3(n7729), .A4(n6648),
    .ZN(n7846) );
CKND2D0 U7319 ( .A1(n7642), .A2(n7912), .ZN(n7848) );
INVD0 U7320 ( .I(n7848), .ZN(n7757) );
OAI211D0 U7321 ( .A1(n6513), .A2(n6512), .B(n7757), .C(n6523)
   , .ZN(n6514) );
AOI22D0 U7322 ( .A1(n9226), .A2(n7846), .B1(n9307), .B2(n6514
   ), .ZN(n6518)
       );
CKND2D0 U7323 ( .A1(n7636), .A2(n7959), .ZN(n7900) );
NR2D0 U7324 ( .A1(n7911), .A2(n7898), .ZN(n7886) );
OAI211D0 U7325 ( .A1(n6515), .A2(n7743), .B(n7886), .C(n7668)
   , .ZN(n6516) );
OAI31D0 U7326 ( .A1(n7766), .A2(n7900), .A3(n6516), .B(n9312)
   , .ZN(n6517) );
OAI211D0 U7327 ( .A1(n7664), .A2(n9300), .B(n6518), .C(n6517)
   , .ZN(n6714) );
INVD0 U7328 ( .I(n6694), .ZN(n7955) );
NR3D0 U7329 ( .A1(n8672), .A2(n7635), .A3(n6519), .ZN(n7568)
   );
AN3D0 U7330 ( .A1(n7955), .A2(n7568), .A3(n7886), .Z(n7916) )
   ;
CKND2D0 U7331 ( .A1(n6520), .A2(n7916), .ZN(n6525) );
NR2D0 U7332 ( .A1(n7819), .A2(n6562), .ZN(n6522) );
ND4D0 U7333 ( .A1(n6666), .A2(n6522), .A3(n6521), .A4(n7780),
    .ZN(n6524) );
CKND2D0 U7334 ( .A1(n7642), .A2(n6523), .ZN(n7811) );
NR2D0 U7335 ( .A1(n6589), .A2(n7911), .ZN(n6650) );
IND3D0 U7336 ( .A1(n7811), .B1(n6650), .B2(n6609), .ZN(n7829)
    );
AOI222D0 U7337 ( .A1(n6525), .A2(n9307), .B1(n6524), .B2(
   n9312), .C1(n7829),
      .C2(n9141), .ZN(n6547) );
AOI31D0 U7338 ( .A1(n7876), .A2(n7581), .A3(n6526), .B(n9274)
   , .ZN(n6535) );
CKND2D0 U7339 ( .A1(n6648), .A2(n6527), .ZN(n7837) );
NR4D0 U7340 ( .A1(n7676), .A2(n7647), .A3(n7837), .A4(n6528),
    .ZN(n6533) );
CKND2D0 U7341 ( .A1(n7826), .A2(n6529), .ZN(n7985) );
NR2D0 U7342 ( .A1(n7985), .A2(n7814), .ZN(n7892) );
INVD0 U7343 ( .I(n7892), .ZN(n7927) );
CKND2D0 U7344 ( .A1(n7577), .A2(n7772), .ZN(n7648) );
OAI31D0 U7345 ( .A1(n6530), .A2(n7927), .A3(n7648), .B(n9141)
   , .ZN(n6532) );
INVD0 U7346 ( .I(n7799), .ZN(n7974) );
CKND2D0 U7347 ( .A1(n7925), .A2(n7974), .ZN(n6626) );
OAI31D0 U7348 ( .A1(n7648), .A2(n6617), .A3(n6626), .B(n9307)
   , .ZN(n6531) );
```

```
OAI211D0 U7349 ( .A1(n6533), .A2(n9318), .B(n6532), .C(n6531)
   , .ZN(n6534) );
OAI21D0 U7350 ( .A1(n6535), .A2(n6534), .B(n8803), .ZN(n6546)
    );
NR2D0 U7351 ( .A1(n7814), .A2(n7766), .ZN(n7858) );
NR2D0 U7352 ( .A1(n6589), .A2(n7854), .ZN(n7935) );
AOI31D0 U7353 ( .A1(n7858), .A2(n7935), .A3(n7641), .B(n9274)
   , .ZN(n6544) );
AOI31D0 U7354 ( .A1(n7687), .A2(n6537), .A3(n6536), .B(n9272)
   , .ZN(n6543) );
NR2D0 U7355 ( .A1(n6620), .A2(n7634), .ZN(n6633) );
IINR4D0 U7356 ( .A1(n6633), .A2(n7989), .B1(n6589), .B2(n7648
   ), .ZN(n6541)
      );
CKND2D0 U7357 ( .A1(n6538), .A2(n7915), .ZN(n7803) );
NR2D0 U7358 ( .A1(n7899), .A2(n6616), .ZN(n7949) );
IND4D0 U7359 ( .A1(n7803), .B1(n7949), .B2(n6650), .B3(n6539)
   , .ZN(n6540) );
MOAI22D0 U7360 ( .A1(n6541), .A2(n9318), .B1(n9141), .B2(
   n6540), .ZN(n6542)
      );
OAI31D0 U7361 ( .A1(n6544), .A2(n6543), .A3(n6542), .B(n8408)
   , .ZN(n6545) );
OAI211D0 U7362 ( .A1(n6547), .A2(n8848), .B(n6546), .C(n6545)
   , .ZN(n6713) );
CKND2D0 U7363 ( .A1(n6548), .A2(n6615), .ZN(n7824) );
CKND2D0 U7364 ( .A1(n7850), .A2(n7758), .ZN(n7910) );
INVD0 U7365 ( .I(n7910), .ZN(n6663) );
NR2D0 U7366 ( .A1(n7804), .A2(n7898), .ZN(n7869) );
IND4D0 U7367 ( .A1(n7824), .B1(n6663), .B2(n7869), .B3(n6648)
   , .ZN(n6554) );
NR4D0 U7368 ( .A1(n7640), .A2(n7659), .A3(n7689), .A4(n7621),
    .ZN(n6549) );
AOI21D0 U7369 ( .A1(n7869), .A2(n6549), .B(n9318), .ZN(n6553)
    );
NR3D0 U7370 ( .A1(n7675), .A2(n6550), .A3(n7811), .ZN(n7805)
   );
NR4D0 U7371 ( .A1(n7585), .A2(n7819), .A3(n6562), .A4(n7640),
    .ZN(n6551) );
OAI22D0 U7372 ( .A1(n7805), .A2(n9272), .B1(n6551), .B2(n9274
   ), .ZN(n6552)
      );
AOI211D0 U7373 ( .A1(n9141), .A2(n6554), .B(n6553), .C(n6552)
   , .ZN(n6711) );
NR2D0 U7374 ( .A1(n6556), .A2(n6555), .ZN(n7960) );
OAI211D0 U7375 ( .A1(n6559), .A2(n6558), .B(n6557), .C(n6648)
   , .ZN(n7727) );
```

```
AOI22D0 U7376 ( .A1(n9226), .A2(n6560), .B1(n9307), .B2(n7727
   ), .ZN(n6564)
      );
CKND2D0 U7377 ( .A1(n7844), .A2(n7873), .ZN(n7963) );
INVD0 U7378 ( .I(n7963), .ZN(n7782) );
ND4D0 U7379 ( .A1(n7782), .A2(n6561), .A3(n7641), .A4(n7912),
    .ZN(n7620) );
OAI21D0 U7380 ( .A1(n6562), .A2(n7620), .B(n9141), .ZN(n6563)
    );
OAI211D0 U7381 ( .A1(n7960), .A2(n9274), .B(n6564), .C(n6563)
   , .ZN(n6608) );
CKND2D0 U7382 ( .A1(n8314), .A2(n7772), .ZN(n7713) );
NR4D0 U7383 ( .A1(n7596), .A2(n7634), .A3(n7766), .A4(n7713),
    .ZN(n6566) );
CKND2D0 U7384 ( .A1(n7939), .A2(n9226), .ZN(n6565) );
AOI31D0 U7385 ( .A1(n7758), .A2(n7868), .A3(n6566), .B(n6565)
   , .ZN(n6607) );
CKND2D0 U7386 ( .A1(n7958), .A2(n7915), .ZN(n7719) );
INVD0 U7387 ( .I(n7719), .ZN(n6657) );
AOI22D0 U7388 ( .A1(n7296), .A2(n6567), .B1(n6573), .B2(n7360
   ), .ZN(n6571)
      );
NR2D0 U7389 ( .A1(n7911), .A2(n6568), .ZN(n7649) );
NR2D0 U7390 ( .A1(n7321), .A2(n6656), .ZN(n6569) );
CKND2D0 U7391 ( .A1(n7285), .A2(n7290), .ZN(n7977) );
INVD0 U7392 ( .I(n7977), .ZN(n6649) );
NR2D0 U7393 ( .A1(n6649), .A2(n6656), .ZN(n7677) );
INR4D0 U7394 ( .A1(n7649), .B1(n7634), .B2(n6569), .B3(n7677)
   , .ZN(n6570) );
ND4D0 U7395 ( .A1(n6657), .A2(n7858), .A3(n6571), .A4(n6570),
    .ZN(n6588) );
OR3D0 U7396 ( .A1(n7770), .A2(n6579), .A3(n7684), .Z(n7995) )
   ;
AOI211D0 U7397 ( .A1(n6573), .A2(n8046), .B(n6572), .C(n7995)
   , .ZN(n6574) );
AOI31D0 U7398 ( .A1(n6575), .A2(n6574), .A3(n7825), .B(n9318)
   , .ZN(n6587) );
CKND2D0 U7399 ( .A1(n7715), .A2(n6581), .ZN(n7952) );
OR4D0 U7400 ( .A1(n6576), .A2(n7814), .A3(n6579), .A4(n7741),
    .Z(n6577) );
OAI31D0 U7401 ( .A1(n7621), .A2(n7952), .A3(n6577), .B(n9141)
   , .ZN(n6585) );
NR2D0 U7402 ( .A1(n8055), .A2(n7972), .ZN(n6660) );
NR3D0 U7403 ( .A1(n9348), .A2(n6592), .A3(n6578), .ZN(n7707)
   );
INR3D0 U7404 ( .A1(n7731), .B1(n6579), .B2(n7707), .ZN(n7681)
    );
```

```
ND4D0 U7405 ( .A1(n7681), .A2(n7642), .A3(n6581), .A4(n6580),
    .ZN(n6582) );
NR4D0 U7406 ( .A1(n6660), .A2(n7813), .A3(n6583), .A4(n6582),
    .ZN(n6584) );
AOI32D0 U7407 ( .A1(n7990), .A2(n6585), .A3(n6584), .B1(n9272
   ), .B2(n6585),
      .ZN(n6586) );
AOI211D0 U7408 ( .A1(n9312), .A2(n6588), .B(n6587), .C(n6586)
   , .ZN(n6605) );
AOI221D0 U7409 ( .A1(n8951), .A2(n7709), .B1(n7413), .B2(
   n7709), .C(n7739),
      .ZN(n6591) );
NR2D0 U7410 ( .A1(n6589), .A2(n7585), .ZN(n7918) );
CKND2D0 U7411 ( .A1(n7795), .A2(n7918), .ZN(n7658) );
NR4D0 U7412 ( .A1(n7819), .A2(n6597), .A3(n7658), .A4(n7818),
    .ZN(n6590) );
OAI211D0 U7413 ( .A1(n6592), .A2(n6656), .B(n6591), .C(n6590)
   , .ZN(n6602) );
NR4D0 U7414 ( .A1(n7767), .A2(n6686), .A3(n7799), .A4(n7811),
    .ZN(n6593) );
AOI21D0 U7415 ( .A1(n7989), .A2(n6593), .B(n9274), .ZN(n6601)
    );
OAI211D0 U7416 ( .A1(n8038), .A2(n6595), .B(n6666), .C(n6594)
   , .ZN(n6596) );
NR4D0 U7417 ( .A1(n7609), .A2(n6662), .A3(n6597), .A4(n6596),
    .ZN(n6599) );
NR4D0 U7418 ( .A1(n6632), .A2(n6620), .A3(n7824), .A4(n6612),
    .ZN(n6598) );
OAI22D0 U7419 ( .A1(n6599), .A2(n9318), .B1(n6598), .B2(n9300
   ), .ZN(n6600)
      );
AOI211D0 U7420 ( .A1(n9307), .A2(n6602), .B(n6601), .C(n6600)
   , .ZN(n6603) );
OAI22D0 U7421 ( .A1(n6605), .A2(n6604), .B1(n6603), .B2(n8797
   ), .ZN(n6606)
      );
AOI211D0 U7422 ( .A1(n8105), .A2(n6608), .B(n6607), .C(n6606)
   , .ZN(n6710) );
ND3D0 U7423 ( .A1(n7938), .A2(n6650), .A3(n7955), .ZN(n7882)
   );
NR2D0 U7424 ( .A1(n7771), .A2(n6620), .ZN(n7893) );
CKND2D0 U7425 ( .A1(n7671), .A2(n6609), .ZN(n7871) );
INR4D0 U7426 ( .A1(n6693), .B1(n7580), .B2(n7871), .B3(n6610)
   , .ZN(n7738) );
OAI22D0 U7427 ( .A1(n7893), .A2(n9318), .B1(n7738), .B2(n9272
   ), .ZN(n6611)
      );
```

```
AOI221D0 U7428 ( .A1(n8672), .A2(n9141), .B1(n7882), .B2(
   n9141), .C(n6611),
      .ZN(n6614) );
AOI211D0 U7429 ( .A1(n6638), .A2(n8983), .B(n6612), .C(n7853)
   , .ZN(n6613) );
AOI32D0 U7430 ( .A1(n6615), .A2(n6614), .A3(n6613), .B1(n9274
   ), .B2(n6614),
      .ZN(n6708) );
INR2D0 U7431 ( .A1(n7642), .B1(n6616), .ZN(n7590) );
NR2D0 U7432 ( .A1(n7770), .A2(n6620), .ZN(n7650) );
ND4D0 U7433 ( .A1(n7590), .A2(n7650), .A3(n7850), .A4(n7959),
    .ZN(n6624) );
CKND2D0 U7434 ( .A1(n7641), .A2(n7678), .ZN(n7889) );
NR2D0 U7435 ( .A1(n7724), .A2(n7889), .ZN(n7978) );
AOI31D0 U7436 ( .A1(n7978), .A2(n7979), .A3(n7989), .B(n9272)
   , .ZN(n6623) );
NR4D0 U7437 ( .A1(n7814), .A2(n7819), .A3(n7770), .A4(n6617),
    .ZN(n7775) );
NR2D0 U7438 ( .A1(n8950), .A2(n6618), .ZN(n8225) );
NR4D0 U7439 ( .A1(n8225), .A2(n6620), .A3(n7910), .A4(n6619),
    .ZN(n6621) );
OAI22D0 U7440 ( .A1(n7775), .A2(n9300), .B1(n6621), .B2(n9318
   ), .ZN(n6622)
      );
AOI211D0 U7441 ( .A1(n9312), .A2(n6624), .B(n6623), .C(n6622)
   , .ZN(n6674) );
INVD0 U7442 ( .I(n7606), .ZN(n7839) );
INR4D0 U7443 ( .A1(n7568), .B1(n7608), .B2(n7839), .B3(n6625)
   , .ZN(n6637) );
INVD0 U7444 ( .I(n6626), .ZN(n7661) );
NR3D0 U7445 ( .A1(n7586), .A2(n7784), .A3(n7812), .ZN(n7625)
   );
INVD0 U7446 ( .I(n7813), .ZN(n7922) );
ND4D0 U7447 ( .A1(n7661), .A2(n7625), .A3(n7844), .A4(n7922),
    .ZN(n6630) );
CKND2D0 U7448 ( .A1(n7858), .A2(n7641), .ZN(n6627) );
INR2D0 U7449 ( .A1(n7590), .B1(n6627), .ZN(n7936) );
ND4D0 U7450 ( .A1(n7936), .A2(n6628), .A3(n7969), .A4(n7729),
    .ZN(n6629) );
AOI22D0 U7451 ( .A1(n9307), .A2(n6630), .B1(n9141), .B2(n6629
   ), .ZN(n6636)
      );
CKND2D0 U7452 ( .A1(n6631), .A2(n7772), .ZN(n7874) );
NR2D0 U7453 ( .A1(n6632), .A2(n7874), .ZN(n7897) );
CKND2D0 U7454 ( .A1(n7959), .A2(n7897), .ZN(n7830) );
ND4D0 U7455 ( .A1(n6633), .A2(n7780), .A3(n7642), .A4(n7795),
    .ZN(n6634) );
```

```
OAI31D0 U7456 ( .A1(n7830), .A2(n7839), .A3(n6634), .B(n9312)
   , .ZN(n6635) );
OAI211D0 U7457 ( .A1(n6637), .A2(n9318), .B(n6636), .C(n6635)
   , .ZN(n6645) );
INVD0 U7458 ( .I(n7872), .ZN(n7982) );
AOI211D0 U7459 ( .A1(n6639), .A2(n6638), .B(n7982), .C(n7871)
   , .ZN(n7792) );
CKND2D0 U7460 ( .A1(n7729), .A2(n7721), .ZN(n7778) );
IND4D0 U7461 ( .A1(n7778), .B1(n7782), .B2(n7869), .B3(n6640)
   , .ZN(n6641) );
CKND2D0 U7462 ( .A1(n7795), .A2(n7912), .ZN(n7735) );
AOI22D0 U7463 ( .A1(n9226), .A2(n6641), .B1(n9312), .B2(n7735
   ), .ZN(n6643)
      );
INVD0 U7464 ( .I(n7900), .ZN(n7603) );
ND3D0 U7465 ( .A1(n7603), .A2(n7772), .A3(n7922), .ZN(n7674)
   );
CKND2D0 U7466 ( .A1(n7925), .A2(n7850), .ZN(n7764) );
OAI21D0 U7467 ( .A1(n7674), .A2(n7764), .B(n9307), .ZN(n6642)
    );
OAI211D0 U7468 ( .A1(n7792), .A2(n9300), .B(n6643), .C(n6642)
   , .ZN(n6644) );
AOI22D0 U7469 ( .A1(n8715), .A2(n6645), .B1(n8656), .B2(n6644
   ), .ZN(n6673)
      );
AOI211D0 U7470 ( .A1(n6647), .A2(n9014), .B(n7722), .C(n6646)
   , .ZN(n7742) );
OAI211D0 U7471 ( .A1(n6649), .A2(n7972), .B(n6648), .C(n7825)
   , .ZN(n7682) );
INR4D0 U7472 ( .A1(n6650), .B1(n7964), .B2(n7647), .B3(n7682)
   , .ZN(n6651) );
AOI31D0 U7473 ( .A1(n6676), .A2(n7742), .A3(n6651), .B(n9318)
   , .ZN(n6671) );
INVD0 U7474 ( .I(n7683), .ZN(n7720) );
OAI21D0 U7475 ( .A1(n8966), .A2(n9329), .B(n7720), .ZN(n7984)
    );
AOI211D0 U7476 ( .A1(n8052), .A2(n7360), .B(n7984), .C(n7647)
   , .ZN(n6652) );
CKND2D0 U7477 ( .A1(n6652), .A2(n7641), .ZN(n7759) );
CKND2D0 U7478 ( .A1(counter[3]), .A2(n6660), .ZN(n7740) );
IOA21D0 U7479 ( .A1(n7360), .A2(n7709), .B(n7740), .ZN(n6653)
    );
NR4D0 U7480 ( .A1(n7767), .A2(n6654), .A3(n7759), .A4(n6653),
    .ZN(n6655) );
OAI21D0 U7481 ( .A1(n7314), .A2(n7414), .B(n7986), .ZN(n7711)
    );
AOI31D0 U7482 ( .A1(n6655), .A2(n7955), .A3(n7711), .B(n9272)
   , .ZN(n6670) );
```

```
NR2D0 U7483 ( .A1(n7743), .A2(n6656), .ZN(n7965) );
NR2D0 U7484 ( .A1(n7314), .A2(n7414), .ZN(n6658) );
OAI211D0 U7485 ( .A1(n6658), .A2(n8054), .B(n6657), .C(n7715)
   , .ZN(n7981) );
CKND2D0 U7486 ( .A1(n8997), .A2(n6878), .ZN(n6659) );
IND4D0 U7487 ( .A1(n6660), .B1(n7925), .B2(n7581), .B3(n6659)
   , .ZN(n6661) );
NR4D0 U7488 ( .A1(n7890), .A2(n7965), .A3(n7981), .A4(n6661),
    .ZN(n6668) );
NR2D0 U7489 ( .A1(n7586), .A2(n6662), .ZN(n7797) );
ND4D0 U7490 ( .A1(n6663), .A2(n7797), .A3(n7711), .A4(n7774),
    .ZN(n6664) );
INR4D0 U7491 ( .A1(n6666), .B1(n7804), .B2(n6665), .B3(n6664)
   , .ZN(n6667) );
OAI22D0 U7492 ( .A1(n6668), .A2(n9300), .B1(n6667), .B2(n9274
   ), .ZN(n6669)
      );
OAI31D0 U7493 ( .A1(n6671), .A2(n6670), .A3(n6669), .B(n8918)
   , .ZN(n6672) );
OAI211D0 U7494 ( .A1(n6674), .A2(n8708), .B(n6673), .C(n6672)
   , .ZN(n6707) );
INVD0 U7495 ( .I(n7784), .ZN(n7617) );
ND4D0 U7496 ( .A1(n7649), .A2(n8394), .A3(n7642), .A4(n7617),
    .ZN(n6682) );
NR2D0 U7497 ( .A1(n7839), .A2(n7899), .ZN(n7866) );
INVD0 U7498 ( .I(n7597), .ZN(n6675) );
AOI31D0 U7499 ( .A1(n6676), .A2(n7866), .A3(n6675), .B(n9318)
   , .ZN(n6681) );
AN4D0 U7500 ( .A1(n7917), .A2(n7850), .A3(n6677), .A4(n7757),
    .Z(n7566) );
OR2D0 U7501 ( .A1(n7724), .A2(n7941), .Z(n7888) );
NR2D0 U7502 ( .A1(n7689), .A2(n6695), .ZN(n7730) );
CKND2D0 U7503 ( .A1(n7730), .A2(n7825), .ZN(n7646) );
NR4D0 U7504 ( .A1(n6678), .A2(n7911), .A3(n7888), .A4(n7646),
    .ZN(n6679) );
OAI22D0 U7505 ( .A1(n7566), .A2(n9300), .B1(n6679), .B2(n9272
   ), .ZN(n6680)
      );
AOI211D0 U7506 ( .A1(n9312), .A2(n6682), .B(n6681), .C(n6680)
   , .ZN(n6705) );
AOI31D0 U7507 ( .A1(n6684), .A2(n7758), .A3(n6683), .B(n9318)
   , .ZN(n6691) );
OA31D0 U7508 ( .A1(n7579), .A2(n6686), .A3(n6685), .B(n9312),
    .Z(n6690) );
ND3D0 U7509 ( .A1(n7757), .A2(n7869), .A3(n7922), .ZN(n7798)
   );
INVD0 U7510 ( .I(n7798), .ZN(n7841) );
```

```
NR4D0 U7511 ( .A1(n7578), .A2(n7634), .A3(n7605), .A4(n6687),
    .ZN(n6688) );
OAI22D0 U7512 ( .A1(n7841), .A2(n9272), .B1(n6688), .B2(n9300
   ), .ZN(n6689)
      );
OAI31D0 U7513 ( .A1(n6691), .A2(n6690), .A3(n6689), .B(n8323)
   , .ZN(n6704) );
CKND2D0 U7514 ( .A1(n8394), .A2(n7842), .ZN(n7584) );
NR4D0 U7515 ( .A1(n7578), .A2(n7735), .A3(n7584), .A4(n7764),
    .ZN(n6692) );
AOI31D0 U7516 ( .A1(n7606), .A2(n7581), .A3(n6692), .B(n9300)
   , .ZN(n6702) );
ND3D0 U7517 ( .A1(n7873), .A2(n7603), .A3(n6693), .ZN(n7908)
   );
OA31D0 U7518 ( .A1(n6694), .A2(n7812), .A3(n7908), .B(n9312),
    .Z(n6701) );
INR4D0 U7519 ( .A1(n6697), .B1(n6696), .B2(n6695), .B3(n7648)
   , .ZN(n6699) );
INVD0 U7520 ( .I(n7978), .ZN(n7896) );
NR4D0 U7521 ( .A1(n7766), .A2(n7896), .A3(n7812), .A4(n7662),
    .ZN(n6698) );
OAI22D0 U7522 ( .A1(n6699), .A2(n9318), .B1(n6698), .B2(n9272
   ), .ZN(n6700)
      );
OAI31D0 U7523 ( .A1(n6702), .A2(n6701), .A3(n6700), .B(n7933)
   , .ZN(n6703) );
OAI211D0 U7524 ( .A1(n6705), .A2(n7593), .B(n6704), .C(n6703)
   , .ZN(n6706) );
AOI211D0 U7525 ( .A1(n8740), .A2(n6708), .B(n6707), .C(n6706)
   , .ZN(n6709) );
OAI211D0 U7526 ( .A1(n6711), .A2(n6986), .B(n6710), .C(n6709)
   , .ZN(n6712) );
AOI211D0 U7527 ( .A1(n8910), .A2(n6714), .B(n6713), .C(n6712)
   , .ZN(n6993) );
AN4D0 U7528 ( .A1(n6716), .A2(n6715), .A3(n6890), .A4(n6774),
    .Z(n6723) );
CKND2D0 U7529 ( .A1(n6776), .A2(n6717), .ZN(n6911) );
OR4D0 U7530 ( .A1(n6911), .A2(n8598), .A3(n6951), .A4(n8648),
    .Z(n6719) );
AOI22D0 U7531 ( .A1(n8582), .A2(n6719), .B1(n8609), .B2(n6718
   ), .ZN(n6722)
      );
CKND2D0 U7532 ( .A1(n6962), .A2(n8702), .ZN(n6924) );
OAI31D0 U7533 ( .A1(n8504), .A2(n6720), .A3(n6924), .B(n8899)
   , .ZN(n6721) );
OAI211D0 U7534 ( .A1(n6723), .A2(n8923), .B(n6722), .C(n6721)
   , .ZN(n6990) );
```

```
ND4D0 U7535 ( .A1(n6775), .A2(n6725), .A3(n6724), .A4(n6926),
    .ZN(n6736) );
ND4D0 U7536 ( .A1(n6727), .A2(n6788), .A3(n6726), .A4(n6935),
    .ZN(n6730) );
NR2D0 U7537 ( .A1(n6863), .A2(n8744), .ZN(n6728) );
ND4D0 U7538 ( .A1(n6749), .A2(n6728), .A3(n8883), .A4(n8813),
    .ZN(n6729) );
AOI22D0 U7539 ( .A1(n8582), .A2(n6730), .B1(n8822), .B2(n6729
   ), .ZN(n6734)
      );
CKND2D0 U7540 ( .A1(n6930), .A2(n8514), .ZN(n6916) );
NR4D0 U7541 ( .A1(n6732), .A2(n8747), .A3(n6731), .A4(n6916),
    .ZN(n6733) );
AOI32D0 U7542 ( .A1(n8703), .A2(n6734), .A3(n6733), .B1(n8923
   ), .B2(n6734),
      .ZN(n6735) );
AOI221D0 U7543 ( .A1(n8748), .A2(n8899), .B1(n6736), .B2(
   n8899), .C(n6735),
      .ZN(n6760) );
AOI31D0 U7544 ( .A1(n6738), .A2(n6880), .A3(n6737), .B(n8923)
   , .ZN(n6748) );
AOI31D0 U7545 ( .A1(n6741), .A2(n6740), .A3(n6739), .B(n8937)
   , .ZN(n6747) );
ND4D0 U7546 ( .A1(n6743), .A2(n6742), .A3(n8788), .A4(n8576),
    .ZN(n6745) );
ND4D0 U7547 ( .A1(n8809), .A2(n8730), .A3(n8813), .A4(n8788),
    .ZN(n6744) );
AO22D0 U7548 ( .A1(n8685), .A2(n6745), .B1(n8609), .B2(n6744)
   , .Z(n6746) );
OAI31D0 U7549 ( .A1(n6748), .A2(n6747), .A3(n6746), .B(n7939)
   , .ZN(n6759) );
AOI31D0 U7550 ( .A1(n6750), .A2(n6749), .A3(n6835), .B(n8919)
   , .ZN(n6757) );
AOI31D0 U7551 ( .A1(n6752), .A2(n6930), .A3(n6751), .B(n8937)
   , .ZN(n6756) );
AN4D0 U7552 ( .A1(n8576), .A2(n8601), .A3(n6789), .A4(n6753),
    .Z(n6754) );
OAI22D0 U7553 ( .A1(n6856), .A2(n8923), .B1(n6754), .B2(n8935
   ), .ZN(n6755)
      );
OAI31D0 U7554 ( .A1(n6757), .A2(n6756), .A3(n6755), .B(n8408)
   , .ZN(n6758) );
OAI211D0 U7555 ( .A1(n6760), .A2(n8230), .B(n6759), .C(n6758)
   , .ZN(n6989) );
CKND2D0 U7556 ( .A1(n8556), .A2(n6857), .ZN(n6782) );
IND4D0 U7557 ( .A1(n6782), .B1(n8518), .B2(n6761), .B3(n8512)
   , .ZN(n6772) );
```

```
AOI31D0 U7558 ( .A1(n6763), .A2(n8844), .A3(n6762), .B(n8919)
   , .ZN(n6771) );
CKND2D0 U7559 ( .A1(n8699), .A2(n6847), .ZN(n6861) );
ND4D0 U7560 ( .A1(n6765), .A2(n8517), .A3(n8520), .A4(n6764),
    .ZN(n6766) );
OAI31D0 U7561 ( .A1(n6861), .A2(n6767), .A3(n6766), .B(n8697)
   , .ZN(n6769) );
NR4D0 U7562 ( .A1(n8632), .A2(n8503), .A3(n6783), .A4(n6861),
    .ZN(n6768) );
AOI32D0 U7563 ( .A1(n6867), .A2(n6769), .A3(n6768), .B1(n8674
   ), .B2(n6769),
      .ZN(n6770) );
AOI211D0 U7564 ( .A1(n8582), .A2(n6772), .B(n6771), .C(n6770)
   , .ZN(n6987) );
ND4D0 U7565 ( .A1(n6775), .A2(n6774), .A3(n6870), .A4(n6773),
    .ZN(n6781) );
AOI31D0 U7566 ( .A1(n8703), .A2(n6777), .A3(n6776), .B(n8923)
   , .ZN(n6780) );
AOI31D0 U7567 ( .A1(n6778), .A2(n6871), .A3(n6786), .B(n8937)
   , .ZN(n6779) );
AOI211D0 U7568 ( .A1(n8582), .A2(n6781), .B(n6780), .C(n6779)
   , .ZN(n6785) );
NR4D0 U7569 ( .A1(n6783), .A2(n6917), .A3(n6862), .A4(n6782),
    .ZN(n6784) );
AOI32D0 U7570 ( .A1(n6954), .A2(n6785), .A3(n6784), .B1(n8919
   ), .B2(n6785),
      .ZN(n6811) );
NR2D0 U7571 ( .A1(n8504), .A2(n6957), .ZN(n6787) );
ND4D0 U7572 ( .A1(n6788), .A2(n6850), .A3(n6787), .A4(n6786),
    .ZN(n6791) );
ND4D0 U7573 ( .A1(n6789), .A2(n6886), .A3(n8512), .A4(n8724),
    .ZN(n6790) );
AOI22D0 U7574 ( .A1(n8822), .A2(n6791), .B1(n8899), .B2(n6790
   ), .ZN(n6798)
      );
OAI31D0 U7575 ( .A1(n6895), .A2(n6793), .A3(n6792), .B(n8582)
   , .ZN(n6797) );
OAI21D0 U7576 ( .A1(n6924), .A2(n6794), .B(n8697), .ZN(n6796)
    );
AOI31D0 U7577 ( .A1(n6798), .A2(n6797), .A3(n6796), .B(n6795)
   , .ZN(n6810) );
ND4D0 U7578 ( .A1(n6871), .A2(n6851), .A3(n6799), .A4(n8617),
    .ZN(n6800) );
AOI22D0 U7579 ( .A1(n8582), .A2(n6801), .B1(n8899), .B2(n6800
   ), .ZN(n6808)
      );
OAI31D0 U7580 ( .A1(n6911), .A2(n6802), .A3(n6861), .B(n8697)
   , .ZN(n6807) );
```

```
OAI31D0 U7581 ( .A1(n6805), .A2(n6804), .A3(n6803), .B(n8609)
   , .ZN(n6806) );
AOI31D0 U7582 ( .A1(n6808), .A2(n6807), .A3(n6806), .B(n7593)
   , .ZN(n6809) );
AOI211D0 U7583 ( .A1(n8656), .A2(n6811), .B(n6810), .C(n6809)
   , .ZN(n6985) );
AOI211D0 U7584 ( .A1(n6812), .A2(n8309), .B(n6956), .C(n8505)
   , .ZN(n6815) );
IOA21D0 U7585 ( .A1(n8725), .A2(n8532), .B(n6813), .ZN(n6821)
    );
ND4D0 U7586 ( .A1(n6816), .A2(n6815), .A3(n6814), .A4(n6821),
    .ZN(n6817) );
NR4D0 U7587 ( .A1(n8755), .A2(n6818), .A3(n6841), .A4(n6817),
    .ZN(n6846) );
NR4D0 U7588 ( .A1(n8566), .A2(n6820), .A3(n6819), .A4(n8772),
    .ZN(n6822) );
ND4D0 U7589 ( .A1(n6828), .A2(n6823), .A3(n6822), .A4(n6821),
    .ZN(n6834) );
INVD0 U7590 ( .I(n6824), .ZN(n6879) );
AOI211D0 U7591 ( .A1(n6879), .A2(n6825), .B(n8771), .C(n6862)
   , .ZN(n6831) );
INR4D0 U7592 ( .A1(n6828), .B1(n6917), .B2(n6827), .B3(n6826)
   , .ZN(n6830) );
CKND2D0 U7593 ( .A1(n8765), .A2(n8764), .ZN(n6829) );
ND4D0 U7594 ( .A1(n6832), .A2(n6831), .A3(n6830), .A4(n6829),
    .ZN(n6833) );
AOI22D0 U7595 ( .A1(n8685), .A2(n6834), .B1(n8697), .B2(n6833
   ), .ZN(n6845)
      );
INVD0 U7596 ( .I(n6835), .ZN(n6843) );
NR4D0 U7597 ( .A1(n8753), .A2(n6837), .A3(n4819), .A4(n6836),
    .ZN(n6839) );
IND4D0 U7598 ( .A1(n6841), .B1(n6840), .B2(n6839), .B3(n6838)
   , .ZN(n6842) );
OAI31D0 U7599 ( .A1(n6843), .A2(n8772), .A3(n6842), .B(n8822)
   , .ZN(n6844) );
OAI211D0 U7600 ( .A1(n6846), .A2(n8937), .B(n6845), .C(n6844)
   , .ZN(n6983) );
ND4D0 U7601 ( .A1(n6886), .A2(n6971), .A3(n6848), .A4(n6847),
    .ZN(n6854) );
AOI31D0 U7602 ( .A1(n8699), .A2(n6849), .A3(n8920), .B(n8935)
   , .ZN(n6853) );
AOI31D0 U7603 ( .A1(n6851), .A2(n8506), .A3(n6850), .B(n8674)
   , .ZN(n6852) );
AOI211D0 U7604 ( .A1(n8609), .A2(n6854), .B(n6853), .C(n6852)
   , .ZN(n6907) );
CKND2D0 U7605 ( .A1(n6927), .A2(n8519), .ZN(n8510) );
```

```
  IINR4D0 U7606 ( .A1(n6954), .A2(n6856), .B1(n8510), .B2(n6855
     ), .ZN(n6866)
       );
  CKAN2D0 U7607 ( .A1(n6857), .A2(n8724), .Z(n6877) );
  ND4D0 U7608 ( .A1(n6877), .A2(n6858), .A3(n8883), .A4(n8691),
      .ZN(n6860) );
  AOI22D0 U7609 ( .A1(n8899), .A2(n6860), .B1(n8697), .B2(n6859
     ), .ZN(n6865)
       );
  OAI31D0 U7610 ( .A1(n6863), .A2(n6862), .A3(n6861), .B(n8582)
     , .ZN(n6864) );
  OAI211D0 U7611 ( .A1(n6866), .A2(n8919), .B(n6865), .C(n6864)
     , .ZN(n6902) );
  ND4D0 U7612 ( .A1(n6869), .A2(n6868), .A3(n6867), .A4(n8788),
      .ZN(n6873) );
  ND4D0 U7613 ( .A1(n6871), .A2(n6886), .A3(n6870), .A4(n8882),
      .ZN(n6872) );
  AOI22D0 U7614 ( .A1(n8609), .A2(n6873), .B1(n8899), .B2(n6872
     ), .ZN(n6884)
       );
  OAI211D0 U7615 ( .A1(n8532), .A2(n8612), .B(n6874), .C(n8575)
     , .ZN(n6875) );
  OAI21D0 U7616 ( .A1(n6925), .A2(n6875), .B(n8697), .ZN(n6883)
      );
  ND4D0 U7617 ( .A1(n6877), .A2(n8836), .A3(n6876), .A4(n8601),
      .ZN(n6943) );
  AOI211D0 U7618 ( .A1(n6879), .A2(n6878), .B(n8762), .C(n6943)
     , .ZN(n6974) );
  ND4D0 U7619 ( .A1(n6880), .A2(n6974), .A3(n8721), .A4(n8575),
      .ZN(n6881) );
  OAI31D0 U7620 ( .A1(n8551), .A2(n6957), .A3(n6881), .B(n8582)
     , .ZN(n6882) );
  AOI31D0 U7621 ( .A1(n6884), .A2(n6883), .A3(n6882), .B(n8797)
     , .ZN(n6901) );
  IND4D0 U7622 ( .A1(n6887), .B1(n6886), .B2(n6885), .B3(n8512)
     , .ZN(n6892) );
  AOI31D0 U7623 ( .A1(n6890), .A2(n6889), .A3(n6888), .B(n8923)
     , .ZN(n6891) );
  AOI221D0 U7624 ( .A1(n6893), .A2(n8822), .B1(n6892), .B2(
     n8822), .C(n6891),
       .ZN(n6899) );
  CKND2D0 U7625 ( .A1(n6894), .A2(n6926), .ZN(n8777) );
  OAI31D0 U7626 ( .A1(n6951), .A2(n8777), .A3(n6895), .B(n8899)
     , .ZN(n6898) );
  OAI31D0 U7627 ( .A1(n8831), .A2(n8744), .A3(n6896), .B(n8685)
     , .ZN(n6897) );
  AOI31D0 U7628 ( .A1(n6899), .A2(n6898), .A3(n6897), .B(n8903)
     , .ZN(n6900) );
```

```
AOI211D0 U7629 ( .A1(n8803), .A2(n6902), .B(n6901), .C(n6900)
   , .ZN(n6906) );
OAI31D0 U7630 ( .A1(n6904), .A2(n8668), .A3(n6903), .B(n8697)
   , .ZN(n6905) );
AOI32D0 U7631 ( .A1(n6907), .A2(n6906), .A3(n6905), .B1(n8687
   ), .B2(n6906),
      .ZN(n6982) );
INVD0 U7632 ( .I(n8777), .ZN(n8751) );
ND4D0 U7633 ( .A1(n8751), .A2(n8678), .A3(n6908), .A4(n7618),
    .ZN(n6922) );
NR4D0 U7634 ( .A1(n8620), .A2(n6911), .A3(n6910), .A4(n6909),
    .ZN(n6912) );
AOI31D0 U7635 ( .A1(n6912), .A2(n8920), .A3(n8691), .B(n8919)
   , .ZN(n6921) );
NR4D0 U7636 ( .A1(n8629), .A2(n6915), .A3(n6914), .A4(n6913),
    .ZN(n6919) );
NR4D0 U7637 ( .A1(n6917), .A2(n8777), .A3(n6916), .A4(n6933),
    .ZN(n6918) );
OAI22D0 U7638 ( .A1(n6919), .A2(n8935), .B1(n6918), .B2(n8937
   ), .ZN(n6920)
      );
AOI211D0 U7639 ( .A1(n8697), .A2(n6922), .B(n6921), .C(n6920)
   , .ZN(n6980) );
NR4D0 U7640 ( .A1(n6925), .A2(n8735), .A3(n6924), .A4(n6923),
    .ZN(n6928) );
AOI31D0 U7641 ( .A1(n6928), .A2(n6927), .A3(n6926), .B(n8935)
   , .ZN(n6940) );
AOI31D0 U7642 ( .A1(n6931), .A2(n6930), .A3(n6929), .B(n8923)
   , .ZN(n6939) );
OAI31D0 U7643 ( .A1(n6934), .A2(n6933), .A3(n6932), .B(n8822)
   , .ZN(n6936) );
AOI32D0 U7644 ( .A1(n6937), .A2(n6936), .A3(n6935), .B1(n8674
   ), .B2(n6936),
      .ZN(n6938) );
OAI31D0 U7645 ( .A1(n6940), .A2(n6939), .A3(n6938), .B(n8715)
   , .ZN(n6979) );
AOI211D0 U7646 ( .A1(n6942), .A2(n6941), .B(n8538), .C(n8831)
   , .ZN(n6950) );
INVD0 U7647 ( .I(n6943), .ZN(n6949) );
CKND2D0 U7648 ( .A1(n8691), .A2(n6944), .ZN(n6946) );
AOI211D0 U7649 ( .A1(n6947), .A2(n8764), .B(n6946), .C(n6945)
   , .ZN(n6948) );
AOI31D0 U7650 ( .A1(n6950), .A2(n6949), .A3(n6948), .B(n8923)
   , .ZN(n6977) );
INVD0 U7651 ( .I(n6951), .ZN(n6959) );
IND4D0 U7652 ( .A1(n8536), .B1(n6954), .B2(n6953), .B3(n6952)
   , .ZN(n6955) );
```

```
NR4D0 U7653 ( .A1(n8504), .A2(n6957), .A3(n6956), .A4(n6955),
    .ZN(n6958) );
AOI31D0 U7654 ( .A1(n6966), .A2(n6959), .A3(n6958), .B(n8937)
   , .ZN(n6976) );
AOI211D0 U7655 ( .A1(n7436), .A2(counter[11]), .B(n8735), .C(
   n8545), .ZN(
      n6961) );
ND4D0 U7656 ( .A1(n6962), .A2(n6961), .A3(n8844), .A4(n6960),
    .ZN(n6963) );
OAI31D0 U7657 ( .A1(n6965), .A2(n6964), .A3(n6963), .B(n8582)
   , .ZN(n6973) );
ND4D0 U7658 ( .A1(n6967), .A2(n6966), .A3(n8882), .A4(n8555),
    .ZN(n6968) );
INR4D0 U7659 ( .A1(n6971), .B1(n6970), .B2(n6969), .B3(n6968)
   , .ZN(n6972) );
AOI32D0 U7660 ( .A1(n6974), .A2(n6973), .A3(n6972), .B1(n8919
   ), .B2(n6973),
      .ZN(n6975) );
OAI31D0 U7661 ( .A1(n6977), .A2(n6976), .A3(n6975), .B(n8639)
   , .ZN(n6978) );
OAI211D0 U7662 ( .A1(n6980), .A2(n8662), .B(n6979), .C(n6978)
   , .ZN(n6981) );
AOI211D0 U7663 ( .A1(n8918), .A2(n6983), .B(n6982), .C(n6981)
   , .ZN(n6984) );
OAI211D0 U7664 ( .A1(n6987), .A2(n6986), .B(n6985), .C(n6984)
   , .ZN(n6988) );
AOI211D0 U7665 ( .A1(n8037), .A2(n6990), .B(n6989), .C(n6988)
   , .ZN(n6991) );
OAI222D0 U7666 ( .A1(n6994), .A2(n7277), .B1(n7000), .B2(
   n6993), .C1(n6992),
      .C2(n6991), .ZN(n4816) );
CKND2D0 U7667 ( .A1(counter[5]), .A2(n9346), .ZN(n9351) );
NR2D0 U7668 ( .A1(n9352), .A2(n9351), .ZN(n9350) );
INVD0 U7669 ( .I(n9350), .ZN(n6999) );
NR2D0 U7670 ( .A1(n9304), .A2(n6999), .ZN(n9339) );
INVD0 U7671 ( .I(n9339), .ZN(n9334) );
CKND2D0 U7672 ( .A1(n8046), .A2(n7310), .ZN(n7323) );
ND3D0 U7673 ( .A1(n6995), .A2(n9328), .A3(n7323), .ZN(n6996)
   );
AOI221D0 U7674 ( .A1(n6997), .A2(n6996), .B1(n9318), .B2(
   n6996), .C(
      counter[7]), .ZN(n6998) );
AOI221D1 U7675 ( .A1(n6998), .A2(n7437), .B1(n9338), .B2(
   n7437), .C(n9329),
      .ZN(n9349) );
AOI221D0 U7676 ( .A1(counter[9]), .A2(n9339), .B1(n9075), .B2
   (n9334), .C(
      n9349), .ZN(N4776) );
```

```
AOI221D0 U7677 ( .A1(n8094), .A2(n8050), .B1(n8983), .B2(
   n8050), .C(n9349),
      .ZN(N4769) );
AOI221D0 U7678 ( .A1(n9346), .A2(counter[5]), .B1(n7948), .B2
   (n7514), .C(
      n9349), .ZN(N4772) );
AOI221D0 U7679 ( .A1(counter[7]), .A2(n9350), .B1(n9153), .B2
   (n6999), .C(
      n9349), .ZN(N4774) );
INVD0 U7680 ( .I(n7000), .ZN(n8466) );
CKND2D0 U7681 ( .A1(n9075), .A2(n8951), .ZN(n9210) );
NR2D0 U7682 ( .A1(n7001), .A2(n9210), .ZN(n7126) );
CKAN2D0 U7683 ( .A1(n7068), .A2(n9075), .Z(n7132) );
OR2D0 U7684 ( .A1(n7126), .A2(n7132), .Z(n7092) );
CKND2D0 U7685 ( .A1(counter[9]), .A2(n7011), .ZN(n7183) );
CKND2D0 U7686 ( .A1(counter[9]), .A2(n8009), .ZN(n7022) );
IND2D0 U7687 ( .A1(n7022), .B1(n8046), .ZN(n7172) );
CKND2D0 U7688 ( .A1(n7183), .A2(n7172), .ZN(n7019) );
INVD0 U7689 ( .I(n7019), .ZN(n7231) );
NR2D0 U7690 ( .A1(n9075), .A2(n8014), .ZN(n7004) );
INVD0 U7691 ( .I(n7004), .ZN(n7094) );
NR2D0 U7692 ( .A1(n8957), .A2(counter[9]), .ZN(n9173) );
INVD0 U7693 ( .I(n9173), .ZN(n9028) );
NR2D0 U7694 ( .A1(n7003), .A2(n9028), .ZN(n7162) );
CKND2D0 U7695 ( .A1(n7004), .A2(n7296), .ZN(n7228) );
CKND2D0 U7696 ( .A1(counter[9]), .A2(n7068), .ZN(n7238) );
CKND2D0 U7697 ( .A1(n7228), .A2(n7238), .ZN(n7076) );
NR2D0 U7698 ( .A1(n7162), .A2(n7076), .ZN(n7112) );
NR2D0 U7699 ( .A1(counter[9]), .A2(n8014), .ZN(n7250) );
CKND2D0 U7700 ( .A1(n7250), .A2(n7296), .ZN(n7189) );
OA211D0 U7701 ( .A1(n7295), .A2(n7094), .B(n7112), .C(n7189),
    .Z(n7142) );
CKND2D0 U7702 ( .A1(n7231), .A2(n7142), .ZN(n7117) );
INVD0 U7703 ( .I(n7117), .ZN(n7224) );
NR2D0 U7704 ( .A1(n8055), .A2(n7094), .ZN(n7134) );
CKND2D0 U7705 ( .A1(n7987), .A2(n7250), .ZN(n7058) );
IND2D0 U7706 ( .A1(n7134), .B1(n7058), .ZN(n7128) );
CKND2D0 U7707 ( .A1(n7004), .A2(n7002), .ZN(n7015) );
CKND2D0 U7708 ( .A1(n7128), .A2(n7015), .ZN(n7104) );
INVD0 U7709 ( .I(n7104), .ZN(n7070) );
NR2D0 U7710 ( .A1(n8953), .A2(counter[9]), .ZN(n9022) );
INVD0 U7711 ( .I(n9022), .ZN(n9111) );
NR2D0 U7712 ( .A1(n7003), .A2(n9111), .ZN(n7105) );
INR2D0 U7713 ( .A1(n7250), .B1(n7295), .ZN(n7163) );
NR3D0 U7714 ( .A1(n7070), .A2(n7105), .A3(n7163), .ZN(n7210)
   );
NR2D0 U7715 ( .A1(n7235), .A2(n7022), .ZN(n7020) );
CKND2D0 U7716 ( .A1(n9075), .A2(n8046), .ZN(n8955) );
```

```
INVD0 U7717 ( .I(n8955), .ZN(n9295) );
CKND2D0 U7718 ( .A1(n7006), .A2(n9295), .ZN(n7096) );
NR3D0 U7719 ( .A1(n8924), .A2(counter[9]), .A3(counter[3]), .
   ZN(n7036) );
INVD0 U7720 ( .I(n7036), .ZN(n7030) );
CKND2D0 U7721 ( .A1(n7987), .A2(n7004), .ZN(n7021) );
ND4D0 U7722 ( .A1(n7059), .A2(n7096), .A3(n7030), .A4(n7021),
    .ZN(n7227) );
NR2D0 U7723 ( .A1(n7020), .A2(n7227), .ZN(n7102) );
INVD0 U7724 ( .I(n7102), .ZN(n7256) );
INR2D0 U7725 ( .A1(n7210), .B1(n7256), .ZN(n7225) );
CKND2D0 U7726 ( .A1(n7004), .A2(n7414), .ZN(n7221) );
INVD0 U7727 ( .I(n7221), .ZN(n7136) );
NR2D0 U7728 ( .A1(n7132), .A2(n7136), .ZN(n7103) );
CKND2D0 U7729 ( .A1(n7225), .A2(n7103), .ZN(n7175) );
INVD0 U7730 ( .I(n7175), .ZN(n7154) );
NR2D0 U7731 ( .A1(n9210), .A2(n7029), .ZN(n7061) );
CKND2D0 U7732 ( .A1(n7004), .A2(n7350), .ZN(n7010) );
CKND2D0 U7733 ( .A1(n7011), .A2(n9075), .ZN(n7005) );
OR2D0 U7734 ( .A1(n7022), .A2(n7321), .Z(n7014) );
CKAN2D0 U7735 ( .A1(n7005), .A2(n7014), .Z(n7152) );
OA211D0 U7736 ( .A1(n7022), .A2(n8055), .B(n7010), .C(n7152),
    .Z(n7045) );
CKND2D0 U7737 ( .A1(n7250), .A2(n7414), .ZN(n7046) );
INVD0 U7738 ( .I(n7046), .ZN(n7110) );
NR2D0 U7739 ( .A1(n8014), .A2(n7285), .ZN(n7062) );
NR2D0 U7740 ( .A1(n7110), .A2(n7062), .ZN(n7048) );
CKND2D0 U7741 ( .A1(n7045), .A2(n7048), .ZN(n7093) );
NR2D0 U7742 ( .A1(n7061), .A2(n7093), .ZN(n7127) );
AOI31D0 U7743 ( .A1(n7224), .A2(n7154), .A3(n7127), .B(
   counter[0]), .ZN(
      n7008) );
INVD0 U7744 ( .I(n9210), .ZN(n8462) );
CKND2D0 U7745 ( .A1(n7006), .A2(n8462), .ZN(n7151) );
INVD0 U7746 ( .I(n7151), .ZN(n7237) );
NR2D0 U7747 ( .A1(n7163), .A2(n7256), .ZN(n7196) );
IND3D0 U7748 ( .A1(n7105), .B1(n7224), .B2(n7196), .ZN(n7193)
    );
NR4D0 U7749 ( .A1(n7134), .A2(n7237), .A3(n7193), .A4(n7093),
    .ZN(n7007) );
OAI21D0 U7750 ( .A1(n7007), .A2(n8970), .B(counter[7]), .ZN(
   n7057) );
AOI211D0 U7751 ( .A1(n9294), .A2(n7092), .B(n7008), .C(n7057)
   , .ZN(n7035) );
NR2D0 U7752 ( .A1(n7162), .A2(n7019), .ZN(n7067) );
INVD0 U7753 ( .I(n7067), .ZN(n7243) );
NR2D0 U7754 ( .A1(n7076), .A2(n7243), .ZN(n7195) );
INVD0 U7755 ( .I(n7195), .ZN(n7013) );
```

```
  NR2D0 U7756 ( .A1(n7061), .A2(n7062), .ZN(n7253) );
  INVD0 U7757 ( .I(n7253), .ZN(n7023) );
  INR2D0 U7758 ( .A1(n7414), .B1(n7022), .ZN(n7249) );
  NR2D0 U7759 ( .A1(n7094), .A2(n7295), .ZN(n7009) );
  NR2D0 U7760 ( .A1(n7126), .A2(n7009), .ZN(n7153) );
  IND2D0 U7761 ( .A1(n7249), .B1(n7153), .ZN(n7024) );
  CKND2D0 U7762 ( .A1(n7046), .A2(n7189), .ZN(n7017) );
  OR2D0 U7763 ( .A1(n7022), .A2(n7290), .Z(n7204) );
  CKND2D0 U7764 ( .A1(n7010), .A2(n7151), .ZN(n7018) );
  INVD0 U7765 ( .I(n7018), .ZN(n7252) );
  CKND2D0 U7766 ( .A1(n7204), .A2(n7252), .ZN(n7259) );
  NR2D0 U7767 ( .A1(n7017), .A2(n7259), .ZN(n7016) );
  INVD0 U7768 ( .I(n7016), .ZN(n7156) );
  AO211D0 U7769 ( .A1(n7011), .A2(n9075), .B(n7024), .C(n7156),
      .Z(n7148) );
  NR2D0 U7770 ( .A1(n7023), .A2(n7148), .ZN(n7101) );
  CKND2D0 U7771 ( .A1(n7014), .A2(n7101), .ZN(n7176) );
  OR2D0 U7772 ( .A1(n7176), .A2(n7105), .Z(n7133) );
  NR2D0 U7773 ( .A1(n7134), .A2(n7133), .ZN(n7095) );
  NR2D0 U7774 ( .A1(n7163), .A2(n7132), .ZN(n7071) );
  CKND2D0 U7775 ( .A1(n7095), .A2(n7071), .ZN(n7039) );
  INR2D0 U7776 ( .A1(n7096), .B1(n7020), .ZN(n7107) );
  CKND2D0 U7777 ( .A1(n7107), .A2(n7021), .ZN(n7206) );
  NR2D0 U7778 ( .A1(n7039), .A2(n7206), .ZN(n7080) );
  CKND2D0 U7779 ( .A1(n7080), .A2(n7022), .ZN(n7012) );
  OAI31D0 U7780 ( .A1(n7036), .A2(n7013), .A3(n7012), .B(n8978)
     , .ZN(n7034) );
  CKND2D0 U7781 ( .A1(n7225), .A2(n7195), .ZN(n7149) );
  CKND2D0 U7782 ( .A1(n7015), .A2(n7014), .ZN(n7078) );
  NR2D0 U7783 ( .A1(n7024), .A2(n7078), .ZN(n7119) );
  IND3D0 U7784 ( .A1(n7149), .B1(n7016), .B2(n7119), .ZN(n7065)
      );
  IND3D0 U7785 ( .A1(n7024), .B1(n7152), .B2(n7103), .ZN(n7060)
      );
  NR3D0 U7786 ( .A1(n7060), .A2(n7017), .A3(n7023), .ZN(n7203)
     );
  CKND2D0 U7787 ( .A1(n7225), .A2(n7203), .ZN(n7242) );
  NR4D0 U7788 ( .A1(n7242), .A2(n7019), .A3(n7076), .A4(n7018),
      .ZN(n7026) );
  CKND2D0 U7789 ( .A1(n7210), .A2(n7195), .ZN(n7044) );
  INVD0 U7790 ( .I(n7044), .ZN(n7198) );
  NR2D0 U7791 ( .A1(counter[5]), .A2(n7030), .ZN(n7073) );
  NR2D0 U7792 ( .A1(n7020), .A2(n7073), .ZN(n7038) );
  ND3D0 U7793 ( .A1(n7103), .A2(n7198), .A3(n7038), .ZN(n7201)
     );
  CKND2D0 U7794 ( .A1(n7022), .A2(n7021), .ZN(n7069) );
  NR3D0 U7795 ( .A1(n7024), .A2(n7023), .A3(n7259), .ZN(n7113)
     );
```

```
CKND2D0 U7796 ( .A1(n7152), .A2(n7113), .ZN(n7025) );
NR4D0 U7797 ( .A1(n7036), .A2(n7201), .A3(n7069), .A4(n7025),
    .ZN(n7097) );
OAI22D0 U7798 ( .A1(n7026), .A2(n9193), .B1(n7097), .B2(n8970
   ), .ZN(n7027)
       );
AOI211D0 U7799 ( .A1(n8978), .A2(n7065), .B(counter[7]), .C(
   n7027), .ZN(
      n7033) );
OAI21D0 U7800 ( .A1(n8957), .A2(n7028), .B(n7095), .ZN(n7037)
    );
NR2D0 U7801 ( .A1(n7029), .A2(n9028), .ZN(n7202) );
CKND2D0 U7802 ( .A1(n7195), .A2(n7107), .ZN(n7031) );
IND2D0 U7803 ( .A1(n7031), .B1(n7030), .ZN(n7135) );
OR2D0 U7804 ( .A1(n7202), .A2(n7135), .Z(n7187) );
OAI31D0 U7805 ( .A1(n7132), .A2(n7037), .A3(n7187), .B(n9025)
   , .ZN(n7032) );
AOI22D0 U7806 ( .A1(n7035), .A2(n7034), .B1(n7033), .B2(n7032
   ), .ZN(n7056)
       );
CKND2D0 U7807 ( .A1(counter[5]), .A2(n7036), .ZN(n7188) );
CKND2D0 U7808 ( .A1(n7059), .A2(n7188), .ZN(n7082) );
NR2D0 U7809 ( .A1(n7037), .A2(n7082), .ZN(n7166) );
CKAN2D0 U7810 ( .A1(n7172), .A2(n7112), .Z(n7129) );
AOI31D0 U7811 ( .A1(n7038), .A2(n7166), .A3(n7129), .B(n7280)
   , .ZN(n7042) );
INR3D0 U7812 ( .A1(n7113), .B1(n7149), .B2(n7132), .ZN(n7077)
    );
CKND2D0 U7813 ( .A1(n7067), .A2(n7228), .ZN(n7217) );
OAI31D0 U7814 ( .A1(n7227), .A2(n7039), .A3(n7217), .B(n9294)
   , .ZN(n7040) );
OAI211D0 U7815 ( .A1(n7077), .A2(n9193), .B(counter[7]), .C(
   n7040), .ZN(
      n7041) );
NR2D0 U7816 ( .A1(n7042), .A2(n7041), .ZN(n7054) );
NR2D0 U7817 ( .A1(n7176), .A2(n7132), .ZN(n7218) );
INVD0 U7818 ( .I(n7218), .ZN(n7043) );
OAI31D0 U7819 ( .A1(n7044), .A2(n7082), .A3(n7043), .B(n8978)
   , .ZN(n7053) );
NR2D0 U7820 ( .A1(n7060), .A2(n7149), .ZN(n7047) );
ND4D0 U7821 ( .A1(n7253), .A2(n7045), .A3(n7047), .A4(n7189),
    .ZN(n7051) );
CKND2D0 U7822 ( .A1(n9025), .A2(n7046), .ZN(n7050) );
IND4D0 U7823 ( .A1(n7259), .B1(n7048), .B2(n7047), .B3(n7280)
   , .ZN(n7049) );
OAI222D0 U7824 ( .A1(n7051), .A2(n7050), .B1(n7051), .B2(
   n9092), .C1(n8978),
      .C2(n7049), .ZN(n7052) );
```

```
AOI22D0 U7825 ( .A1(n7054), .A2(n7053), .B1(n9153), .B2(n7052
   ), .ZN(n7055)
      );
AOI221D0 U7826 ( .A1(n7056), .A2(counter[8]), .B1(n7055), .B2
   (n9338), .C(
      n9352), .ZN(n7276) );
INVD0 U7827 ( .I(n7057), .ZN(n7091) );
CKND2D0 U7828 ( .A1(n7058), .A2(n7172), .ZN(n7081) );
INVD0 U7829 ( .I(n7183), .ZN(n7109) );
NR2D0 U7830 ( .A1(n7109), .A2(n7073), .ZN(n7164) );
ND3D0 U7831 ( .A1(n7059), .A2(n7112), .A3(n7164), .ZN(n7205)
   );
INVD0 U7832 ( .I(n7205), .ZN(n7106) );
CKND2D0 U7833 ( .A1(n7080), .A2(n7106), .ZN(n7143) );
NR2D0 U7834 ( .A1(n7060), .A2(n7156), .ZN(n7108) );
INR2D0 U7835 ( .A1(n7108), .B1(n7061), .ZN(n7247) );
AOI21D0 U7836 ( .A1(counter[9]), .A2(n7062), .B(n7227), .ZN(
   n7158) );
AOI31D0 U7837 ( .A1(n7198), .A2(n7247), .A3(n7158), .B(n9092)
   , .ZN(n7063) );
AOI221D0 U7838 ( .A1(n7081), .A2(n9025), .B1(n7143), .B2(
   n9025), .C(n7063),
      .ZN(n7090) );
CKND2D0 U7839 ( .A1(n7250), .A2(n7064), .ZN(n7244) );
INVD0 U7840 ( .I(n7244), .ZN(n7066) );
OAI31D0 U7841 ( .A1(n7132), .A2(n7066), .A3(n7065), .B(n8968)
   , .ZN(n7089) );
IND4D0 U7842 ( .A1(n7068), .B1(n7067), .B2(n7095), .B3(n7196)
   , .ZN(n7087) );
INVD0 U7843 ( .I(n7069), .ZN(n7075) );
NR2D0 U7844 ( .A1(n7070), .A2(n7176), .ZN(n7194) );
CKAN2D0 U7845 ( .A1(n7112), .A2(n7194), .Z(n7184) );
CKND2D0 U7846 ( .A1(n7071), .A2(n7184), .ZN(n7072) );
NR2D0 U7847 ( .A1(n7073), .A2(n7072), .ZN(n7074) );
AOI31D0 U7848 ( .A1(n7107), .A2(n7075), .A3(n7074), .B(n8970)
   , .ZN(n7086) );
INVD0 U7849 ( .I(n7076), .ZN(n7173) );
INVD0 U7850 ( .I(n7077), .ZN(n7079) );
AOI211D0 U7851 ( .A1(n8968), .A2(n7079), .B(n7078), .C(
   counter[7]), .ZN(
      n7084) );
INVD0 U7852 ( .I(n7080), .ZN(n7186) );
NR4D0 U7853 ( .A1(n7109), .A2(n7186), .A3(n7082), .A4(n7081),
    .ZN(n7083) );
AOI32D0 U7854 ( .A1(n7173), .A2(n7084), .A3(n7083), .B1(n9092
   ), .B2(n7084),
      .ZN(n7085) );
```

```
AOI211D0 U7855 ( .A1(n9025), .A2(n7087), .B(n7086), .C(n7085)
   , .ZN(n7088) );
AOI31D0 U7856 ( .A1(n7091), .A2(n7090), .A3(n7089), .B(n7088)
   , .ZN(n7125) );
INR4D0 U7857 ( .A1(n7189), .B1(n7149), .B2(n7093), .B3(n7092)
   , .ZN(n7144) );
CKND2D0 U7858 ( .A1(n7144), .A2(n7094), .ZN(n7100) );
AOI31D0 U7859 ( .A1(n7102), .A2(n7195), .A3(n7095), .B(n8970)
   , .ZN(n7099) );
AOI31D0 U7860 ( .A1(n7097), .A2(n7096), .A3(n7189), .B(n9092)
   , .ZN(n7098) );
AOI211D0 U7861 ( .A1(n9025), .A2(n7100), .B(n7099), .C(n7098)
   , .ZN(n7123) );
INVD0 U7862 ( .I(n7101), .ZN(n7200) );
CKAN2D0 U7863 ( .A1(n7103), .A2(n7102), .Z(n7199) );
ND3D0 U7864 ( .A1(n7195), .A2(n7199), .A3(n7104), .ZN(n7248)
   );
OAI31D0 U7865 ( .A1(n7105), .A2(n7200), .A3(n7248), .B(n8968)
   , .ZN(n7122) );
OAI211D0 U7866 ( .A1(counter[3]), .A2(n7172), .B(n7107), .C(
   n7106), .ZN(
      n7255) );
CKND2D0 U7867 ( .A1(n7210), .A2(n7247), .ZN(n7160) );
ND3D0 U7868 ( .A1(n7244), .A2(n7154), .A3(n7152), .ZN(n7116)
   );
AOI211D0 U7869 ( .A1(n7367), .A2(n7285), .B(n8276), .C(n9075)
   , .ZN(n7219) );
INVD0 U7870 ( .I(n7219), .ZN(n7165) );
IND4D0 U7871 ( .A1(n7116), .B1(n7108), .B2(n7183), .B3(n7165)
   , .ZN(n7115) );
NR4D0 U7872 ( .A1(n7110), .A2(n7109), .A3(n7175), .A4(n7170),
    .ZN(n7111) );
AOI31D0 U7873 ( .A1(n7113), .A2(n7112), .A3(n7111), .B(n7280)
   , .ZN(n7114) );
AOI211D0 U7874 ( .A1(n8978), .A2(n7115), .B(counter[7]), .C(
   n7114), .ZN(
      n7118) );
NR4D0 U7875 ( .A1(n7249), .A2(n7259), .A3(n7117), .A4(n7116),
    .ZN(n7140) );
AOI32D0 U7876 ( .A1(n7119), .A2(n7118), .A3(n7140), .B1(n8970
   ), .B2(n7118),
      .ZN(n7120) );
AOI221D0 U7877 ( .A1(n7255), .A2(n8968), .B1(n7160), .B2(
   n8968), .C(n7120),
      .ZN(n7121) );
AOI31D0 U7878 ( .A1(counter[7]), .A2(n7123), .A3(n7122), .B(
   n7121), .ZN(
      n7124) );
```

```
AOI221D0 U7879 ( .A1(n7125), .A2(counter[8]), .B1(n7124), .B2
   (n9338), .C(
      counter[6]), .ZN(n7275) );
CKND2D0 U7880 ( .A1(n9329), .A2(n7154), .ZN(n7147) );
INR2D0 U7881 ( .A1(n7127), .B1(n7126), .ZN(n7141) );
INVD0 U7882 ( .I(n7141), .ZN(n7191) );
NR2D0 U7883 ( .A1(n7191), .A2(n7237), .ZN(n7197) );
INR3D0 U7884 ( .A1(n7197), .B1(n7128), .B2(n7255), .ZN(n7190)
    );
IND2D0 U7885 ( .A1(n7147), .B1(n7190), .ZN(n7131) );
CKND2D0 U7886 ( .A1(n9329), .A2(n7129), .ZN(n7159) );
OA31D0 U7887 ( .A1(n7175), .A2(n7200), .A3(n7159), .B(n8968),
    .Z(n7130) );
AOI211D0 U7888 ( .A1(n9294), .A2(n7131), .B(counter[7]), .C(
   n7130), .ZN(
      n7139) );
NR3D0 U7889 ( .A1(n7133), .A2(n7250), .A3(n7132), .ZN(n7230)
   );
CKND2D0 U7890 ( .A1(n7230), .A2(n9329), .ZN(n7254) );
OAI31D0 U7891 ( .A1(n7134), .A2(n7187), .A3(n7254), .B(n9025)
   , .ZN(n7138) );
OAI31D0 U7892 ( .A1(n7136), .A2(n7135), .A3(n7254), .B(n8978)
   , .ZN(n7137) );
AOI31D0 U7893 ( .A1(n7139), .A2(n7138), .A3(n7137), .B(n9338)
   , .ZN(n7273) );
NR2D0 U7894 ( .A1(counter[6]), .A2(n9193), .ZN(n7361) );
INVD0 U7895 ( .I(n7140), .ZN(n7272) );
AOI31D0 U7896 ( .A1(n7225), .A2(n7142), .A3(n7141), .B(n9092)
   , .ZN(n7146) );
MOAI22D0 U7897 ( .A1(n7144), .A2(n8970), .B1(n7143), .B2(
   n9025), .ZN(n7145)
      );
OAI31D0 U7898 ( .A1(n7146), .A2(n9153), .A3(n7145), .B(n7273)
   , .ZN(n7270) );
OAI31D0 U7899 ( .A1(n7149), .A2(n7148), .A3(n7147), .B(n9294)
   , .ZN(n7150) );
OAI211D0 U7900 ( .A1(n7151), .A2(n7280), .B(counter[7]), .C(
   n7150), .ZN(
      n7182) );
ND3D0 U7901 ( .A1(n7153), .A2(n7152), .A3(n7253), .ZN(n7222)
   );
IND2D0 U7902 ( .A1(n7222), .B1(n7154), .ZN(n7155) );
NR4D0 U7903 ( .A1(counter[11]), .A2(n7243), .A3(n7156), .A4(
   n7155), .ZN(
      n7157) );
OAI22D0 U7904 ( .A1(counter[0]), .A2(n7259), .B1(n7157), .B2(
   n9092), .ZN(
      n7181) );
```

```
INVD0 U7905 ( .I(n7158), .ZN(n7161) );
AOI211D0 U7906 ( .A1(n7164), .A2(n7161), .B(n7160), .C(n7159)
   , .ZN(n7179) );
INR4D0 U7907 ( .A1(n7164), .B1(counter[11]), .B2(n7163), .B3(
   n7162), .ZN(
      n7167) );
AOI31D0 U7908 ( .A1(n7167), .A2(n7166), .A3(n7165), .B(n9193)
   , .ZN(n7168) );
AOI211D0 U7909 ( .A1(n9294), .A2(n7173), .B(counter[7]), .C(
   n7168), .ZN(
      n7178) );
CKND2D0 U7910 ( .A1(n8983), .A2(n7169), .ZN(n7971) );
NR2D0 U7911 ( .A1(n8765), .A2(n7971), .ZN(n7171) );
OAI22D0 U7912 ( .A1(n7171), .A2(n8041), .B1(n7170), .B2(n8116
   ), .ZN(n7239)
       );
NR2D0 U7913 ( .A1(n7350), .A2(n7239), .ZN(n7220) );
ND4D0 U7914 ( .A1(n7173), .A2(n7220), .A3(n7172), .A4(n8116),
    .ZN(n7174) );
OAI31D0 U7915 ( .A1(n7176), .A2(n7175), .A3(n7174), .B(n8978)
   , .ZN(n7177) );
OAI211D0 U7916 ( .A1(n7179), .A2(n7280), .B(n7178), .C(n7177)
   , .ZN(n7180) );
OAI211D0 U7917 ( .A1(n7182), .A2(n7181), .B(n9338), .C(n7180)
   , .ZN(n7269) );
AOI31D0 U7918 ( .A1(n7184), .A2(n7199), .A3(n7183), .B(n8970)
   , .ZN(n7185) );
AOI221D0 U7919 ( .A1(n7187), .A2(n8968), .B1(n7186), .B2(
   n8968), .C(n7185),
      .ZN(n7216) );
AOI31D0 U7920 ( .A1(n7190), .A2(n7189), .A3(n7188), .B(n9092)
   , .ZN(n7192) );
AOI211D0 U7921 ( .A1(n9025), .A2(n7193), .B(n7192), .C(n7191)
   , .ZN(n7215) );
ND4D0 U7922 ( .A1(n7196), .A2(n7195), .A3(n7194), .A4(n8116),
    .ZN(n7213) );
AOI31D0 U7923 ( .A1(n7199), .A2(n7198), .A3(n7197), .B(n8970)
   , .ZN(n7212) );
OAI31D0 U7924 ( .A1(n7202), .A2(n7201), .A3(n7200), .B(n9025)
   , .ZN(n7209) );
INVD0 U7925 ( .I(n7203), .ZN(n7207) );
CKND2D0 U7926 ( .A1(n7204), .A2(n8116), .ZN(n7240) );
NR4D0 U7927 ( .A1(n7207), .A2(n7206), .A3(n7205), .A4(n7240),
    .ZN(n7208) );
AOI32D0 U7928 ( .A1(n7210), .A2(n7209), .A3(n7208), .B1(n9092
   ), .B2(n7209),
      .ZN(n7211) );
```

```
AOI211D0 U7929 ( .A1(n8968), .A2(n7213), .B(n7212), .C(n7211)
   , .ZN(n7214) );
AOI32D0 U7930 ( .A1(n7216), .A2(counter[7]), .A3(n7215), .B1(
   n7214), .B2(
      n9153), .ZN(n7267) );
INVD0 U7931 ( .I(n7217), .ZN(n7246) );
CKAN2D0 U7932 ( .A1(n7225), .A2(n9329), .Z(n7245) );
OAI211D0 U7933 ( .A1(n7246), .A2(n7219), .B(n7218), .C(n7245)
   , .ZN(n7234) );
ND3D0 U7934 ( .A1(n8041), .A2(n7221), .A3(n7220), .ZN(n7226)
   );
NR4D0 U7935 ( .A1(n9173), .A2(n7259), .A3(n7226), .A4(n7222),
    .ZN(n7223) );
AOI31D0 U7936 ( .A1(n7225), .A2(n7224), .A3(n7223), .B(n7280)
   , .ZN(n7233) );
INR4D0 U7937 ( .A1(n7228), .B1(n9295), .B2(n7227), .B3(n7226)
   , .ZN(n7229) );
AOI31D0 U7938 ( .A1(n7231), .A2(n7230), .A3(n7229), .B(n8970)
   , .ZN(n7232) );
AOI211D0 U7939 ( .A1(n8978), .A2(n7234), .B(n7233), .C(n7232)
   , .ZN(n7265) );
NR2D0 U7940 ( .A1(n7235), .A2(n8041), .ZN(n7236) );
NR2D0 U7941 ( .A1(n7237), .A2(n7236), .ZN(n7257) );
IIND4D0 U7942 ( .A1(n7240), .A2(n7239), .B1(n7257), .B2(n7238
   ), .ZN(n7241)
     );
OAI31D0 U7943 ( .A1(n7243), .A2(n7242), .A3(n7241), .B(n8968)
   , .ZN(n7264) );
ND4D0 U7944 ( .A1(n7247), .A2(n7246), .A3(n7245), .A4(n7244),
    .ZN(n7262) );
NR4D0 U7945 ( .A1(counter[11]), .A2(n7250), .A3(n7249), .A4(
   n7248), .ZN(
      n7251) );
AOI31D0 U7946 ( .A1(n7253), .A2(n7252), .A3(n7251), .B(n9092)
   , .ZN(n7261) );
OAI31D0 U7947 ( .A1(n7256), .A2(n7255), .A3(n7254), .B(n9294)
   , .ZN(n7258) );
AOI32D0 U7948 ( .A1(n7259), .A2(n7258), .A3(n7257), .B1(n7280
   ), .B2(n7258),
      .ZN(n7260) );
AOI211D0 U7949 ( .A1(n8968), .A2(n7262), .B(n7261), .C(n7260)
   , .ZN(n7263) );
AOI32D0 U7950 ( .A1(n7265), .A2(n9153), .A3(n7264), .B1(
   counter[7]), .B2(
      n7263), .ZN(n7266) );
AOI22D0 U7951 ( .A1(counter[8]), .A2(n7267), .B1(n7266), .B2(
   n9338), .ZN(
      n7268) );
```

```
AOI32D0 U7952 ( .A1(n7270), .A2(n9352), .A3(n7269), .B1(
   counter[6]), .B2(
      n7268), .ZN(n7271) );
AOI31D0 U7953 ( .A1(n7273), .A2(n7361), .A3(n7272), .B(n7271)
   , .ZN(n7274) );
OAI32D0 U7954 ( .A1(n9342), .A2(n7276), .A3(n7275), .B1(n7274
   ), .B2(
      counter[10]), .ZN(n7278) );
INVD0 U7955 ( .I(n7277), .ZN(n8947) );
AOI22D0 U7956 ( .A1(n8466), .A2(n7278), .B1(n8947), .B2(q_out
   [5]), .ZN(n7565) );
AOI22D0 U7957 ( .A1(n8968), .A2(n7413), .B1(n7283), .B2(n7412
   ), .ZN(n7279)
       );
NR2D0 U7958 ( .A1(n8463), .A2(n7322), .ZN(n7358) );
INR2D0 U7959 ( .A1(n7279), .B1(n7358), .ZN(n7306) );
NR2D0 U7960 ( .A1(n7294), .A2(n7422), .ZN(n7391) );
NR2D0 U7961 ( .A1(n7280), .A2(counter[3]), .ZN(n9090) );
INVD0 U7962 ( .I(n9090), .ZN(n9048) );
AOI22D0 U7963 ( .A1(n7359), .A2(n8964), .B1(n9299), .B2(n9352
   ), .ZN(n7468)
       );
INVD0 U7964 ( .I(n8463), .ZN(n8979) );
OAI21D0 U7965 ( .A1(n8979), .A2(n9090), .B(n7369), .ZN(n7281)
    );
OAI211D0 U7966 ( .A1(n7282), .A2(n7366), .B(n7468), .C(n7281)
   , .ZN(n7345) );
NR2D0 U7967 ( .A1(n9011), .A2(n7322), .ZN(n7403) );
INVD0 U7968 ( .I(n7403), .ZN(n7449) );
CKND2D0 U7969 ( .A1(n7283), .A2(n7361), .ZN(n7481) );
OA211D0 U7970 ( .A1(n7322), .A2(n8982), .B(n7449), .C(n7481),
    .Z(n7410) );
CKND2D0 U7971 ( .A1(n7295), .A2(n8950), .ZN(n7287) );
CKND2D0 U7972 ( .A1(n7396), .A2(n8971), .ZN(n7516) );
INVD0 U7973 ( .I(n7516), .ZN(n7318) );
CKND2D0 U7974 ( .A1(n7369), .A2(n8971), .ZN(n7480) );
CKND2D0 U7975 ( .A1(n9090), .A2(n7284), .ZN(n7374) );
NR2D0 U7976 ( .A1(n7285), .A2(n7293), .ZN(n7557) );
CKND2D0 U7977 ( .A1(counter[0]), .A2(n7557), .ZN(n7454) );
AN3D0 U7978 ( .A1(n7480), .A2(n7374), .A3(n7454), .Z(n7343) )
   ;
CKND2D0 U7979 ( .A1(n7359), .A2(n8979), .ZN(n7286) );
OAI211D0 U7980 ( .A1(n7444), .A2(n8970), .B(n7343), .C(n7286)
   , .ZN(n7474) );
AOI211D0 U7981 ( .A1(n7412), .A2(n7287), .B(n7318), .C(n7474)
   , .ZN(n7417) );
CKND2D0 U7982 ( .A1(n9189), .A2(n7288), .ZN(n9336) );
CKND2D0 U7983 ( .A1(n7359), .A2(n9129), .ZN(n7513) );
```

```
CKND2D0 U7984 ( .A1(n7359), .A2(n9090), .ZN(n7482) );
CKND2D0 U7985 ( .A1(n7513), .A2(n7482), .ZN(n7346) );
NR2D0 U7986 ( .A1(n7294), .A2(n7388), .ZN(n7406) );
OR2D0 U7987 ( .A1(n7348), .A2(n7406), .Z(n7373) );
NR2D0 U7988 ( .A1(n7346), .A2(n7373), .ZN(n7303) );
INVD0 U7989 ( .I(n7303), .ZN(n7476) );
AOI31D0 U7990 ( .A1(n8966), .A2(n9336), .A3(n7289), .B(n7476)
   , .ZN(n7411) );
NR2D0 U7991 ( .A1(n7290), .A2(n7422), .ZN(n7302) );
CKND2D0 U7992 ( .A1(n8078), .A2(n7291), .ZN(n7377) );
NR2D0 U7993 ( .A1(counter[0]), .A2(n7444), .ZN(n7292) );
CKND2D0 U7994 ( .A1(n7292), .A2(n9029), .ZN(n7518) );
CKND2D0 U7995 ( .A1(n7377), .A2(n7518), .ZN(n7539) );
CKND2D0 U7996 ( .A1(counter[3]), .A2(n7539), .ZN(n7420) );
AOI32D0 U7997 ( .A1(n7294), .A2(n7420), .A3(n7950), .B1(n7293
   ), .B2(n7420),
      .ZN(n7427) );
INR3D0 U7998 ( .A1(n7411), .B1(n7302), .B2(n7427), .ZN(n7438)
    );
ND3D0 U7999 ( .A1(n7410), .A2(n7417), .A3(n7438), .ZN(n7462)
   );
NR2D0 U8000 ( .A1(n7345), .A2(n7462), .ZN(n7447) );
NR2D0 U8001 ( .A1(n7295), .A2(n7422), .ZN(n7523) );
INVD0 U8002 ( .I(n7523), .ZN(n7311) );
OAI211D0 U8003 ( .A1(n7309), .A2(n9048), .B(n7447), .C(n7311)
   , .ZN(n7536) );
NR2D0 U8004 ( .A1(n8463), .A2(n7319), .ZN(n7347) );
NR2D0 U8005 ( .A1(n9311), .A2(n7322), .ZN(n7408) );
CKND2D0 U8006 ( .A1(n9014), .A2(n7327), .ZN(n7500) );
CKND2D0 U8007 ( .A1(n8964), .A2(n7396), .ZN(n7531) );
OAI211D0 U8008 ( .A1(n8962), .A2(n7328), .B(n7500), .C(n7531)
   , .ZN(n7467) );
CKND2D0 U8009 ( .A1(n8964), .A2(n7326), .ZN(n7393) );
CKND2D0 U8010 ( .A1(n7296), .A2(n7412), .ZN(n7418) );
CKND2D0 U8011 ( .A1(n7393), .A2(n7418), .ZN(n7473) );
AO31D0 U8012 ( .A1(n7297), .A2(n7987), .A3(n7313), .B(n7473),
    .Z(n7298) );
NR3D0 U8013 ( .A1(n7408), .A2(n7467), .A3(n7298), .ZN(n7308)
   );
NR2D0 U8014 ( .A1(n7322), .A2(n9048), .ZN(n7464) );
INVD0 U8015 ( .I(n7464), .ZN(n7375) );
IND4D0 U8016 ( .A1(n7347), .B1(n7308), .B2(n7375), .B3(n7299)
   , .ZN(n7522) );
INR4D0 U8017 ( .A1(n7306), .B1(n7391), .B2(n7536), .B3(n7522)
   , .ZN(n7356) );
NR2D0 U8018 ( .A1(n9153), .A2(counter[5]), .ZN(n7494) );
INVD0 U8019 ( .I(n7494), .ZN(n7544) );
NR2D0 U8020 ( .A1(n7300), .A2(n8982), .ZN(n7497) );
```

```
OAI21D0 U8021 ( .A1(n8962), .A2(n7301), .B(n9048), .ZN(n7325)
    );
CKND2D0 U8022 ( .A1(n7414), .A2(n7412), .ZN(n7329) );
INVD0 U8023 ( .I(n7302), .ZN(n7491) );
ND3D0 U8024 ( .A1(n7329), .A2(n7491), .A3(n7303), .ZN(n7363)
   );
NR2D0 U8025 ( .A1(n9178), .A2(n7444), .ZN(n7357) );
NR2D0 U8026 ( .A1(n7672), .A2(n7304), .ZN(n7710) );
CKND2D0 U8027 ( .A1(n7361), .A2(n7708), .ZN(n7501) );
CKND2D0 U8028 ( .A1(n7501), .A2(n8966), .ZN(n7305) );
AOI22D0 U8029 ( .A1(counter[6]), .A2(n7710), .B1(n8311), .B2(
   n7305), .ZN(
      n7443) );
CKND2D0 U8030 ( .A1(n8979), .A2(n7396), .ZN(n7512) );
NR2D0 U8031 ( .A1(n8982), .A2(n7328), .ZN(n7331) );
INR2D0 U8032 ( .A1(n7512), .B1(n7331), .ZN(n7365) );
CKND2D0 U8033 ( .A1(n7369), .A2(n8997), .ZN(n7543) );
ND4D0 U8034 ( .A1(n7443), .A2(n7306), .A3(n7365), .A4(n7543),
    .ZN(n7537) );
NR2D0 U8035 ( .A1(n8964), .A2(n8997), .ZN(n7316) );
NR2D0 U8036 ( .A1(n8463), .A2(n7444), .ZN(n7498) );
INVD0 U8037 ( .I(n7498), .ZN(n7307) );
OAI211D0 U8038 ( .A1(n7309), .A2(n9048), .B(n7308), .C(n7307)
   , .ZN(n7315) );
CKND2D0 U8039 ( .A1(n8156), .A2(n7310), .ZN(n7483) );
NR2D0 U8040 ( .A1(n7328), .A2(n8954), .ZN(n7524) );
INVD0 U8041 ( .I(n7524), .ZN(n7337) );
CKND2D0 U8042 ( .A1(n7337), .A2(n7375), .ZN(n7385) );
CKND2D0 U8043 ( .A1(n7332), .A2(n9129), .ZN(n7339) );
ND4D0 U8044 ( .A1(n7481), .A2(n7377), .A3(n7311), .A4(n7339),
    .ZN(n7312) );
OR2D0 U8045 ( .A1(n7385), .A2(n7312), .Z(n7475) );
NR2D0 U8046 ( .A1(n9352), .A2(n7313), .ZN(n7349) );
CKAN2D0 U8047 ( .A1(n7349), .A2(n7314), .Z(n7342) );
AO21D0 U8048 ( .A1(n8971), .A2(n7359), .B(n7342), .Z(n7376) )
   ;
NR2D0 U8049 ( .A1(n8982), .A2(n7336), .ZN(n7392) );
INR4D0 U8050 ( .A1(n7480), .B1(n7475), .B2(n7376), .B3(n7392)
   , .ZN(n7382) );
IND4D0 U8051 ( .A1(n7315), .B1(n7449), .B2(n7483), .B3(n7382)
   , .ZN(n7364) );
INVD0 U8052 ( .I(n7364), .ZN(n7455) );
OAI211D0 U8053 ( .A1(n7316), .A2(n7319), .B(n7455), .C(n7374)
   , .ZN(n7317) );
OR4D0 U8054 ( .A1(n7318), .A2(n7357), .A3(n7537), .A4(n7317),
    .Z(n7344) );
CKND2D0 U8055 ( .A1(n7319), .A2(n7328), .ZN(n7409) );
CKND2D0 U8056 ( .A1(n8979), .A2(n7409), .ZN(n7515) );
```

```
INVD0 U8057 ( .I(n7515), .ZN(n7386) );
OR4D0 U8058 ( .A1(n7325), .A2(n7363), .A3(n7344), .A4(n7386),
    .Z(n7320) );
AOI221D0 U8059 ( .A1(n7497), .A2(n7529), .B1(n7320), .B2(
   n7529), .C(
      counter[10]), .ZN(n7355) );
CKND2D0 U8060 ( .A1(n9153), .A2(counter[5]), .ZN(n7448) );
INVD0 U8061 ( .I(n7448), .ZN(n7534) );
CKND2D0 U8062 ( .A1(n8156), .A2(n8971), .ZN(n8965) );
AOI21D0 U8063 ( .A1(n8013), .A2(n8965), .B(counter[6]), .ZN(
   n7341) );
NR2D0 U8064 ( .A1(n7321), .A2(n7366), .ZN(n7465) );
AOI32D0 U8065 ( .A1(n7492), .A2(n7323), .A3(n7322), .B1(n8982
   ), .B2(n7323),
      .ZN(n7324) );
AOI211D0 U8066 ( .A1(n7326), .A2(n7325), .B(n7465), .C(n7324)
   , .ZN(n7461) );
OAI211D0 U8067 ( .A1(n7328), .A2(n9048), .B(n7461), .C(n7443)
   , .ZN(n7510) );
CKND2D0 U8068 ( .A1(n9172), .A2(n7332), .ZN(n7458) );
CKND2D0 U8069 ( .A1(n8971), .A2(n7327), .ZN(n7532) );
OAI211D0 U8070 ( .A1(n8962), .A2(n7328), .B(n7458), .C(n7532)
   , .ZN(n7499) );
OAI211D0 U8071 ( .A1(n8982), .A2(n8953), .B(n7543), .C(n7329)
   , .ZN(n7459) );
CKND2D0 U8072 ( .A1(n7436), .A2(n8971), .ZN(n7330) );
CKND2D0 U8073 ( .A1(n7512), .A2(n7330), .ZN(n7390) );
AOI221D0 U8074 ( .A1(n7332), .A2(n8997), .B1(n7396), .B2(
   n8997), .C(n7331),
      .ZN(n7457) );
CKND2D0 U8075 ( .A1(n8156), .A2(n8979), .ZN(n9004) );
CKND2D0 U8076 ( .A1(n7457), .A2(n9004), .ZN(n7333) );
INR4D0 U8077 ( .A1(n7500), .B1(n7334), .B2(n7390), .B3(n7333)
   , .ZN(n7446) );
ND4D0 U8078 ( .A1(n7446), .A2(n7449), .A3(n7516), .A4(n7491),
    .ZN(n7335) );
NR4D0 U8079 ( .A1(n7510), .A2(n7499), .A3(n7459), .A4(n7335),
    .ZN(n7370) );
NR2D0 U8080 ( .A1(n8463), .A2(n7336), .ZN(n7340) );
AOI211D0 U8081 ( .A1(n8156), .A2(n7412), .B(n7408), .C(n7392)
   , .ZN(n7338) );
CKND2D0 U8082 ( .A1(n7338), .A2(n7337), .ZN(n7533) );
CKND2D0 U8083 ( .A1(n7339), .A2(n7393), .ZN(n7378) );
NR4D0 U8084 ( .A1(n7391), .A2(n7340), .A3(n7533), .A4(n7378),
    .ZN(n7463) );
CKND2D0 U8085 ( .A1(n7370), .A2(n7463), .ZN(n7484) );
INR4D0 U8086 ( .A1(n7513), .B1(n7342), .B2(n7341), .B3(n7484)
   , .ZN(n7404) );
```

```
OAI211D0 U8087 ( .A1(n8970), .A2(n8769), .B(n7343), .C(n7404)
   , .ZN(n7353) );
NR3D0 U8088 ( .A1(n7346), .A2(n7345), .A3(n7344), .ZN(n7493)
   );
AOI211D0 U8089 ( .A1(n7350), .A2(n7349), .B(n7348), .C(n7347)
   , .ZN(n7351) );
NR2D0 U8090 ( .A1(n9153), .A2(n7514), .ZN(n7503) );
INVD0 U8091 ( .I(n7503), .ZN(n7435) );
AOI31D0 U8092 ( .A1(n7493), .A2(n7351), .A3(n7454), .B(n7435)
   , .ZN(n7352) );
AOI211D0 U8093 ( .A1(n7534), .A2(n7353), .B(n7464), .C(n7352)
   , .ZN(n7354) );
OAI211D0 U8094 ( .A1(n7356), .A2(n7544), .B(n7355), .C(n7354)
   , .ZN(n7434) );
AOI211D0 U8095 ( .A1(n9014), .A2(n7359), .B(n7358), .C(n7357)
   , .ZN(n7372) );
CKND2D0 U8096 ( .A1(n7361), .A2(n7360), .ZN(n7371) );
ND4D0 U8097 ( .A1(n7543), .A2(n7458), .A3(n7371), .A4(n7454),
    .ZN(n7362) );
NR4D0 U8098 ( .A1(n7510), .A2(n7364), .A3(n7363), .A4(n7362),
    .ZN(n7517) );
ND4D0 U8099 ( .A1(n7365), .A2(n7372), .A3(n7517), .A4(n8016),
    .ZN(n7380) );
NR2D0 U8100 ( .A1(n7367), .A2(n7366), .ZN(n7368) );
AOI211D0 U8101 ( .A1(n7369), .A2(n8979), .B(n7408), .C(n7368)
   , .ZN(n7453) );
CKND2D0 U8102 ( .A1(n7370), .A2(n7453), .ZN(n7381) );
OAI211D0 U8103 ( .A1(n7444), .A2(n9110), .B(n7372), .C(n7371)
   , .ZN(n7384) );
INR3D0 U8104 ( .A1(n7374), .B1(n7384), .B2(n7373), .ZN(n7558)
    );
ND3D0 U8105 ( .A1(n7558), .A2(n7375), .A3(n7513), .ZN(n7486)
   );
INR4D0 U8106 ( .A1(n7418), .B1(n7376), .B2(n7381), .B3(n7486)
   , .ZN(n7488) );
IIND4D0 U8107 ( .A1(n7378), .A2(n7392), .B1(n7488), .B2(n7377
   ), .ZN(n7379)
      );
AOI22D0 U8108 ( .A1(n7529), .A2(n7380), .B1(n7494), .B2(n7379
   ), .ZN(n7401)
      );
INVD0 U8109 ( .I(n7381), .ZN(n7479) );
ND3D0 U8110 ( .A1(n7479), .A2(n7382), .A3(n7482), .ZN(n7555)
   );
ND3D0 U8111 ( .A1(n8463), .A2(n7483), .A3(n7454), .ZN(n7383)
   );
OAI31D0 U8112 ( .A1(n7555), .A2(n7384), .A3(n7383), .B(n7534)
   , .ZN(n7400) );
```

```
  NR2D0 U8113 ( .A1(n8156), .A2(n7414), .ZN(n7389) );
  IINR4D0 U8114 ( .A1(n7501), .A2(n7543), .B1(n7386), .B2(n7385
     ), .ZN(n7387)
       );
  OAI211D0 U8115 ( .A1(n7389), .A2(n7388), .B(n7468), .C(n7387)
     , .ZN(n7407) );
  INVD0 U8116 ( .I(n7390), .ZN(n7419) );
  INR4D0 U8117 ( .A1(n7393), .B1(n7465), .B2(n7392), .B3(n7391)
     , .ZN(n7394) );
  OAI21D0 U8118 ( .A1(counter[0]), .A2(n7444), .B(n7394), .ZN(
     n7395) );
  AOI31D0 U8119 ( .A1(n7396), .A2(n9345), .A3(n9193), .B(n7395)
     , .ZN(n7415) );
  NR2D0 U8120 ( .A1(n7489), .A2(n8277), .ZN(n7397) );
  INR2D0 U8121 ( .A1(n7532), .B1(n7397), .ZN(n7442) );
  ND4D0 U8122 ( .A1(n7419), .A2(n7415), .A3(n7442), .A4(n7458),
      .ZN(n7398) );
  OAI31D0 U8123 ( .A1(n7407), .A2(n7462), .A3(n7398), .B(n7503)
     , .ZN(n7399) );
  ND4D0 U8124 ( .A1(n7401), .A2(n9075), .A3(n7400), .A4(n7399),
      .ZN(n7433) );
  CKND2D0 U8125 ( .A1(n7458), .A2(n7481), .ZN(n7441) );
  OA31D0 U8126 ( .A1(n7556), .A2(n7441), .A3(n7403), .B(n7402),
      .Z(n7431) );
  INVD0 U8127 ( .I(n7404), .ZN(n7405) );
  OAI31D0 U8128 ( .A1(n7406), .A2(n8426), .A3(n7405), .B(n7534)
     , .ZN(n7430) );
  AOI211D0 U8129 ( .A1(n9090), .A2(n7409), .B(n7408), .C(n7407)
     , .ZN(n7439) );
  ND4D0 U8130 ( .A1(n7411), .A2(n7439), .A3(n7410), .A4(n7458),
      .ZN(n7424) );
  OAI21D0 U8131 ( .A1(n7414), .A2(n7413), .B(n7412), .ZN(n7525)
      );
  OAI211D0 U8132 ( .A1(n8055), .A2(n8982), .B(n7415), .C(n7525)
     , .ZN(n7416) );
  INR2D0 U8133 ( .A1(n7417), .B1(n7416), .ZN(n7425) );
  ND3D0 U8134 ( .A1(n7425), .A2(n7419), .A3(n7418), .ZN(n7440)
     );
  OAI211D0 U8135 ( .A1(n7422), .A2(n7421), .B(n7532), .C(n7420)
     , .ZN(n7423) );
  OAI31D0 U8136 ( .A1(n7424), .A2(n7440), .A3(n7423), .B(n7503)
     , .ZN(n7429) );
  INVD0 U8137 ( .I(n7424), .ZN(n7561) );
  ND4D0 U8138 ( .A1(n7561), .A2(n7425), .A3(n7442), .A4(n7500),
      .ZN(n7426) );
  OAI21D0 U8139 ( .A1(n7427), .A2(n7426), .B(n7494), .ZN(n7428)
      );
```

```
ND4D0 U8140 ( .A1(n7431), .A2(n7430), .A3(n7429), .A4(n7428),
    .ZN(n7432) );
AOI31D0 U8141 ( .A1(n7434), .A2(n7433), .A3(n7432), .B(n7437)
   , .ZN(n7554) );
AOI32D0 U8142 ( .A1(n9014), .A2(n7437), .A3(n7436), .B1(n7435
   ), .B2(n7437),
      .ZN(n7452) );
IND3D0 U8143 ( .A1(n7440), .B1(n7439), .B2(n7438), .ZN(n7472)
    );
INR3D0 U8144 ( .A1(n7442), .B1(n7472), .B2(n7441), .ZN(n7541)
    );
AN3D0 U8145 ( .A1(n7444), .A2(n7463), .A3(n7443), .Z(n7445) )
   ;
AO31D0 U8146 ( .A1(n7447), .A2(n7446), .A3(n7445), .B(n7556),
    .Z(n7450) );
AOI32D0 U8147 ( .A1(n7541), .A2(n7450), .A3(n7449), .B1(n7448
   ), .B2(n7450),
      .ZN(n7451) );
AOI211D0 U8148 ( .A1(n7453), .A2(n7494), .B(n7452), .C(n7451)
   , .ZN(n7553) );
ND4D0 U8149 ( .A1(n7558), .A2(n7455), .A3(n7515), .A4(n7454),
    .ZN(n7456) );
INR2D0 U8150 ( .A1(n7457), .B1(n7456), .ZN(n7511) );
IND4D0 U8151 ( .A1(n7459), .B1(n7511), .B2(n7482), .B3(n7458)
   , .ZN(n7460) );
INR2D0 U8152 ( .A1(n7461), .B1(n7460), .ZN(n7470) );
NR2D0 U8153 ( .A1(n7462), .A2(n7537), .ZN(n7526) );
INVD0 U8154 ( .I(n7463), .ZN(n7505) );
OR4D0 U8155 ( .A1(n7465), .A2(n7505), .A3(n7464), .A4(n7498),
    .Z(n7466) );
IINR4D0 U8156 ( .A1(n7526), .A2(n7468), .B1(n7467), .B2(n7466
   ), .ZN(n7469)
      );
AOI221D0 U8157 ( .A1(counter[7]), .A2(n7470), .B1(n9153), .B2
   (n7469), .C(
      n7514), .ZN(n7471) );
AOI211D0 U8158 ( .A1(n7529), .A2(n7472), .B(n9075), .C(n7471)
   , .ZN(n7478) );
NR4D0 U8159 ( .A1(n7476), .A2(n7475), .A3(n7474), .A4(n7473),
    .ZN(n7477) );
AOI32D0 U8160 ( .A1(n7479), .A2(n7478), .A3(n7477), .B1(n7544
   ), .B2(n7478),
      .ZN(n7551) );
AN3D0 U8161 ( .A1(n7482), .A2(n7481), .A3(n7480), .Z(n7487) )
   ;
IND4D0 U8162 ( .A1(n7484), .B1(n7487), .B2(n9193), .B3(n7483)
   , .ZN(n7485) );
```

```
AOI221D0 U8163 ( .A1(n7486), .A2(n7534), .B1(n7485), .B2(
   n7534), .C(
      counter[9]), .ZN(n7509) );
OAI211D0 U8164 ( .A1(n7489), .A2(n8277), .B(n7488), .C(n7487)
   , .ZN(n7490) );
OAI31D0 U8165 ( .A1(n7524), .A2(n7557), .A3(n7490), .B(n7529)
   , .ZN(n7508) );
CKND2D0 U8166 ( .A1(n8046), .A2(n9014), .ZN(n9076) );
OAI211D0 U8167 ( .A1(n8463), .A2(n7492), .B(n7491), .C(n9076)
   , .ZN(n7496) );
INVD0 U8168 ( .I(n7493), .ZN(n7495) );
OAI31D0 U8169 ( .A1(n7497), .A2(n7496), .A3(n7495), .B(n7494)
   , .ZN(n7507) );
NR3D0 U8170 ( .A1(n7499), .A2(n8971), .A3(n7498), .ZN(n7502)
   );
ND4D0 U8171 ( .A1(n7502), .A2(n7543), .A3(n7501), .A4(n7500),
    .ZN(n7504) );
OAI31D0 U8172 ( .A1(n7536), .A2(n7505), .A3(n7504), .B(n7503)
   , .ZN(n7506) );
ND4D0 U8173 ( .A1(n7509), .A2(n7508), .A3(n7507), .A4(n7506),
    .ZN(n7550) );
INVD0 U8174 ( .I(n7510), .ZN(n7521) );
AN4D0 U8175 ( .A1(n7514), .A2(n7513), .A3(n7512), .A4(n7511),
    .Z(n7520) );
AN4D0 U8176 ( .A1(n7517), .A2(n7516), .A3(n7515), .A4(counter
   [5]), .Z(n7519)
      );
AOI22D0 U8177 ( .A1(n7521), .A2(n7520), .B1(n7519), .B2(n7518
   ), .ZN(n7530)
      );
INR4D0 U8178 ( .A1(n7525), .B1(n7524), .B2(n7523), .B3(n7522)
   , .ZN(n7527) );
CKND2D0 U8179 ( .A1(n7527), .A2(n7526), .ZN(n7528) );
AOI22D0 U8180 ( .A1(counter[7]), .A2(n7530), .B1(n7529), .B2(
   n7528), .ZN(
      n7548) );
IND4D0 U8181 ( .A1(n7533), .B1(n9004), .B2(n7532), .B3(n7531)
   , .ZN(n7535) );
OAI31D0 U8182 ( .A1(n7537), .A2(n7536), .A3(n7535), .B(n7534)
   , .ZN(n7547) );
CKAN2D0 U8183 ( .A1(n7539), .A2(n7538), .Z(n7540) );
AOI31D0 U8184 ( .A1(n7542), .A2(counter[5]), .A3(n7541), .B(
   n7540), .ZN(
      n7545) );
OAI22D0 U8185 ( .A1(counter[7]), .A2(n7545), .B1(n7544), .B2(
   n7543), .ZN(
      n7546) );
```

```
AOI32D0 U8186 ( .A1(n7548), .A2(n9075), .A3(n7547), .B1(
   counter[9]), .B2(
      n7546), .ZN(n7549) );
AOI32D0 U8187 ( .A1(n7551), .A2(n9342), .A3(n7550), .B1(
   counter[10]), .B2(
      n7549), .ZN(n7552) );
OAI32D0 U8188 ( .A1(counter[8]), .A2(n7554), .A3(n7553), .B1(
   n9338), .B2(
      n7552), .ZN(n7563) );
INVD0 U8189 ( .I(n7555), .ZN(n7560) );
INR4D0 U8190 ( .A1(n7558), .B1(counter[8]), .B2(n7557), .B3(
   n7556), .ZN(
      n7559) );
AOI22D0 U8191 ( .A1(counter[8]), .A2(n7561), .B1(n7560), .B2(
   n7559), .ZN(
      n7562) );
OAI221D0 U8192 ( .A1(counter[11]), .A2(n7563), .B1(n9329), .
   B2(n7562), .C(
      n8949), .ZN(n7564) );
CKND2D0 U8193 ( .A1(n7565), .A2(n7564), .ZN(n4817) );
INVD0 U8194 ( .I(reset_nanochip), .ZN(n4811) );
INVD0 U8195 ( .I(n7735), .ZN(n7776) );
INVD0 U8196 ( .I(n7647), .ZN(n7851) );
NR2D0 U8197 ( .A1(n7608), .A2(n7652), .ZN(n7845) );
IND4D0 U8198 ( .A1(n7658), .B1(n7851), .B2(n7566), .B3(n7845)
   , .ZN(n7573) );
NR4D0 U8199 ( .A1(n7911), .A2(n7814), .A3(n7890), .A4(n7798),
    .ZN(n7567) );
AOI31D0 U8200 ( .A1(n7568), .A2(n7687), .A3(n7567), .B(n9274)
   , .ZN(n7572) );
NR2D0 U8201 ( .A1(n7659), .A2(n7724), .ZN(n7921) );
CKND2D0 U8202 ( .A1(n7867), .A2(n7617), .ZN(n7598) );
NR4D0 U8203 ( .A1(n7854), .A2(n7569), .A3(n7647), .A4(n7598),
    .ZN(n7570) );
AOI31D0 U8204 ( .A1(n7893), .A2(n7921), .A3(n7570), .B(n9300)
   , .ZN(n7571) );
AOI211D0 U8205 ( .A1(n9226), .A2(n7573), .B(n7572), .C(n7571)
   , .ZN(n7576) );
NR4D0 U8206 ( .A1(n7584), .A2(n7605), .A3(n7874), .A4(n7574),
    .ZN(n7575) );
AOI32D0 U8207 ( .A1(n7776), .A2(n7576), .A3(n7575), .B1(n9272
   ), .B2(n7576),
      .ZN(n7615) );
INVD0 U8208 ( .I(n7918), .ZN(n7690) );
NR3D0 U8209 ( .A1(n7690), .A2(n7824), .A3(n7963), .ZN(n7670)
   );
CKND2D0 U8210 ( .A1(n7867), .A2(n7577), .ZN(n7884) );
```

```
INR4D0 U8211 ( .A1(n7670), .B1(n7578), .B2(n7676), .B3(n7884)
   , .ZN(n7595) );
CKND2D0 U8212 ( .A1(n7636), .A2(n7892), .ZN(n7823) );
OR4D0 U8213 ( .A1(n7823), .A2(n7910), .A3(n7580), .A4(n7579),
    .Z(n7592) );
CKND2D0 U8214 ( .A1(n7721), .A2(n7641), .ZN(n7622) );
INVD0 U8215 ( .I(n7622), .ZN(n7970) );
ND4D0 U8216 ( .A1(n7582), .A2(n7970), .A3(n7581), .A4(n7949),
    .ZN(n7583) );
OAI31D0 U8217 ( .A1(n7585), .A2(n7584), .A3(n7583), .B(n9307)
   , .ZN(n7589) );
CKND2D0 U8218 ( .A1(n7776), .A2(n7687), .ZN(n7765) );
NR4D0 U8219 ( .A1(n7587), .A2(n7586), .A3(n7846), .A4(n7765),
    .ZN(n7588) );
AOI32D0 U8220 ( .A1(n7590), .A2(n7589), .A3(n7588), .B1(n9318
   ), .B2(n7589),
      .ZN(n7591) );
AOI221D0 U8221 ( .A1(n7735), .A2(n9141), .B1(n7592), .B2(
   n9141), .C(n7591),
      .ZN(n7594) );
AOI221D0 U8222 ( .A1(n7595), .A2(n7594), .B1(n9274), .B2(
   n7594), .C(n7593),
      .ZN(n7614) );
INVD0 U8223 ( .I(n7650), .ZN(n7639) );
NR3D0 U8224 ( .A1(n7597), .A2(n7596), .A3(n7639), .ZN(n7619)
   );
ND4D0 U8225 ( .A1(n7619), .A2(n7897), .A3(n7858), .A4(n7923),
    .ZN(n7602) );
INVD0 U8226 ( .I(n7616), .ZN(n7700) );
NR4D0 U8227 ( .A1(n7635), .A2(n7724), .A3(n7598), .A4(n7700),
    .ZN(n7599) );
AOI31D0 U8228 ( .A1(n7775), .A2(n7600), .A3(n7599), .B(n9318)
   , .ZN(n7601) );
AOI221D0 U8229 ( .A1(n7779), .A2(n9141), .B1(n7602), .B2(
   n9141), .C(n7601),
      .ZN(n7612) );
IND4D0 U8230 ( .A1(n7888), .B1(n7970), .B2(n7974), .B3(n7603)
   , .ZN(n7604) );
OAI21D0 U8231 ( .A1(n7605), .A2(n7604), .B(n9312), .ZN(n7611)
    );
IND4D0 U8232 ( .A1(n7765), .B1(n7974), .B2(n7893), .B3(n7606)
   , .ZN(n7607) );
OAI31D0 U8233 ( .A1(n7609), .A2(n7608), .A3(n7607), .B(n9307)
   , .ZN(n7610) );
AOI31D0 U8234 ( .A1(n7612), .A2(n7611), .A3(n7610), .B(n8687)
   , .ZN(n7613) );
AOI211D0 U8235 ( .A1(n8939), .A2(n7615), .B(n7614), .C(n7613)
   , .ZN(n8008) );
```

```
ND3D0 U8236 ( .A1(n7671), .A2(n7616), .A3(n7866), .ZN(n7802)
   );
INVD0 U8237 ( .I(n7802), .ZN(n7632) );
ND4D0 U8238 ( .A1(n7619), .A2(n7797), .A3(n7618), .A4(n7617),
    .ZN(n7627) );
OAI31D0 U8239 ( .A1(n7853), .A2(n7818), .A3(n7620), .B(n9141)
   , .ZN(n7624) );
NR4D0 U8240 ( .A1(n7622), .A2(n7765), .A3(n7621), .A4(n7837),
    .ZN(n7623) );
AOI32D0 U8241 ( .A1(n7625), .A2(n7624), .A3(n7623), .B1(n9318
   ), .B2(n7624),
      .ZN(n7626) );
AOI221D0 U8242 ( .A1(n7690), .A2(n9312), .B1(n7627), .B2(
   n9312), .C(n7626),
      .ZN(n7631) );
ND4D0 U8243 ( .A1(n7873), .A2(n7776), .A3(n7990), .A4(n7628),
    .ZN(n7629) );
INR4D0 U8244 ( .A1(n7893), .B1(n7659), .B2(n7634), .B3(n7629)
   , .ZN(n7630) );
AOI32D0 U8245 ( .A1(n7632), .A2(n7631), .A3(n7630), .B1(n9272
   ), .B2(n7631),
      .ZN(n7706) );
NR2D0 U8246 ( .A1(n7911), .A2(n7814), .ZN(n7638) );
CKND2D0 U8247 ( .A1(n7797), .A2(n7959), .ZN(n7698) );
NR4D0 U8248 ( .A1(n7635), .A2(n7634), .A3(n7633), .A4(n7698),
    .ZN(n7637) );
ND4D0 U8249 ( .A1(n7638), .A2(n7637), .A3(n7636), .A4(n7922),
    .ZN(n7645) );
NR4D0 U8250 ( .A1(n7819), .A2(n7640), .A3(n7647), .A4(n7639),
    .ZN(n7643) );
ND4D0 U8251 ( .A1(n7897), .A2(n7643), .A3(n7642), .A4(n7641),
    .ZN(n7644) );
AOI22D0 U8252 ( .A1(n9226), .A2(n7645), .B1(n9141), .B2(n7644
   ), .ZN(n7655)
       );
OAI31D0 U8253 ( .A1(n7648), .A2(n7647), .A3(n7646), .B(n9312)
   , .ZN(n7654) );
INVD0 U8254 ( .I(n7830), .ZN(n7793) );
ND4D0 U8255 ( .A1(n7650), .A2(n7793), .A3(n7649), .A4(n7668),
    .ZN(n7651) );
OAI31D0 U8256 ( .A1(n7652), .A2(n7735), .A3(n7651), .B(n9307)
   , .ZN(n7653) );
AOI31D0 U8257 ( .A1(n7655), .A2(n7654), .A3(n7653), .B(n8945)
   , .ZN(n7705) );
ND4D0 U8258 ( .A1(n7949), .A2(n7699), .A3(n7657), .A4(n7656),
    .ZN(n7667) );
NR4D0 U8259 ( .A1(n7659), .A2(n7658), .A3(n7871), .A4(n7803),
    .ZN(n7660) );
```

```
AOI31D0 U8260 ( .A1(n7661), .A2(n7805), .A3(n7660), .B(n9274)
   , .ZN(n7666) );
NR4D0 U8261 ( .A1(n7927), .A2(n7700), .A3(n7662), .A4(n7727),
    .ZN(n7663) );
AOI31D0 U8262 ( .A1(n7664), .A2(n7797), .A3(n7663), .B(n9300)
   , .ZN(n7665) );
AOI211D0 U8263 ( .A1(n9307), .A2(n7667), .B(n7666), .C(n7665)
   , .ZN(n7703) );
AOI31D0 U8264 ( .A1(n7670), .A2(n7669), .A3(n7668), .B(n9272)
   , .ZN(n7697) );
OAI211D0 U8265 ( .A1(n7673), .A2(n7672), .B(n7671), .C(n7740)
   , .ZN(n7953) );
NR4D0 U8266 ( .A1(n7676), .A2(n7675), .A3(n7953), .A4(n7674),
    .ZN(n7686) );
AOI211D0 U8267 ( .A1(n7709), .A2(n8046), .B(n7749), .C(n7677)
   , .ZN(n7679) );
ND4D0 U8268 ( .A1(n7681), .A2(n7680), .A3(n7679), .A4(n7678),
    .ZN(n7957) );
NR4D0 U8269 ( .A1(n7684), .A2(n7683), .A3(n7957), .A4(n7682),
    .ZN(n7685) );
AOI21D0 U8270 ( .A1(n7686), .A2(n7685), .B(n9318), .ZN(n7696)
    );
ND4D0 U8271 ( .A1(n7687), .A2(n7974), .A3(n7990), .A4(n7858),
    .ZN(n7688) );
NR4D0 U8272 ( .A1(n7689), .A2(n7997), .A3(n7901), .A4(n7688),
    .ZN(n7693) );
NR2D0 U8273 ( .A1(n7690), .A2(n7941), .ZN(n7968) );
IINR4D0 U8274 ( .A1(n7968), .A2(n7845), .B1(n7813), .B2(n7691
   ), .ZN(n7692)
       );
OAI22D0 U8275 ( .A1(n7693), .A2(n9300), .B1(n7692), .B2(n9274
   ), .ZN(n7695)
       );
OAI31D0 U8276 ( .A1(n7697), .A2(n7696), .A3(n7695), .B(n7694)
   , .ZN(n7702) );
INVD0 U8277 ( .I(n7698), .ZN(n7937) );
ND3D0 U8278 ( .A1(n7937), .A2(n7827), .A3(n7699), .ZN(n7914)
   );
OAI31D0 U8279 ( .A1(n7700), .A2(n7779), .A3(n7914), .B(n9226)
   , .ZN(n7701) );
AOI32D0 U8280 ( .A1(n7703), .A2(n7702), .A3(n7701), .B1(n8903
   ), .B2(n7702),
      .ZN(n7704) );
AOI211D0 U8281 ( .A1(n8037), .A2(n7706), .B(n7705), .C(n7704)
   , .ZN(n8007) );
AOI211D0 U8282 ( .A1(n7709), .A2(n7708), .B(n7707), .C(n7855)
   , .ZN(n7763) );
INR2D0 U8283 ( .A1(n7711), .B1(n7710), .ZN(n7961) );
```

```
  NR2D0 U8284 ( .A1(n7804), .A2(n7839), .ZN(n7913) );
  INR4D0 U8285 ( .A1(n7715), .B1(n7714), .B2(n7713), .B3(n7712)
     , .ZN(n7716) );
  CKND2D0 U8286 ( .A1(n7913), .A2(n7716), .ZN(n7760) );
  OAI22D0 U8287 ( .A1(n8962), .A2(n8768), .B1(n8991), .B2(n7717
     ), .ZN(n7718)
       );
  NR3D0 U8288 ( .A1(n7741), .A2(n7719), .A3(n7718), .ZN(n7736)
     );
  IND4D0 U8289 ( .A1(n7722), .B1(n7736), .B2(n7721), .B3(n7720)
     , .ZN(n7723) );
  INR4D0 U8290 ( .A1(n7961), .B1(n7724), .B2(n7760), .B3(n7723)
     , .ZN(n7726) );
  IND4D0 U8291 ( .A1(n7727), .B1(n7726), .B2(n7725), .B3(n7850)
     , .ZN(n7754) );
  ND4D0 U8292 ( .A1(n7731), .A2(n7730), .A3(n7729), .A4(n7728),
      .ZN(n7732) );
  NR2D0 U8293 ( .A1(n7733), .A2(n7732), .ZN(n7954) );
  OAI22D0 U8294 ( .A1(counter[1]), .A2(n7972), .B1(n8725), .B2(
     n9029), .ZN(
        n7734) );
  INR4D0 U8295 ( .A1(n7736), .B1(n7735), .B2(n7941), .B3(n7734)
     , .ZN(n7737) );
  AOI31D0 U8296 ( .A1(n7738), .A2(n7954), .A3(n7737), .B(n9300)
     , .ZN(n7753) );
  INR4D0 U8297 ( .A1(n7740), .B1(n7911), .B2(n7739), .B3(n7896)
     , .ZN(n7751) );
  INVD0 U8298 ( .I(n7741), .ZN(n7973) );
  OAI211D0 U8299 ( .A1(n7744), .A2(n7743), .B(n7742), .C(n7973)
     , .ZN(n7745) );
  AOI211D0 U8300 ( .A1(counter[11]), .A2(n9299), .B(n7746), .C(
     n7745), .ZN(
        n7980) );
  ND4D0 U8301 ( .A1(n7873), .A2(n7938), .A3(n7850), .A4(n7774),
      .ZN(n7747) );
  NR4D0 U8302 ( .A1(n7749), .A2(n7748), .A3(n7760), .A4(n7747),
      .ZN(n7750) );
  AOI31D0 U8303 ( .A1(n7751), .A2(n7980), .A3(n7750), .B(n9274)
     , .ZN(n7752) );
  AOI211D0 U8304 ( .A1(n9307), .A2(n7754), .B(n7753), .C(n7752)
     , .ZN(n7762) );
  OAI211D0 U8305 ( .A1(n7756), .A2(n9311), .B(n7938), .C(n7755)
     , .ZN(n7996) );
  CKND2D0 U8306 ( .A1(n7758), .A2(n7757), .ZN(n7887) );
  NR4D0 U8307 ( .A1(n7996), .A2(n7887), .A3(n7760), .A4(n7759),
      .ZN(n7761) );
  AOI32D0 U8308 ( .A1(n7763), .A2(n7762), .A3(n7761), .B1(n9318
     ), .B2(n7762),
```

```
      .ZN(n7947) );
NR4D0 U8309 ( .A1(n7767), .A2(n7766), .A3(n7765), .A4(n7764),
    .ZN(n7768) );
AOI31D0 U8310 ( .A1(n7886), .A2(n7938), .A3(n7768), .B(n9300)
   , .ZN(n7791) );
NR4D0 U8311 ( .A1(n7771), .A2(n7770), .A3(n7941), .A4(n7769),
    .ZN(n7773) );
AOI31D0 U8312 ( .A1(n7773), .A2(n8314), .A3(n7772), .B(n9274)
   , .ZN(n7790) );
ND4D0 U8313 ( .A1(n7776), .A2(n7775), .A3(n7923), .A4(n7774),
    .ZN(n7777) );
NR4D0 U8314 ( .A1(n7900), .A2(n7779), .A3(n7778), .A4(n7777),
    .ZN(n7788) );
ND4D0 U8315 ( .A1(n7782), .A2(n7893), .A3(n7781), .A4(n7780),
    .ZN(n7783) );
AOI211D0 U8316 ( .A1(n7786), .A2(n7785), .B(n7784), .C(n7783)
   , .ZN(n7787) );
OAI22D0 U8317 ( .A1(n7788), .A2(n9318), .B1(n7787), .B2(n9272
   ), .ZN(n7789)
      );
OAI31D0 U8318 ( .A1(n7791), .A2(n7790), .A3(n7789), .B(n8715)
   , .ZN(n7865) );
AOI31D0 U8319 ( .A1(n7794), .A2(n7793), .A3(n7792), .B(n9318)
   , .ZN(n7810) );
ND3D0 U8320 ( .A1(n7797), .A2(n7796), .A3(n7795), .ZN(n7822)
   );
NR4D0 U8321 ( .A1(n7819), .A2(n7799), .A3(n7798), .A4(n7822),
    .ZN(n7800) );
AOI21D0 U8322 ( .A1(n7866), .A2(n7800), .B(n9300), .ZN(n7809)
    );
NR4D0 U8323 ( .A1(n7941), .A2(n7803), .A3(n7802), .A4(n7801),
    .ZN(n7807) );
INR4D0 U8324 ( .A1(n7805), .B1(n8225), .B2(n7804), .B3(n7823)
   , .ZN(n7806) );
OAI22D0 U8325 ( .A1(n7807), .A2(n9272), .B1(n7806), .B2(n9274
   ), .ZN(n7808)
      );
OAI31D0 U8326 ( .A1(n7810), .A2(n7809), .A3(n7808), .B(n8740)
   , .ZN(n7864) );
NR4D0 U8327 ( .A1(n7814), .A2(n7813), .A3(n7812), .A4(n7811),
    .ZN(n7817) );
CKND2D0 U8328 ( .A1(n7815), .A2(n8461), .ZN(n7816) );
AOI31D0 U8329 ( .A1(n7817), .A2(n7974), .A3(n7816), .B(n9318)
   , .ZN(n7835) );
NR4D0 U8330 ( .A1(n7819), .A2(n7985), .A3(n7896), .A4(n7818),
    .ZN(n7820) );
AOI31D0 U8331 ( .A1(n7989), .A2(n7872), .A3(n7820), .B(n9274)
   , .ZN(n7834) );
```

```
NR4D0 U8332 ( .A1(n7824), .A2(n7823), .A3(n7822), .A4(n7821),
    .ZN(n7832) );
ND4D0 U8333 ( .A1(n7827), .A2(n8394), .A3(n7826), .A4(n7825),
    .ZN(n7828) );
NR4D0 U8334 ( .A1(n7941), .A2(n7830), .A3(n7829), .A4(n7828),
    .ZN(n7831) );
OAI22D0 U8335 ( .A1(n7832), .A2(n9300), .B1(n7831), .B2(n9272
   ), .ZN(n7833)
       );
OAI31D0 U8336 ( .A1(n7835), .A2(n7834), .A3(n7833), .B(n8803)
   , .ZN(n7863) );
CKND2D0 U8337 ( .A1(n7850), .A2(n7915), .ZN(n7838) );
NR4D0 U8338 ( .A1(n7839), .A2(n7838), .A3(n7837), .A4(n7836),
    .ZN(n7840) );
AOI31D0 U8339 ( .A1(n7921), .A2(n7841), .A3(n7840), .B(n9274)
   , .ZN(n7861) );
INR4D0 U8340 ( .A1(n7913), .B1(n7997), .B2(n7853), .B3(n7887)
   , .ZN(n7843) );
AOI31D0 U8341 ( .A1(n7843), .A2(n7842), .A3(n7850), .B(n9318)
   , .ZN(n7860) );
IND4D0 U8342 ( .A1(n7846), .B1(n7845), .B2(n7844), .B3(n7922)
   , .ZN(n7847) );
OAI31D0 U8343 ( .A1(n8011), .A2(n7848), .A3(n7847), .B(n9307)
   , .ZN(n7857) );
ND4D0 U8344 ( .A1(n7851), .A2(n7850), .A3(n7915), .A4(n7849),
    .ZN(n7852) );
NR4D0 U8345 ( .A1(n7855), .A2(n7854), .A3(n7853), .A4(n7852),
    .ZN(n7856) );
AOI32D0 U8346 ( .A1(n7858), .A2(n7857), .A3(n7856), .B1(n9300
   ), .B2(n7857),
      .ZN(n7859) );
OAI31D0 U8347 ( .A1(n7861), .A2(n7860), .A3(n7859), .B(n8910)
   , .ZN(n7862) );
ND4D0 U8348 ( .A1(n7865), .A2(n7864), .A3(n7863), .A4(n7862),
    .ZN(n7946) );
AN3D0 U8349 ( .A1(n7866), .A2(n7969), .A3(n7922), .Z(n7885) )
   ;
ND4D0 U8350 ( .A1(n7869), .A2(n7885), .A3(n7868), .A4(n7867),
    .ZN(n7881) );
OR3D0 U8351 ( .A1(n7871), .A2(n7870), .A3(n7890), .Z(n7909) )
   ;
CKND2D0 U8352 ( .A1(n7873), .A2(n7872), .ZN(n7904) );
OR4D0 U8353 ( .A1(n7985), .A2(n7874), .A3(n7909), .A4(n7904),
    .Z(n7880) );
ND3D0 U8354 ( .A1(n7877), .A2(n7876), .A3(n7875), .ZN(n7883)
   );
IND4D0 U8355 ( .A1(n7883), .B1(n7918), .B2(n7990), .B3(n7878)
   , .ZN(n7879) );
```

```
AOI222D0 U8356 ( .A1(n7881), .A2(n9307), .B1(n7880), .B2(
   n9312), .C1(n7879),
      .C2(n9141), .ZN(n7944) );
NR4D0 U8357 ( .A1(n7890), .A2(n7884), .A3(n7883), .A4(n7882),
    .ZN(n7907) );
ND4D0 U8358 ( .A1(n7897), .A2(n7886), .A3(n7885), .A4(n7912),
    .ZN(n7895) );
NR4D0 U8359 ( .A1(n7890), .A2(n7889), .A3(n7888), .A4(n7887),
    .ZN(n7891) );
AOI31D0 U8360 ( .A1(n7893), .A2(n7892), .A3(n7891), .B(n9300)
   , .ZN(n7894) );
AOI221D0 U8361 ( .A1(n7896), .A2(n9312), .B1(n7895), .B2(
   n9312), .C(n7894),
      .ZN(n7906) );
INVD0 U8362 ( .I(n7897), .ZN(n7903) );
OR4D0 U8363 ( .A1(n7901), .A2(n7900), .A3(n7899), .A4(n7898),
    .Z(n7902) );
OAI31D0 U8364 ( .A1(n7904), .A2(n7903), .A3(n7902), .B(n9307)
   , .ZN(n7905) );
OAI211D0 U8365 ( .A1(n7907), .A2(n9318), .B(n7906), .C(n7905)
   , .ZN(n7934) );
NR4D0 U8366 ( .A1(n7911), .A2(n7910), .A3(n7909), .A4(n7908),
    .ZN(n7931) );
IND4D0 U8367 ( .A1(n7914), .B1(n7968), .B2(n7913), .B3(n7912)
   , .ZN(n7920) );
ND4D0 U8368 ( .A1(n7918), .A2(n7917), .A3(n7916), .A4(n7915),
    .ZN(n7919) );
AOI22D0 U8369 ( .A1(n9226), .A2(n7920), .B1(n9312), .B2(n7919
   ), .ZN(n7930)
      );
INVD0 U8370 ( .I(n7921), .ZN(n7928) );
ND4D0 U8371 ( .A1(n7925), .A2(n7924), .A3(n7923), .A4(n7922),
    .ZN(n7926) );
OAI31D0 U8372 ( .A1(n7928), .A2(n7927), .A3(n7926), .B(n9141)
   , .ZN(n7929) );
OAI211D0 U8373 ( .A1(n7931), .A2(n9272), .B(n7930), .C(n7929)
   , .ZN(n7932) );
AOI22D0 U8374 ( .A1(n8656), .A2(n7934), .B1(n7933), .B2(n7932
   ), .ZN(n7943)
      );
ND4D0 U8375 ( .A1(n7938), .A2(n7937), .A3(n7936), .A4(n7935),
    .ZN(n7940) );
OAI211D0 U8376 ( .A1(n7941), .A2(n7940), .B(n9226), .C(n7939)
   , .ZN(n7942) );
OAI211D0 U8377 ( .A1(n7944), .A2(n8848), .B(n7943), .C(n7942)
   , .ZN(n7945) );
AOI211D0 U8378 ( .A1(n8639), .A2(n7947), .B(n7946), .C(n7945)
   , .ZN(n8006) );
```

```
OAI211D0 U8379 ( .A1(n7950), .A2(n9348), .B(n7949), .C(n7948)
   , .ZN(n7951) );
INR4D0 U8380 ( .A1(n7954), .B1(n7953), .B2(n7952), .B3(n7951)
   , .ZN(n7956) );
AOI31D0 U8381 ( .A1(n7956), .A2(n7955), .A3(n7958), .B(n9274)
   , .ZN(n8004) );
INVD0 U8382 ( .I(n7957), .ZN(n7967) );
ND4D0 U8383 ( .A1(n7961), .A2(n7960), .A3(n7959), .A4(n7958),
    .ZN(n7962) );
NR4D0 U8384 ( .A1(n7965), .A2(n7964), .A3(n7963), .A4(n7962),
    .ZN(n7966) );
AOI31D0 U8385 ( .A1(n7968), .A2(n7967), .A3(n7966), .B(n9272)
   , .ZN(n8003) );
OAI211D0 U8386 ( .A1(n7972), .A2(n7971), .B(n7970), .C(n7969)
   , .ZN(n7976) );
IND4D0 U8387 ( .A1(n8225), .B1(n7974), .B2(n8039), .B3(n7973)
   , .ZN(n7975) );
AOI211D0 U8388 ( .A1(n8052), .A2(n7977), .B(n7976), .C(n7975)
   , .ZN(n8001) );
CKND2D0 U8389 ( .A1(n7979), .A2(n7978), .ZN(n7994) );
INVD0 U8390 ( .I(n7980), .ZN(n7993) );
NR4D0 U8391 ( .A1(n8011), .A2(n7983), .A3(n7982), .A4(n7981),
    .ZN(n7991) );
AOI211D0 U8392 ( .A1(n7987), .A2(n7986), .B(n7985), .C(n7984)
   , .ZN(n7988) );
ND4D0 U8393 ( .A1(n7991), .A2(n7990), .A3(n7989), .A4(n7988),
    .ZN(n7992) );
OAI31D0 U8394 ( .A1(n7994), .A2(n7993), .A3(n7992), .B(n9141)
   , .ZN(n8000) );
NR4D0 U8395 ( .A1(n7998), .A2(n7997), .A3(n7996), .A4(n7995),
    .ZN(n7999) );
AOI32D0 U8396 ( .A1(n8001), .A2(n8000), .A3(n7999), .B1(n9318
   ), .B2(n8000),
      .ZN(n8002) );
OAI31D0 U8397 ( .A1(n8004), .A2(n8003), .A3(n8002), .B(n8918)
   , .ZN(n8005) );
ND4D0 U8398 ( .A1(n8008), .A2(n8007), .A3(n8006), .A4(n8005),
    .ZN(n8457) );
CKND2D0 U8399 ( .A1(n9348), .A2(n8009), .ZN(n8209) );
NR2D0 U8400 ( .A1(n8050), .A2(n8209), .ZN(n8182) );
NR2D0 U8401 ( .A1(n9348), .A2(n8021), .ZN(n8290) );
NR2D0 U8402 ( .A1(n8182), .A2(n8290), .ZN(n8259) );
INVD0 U8403 ( .I(n8259), .ZN(n8177) );
NR2D0 U8404 ( .A1(n9348), .A2(n8276), .ZN(n8184) );
CKND2D0 U8405 ( .A1(n8130), .A2(n8184), .ZN(n8213) );
INVD0 U8406 ( .I(n8209), .ZN(n8385) );
CKND2D0 U8407 ( .A1(n8126), .A2(n8385), .ZN(n8400) );
CKND2D0 U8408 ( .A1(n8213), .A2(n8400), .ZN(n8091) );
```

```
  INVD0 U8409 ( .I(n8091), .ZN(n8307) );
  CKND2D0 U8410 ( .A1(counter[4]), .A2(n8420), .ZN(n8295) );
  CKND2D0 U8411 ( .A1(n8307), .A2(n8295), .ZN(n8435) );
  NR2D0 U8412 ( .A1(n8014), .A2(counter[4]), .ZN(n8155) );
  INVD0 U8413 ( .I(n8155), .ZN(n8027) );
  NR2D0 U8414 ( .A1(n9344), .A2(n8027), .ZN(n8370) );
  CKND2D0 U8415 ( .A1(n8042), .A2(n8385), .ZN(n8282) );
  INVD0 U8416 ( .I(n8282), .ZN(n8377) );
  OR2D0 U8417 ( .A1(n8370), .A2(n8377), .Z(n8304) );
  NR2D0 U8418 ( .A1(n8050), .A2(n8027), .ZN(n8222) );
  INVD0 U8419 ( .I(n8184), .ZN(n8115) );
  NR2D0 U8420 ( .A1(n8143), .A2(n8115), .ZN(n8415) );
  NR3D0 U8421 ( .A1(n8304), .A2(n8222), .A3(n8415), .ZN(n8114)
     );
  INVD0 U8422 ( .I(n8114), .ZN(n8387) );
  ND3D0 U8423 ( .A1(n9329), .A2(n8822), .A3(n8141), .ZN(n8439)
     );
  CKND2D0 U8424 ( .A1(n8130), .A2(n8155), .ZN(n8163) );
  CKND2D0 U8425 ( .A1(n8439), .A2(n8163), .ZN(n8265) );
  INVD0 U8426 ( .I(n8265), .ZN(n8236) );
  NR2D0 U8427 ( .A1(n8143), .A2(n8209), .ZN(n8221) );
  CKND2D0 U8428 ( .A1(n8822), .A2(n8010), .ZN(n8154) );
  NR2D0 U8429 ( .A1(n8154), .A2(counter[0]), .ZN(n8076) );
  INVD0 U8430 ( .I(n8076), .ZN(n8328) );
  CKND2D0 U8431 ( .A1(counter[5]), .A2(n8011), .ZN(n8393) );
  CKND2D0 U8432 ( .A1(n8328), .A2(n8393), .ZN(n8093) );
  NR2D0 U8433 ( .A1(n8221), .A2(n8093), .ZN(n8275) );
  NR2D0 U8434 ( .A1(n8142), .A2(n8115), .ZN(n8351) );
  INVD0 U8435 ( .I(n8351), .ZN(n8313) );
  NR2D0 U8436 ( .A1(n8277), .A2(n8209), .ZN(n8383) );
  INVD0 U8437 ( .I(n8383), .ZN(n8359) );
  ND4D0 U8438 ( .A1(n8236), .A2(n8275), .A3(n8313), .A4(n8359),
      .ZN(n8012) );
  NR4D0 U8439 ( .A1(n8177), .A2(n8435), .A3(n8387), .A4(n8012),
      .ZN(n8025) );
  NR2D0 U8440 ( .A1(n8014), .A2(n8013), .ZN(n8201) );
  CKND2D0 U8441 ( .A1(counter[4]), .A2(n8280), .ZN(n8356) );
  CKND2D0 U8442 ( .A1(n8163), .A2(n8356), .ZN(n8349) );
  NR2D0 U8443 ( .A1(n8201), .A2(n8349), .ZN(n8218) );
  CKND2D0 U8444 ( .A1(n8078), .A2(n8385), .ZN(n8358) );
  INVD0 U8445 ( .I(n8358), .ZN(n8289) );
  CKND2D0 U8446 ( .A1(n8385), .A2(n8384), .ZN(n8015) );
  CKND2D0 U8447 ( .A1(n8393), .A2(n8015), .ZN(n8325) );
  NR2D0 U8448 ( .A1(n8289), .A2(n8325), .ZN(n8212) );
  CKND2D0 U8449 ( .A1(n8218), .A2(n8212), .ZN(n8026) );
  CKND2D0 U8450 ( .A1(n8394), .A2(n8328), .ZN(n8261) );
  NR3D0 U8451 ( .A1(n8186), .A2(n8919), .A3(counter[11]), .ZN(
     n8257) );
```

```
  NR2D0 U8452 ( .A1(n8257), .A2(n8289), .ZN(n8208) );
  CKND2D0 U8453 ( .A1(n8439), .A2(n8208), .ZN(n8299) );
  INVD0 U8454 ( .I(n8299), .ZN(n8338) );
  CKND2D0 U8455 ( .A1(n8078), .A2(n8184), .ZN(n8132) );
  CKND2D0 U8456 ( .A1(n8126), .A2(n8155), .ZN(n8430) );
  CKND2D0 U8457 ( .A1(n8132), .A2(n8430), .ZN(n8237) );
  INVD0 U8458 ( .I(n8237), .ZN(n8353) );
  NR2D0 U8459 ( .A1(n8016), .A2(n8115), .ZN(n8300) );
  INVD0 U8460 ( .I(n8300), .ZN(n8223) );
  IND4D0 U8461 ( .A1(n8261), .B1(n8338), .B2(n8353), .B3(n8223)
     , .ZN(n8019) );
  CKND2D0 U8462 ( .A1(n8141), .A2(n8184), .ZN(n8253) );
  CKND2D0 U8463 ( .A1(counter[1]), .A2(n8201), .ZN(n8294) );
  CKND2D0 U8464 ( .A1(n8253), .A2(n8294), .ZN(n8386) );
  NR2D0 U8465 ( .A1(n8386), .A2(n8257), .ZN(n8136) );
  INVD0 U8466 ( .I(n8136), .ZN(n8274) );
  INVD0 U8467 ( .I(n8182), .ZN(n8129) );
  CKND2D0 U8468 ( .A1(n8275), .A2(n8129), .ZN(n8436) );
  NR2D0 U8469 ( .A1(n8016), .A2(n8027), .ZN(n8382) );
  NR2D0 U8470 ( .A1(n9344), .A2(n8209), .ZN(n8243) );
  NR2D0 U8471 ( .A1(n8382), .A2(n8243), .ZN(n8360) );
  INVD0 U8472 ( .I(n8360), .ZN(n8192) );
  INR4D0 U8473 ( .A1(n8213), .B1(n8274), .B2(n8436), .B3(n8192)
     , .ZN(n8017) );
  CKND2D0 U8474 ( .A1(n8130), .A2(n8385), .ZN(n8431) );
  AOI31D0 U8475 ( .A1(n8017), .A2(n8431), .A3(n8430), .B(n8416)
     , .ZN(n8018) );
  AOI221D0 U8476 ( .A1(n8026), .A2(n8311), .B1(n8019), .B2(
     n8311), .C(n8018),
        .ZN(n8024) );
  NR2D0 U8477 ( .A1(n8020), .A2(n8383), .ZN(n8287) );
  ND3D0 U8478 ( .A1(n8287), .A2(n8356), .A3(n8400), .ZN(n8318)
     );
  NR2D0 U8479 ( .A1(n8277), .A2(n8115), .ZN(n8140) );
  NR2D0 U8480 ( .A1(n8222), .A2(n8140), .ZN(n8395) );
  CKND2D0 U8481 ( .A1(n8395), .A2(n8208), .ZN(n8424) );
  INVD0 U8482 ( .I(n8415), .ZN(n8262) );
  CKND2D0 U8483 ( .A1(n8262), .A2(n8430), .ZN(n8256) );
  INVD0 U8484 ( .I(n8256), .ZN(n8164) );
  IND4D0 U8485 ( .A1(n8424), .B1(n8164), .B2(n8314), .B3(n8021)
     , .ZN(n8022) );
  OAI31D0 U8486 ( .A1(n8436), .A2(n8318), .A3(n8022), .B(n8316)
     , .ZN(n8023) );
  OAI211D0 U8487 ( .A1(n8025), .A2(n8441), .B(n8024), .C(n8023)
     , .ZN(n8036) );
  INVD0 U8488 ( .I(n8253), .ZN(n8364) );
  CKND2D0 U8489 ( .A1(n8259), .A2(n8132), .ZN(n8425) );
  NR2D0 U8490 ( .A1(n9344), .A2(n8115), .ZN(n8153) );
```

```
INVD0 U8491 ( .I(n8153), .ZN(n8090) );
CKND2D0 U8492 ( .A1(n8090), .A2(n8328), .ZN(n8376) );
NR4D0 U8493 ( .A1(n8364), .A2(n8425), .A3(n8376), .A4(n8026),
    .ZN(n8034) );
CKND2D0 U8494 ( .A1(n8287), .A2(n8154), .ZN(n8350) );
INVD0 U8495 ( .I(n8325), .ZN(n8367) );
NR2D0 U8496 ( .A1(n8175), .A2(n8027), .ZN(n8215) );
NR2D0 U8497 ( .A1(n8300), .A2(n8215), .ZN(n8239) );
CKND2D0 U8498 ( .A1(n8239), .A2(n8395), .ZN(n8363) );
INVD0 U8499 ( .I(n8363), .ZN(n8099) );
NR2D0 U8500 ( .A1(n8919), .A2(n8028), .ZN(n8388) );
INVD0 U8501 ( .I(n8430), .ZN(n8419) );
NR2D0 U8502 ( .A1(n8388), .A2(n8419), .ZN(n8268) );
ND4D0 U8503 ( .A1(n8367), .A2(n8099), .A3(n8268), .A4(n8394),
    .ZN(n8031) );
INVD0 U8504 ( .I(n8132), .ZN(n8092) );
NR2D0 U8505 ( .A1(n8092), .A2(n8221), .ZN(n8166) );
OAI211D0 U8506 ( .A1(n8278), .A2(n8919), .B(n8166), .C(n8213)
   , .ZN(n8097) );
NR2D0 U8507 ( .A1(n8201), .A2(n8097), .ZN(n8361) );
INVD0 U8508 ( .I(n8239), .ZN(n8404) );
INVD0 U8509 ( .I(n8208), .ZN(n8098) );
NR2D0 U8510 ( .A1(n8243), .A2(n8377), .ZN(n8421) );
INVD0 U8511 ( .I(n8421), .ZN(n8064) );
NR4D0 U8512 ( .A1(n8290), .A2(n8404), .A3(n8098), .A4(n8064),
    .ZN(n8029) );
CKND2D0 U8513 ( .A1(n8155), .A2(n8384), .ZN(n8312) );
AOI31D0 U8514 ( .A1(n8361), .A2(n8029), .A3(n8312), .B(n8612)
   , .ZN(n8030) );
AOI221D0 U8515 ( .A1(n8350), .A2(n8309), .B1(n8031), .B2(
   n8309), .C(n8030),
      .ZN(n8033) );
NR2D0 U8516 ( .A1(n8221), .A2(n8363), .ZN(n8332) );
INVD0 U8517 ( .I(n8257), .ZN(n8392) );
CKND2D0 U8518 ( .A1(n8359), .A2(n8392), .ZN(n8074) );
INR2D0 U8519 ( .A1(n8332), .B1(n8074), .ZN(n8306) );
INR2D0 U8520 ( .A1(n8356), .B1(n8376), .ZN(n8390) );
AO31D0 U8521 ( .A1(n8306), .A2(n8390), .A3(n8361), .B(n8416),
    .Z(n8032) );
OAI211D0 U8522 ( .A1(n8034), .A2(n8426), .B(n8033), .C(n8032)
   , .ZN(n8035) );
AOI22D0 U8523 ( .A1(n8037), .A2(n8036), .B1(n8740), .B2(n8035
   ), .ZN(n8455)
      );
CKND2D0 U8524 ( .A1(n8356), .A2(n8294), .ZN(n8438) );
NR2D0 U8525 ( .A1(n8438), .A2(n8351), .ZN(n8303) );
OR2D0 U8526 ( .A1(n8116), .A2(counter[4]), .Z(n8185) );
OR2D0 U8527 ( .A1(n8185), .A2(n8277), .Z(n8173) );
```

```
NR2D0 U8528 ( .A1(counter[4]), .A2(n8041), .ZN(n8133) );
NR2D0 U8529 ( .A1(n8935), .A2(n8038), .ZN(n8134) );
CKAN2D0 U8530 ( .A1(n8134), .A2(n8046), .Z(n8188) );
AOI211D0 U8531 ( .A1(n8130), .A2(n8133), .B(n8188), .C(n8300)
   , .ZN(n8197) );
NR2D0 U8532 ( .A1(n9348), .A2(n8116), .ZN(n8139) );
INVD0 U8533 ( .I(n8133), .ZN(n8118) );
NR2D0 U8534 ( .A1(n8142), .A2(n8118), .ZN(n8040) );
NR2D0 U8535 ( .A1(counter[5]), .A2(n8039), .ZN(n8178) );
AOI211D0 U8536 ( .A1(n9215), .A2(n8139), .B(n8040), .C(n8178)
   , .ZN(n8044) );
NR2D0 U8537 ( .A1(n9348), .A2(n8041), .ZN(n8194) );
OAI21D0 U8538 ( .A1(n8385), .A2(n8194), .B(n8042), .ZN(n8043)
    );
ND4D0 U8539 ( .A1(n8173), .A2(n8197), .A3(n8044), .A4(n8043),
    .ZN(n8147) );
OAI22D0 U8540 ( .A1(n8186), .A2(n8185), .B1(n9344), .B2(n8118
   ), .ZN(n8063)
       );
NR2D0 U8541 ( .A1(n8140), .A2(n8370), .ZN(n8242) );
CKND2D0 U8542 ( .A1(n8045), .A2(n8133), .ZN(n8047) );
CKND2D0 U8543 ( .A1(n8046), .A2(n8139), .ZN(n8058) );
IND4D0 U8544 ( .A1(n8063), .B1(n8242), .B2(n8047), .B3(n8058)
   , .ZN(n8191) );
AOI211D0 U8545 ( .A1(n9216), .A2(n8194), .B(n8215), .C(n8097)
   , .ZN(n8082) );
CKND2D0 U8546 ( .A1(n8078), .A2(n8194), .ZN(n8048) );
ND4D0 U8547 ( .A1(n8338), .A2(n8082), .A3(n8090), .A4(n8048),
    .ZN(n8049) );
INR4D0 U8548 ( .A1(n8303), .B1(n8147), .B2(n8191), .B3(n8049)
   , .ZN(n8088) );
INVD0 U8549 ( .I(n8222), .ZN(n8283) );
NR2D0 U8550 ( .A1(n8351), .A2(n8388), .ZN(n8407) );
CKND2D0 U8551 ( .A1(n8283), .A2(n8407), .ZN(n8334) );
INVD0 U8552 ( .I(n8334), .ZN(n8297) );
AOI21D0 U8553 ( .A1(n8186), .A2(n8050), .B(n8118), .ZN(n8051)
    );
AOI211D0 U8554 ( .A1(n8066), .A2(n8139), .B(n8256), .C(n8051)
   , .ZN(n8081) );
AOI22D0 U8555 ( .A1(n8053), .A2(n8052), .B1(n8130), .B2(n8697
   ), .ZN(n8107)
       );
OAI22D0 U8556 ( .A1(n8056), .A2(n8185), .B1(n8055), .B2(n8054
   ), .ZN(n8057)
       );
INR4D0 U8557 ( .A1(n8107), .B1(n8280), .B2(n8153), .B3(n8057)
   , .ZN(n8059) );
```

```
IND4D0 U8558 ( .A1(n8221), .B1(n8081), .B2(n8059), .B3(n8058)
   , .ZN(n8176) );
OAI22D0 U8559 ( .A1(n8186), .A2(n8919), .B1(n8175), .B2(n8118
   ), .ZN(n8060)
      );
NR4D0 U8560 ( .A1(n8075), .A2(n8188), .A3(n8176), .A4(n8060),
    .ZN(n8061) );
ND4D0 U8561 ( .A1(n8297), .A2(n8421), .A3(n8061), .A4(n8328),
    .ZN(n8073) );
CKND2D0 U8562 ( .A1(n9216), .A2(n8155), .ZN(n8401) );
CKND2D0 U8563 ( .A1(n8090), .A2(n8401), .ZN(n8335) );
NR2D0 U8564 ( .A1(n8062), .A2(n8382), .ZN(n8079) );
INVD0 U8565 ( .I(n8079), .ZN(n8414) );
NR4D0 U8566 ( .A1(n8335), .A2(n8414), .A3(n8064), .A4(n8063),
    .ZN(n8071) );
INVD0 U8567 ( .I(n8400), .ZN(n8224) );
INVD0 U8568 ( .I(n8194), .ZN(n8174) );
CKND2D0 U8569 ( .A1(n8130), .A2(n8139), .ZN(n8195) );
NR2D0 U8570 ( .A1(n8143), .A2(n8174), .ZN(n8065) );
INR4D0 U8571 ( .A1(n8195), .B1(n8178), .B2(n8065), .B3(n8349)
   , .ZN(n8123) );
AOI22D0 U8572 ( .A1(n8066), .A2(n8133), .B1(n9216), .B2(n8139
   ), .ZN(n8067)
      );
OAI211D0 U8573 ( .A1(n8068), .A2(n8174), .B(n8123), .C(n8067)
   , .ZN(n8084) );
NR4D0 U8574 ( .A1(n8069), .A2(n8388), .A3(n8224), .A4(n8084),
    .ZN(n8070) );
ND4D0 U8575 ( .A1(n8099), .A2(n8338), .A3(n8071), .A4(n8070),
    .ZN(n8072) );
AOI22D0 U8576 ( .A1(n8396), .A2(n8073), .B1(n8309), .B2(n8072
   ), .ZN(n8087)
      );
AO211D0 U8577 ( .A1(n8126), .A2(n8133), .B(n8075), .C(n8074),
    .Z(n8077) );
AOI211D0 U8578 ( .A1(n8078), .A2(n8139), .B(n8077), .C(n8076)
   , .ZN(n8122) );
INVD0 U8579 ( .I(n8122), .ZN(n8085) );
CKND2D0 U8580 ( .A1(n8079), .A2(n8431), .ZN(n8333) );
NR3D0 U8581 ( .A1(n8140), .A2(n8289), .A3(n8333), .ZN(n8080)
   );
ND4D0 U8582 ( .A1(n8082), .A2(n8081), .A3(n8080), .A4(n8295),
    .ZN(n8083) );
OAI31D0 U8583 ( .A1(n8085), .A2(n8084), .A3(n8083), .B(n8316)
   , .ZN(n8086) );
OAI211D0 U8584 ( .A1(n8088), .A2(n8426), .B(n8087), .C(n8086)
   , .ZN(n8106) );
INVD0 U8585 ( .I(n8154), .ZN(n8162) );
```

```
  NR2D0 U8586 ( .A1(n8117), .A2(n8115), .ZN(n8089) );
  CKND2D0 U8587 ( .A1(n8439), .A2(n8358), .ZN(n8434) );
  NR4D0 U8588 ( .A1(n8162), .A2(n8089), .A3(n8386), .A4(n8434),
      .ZN(n8103) );
  CKND2D0 U8589 ( .A1(n8090), .A2(n8294), .ZN(n8398) );
  NR2D0 U8590 ( .A1(n8177), .A2(n8398), .ZN(n8336) );
  NR3D0 U8591 ( .A1(n8420), .A2(n8415), .A3(n8091), .ZN(n8241)
     );
  ND4D0 U8592 ( .A1(n8336), .A2(n8166), .A3(n8241), .A4(n8421),
      .ZN(n8096) );
  CKND2D0 U8593 ( .A1(n8223), .A2(n8359), .ZN(n8354) );
  NR3D0 U8594 ( .A1(n8201), .A2(n8092), .A3(n8354), .ZN(n8254)
     );
  INVD0 U8595 ( .I(n8243), .ZN(n8329) );
  ND3D0 U8596 ( .A1(n8313), .A2(n8431), .A3(n8329), .ZN(n8305)
     );
  NR2D0 U8597 ( .A1(n8305), .A2(n8349), .ZN(n8160) );
  CKND2D0 U8598 ( .A1(n8338), .A2(n8213), .ZN(n8348) );
  AOI211D0 U8599 ( .A1(n8094), .A2(n8155), .B(n8093), .C(n8348)
     , .ZN(n8165) );
  INVD0 U8600 ( .I(n8140), .ZN(n8339) );
  ND4D0 U8601 ( .A1(n8254), .A2(n8160), .A3(n8165), .A4(n8339),
      .ZN(n8095) );
  AOI22D0 U8602 ( .A1(n8311), .A2(n8096), .B1(n8309), .B2(n8095
     ), .ZN(n8102)
         );
  INVD0 U8603 ( .I(n8097), .ZN(n8315) );
  NR2D0 U8604 ( .A1(n8335), .A2(n8098), .ZN(n8365) );
  ND4D0 U8605 ( .A1(n8099), .A2(n8315), .A3(n8164), .A4(n8365),
      .ZN(n8100) );
  OAI21D0 U8606 ( .A1(n8333), .A2(n8100), .B(n8396), .ZN(n8101)
      );
  OAI211D0 U8607 ( .A1(n8103), .A2(n8612), .B(n8102), .C(n8101)
     , .ZN(n8104) );
  AOI22D0 U8608 ( .A1(n8639), .A2(n8106), .B1(n8105), .B2(n8104
     ), .ZN(n8454)
         );
  OAI211D0 U8609 ( .A1(n8935), .A2(n8108), .B(n8407), .C(n8107)
     , .ZN(n8113) );
  INVD0 U8610 ( .I(n8218), .ZN(n8412) );
  CKND2D0 U8611 ( .A1(n8393), .A2(n8295), .ZN(n8109) );
  NR2D0 U8612 ( .A1(n8289), .A2(n8109), .ZN(n8272) );
  AOI211D0 U8613 ( .A1(n8110), .A2(n8899), .B(n8383), .C(n8191)
     , .ZN(n8111) );
  ND4D0 U8614 ( .A1(n8272), .A2(n8111), .A3(n8394), .A4(n8329),
      .ZN(n8112) );
  OAI31D0 U8615 ( .A1(n8113), .A2(n8412), .A3(n8112), .B(n8311)
     , .ZN(n8152) );
```

```
OAI211D0 U8616 ( .A1(n8117), .A2(n8115), .B(n8114), .C(n8314)
   , .ZN(n8125) );
AOI21D0 U8617 ( .A1(n8141), .A2(n8139), .B(n8435), .ZN(n8135)
    );
AOI211D0 U8618 ( .A1(n8117), .A2(n9189), .B(counter[4]), .C(
   n8116), .ZN(
      n8120) );
NR2D0 U8619 ( .A1(n8277), .A2(n8118), .ZN(n8119) );
AOI211D0 U8620 ( .A1(n8139), .A2(n8126), .B(n8120), .C(n8119)
   , .ZN(n8121) );
ND4D0 U8621 ( .A1(n8123), .A2(n8122), .A3(n8135), .A4(n8121),
    .ZN(n8124) );
OAI31D0 U8622 ( .A1(n8201), .A2(n8125), .A3(n8124), .B(n8309)
   , .ZN(n8151) );
CKND2D0 U8623 ( .A1(n8141), .A2(n8133), .ZN(n8128) );
CKND2D0 U8624 ( .A1(n8126), .A2(n8194), .ZN(n8127) );
ND4D0 U8625 ( .A1(n8129), .A2(n8393), .A3(n8128), .A4(n8127),
    .ZN(n8138) );
OAI22D0 U8626 ( .A1(n8951), .A2(n8130), .B1(n8134), .B2(n8194
   ), .ZN(n8131)
      );
ND3D0 U8627 ( .A1(n8268), .A2(n8132), .A3(n8131), .ZN(n8199)
   );
AOI22D0 U8628 ( .A1(n8156), .A2(n8134), .B1(n9216), .B2(n8133
   ), .ZN(n8181)
      );
IND4D0 U8629 ( .A1(n8434), .B1(n8136), .B2(n8135), .B3(n8181)
   , .ZN(n8137) );
OAI31D0 U8630 ( .A1(n8138), .A2(n8199), .A3(n8137), .B(n8316)
   , .ZN(n8150) );
INVD0 U8631 ( .I(n8165), .ZN(n8148) );
NR2D0 U8632 ( .A1(n8290), .A2(n8364), .ZN(n8366) );
AOI22D0 U8633 ( .A1(n8183), .A2(n8139), .B1(n9216), .B2(n8899
   ), .ZN(n8145)
      );
AOI21D0 U8634 ( .A1(n8141), .A2(n8194), .B(n8140), .ZN(n8180)
    );
OA221D0 U8635 ( .A1(n8143), .A2(n8174), .B1(n8142), .B2(n8174
   ), .C(n8180),
      .Z(n8144) );
ND4D0 U8636 ( .A1(n8366), .A2(n8268), .A3(n8145), .A4(n8144),
    .ZN(n8146) );
OAI31D0 U8637 ( .A1(n8148), .A2(n8147), .A3(n8146), .B(n8396)
   , .ZN(n8149) );
ND4D0 U8638 ( .A1(n8152), .A2(n8151), .A3(n8150), .A4(n8149),
    .ZN(n8235) );
NR3D0 U8639 ( .A1(n8425), .A2(n8351), .A3(n8415), .ZN(n8245)
   );
```

```
INVD0 U8640 ( .I(n8245), .ZN(n8326) );
NR2D0 U8641 ( .A1(n8326), .A2(n8412), .ZN(n8207) );
NR2D0 U8642 ( .A1(n8153), .A2(n8377), .ZN(n8211) );
ND4D0 U8643 ( .A1(n8207), .A2(n8211), .A3(n8154), .A4(n8359),
    .ZN(n8159) );
AOI21D0 U8644 ( .A1(n8156), .A2(n8155), .B(n8304), .ZN(n8296)
    );
ND4D0 U8645 ( .A1(n8296), .A2(n8303), .A3(n8268), .A4(n8157),
    .ZN(n8158) );
AOI22D0 U8646 ( .A1(n8396), .A2(n8159), .B1(n8316), .B2(n8158
   ), .ZN(n8172)
       );
INVD0 U8647 ( .I(n8335), .ZN(n8167) );
CKND2D0 U8648 ( .A1(n8167), .A2(n8160), .ZN(n8249) );
INVD0 U8649 ( .I(n8215), .ZN(n8427) );
IND4D0 U8650 ( .A1(n8249), .B1(n8315), .B2(n8394), .B3(n8427)
   , .ZN(n8161) );
OAI31D0 U8651 ( .A1(n8415), .A2(n8162), .A3(n8161), .B(n8309)
   , .ZN(n8171) );
CKND2D0 U8652 ( .A1(n8164), .A2(n8163), .ZN(n8255) );
ND4D0 U8653 ( .A1(n8167), .A2(n8259), .A3(n8166), .A4(n8165),
    .ZN(n8168) );
OAI21D0 U8654 ( .A1(n8255), .A2(n8168), .B(n8311), .ZN(n8170)
    );
AOI31D0 U8655 ( .A1(n8172), .A2(n8171), .A3(n8170), .B(n8169)
   , .ZN(n8234) );
OAI211D0 U8656 ( .A1(n8175), .A2(n8174), .B(n8173), .C(n8212)
   , .ZN(n8200) );
INR4D0 U8657 ( .A1(n8294), .B1(n8178), .B2(n8177), .B3(n8176)
   , .ZN(n8179) );
IND4D0 U8658 ( .A1(n8200), .B1(n8181), .B2(n8180), .B3(n8179)
   , .ZN(n8206) );
AOI21D0 U8659 ( .A1(n8184), .A2(n8183), .B(n8182), .ZN(n8302)
    );
NR2D0 U8660 ( .A1(n8186), .A2(n8185), .ZN(n8187) );
INR4D0 U8661 ( .A1(n8361), .B1(n8188), .B2(n8187), .B3(n8261)
   , .ZN(n8190) );
AOI31D0 U8662 ( .A1(n8302), .A2(n8190), .A3(n8189), .B(n8416)
   , .ZN(n8205) );
AOI211D0 U8663 ( .A1(n8194), .A2(n8193), .B(n8192), .C(n8191)
   , .ZN(n8196) );
INVD0 U8664 ( .I(n8201), .ZN(n8240) );
ND4D0 U8665 ( .A1(n8197), .A2(n8196), .A3(n8240), .A4(n8195),
    .ZN(n8198) );
OAI31D0 U8666 ( .A1(n8200), .A2(n8199), .A3(n8198), .B(n8316)
   , .ZN(n8203) );
NR2D0 U8667 ( .A1(n8265), .A2(n8201), .ZN(n8380) );
INVD0 U8668 ( .I(n8380), .ZN(n8327) );
```

```
INR4D0 U8669 ( .A1(n8395), .B1(n8388), .B2(n8333), .B3(n8327)
   , .ZN(n8202) );
AOI32D0 U8670 ( .A1(n8241), .A2(n8203), .A3(n8202), .B1(n8441
   ), .B2(n8203),
      .ZN(n8204) );
AOI211D0 U8671 ( .A1(n8311), .A2(n8206), .B(n8205), .C(n8204)
   , .ZN(n8232) );
OAI211D0 U8672 ( .A1(n8210), .A2(n8209), .B(n8208), .C(n8207)
   , .ZN(n8229) );
CKND2D0 U8673 ( .A1(n8332), .A2(n8303), .ZN(n8433) );
ND4D0 U8674 ( .A1(n8236), .A2(n8212), .A3(n8211), .A4(n8295),
    .ZN(n8220) );
CKND2D0 U8675 ( .A1(n8214), .A2(n8213), .ZN(n8369) );
AOI211D0 U8676 ( .A1(n8216), .A2(n8609), .B(n8215), .C(n8369)
   , .ZN(n8217) );
AOI31D0 U8677 ( .A1(n8360), .A2(n8218), .A3(n8217), .B(n8426)
   , .ZN(n8219) );
AOI221D0 U8678 ( .A1(n8433), .A2(n8309), .B1(n8220), .B2(
   n8309), .C(n8219),
      .ZN(n8227) );
NR2D0 U8679 ( .A1(n8222), .A2(n8221), .ZN(n8381) );
ND4D0 U8680 ( .A1(n8381), .A2(n8393), .A3(n8223), .A4(n8295),
    .ZN(n8248) );
NR4D0 U8681 ( .A1(n8225), .A2(n8224), .A3(n8376), .A4(n8248),
    .ZN(n8226) );
AOI32D0 U8682 ( .A1(n8303), .A2(n8227), .A3(n8226), .B1(n8612
   ), .B2(n8227),
      .ZN(n8228) );
AOI221D0 U8683 ( .A1(n8300), .A2(n8396), .B1(n8229), .B2(
   n8396), .C(n8228),
      .ZN(n8231) );
OAI22D0 U8684 ( .A1(n8232), .A2(n8797), .B1(n8231), .B2(n8230
   ), .ZN(n8233)
       );
AOI211D0 U8685 ( .A1(n8918), .A2(n8235), .B(n8234), .C(n8233)
   , .ZN(n8453) );
CKND2D0 U8686 ( .A1(n8394), .A2(n8401), .ZN(n8238) );
ND3D0 U8687 ( .A1(n8239), .A2(n8329), .A3(n8295), .ZN(n8413)
   );
CKND2D0 U8688 ( .A1(n8236), .A2(n8307), .ZN(n8399) );
NR4D0 U8689 ( .A1(n8238), .A2(n8413), .A3(n8399), .A4(n8237),
    .ZN(n8252) );
CKND2D0 U8690 ( .A1(n8239), .A2(n8287), .ZN(n8273) );
ND4D0 U8691 ( .A1(n8366), .A2(n8241), .A3(n8240), .A4(n8393),
    .ZN(n8247) );
INVD0 U8692 ( .I(n8242), .ZN(n8402) );
NR3D0 U8693 ( .A1(n8355), .A2(n8243), .A3(n8402), .ZN(n8244)
   );
```

```
AOI31D0 U8694 ( .A1(n8272), .A2(n8245), .A3(n8244), .B(n8612)
   , .ZN(n8246) );
AOI221D0 U8695 ( .A1(n8273), .A2(n8396), .B1(n8247), .B2(
   n8396), .C(n8246),
      .ZN(n8251) );
OAI31D0 U8696 ( .A1(n8364), .A2(n8249), .A3(n8248), .B(n8309)
   , .ZN(n8250) );
OAI211D0 U8697 ( .A1(n8252), .A2(n8426), .B(n8251), .C(n8250)
   , .ZN(n8451) );
NR2D0 U8698 ( .A1(n8305), .A2(n8435), .ZN(n8281) );
IND4D0 U8699 ( .A1(n8255), .B1(n8281), .B2(n8254), .B3(n8253)
   , .ZN(n8271) );
CKND2D0 U8700 ( .A1(n8431), .A2(n8439), .ZN(n8301) );
NR4D0 U8701 ( .A1(n8382), .A2(n8257), .A3(n8301), .A4(n8256),
    .ZN(n8258) );
AOI31D0 U8702 ( .A1(n8332), .A2(n8259), .A3(n8258), .B(n8426)
   , .ZN(n8270) );
ND4D0 U8703 ( .A1(n8439), .A2(n8393), .A3(n8339), .A4(n8392),
    .ZN(n8260) );
OAI31D0 U8704 ( .A1(n8261), .A2(n8318), .A3(n8260), .B(n8309)
   , .ZN(n8267) );
CKND2D0 U8705 ( .A1(n8263), .A2(n8262), .ZN(n8264) );
INR4D0 U8706 ( .A1(n8336), .B1(n8265), .B2(n8354), .B3(n8264)
   , .ZN(n8266) );
AOI32D0 U8707 ( .A1(n8268), .A2(n8267), .A3(n8266), .B1(n8416
   ), .B2(n8267),
      .ZN(n8269) );
AOI211D0 U8708 ( .A1(n8316), .A2(n8271), .B(n8270), .C(n8269)
   , .ZN(n8347) );
INVD0 U8709 ( .I(n8272), .ZN(n8403) );
NR4D0 U8710 ( .A1(n8403), .A2(n8387), .A3(n8274), .A4(n8273),
    .ZN(n8293) );
OAI211D0 U8711 ( .A1(n8277), .A2(n8276), .B(n8275), .C(n8613)
   , .ZN(n8286) );
NR2D0 U8712 ( .A1(n8919), .A2(n8278), .ZN(n8279) );
INR4D0 U8713 ( .A1(n8281), .B1(n8280), .B2(n8290), .B3(n8279)
   , .ZN(n8284) );
AOI31D0 U8714 ( .A1(n8284), .A2(n8283), .A3(n8282), .B(n8426)
   , .ZN(n8285) );
AOI221D0 U8715 ( .A1(n8398), .A2(n8396), .B1(n8286), .B2(
   n8396), .C(n8285),
      .ZN(n8292) );
ND4D0 U8716 ( .A1(n8307), .A2(n8287), .A3(n8394), .A4(n8401),
    .ZN(n8288) );
OAI31D0 U8717 ( .A1(n8290), .A2(n8289), .A3(n8288), .B(n8316)
   , .ZN(n8291) );
OAI211D0 U8718 ( .A1(n8293), .A2(n8441), .B(n8292), .C(n8291)
   , .ZN(n8324) );
```

```
ND4D0 U8719 ( .A1(n8297), .A2(n8296), .A3(n8295), .A4(n8294),
    .ZN(n8298) );
NR4D0 U8720 ( .A1(n8415), .A2(n8300), .A3(n8299), .A4(n8298),
    .ZN(n8321) );
INVD0 U8721 ( .I(n8301), .ZN(n8429) );
IND4D0 U8722 ( .A1(n8350), .B1(n8303), .B2(n8302), .B3(n8429)
   , .ZN(n8310) );
NR3D0 U8723 ( .A1(n8419), .A2(n8305), .A3(n8304), .ZN(n8337)
   );
ND4D0 U8724 ( .A1(n8366), .A2(n8337), .A3(n8307), .A4(n8306),
    .ZN(n8308) );
AOI22D0 U8725 ( .A1(n8311), .A2(n8310), .B1(n8309), .B2(n8308
   ), .ZN(n8320)
       );
ND4D0 U8726 ( .A1(n8315), .A2(n8314), .A3(n8313), .A4(n8312),
    .ZN(n8317) );
OAI31D0 U8727 ( .A1(n8318), .A2(n8387), .A3(n8317), .B(n8316)
   , .ZN(n8319) );
OAI211D0 U8728 ( .A1(n8321), .A2(n8416), .B(n8320), .C(n8319)
   , .ZN(n8322) );
AOI22D0 U8729 ( .A1(n8939), .A2(n8324), .B1(n8323), .B2(n8322
   ), .ZN(n8346)
       );
NR4D0 U8730 ( .A1(n8327), .A2(n8326), .A3(n8325), .A4(n8354),
    .ZN(n8330) );
AOI31D0 U8731 ( .A1(n8330), .A2(n8329), .A3(n8328), .B(n8612)
   , .ZN(n8344) );
INR4D0 U8732 ( .A1(n8390), .B1(n8420), .B2(n8378), .B3(n8419)
   , .ZN(n8331) );
AOI31D0 U8733 ( .A1(n8366), .A2(n8332), .A3(n8331), .B(n8441)
   , .ZN(n8343) );
NR4D0 U8734 ( .A1(n8378), .A2(n8335), .A3(n8334), .A4(n8333),
    .ZN(n8341) );
AN4D0 U8735 ( .A1(n8339), .A2(n8338), .A3(n8337), .A4(n8336),
    .Z(n8340) );
OAI22D0 U8736 ( .A1(n8341), .A2(n8416), .B1(n8340), .B2(n8426
   ), .ZN(n8342)
       );
OAI31D0 U8737 ( .A1(n8344), .A2(n8343), .A3(n8342), .B(n8715)
   , .ZN(n8345) );
OAI211D0 U8738 ( .A1(n8347), .A2(n8848), .B(n8346), .C(n8345)
   , .ZN(n8450) );
INVD0 U8739 ( .I(n8348), .ZN(n8418) );
NR4D0 U8740 ( .A1(n8351), .A2(n8382), .A3(n8350), .A4(n8349),
    .ZN(n8352) );
ND4D0 U8741 ( .A1(n8353), .A2(n8381), .A3(n8418), .A4(n8352),
    .ZN(n8375) );
```

```
INR4D0 U8742 ( .A1(n8401), .B1(n8355), .B2(n8425), .B3(n8354)
   , .ZN(n8357) );
AOI21D0 U8743 ( .A1(n8357), .A2(n8356), .B(n8441), .ZN(n8374)
    );
ND4D0 U8744 ( .A1(n8361), .A2(n8360), .A3(n8359), .A4(n8358),
    .ZN(n8362) );
NR4D0 U8745 ( .A1(n8420), .A2(n8364), .A3(n8363), .A4(n8362),
    .ZN(n8372) );
ND4D0 U8746 ( .A1(n8407), .A2(n8367), .A3(n8366), .A4(n8365),
    .ZN(n8368) );
NR4D0 U8747 ( .A1(n8383), .A2(n8370), .A3(n8369), .A4(n8368),
    .ZN(n8371) );
OAI22D0 U8748 ( .A1(n8372), .A2(n8426), .B1(n8371), .B2(n8612
   ), .ZN(n8373)
      );
AOI211D0 U8749 ( .A1(n8396), .A2(n8375), .B(n8374), .C(n8373)
   , .ZN(n8448) );
NR4D0 U8750 ( .A1(n8378), .A2(n8415), .A3(n8377), .A4(n8376),
    .ZN(n8379) );
AOI31D0 U8751 ( .A1(n8381), .A2(n8380), .A3(n8379), .B(n8426)
   , .ZN(n8411) );
AOI211D0 U8752 ( .A1(n8385), .A2(n8384), .B(n8383), .C(n8382)
   , .ZN(n8391) );
NR4D0 U8753 ( .A1(n8388), .A2(n8404), .A3(n8387), .A4(n8386),
    .ZN(n8389) );
AOI31D0 U8754 ( .A1(n8391), .A2(n8390), .A3(n8389), .B(n8612)
   , .ZN(n8410) );
ND4D0 U8755 ( .A1(n8395), .A2(n8394), .A3(n8393), .A4(n8392),
    .ZN(n8397) );
OAI31D0 U8756 ( .A1(n8399), .A2(n8398), .A3(n8397), .B(n8396)
   , .ZN(n8406) );
CKND2D0 U8757 ( .A1(n8401), .A2(n8400), .ZN(n8423) );
NR4D0 U8758 ( .A1(n8404), .A2(n8403), .A3(n8402), .A4(n8423),
    .ZN(n8405) );
AOI32D0 U8759 ( .A1(n8407), .A2(n8406), .A3(n8405), .B1(n8441
   ), .B2(n8406),
     .ZN(n8409) );
OAI31D0 U8760 ( .A1(n8411), .A2(n8410), .A3(n8409), .B(n8408)
   , .ZN(n8447) );
NR4D0 U8761 ( .A1(n8415), .A2(n8414), .A3(n8413), .A4(n8412),
    .ZN(n8417) );
AOI21D0 U8762 ( .A1(n8418), .A2(n8417), .B(n8416), .ZN(n8445)
    );
NR2D0 U8763 ( .A1(n8420), .A2(n8419), .ZN(n8422) );
CKND2D0 U8764 ( .A1(n8422), .A2(n8421), .ZN(n8437) );
NR4D0 U8765 ( .A1(n8425), .A2(n8424), .A3(n8437), .A4(n8423),
    .ZN(n8428) );
```

```
  AOI31D0 U8766 ( .A1(n8429), .A2(n8428), .A3(n8427), .B(n8426)
     , .ZN(n8444) );
  CKND2D0 U8767 ( .A1(n8431), .A2(n8430), .ZN(n8432) );
  NR4D0 U8768 ( .A1(n8435), .A2(n8434), .A3(n8433), .A4(n8432),
      .ZN(n8442) );
  INR4D0 U8769 ( .A1(n8439), .B1(n8438), .B2(n8437), .B3(n8436)
     , .ZN(n8440) );
  OAI22D0 U8770 ( .A1(n8442), .A2(n8441), .B1(n8440), .B2(n8612
     ), .ZN(n8443)
        );
  OAI31D0 U8771 ( .A1(n8445), .A2(n8444), .A3(n8443), .B(n8656)
     , .ZN(n8446) );
  OAI211D0 U8772 ( .A1(n8448), .A2(n8662), .B(n8447), .C(n8446)
     , .ZN(n8449) );
  AOI211D0 U8773 ( .A1(n8895), .A2(n8451), .B(n8450), .C(n8449)
     , .ZN(n8452) );
  ND4D0 U8774 ( .A1(n8455), .A2(n8454), .A3(n8453), .A4(n8452),
      .ZN(n8456) );
  AO222D0 U8775 ( .A1(n8457), .A2(n8949), .B1(n8456), .B2(n8466
     ), .C1(q_out[2]), .C2(n8947), .Z(n4814) );
  NR2D0 U8776 ( .A1(counter[9]), .A2(n8965), .ZN(n9009) );
  CKND2D0 U8777 ( .A1(n9173), .A2(n8964), .ZN(n8972) );
  INVD0 U8778 ( .I(n8972), .ZN(n9214) );
  NR2D0 U8779 ( .A1(n9009), .A2(n9214), .ZN(n9046) );
  CKND2D0 U8780 ( .A1(n9022), .A2(n9014), .ZN(n9163) );
  INVD0 U8781 ( .I(n9163), .ZN(n9043) );
  NR2D0 U8782 ( .A1(n9028), .A2(n9110), .ZN(n9050) );
  CKND2D0 U8783 ( .A1(n9022), .A2(n8461), .ZN(n9104) );
  CKND2D0 U8784 ( .A1(n9295), .A2(n9172), .ZN(n8458) );
  CKND2D0 U8785 ( .A1(n9104), .A2(n8458), .ZN(n9188) );
  CKND2D0 U8786 ( .A1(n8462), .A2(n8971), .ZN(n8975) );
  NR2D0 U8787 ( .A1(n8966), .A2(counter[9]), .ZN(n8483) );
  INVD0 U8788 ( .I(n8483), .ZN(n9174) );
  CKND2D0 U8789 ( .A1(n8975), .A2(n9174), .ZN(n9095) );
  NR2D0 U8790 ( .A1(n9210), .A2(n8954), .ZN(n9127) );
  INVD0 U8791 ( .I(n9127), .ZN(n9199) );
  CKND2D0 U8792 ( .A1(n9295), .A2(n9014), .ZN(n9289) );
  CKND2D0 U8793 ( .A1(n9199), .A2(n9289), .ZN(n9105) );
  CKND2D0 U8794 ( .A1(n8462), .A2(n8964), .ZN(n9198) );
  INVD0 U8795 ( .I(n9198), .ZN(n9229) );
  CKND2D0 U8796 ( .A1(n8462), .A2(n8979), .ZN(n9310) );
  INVD0 U8797 ( .I(n9310), .ZN(n9249) );
  NR2D0 U8798 ( .A1(n9028), .A2(n9178), .ZN(n9254) );
  NR2D0 U8799 ( .A1(counter[9]), .A2(n9004), .ZN(n9221) );
  NR4D0 U8800 ( .A1(n9229), .A2(n9249), .A3(n9254), .A4(n9221),
      .ZN(n9196) );
  OAI21D0 U8801 ( .A1(n9025), .A2(n8997), .B(n9295), .ZN(n8459)
      );
```

```
  CKND2D0 U8802 ( .A1(n9022), .A2(n9129), .ZN(n8996) );
  CKND2D0 U8803 ( .A1(n9173), .A2(n9014), .ZN(n8999) );
  ND4D0 U8804 ( .A1(n9196), .A2(n8459), .A3(n8996), .A4(n8999),
      .ZN(n8460) );
  NR4D0 U8805 ( .A1(n9188), .A2(n9095), .A3(n9105), .A4(n8460),
      .ZN(n8469) );
  CKND2D0 U8806 ( .A1(n8462), .A2(n9129), .ZN(n9267) );
  NR2D0 U8807 ( .A1(n9111), .A2(n8982), .ZN(n8958) );
  INVD0 U8808 ( .I(n8958), .ZN(n9279) );
  IND4D0 U8809 ( .A1(n9050), .B1(n8469), .B2(n9267), .B3(n9279)
     , .ZN(n8464) );
  CKND2D0 U8810 ( .A1(n9295), .A2(n8971), .ZN(n9245) );
  INVD0 U8811 ( .I(n9245), .ZN(n9114) );
  CKND2D0 U8812 ( .A1(n9173), .A2(n8997), .ZN(n9206) );
  INVD0 U8813 ( .I(n9206), .ZN(n8493) );
  NR2D0 U8814 ( .A1(n9114), .A2(n8493), .ZN(n9138) );
  CKND2D0 U8815 ( .A1(n9295), .A2(n9129), .ZN(n8956) );
  NR2D0 U8816 ( .A1(n9028), .A2(n9048), .ZN(n9096) );
  INVD0 U8817 ( .I(n9096), .ZN(n9003) );
  CKND2D0 U8818 ( .A1(n8462), .A2(n9172), .ZN(n9099) );
  ND4D0 U8819 ( .A1(n9138), .A2(n8956), .A3(n9003), .A4(n9099),
      .ZN(n9072) );
  NR3D0 U8820 ( .A1(n9043), .A2(n8464), .A3(n9072), .ZN(n8490)
     );
  CKND2D0 U8821 ( .A1(n9173), .A2(n8979), .ZN(n9278) );
  AOI31D0 U8822 ( .A1(n9046), .A2(n8490), .A3(n9278), .B(n9272)
     , .ZN(n8468) );
  CKND2D0 U8823 ( .A1(n8462), .A2(n8461), .ZN(n9047) );
  CKND2D0 U8824 ( .A1(n9245), .A2(n9047), .ZN(n9284) );
  NR2D0 U8825 ( .A1(n9284), .A2(n9009), .ZN(n9185) );
  NR2D0 U8826 ( .A1(n9210), .A2(n9048), .ZN(n9077) );
  INVD0 U8827 ( .I(n9077), .ZN(n8475) );
  NR2D0 U8828 ( .A1(n8955), .A2(n8463), .ZN(n8963) );
  INVD0 U8829 ( .I(n8963), .ZN(n8977) );
  CKND2D0 U8830 ( .A1(n8475), .A2(n8977), .ZN(n9038) );
  IINR4D0 U8831 ( .A1(n9185), .A2(n8956), .B1(n8464), .B2(n9038
     ), .ZN(n8465)
       );
  INVD0 U8832 ( .I(n9188), .ZN(n9200) );
  OAI211D0 U8833 ( .A1(n8465), .A2(n9318), .B(n9200), .C(
     counter[8]), .ZN(
        n8467) );
  OAI211D0 U8834 ( .A1(n8468), .A2(n8467), .B(n8466), .C(
     counter[11]), .ZN(
        n9333) );
  OAI211D0 U8835 ( .A1(n9029), .A2(n9028), .B(n8469), .C(n8977)
     , .ZN(n8495) );
```

```
NR4D0 U8836 ( .A1(n9050), .A2(n9043), .A3(n9284), .A4(n8495),
    .ZN(n8470) );
AOI31D0 U8837 ( .A1(n8470), .A2(n8475), .A3(n9003), .B(n9274)
   , .ZN(n8502) );
CKND2D0 U8838 ( .A1(n9295), .A2(n9090), .ZN(n9033) );
CKND2D0 U8839 ( .A1(n8972), .A2(n9033), .ZN(n8994) );
OAI21D0 U8840 ( .A1(counter[1]), .A2(n8962), .B(n9185), .ZN(
   n8471) );
AOI211D0 U8841 ( .A1(n8968), .A2(n9173), .B(n8994), .C(n8471)
   , .ZN(n8477) );
AOI21D0 U8842 ( .A1(n8962), .A2(n8472), .B(n9210), .ZN(n8473)
    );
AOI211D0 U8843 ( .A1(n9173), .A2(n9294), .B(n8963), .C(n8473)
   , .ZN(n8478) );
ND4D0 U8844 ( .A1(n8477), .A2(n8478), .A3(n9310), .A4(n8956),
    .ZN(n8480) );
NR4D0 U8845 ( .A1(n9221), .A2(n9127), .A3(n8493), .A4(n8480),
    .ZN(n8487) );
OAI21D0 U8846 ( .A1(n8964), .A2(n9014), .B(n9295), .ZN(n8474)
    );
ND3D0 U8847 ( .A1(n9110), .A2(n8475), .A3(n8474), .ZN(n8986)
   );
NR2D0 U8848 ( .A1(n8483), .A2(n8986), .ZN(n8476) );
AOI21D0 U8849 ( .A1(n8487), .A2(n8476), .B(n9318), .ZN(n8501)
    );
CKND2D0 U8850 ( .A1(n9295), .A2(n8997), .ZN(n9119) );
CKND2D0 U8851 ( .A1(n9119), .A2(n9279), .ZN(n8482) );
NR2D0 U8852 ( .A1(n9096), .A2(n8482), .ZN(n8488) );
AN3D0 U8853 ( .A1(n8488), .A2(n8477), .A3(n8476), .Z(n8489) )
   ;
AN4D0 U8854 ( .A1(n9163), .A2(n9206), .A3(n8478), .A4(n8489),
    .Z(n8486) );
NR2D0 U8855 ( .A1(n9077), .A2(n9096), .ZN(n8479) );
IND4D0 U8856 ( .A1(n8480), .B1(n9200), .B2(n8479), .B3(n9289)
   , .ZN(n8481) );
OAI31D0 U8857 ( .A1(n8483), .A2(n8482), .A3(n8481), .B(n9307)
   , .ZN(n8484) );
OAI211D0 U8858 ( .A1(n8486), .A2(n9300), .B(n8485), .C(n8484)
   , .ZN(n8500) );
NR2D0 U8859 ( .A1(n8955), .A2(n8982), .ZN(n9073) );
NR2D0 U8860 ( .A1(n9043), .A2(n9073), .ZN(n8961) );
AN4D0 U8861 ( .A1(n9104), .A2(n8488), .A3(n8487), .A4(n8961),
    .Z(n8498) );
ND4D0 U8862 ( .A1(n9196), .A2(n8489), .A3(n8956), .A4(n9199),
    .ZN(n8492) );
NR2D0 U8863 ( .A1(n9028), .A2(n8962), .ZN(n8987) );
INVD0 U8864 ( .I(n8987), .ZN(n9157) );
```

```
AOI31D0 U8865 ( .A1(n8490), .A2(n8977), .A3(n9157), .B(n9300)
   , .ZN(n8491) );
AOI211D0 U8866 ( .A1(n9226), .A2(n8492), .B(n8491), .C(n9153)
   , .ZN(n8497) );
NR2D0 U8867 ( .A1(n9077), .A2(n8493), .ZN(n9258) );
CKND2D0 U8868 ( .A1(n9258), .A2(n9267), .ZN(n9088) );
NR2D0 U8869 ( .A1(n9210), .A2(n9311), .ZN(n9113) );
IND4D0 U8870 ( .A1(n9113), .B1(n9046), .B2(n9279), .B3(n8956)
   , .ZN(n8494) );
OAI31D0 U8871 ( .A1(n9088), .A2(n8495), .A3(n8494), .B(n9312)
   , .ZN(n8496) );
OAI211D0 U8872 ( .A1(n8498), .A2(n9272), .B(n8497), .C(n8496)
   , .ZN(n8499) );
OAI31D0 U8873 ( .A1(n8502), .A2(n8501), .A3(n8500), .B(n8499)
   , .ZN(n9332) );
NR2D0 U8874 ( .A1(n8503), .A2(n8620), .ZN(n8719) );
INVD0 U8875 ( .I(n8719), .ZN(n8929) );
NR2D0 U8876 ( .A1(n8504), .A2(n8929), .ZN(n8892) );
NR2D0 U8877 ( .A1(n8551), .A2(n8505), .ZN(n8793) );
OR2D0 U8878 ( .A1(n5555), .A2(n8507), .Z(n8824) );
NR3D0 U8879 ( .A1(n8824), .A2(n8860), .A3(n8508), .ZN(n8873)
   );
NR2D0 U8880 ( .A1(n8553), .A2(n8876), .ZN(n8872) );
CKND2D0 U8881 ( .A1(n8844), .A2(n8872), .ZN(n8681) );
NR2D0 U8882 ( .A1(n8681), .A2(n8754), .ZN(n8738) );
INVD0 U8883 ( .I(n8738), .ZN(n8604) );
NR2D0 U8884 ( .A1(n8531), .A2(n8604), .ZN(n8677) );
INVD0 U8885 ( .I(n8677), .ZN(n8889) );
NR2D0 U8886 ( .A1(n8889), .A2(n8735), .ZN(n8861) );
CKND2D0 U8887 ( .A1(n8873), .A2(n8861), .ZN(n8807) );
NR2D0 U8888 ( .A1(n8807), .A2(n8672), .ZN(n8595) );
INVD0 U8889 ( .I(n8595), .ZN(n8686) );
NR2D0 U8890 ( .A1(n8808), .A2(n8686), .ZN(n8926) );
NR2D0 U8891 ( .A1(n8878), .A2(n8733), .ZN(n8594) );
ND4D0 U8892 ( .A1(n8892), .A2(n8793), .A3(n8926), .A4(n8594),
    .ZN(n8528) );
NR2D0 U8893 ( .A1(n4819), .A2(n8509), .ZN(n8834) );
INVD0 U8894 ( .I(n8834), .ZN(n8563) );
NR2D0 U8895 ( .A1(n8808), .A2(n8510), .ZN(n8862) );
ND4D0 U8896 ( .A1(n8511), .A2(n8862), .A3(n8730), .A4(n8883),
    .ZN(n8930) );
INVD0 U8897 ( .I(n8930), .ZN(n8645) );
CKND2D0 U8898 ( .A1(n8512), .A2(n8645), .ZN(n8746) );
IND2D0 U8899 ( .A1(n8746), .B1(n8721), .ZN(n8707) );
NR2D0 U8900 ( .A1(n8563), .A2(n8707), .ZN(n8665) );
CKND2D0 U8901 ( .A1(n8513), .A2(n8719), .ZN(n8875) );
NR2D0 U8902 ( .A1(n8643), .A2(n8748), .ZN(n8549) );
INVD0 U8903 ( .I(n8549), .ZN(n8934) );
```

```
NR3D0 U8904 ( .A1(n8875), .A2(n8681), .A3(n8934), .ZN(n8693)
   );
AOI31D0 U8905 ( .A1(n8665), .A2(n8693), .A3(n8514), .B(n8935)
   , .ZN(n8527) );
NR2D0 U8906 ( .A1(n8516), .A2(n8515), .ZN(n8880) );
INVD0 U8907 ( .I(n8880), .ZN(n8671) );
NR2D0 U8908 ( .A1(n8643), .A2(n8671), .ZN(n8835) );
CKND2D0 U8909 ( .A1(n8517), .A2(n8677), .ZN(n8796) );
INVD0 U8910 ( .I(n8796), .ZN(n8775) );
NR2D0 U8911 ( .A1(n8563), .A2(n8875), .ZN(n8602) );
ND3D0 U8912 ( .A1(n8519), .A2(n8518), .A3(n8602), .ZN(n8599)
   );
NR2D0 U8913 ( .A1(n8682), .A2(n8599), .ZN(n8838) );
AO31D0 U8914 ( .A1(n8835), .A2(n8775), .A3(n8838), .B(n8937),
    .Z(n8525) );
INVD0 U8915 ( .I(n8653), .ZN(n8810) );
ND4D0 U8916 ( .A1(n8521), .A2(n8520), .A3(n8884), .A4(n8810),
    .ZN(n8853) );
IND4D0 U8917 ( .A1(n8853), .B1(n8839), .B2(n8522), .B3(n5576)
   , .ZN(n8874) );
NR4D0 U8918 ( .A1(n8808), .A2(n8610), .A3(n8523), .A4(n8874),
    .ZN(n8524) );
AOI32D0 U8919 ( .A1(n8835), .A2(n8525), .A3(n8524), .B1(n8919
   ), .B2(n8525),
      .ZN(n8526) );
AOI211D0 U8920 ( .A1(n8697), .A2(n8528), .B(n8527), .C(n8526)
   , .ZN(n8946) );
INVD0 U8921 ( .I(n8873), .ZN(n8734) );
NR4D0 U8922 ( .A1(n8531), .A2(n8734), .A3(n8530), .A4(n8529),
    .ZN(n8842) );
NR2D0 U8923 ( .A1(n8796), .A2(n8777), .ZN(n8694) );
INVD0 U8924 ( .I(n8694), .ZN(n8584) );
NR2D0 U8925 ( .A1(n8533), .A2(n8532), .ZN(n8574) );
NR4D0 U8926 ( .A1(n8763), .A2(n8772), .A3(n8759), .A4(n8574),
    .ZN(n8623) );
OAI22D0 U8927 ( .A1(n9352), .A2(n8535), .B1(n8612), .B2(n8534
   ), .ZN(n8618)
      );
NR3D0 U8928 ( .A1(n8536), .A2(n8618), .A3(n8573), .ZN(n8628)
   );
CKND2D0 U8929 ( .A1(n8628), .A2(n8537), .ZN(n8634) );
NR4D0 U8930 ( .A1(n8564), .A2(n8539), .A3(n8538), .A4(n8634),
    .ZN(n8540) );
AN3D0 U8931 ( .A1(n8623), .A2(n8541), .A3(n8540), .Z(n8587) )
   ;
INR2D0 U8932 ( .A1(n8543), .B1(n8542), .ZN(n8568) );
ND3D0 U8933 ( .A1(n8544), .A2(n8587), .A3(n8568), .ZN(n8560)
   );
```

```
AOI21D0 U8934 ( .A1(n8547), .A2(n8546), .B(n8545), .ZN(n8588)
    );
CKND2D0 U8935 ( .A1(n8549), .A2(n8548), .ZN(n8717) );
NR2D0 U8936 ( .A1(n8753), .A2(n8717), .ZN(n8792) );
NR4D0 U8937 ( .A1(n8551), .A2(n8566), .A3(n8550), .A4(n8650),
    .ZN(n8552) );
IND4D0 U8938 ( .A1(n8560), .B1(n8588), .B2(n8792), .B3(n8552)
   , .ZN(n8583) );
INVD0 U8939 ( .I(n8875), .ZN(n8863) );
NR2D0 U8940 ( .A1(n8554), .A2(n8553), .ZN(n8669) );
ND4D0 U8941 ( .A1(n8863), .A2(n8669), .A3(n8556), .A4(n8555),
    .ZN(n8559) );
ND4D0 U8942 ( .A1(n8810), .A2(n8835), .A3(n8557), .A4(n8561),
    .ZN(n8558) );
NR4D0 U8943 ( .A1(n8707), .A2(n8560), .A3(n8559), .A4(n8558),
    .ZN(n8580) );
INVD0 U8944 ( .I(n8835), .ZN(n8847) );
NR2D0 U8945 ( .A1(n8847), .A2(n8733), .ZN(n8858) );
INVD0 U8946 ( .I(n8561), .ZN(n8586) );
NR2D0 U8947 ( .A1(n8563), .A2(n8562), .ZN(n8569) );
NR4D0 U8948 ( .A1(n8566), .A2(n8565), .A3(n8564), .A4(n8591),
    .ZN(n8567) );
ND4D0 U8949 ( .A1(n8569), .A2(n8588), .A3(n8568), .A4(n8567),
    .ZN(n8615) );
NR3D0 U8950 ( .A1(n8759), .A2(n8586), .A3(n8615), .ZN(n8636)
   );
ND4D0 U8951 ( .A1(n8636), .A2(n8891), .A3(n8571), .A4(n8570),
    .ZN(n8625) );
NR4D0 U8952 ( .A1(n8610), .A2(n8574), .A3(n8573), .A4(n8572),
    .ZN(n8577) );
ND4D0 U8953 ( .A1(n8577), .A2(n8633), .A3(n8576), .A4(n8575),
    .ZN(n8578) );
INR4D0 U8954 ( .A1(n8858), .B1(n8796), .B2(n8625), .B3(n8578)
   , .ZN(n8579) );
OAI22D0 U8955 ( .A1(n8580), .A2(n8937), .B1(n8579), .B2(n8919
   ), .ZN(n8581)
      );
AOI221D0 U8956 ( .A1(n8584), .A2(n8582), .B1(n8583), .B2(
   n8582), .C(n8581),
     .ZN(n8593) );
CKND2D0 U8957 ( .A1(n8844), .A2(n8920), .ZN(n8585) );
NR2D0 U8958 ( .A1(n8586), .A2(n8585), .ZN(n8621) );
ND4D0 U8959 ( .A1(n8835), .A2(n8588), .A3(n8621), .A4(n8587),
    .ZN(n8589) );
NR4D0 U8960 ( .A1(n8929), .A2(n8591), .A3(n8590), .A4(n8589),
    .ZN(n8592) );
AOI32D0 U8961 ( .A1(n8842), .A2(n8593), .A3(n8592), .B1(n8923
   ), .B2(n8593),
```

```
      .ZN(n8917) );
ND4D0 U8962 ( .A1(n8863), .A2(n8595), .A3(n8594), .A4(n8699),
    .ZN(n8608) );
INVD0 U8963 ( .I(n8602), .ZN(n8758) );
NR4D0 U8964 ( .A1(n8596), .A2(n8746), .A3(n8758), .A4(n8664),
    .ZN(n8597) );
AOI21D0 U8965 ( .A1(n8677), .A2(n8597), .B(n8674), .ZN(n8607)
    );
NR2D0 U8966 ( .A1(n8671), .A2(n8598), .ZN(n8925) );
INVD0 U8967 ( .I(n8752), .ZN(n8928) );
IINR4D0 U8968 ( .A1(n8925), .A2(n8600), .B1(n8599), .B2(n8928
   ), .ZN(n8605)
      );
CKND2D0 U8969 ( .A1(n8678), .A2(n8858), .ZN(n8859) );
INVD0 U8970 ( .I(n8731), .ZN(n8843) );
IND4D0 U8971 ( .A1(n8859), .B1(n8602), .B2(n8843), .B3(n8601)
   , .ZN(n8603) );
NR2D0 U8972 ( .A1(n8604), .A2(n8603), .ZN(n8722) );
OAI22D0 U8973 ( .A1(n8605), .A2(n8935), .B1(n8722), .B2(n8923
   ), .ZN(n8606)
      );
AOI211D0 U8974 ( .A1(n8609), .A2(n8608), .B(n8607), .C(n8606)
   , .ZN(n8663) );
NR2D0 U8975 ( .A1(n8610), .A2(n8746), .ZN(n8812) );
OA211D0 U8976 ( .A1(n8613), .A2(n8612), .B(n8611), .C(n8633),
    .Z(n8614) );
IND4D0 U8977 ( .A1(n8615), .B1(n8812), .B2(n8703), .B3(n8614)
   , .ZN(n8616) );
NR2D0 U8978 ( .A1(n8671), .A2(n8616), .ZN(n8630) );
CKND2D0 U8979 ( .A1(n8617), .A2(n8721), .ZN(n8619) );
INR4D0 U8980 ( .A1(n8621), .B1(n8620), .B2(n8619), .B3(n8618)
   , .ZN(n8622) );
AOI31D0 U8981 ( .A1(n8630), .A2(n8623), .A3(n8622), .B(n8935)
   , .ZN(n8642) );
CKND2D0 U8982 ( .A1(n8624), .A2(n8863), .ZN(n8901) );
NR2D0 U8983 ( .A1(n8796), .A2(n8901), .ZN(n8644) );
INVD0 U8984 ( .I(n8644), .ZN(n8647) );
INVD0 U8985 ( .I(n8792), .ZN(n8806) );
NR3D0 U8986 ( .A1(n8647), .A2(n8806), .A3(n8625), .ZN(n8627)
   );
AOI31D0 U8987 ( .A1(n8628), .A2(n8627), .A3(n8626), .B(n8937)
   , .ZN(n8641) );
INR4D0 U8988 ( .A1(n8630), .B1(n8928), .B2(n8629), .B3(n8634)
   , .ZN(n8638) );
NR4D0 U8989 ( .A1(n8782), .A2(n8632), .A3(n8847), .A4(n8631),
    .ZN(n8870) );
CKND2D0 U8990 ( .A1(n8870), .A2(n8882), .ZN(n8787) );
```

```
IND4D0 U8991 ( .A1(n8634), .B1(n8892), .B2(n8836), .B3(n8633)
   , .ZN(n8635) );
INR4D0 U8992 ( .A1(n8636), .B1(n8707), .B2(n8787), .B3(n8635)
   , .ZN(n8637) );
OAI22D0 U8993 ( .A1(n8638), .A2(n8923), .B1(n8637), .B2(n8919
   ), .ZN(n8640)
      );
OAI31D0 U8994 ( .A1(n8642), .A2(n8641), .A3(n8640), .B(n8639)
   , .ZN(n8661) );
INVD0 U8995 ( .I(n8812), .ZN(n8829) );
NR2D0 U8996 ( .A1(n8643), .A2(n8829), .ZN(n8675) );
AOI31D0 U8997 ( .A1(n8675), .A2(n8793), .A3(n8694), .B(n8935)
   , .ZN(n8659) );
NR2D0 U8998 ( .A1(n8878), .A2(n8701), .ZN(n8646) );
AOI31D0 U8999 ( .A1(n8646), .A2(n8645), .A3(n8644), .B(n8919)
   , .ZN(n8658) );
INR4D0 U9000 ( .A1(n8884), .B1(n5555), .B2(n8717), .B3(n8647)
   , .ZN(n8655) );
INVD0 U9001 ( .I(n8872), .ZN(n8830) );
NR3D0 U9002 ( .A1(n8808), .A2(n8648), .A3(n8830), .ZN(n8922)
   );
NR2D0 U9003 ( .A1(n8650), .A2(n8649), .ZN(n8651) );
ND4D0 U9004 ( .A1(n8652), .A2(n8922), .A3(n8792), .A4(n8651),
    .ZN(n8900) );
NR4D0 U9005 ( .A1(n8747), .A2(n8664), .A3(n8653), .A4(n8900),
    .ZN(n8654) );
OAI22D0 U9006 ( .A1(n8655), .A2(n8923), .B1(n8654), .B2(n8674
   ), .ZN(n8657)
      );
OAI31D0 U9007 ( .A1(n8659), .A2(n8658), .A3(n8657), .B(n8656)
   , .ZN(n8660) );
OAI211D0 U9008 ( .A1(n8663), .A2(n8662), .B(n8661), .C(n8660)
   , .ZN(n8916) );
ND3D0 U9009 ( .A1(n8834), .A2(n8861), .A3(n8751), .ZN(n8745)
   );
INR3D0 U9010 ( .A1(n8675), .B1(n8664), .B2(n8745), .ZN(n8673)
    );
INVD0 U9011 ( .I(n8665), .ZN(n8743) );
IND4D0 U9012 ( .A1(n8901), .B1(n8669), .B2(n8844), .B3(n8666)
   , .ZN(n8667) );
NR2D0 U9013 ( .A1(n8743), .A2(n8667), .ZN(n8726) );
NR4D0 U9014 ( .A1(n8735), .A2(n8747), .A3(n8734), .A4(n8668),
    .ZN(n8845) );
CKND2D0 U9015 ( .A1(n8669), .A2(n8845), .ZN(n8720) );
NR4D0 U9016 ( .A1(n8672), .A2(n8671), .A3(n8670), .A4(n8720),
    .ZN(n8816) );
OAI222D0 U9017 ( .A1(n8674), .A2(n8673), .B1(n8923), .B2(
   n8726), .C1(n8919),
```

```
      .C2(n8816), .ZN(n8714) );
ND4D0 U9018 ( .A1(n8677), .A2(n8676), .A3(n8675), .A4(n8811),
    .ZN(n8680) );
ND4D0 U9019 ( .A1(n8835), .A2(n8926), .A3(n8678), .A4(n8794),
    .ZN(n8679) );
AOI22D0 U9020 ( .A1(n8822), .A2(n8680), .B1(n8697), .B2(n8679
   ), .ZN(n8690)
       );
ND3D0 U9021 ( .A1(n8819), .A2(n8873), .A3(n8699), .ZN(n8931)
   );
OR4D0 U9022 ( .A1(n8682), .A2(n8681), .A3(n8931), .A4(n8831),
    .Z(n8683) );
OAI31D0 U9023 ( .A1(n8717), .A2(n8684), .A3(n8683), .B(n8899)
   , .ZN(n8689) );
OAI21D0 U9024 ( .A1(n8686), .A2(n8859), .B(n8685), .ZN(n8688)
    );
AOI31D0 U9025 ( .A1(n8690), .A2(n8689), .A3(n8688), .B(n8687)
   , .ZN(n8713) );
ND4D0 U9026 ( .A1(n8693), .A2(n8793), .A3(n8692), .A4(n8691),
    .ZN(n8696) );
NR2D0 U9027 ( .A1(n8847), .A2(n8930), .ZN(n8784) );
AOI31D0 U9028 ( .A1(n8694), .A2(n8809), .A3(n8784), .B(n8935)
   , .ZN(n8695) );
AOI211D0 U9029 ( .A1(n8697), .A2(n8696), .B(n8930), .C(n8695)
   , .ZN(n8711) );
ND4D0 U9030 ( .A1(n8775), .A2(n8699), .A3(n8812), .A4(n8698),
    .ZN(n8700) );
OAI31D0 U9031 ( .A1(n8878), .A2(n8701), .A3(n8700), .B(n8822)
   , .ZN(n8710) );
ND4D0 U9032 ( .A1(n8705), .A2(n8704), .A3(n8703), .A4(n8702),
    .ZN(n8706) );
OAI31D0 U9033 ( .A1(n8707), .A2(n8901), .A3(n8706), .B(n8899)
   , .ZN(n8709) );
AOI31D0 U9034 ( .A1(n8711), .A2(n8710), .A3(n8709), .B(n8708)
   , .ZN(n8712) );
AOI211D0 U9035 ( .A1(n8715), .A2(n8714), .B(n8713), .C(n8712)
   , .ZN(n8914) );
IND4D0 U9036 ( .A1(n8717), .B1(n8863), .B2(n8926), .B3(n8716)
   , .ZN(n8741) );
IND4D0 U9037 ( .A1(n8720), .B1(n8835), .B2(n8719), .B3(n8718)
   , .ZN(n8729) );
AOI31D0 U9038 ( .A1(n8723), .A2(n8722), .A3(n8721), .B(n8919)
   , .ZN(n8728) );
AOI31D0 U9039 ( .A1(n8726), .A2(n8725), .A3(n8724), .B(n8923)
   , .ZN(n8727) );
AOI211D0 U9040 ( .A1(n8899), .A2(n8729), .B(n8728), .C(n8727)
   , .ZN(n8737) );
CKND2D0 U9041 ( .A1(n8730), .A2(n8883), .ZN(n8732) );
```

```
OR2D0 U9042 ( .A1(n8732), .A2(n8731), .Z(n8932) );
NR2D0 U9043 ( .A1(n8932), .A2(n8733), .ZN(n8820) );
CKND2D0 U9044 ( .A1(n8863), .A2(n8793), .ZN(n8805) );
INR4D0 U9045 ( .A1(n8820), .B1(n8735), .B2(n8734), .B3(n8805)
   , .ZN(n8736) );
AOI32D0 U9046 ( .A1(n8738), .A2(n8737), .A3(n8736), .B1(n8935
   ), .B2(n8737),
      .ZN(n8739) );
AOI22D0 U9047 ( .A1(n8742), .A2(n8741), .B1(n8740), .B2(n8739
   ), .ZN(n8913)
      );
NR4D0 U9048 ( .A1(n8868), .A2(n8744), .A3(n8743), .A4(n8787),
    .ZN(n8750) );
NR4D0 U9049 ( .A1(n8748), .A2(n8747), .A3(n8746), .A4(n8745),
    .ZN(n8749) );
OAI22D0 U9050 ( .A1(n8750), .A2(n8923), .B1(n8749), .B2(n8937
   ), .ZN(n8804)
      );
CKND2D0 U9051 ( .A1(n8752), .A2(n8751), .ZN(n8933) );
CKND2D0 U9052 ( .A1(n8862), .A2(n8858), .ZN(n8885) );
NR4D0 U9053 ( .A1(n8753), .A2(n8853), .A3(n8933), .A4(n8885),
    .ZN(n8757) );
NR3D0 U9054 ( .A1(n8853), .A2(n8875), .A3(n8754), .ZN(n8857)
   );
INR4D0 U9055 ( .A1(n8857), .B1(n8755), .B2(n8776), .B3(n8885)
   , .ZN(n8756) );
OAI22D0 U9056 ( .A1(n8757), .A2(n8935), .B1(n8756), .B2(n8919
   ), .ZN(n8802)
      );
AOI211D0 U9057 ( .A1(n8761), .A2(n8760), .B(n8759), .C(n8758)
   , .ZN(n8774) );
AOI211D0 U9058 ( .A1(n8765), .A2(n8764), .B(n8763), .C(n8762)
   , .ZN(n8766) );
OAI211D0 U9059 ( .A1(n8769), .A2(n8768), .B(n8767), .C(n8766)
   , .ZN(n8781) );
NR4D0 U9060 ( .A1(n8772), .A2(n8771), .A3(n8770), .A4(n8781),
    .ZN(n8773) );
ND4D0 U9061 ( .A1(n8775), .A2(n8925), .A3(n8774), .A4(n8773),
    .ZN(n8786) );
NR3D0 U9062 ( .A1(n8778), .A2(n8777), .A3(n8776), .ZN(n8821)
   );
NR4D0 U9063 ( .A1(n8782), .A2(n8781), .A3(n8780), .A4(n8779),
    .ZN(n8783) );
ND4D0 U9064 ( .A1(n8810), .A2(n8784), .A3(n8821), .A4(n8783),
    .ZN(n8785) );
AOI22D0 U9065 ( .A1(n8582), .A2(n8786), .B1(n8697), .B2(n8785
   ), .ZN(n8800)
      );
```

```
INVD0 U9066 ( .I(n8787), .ZN(n8837) );
ND4D0 U9067 ( .A1(n8812), .A2(n8837), .A3(n8789), .A4(n8788),
    .ZN(n8790) );
OAI21D0 U9068 ( .A1(n8876), .A2(n8790), .B(n8899), .ZN(n8799)
    );
ND4D0 U9069 ( .A1(n8794), .A2(n8793), .A3(n8792), .A4(n8791),
    .ZN(n8795) );
OAI31D0 U9070 ( .A1(n8824), .A2(n8796), .A3(n8795), .B(n8822)
   , .ZN(n8798) );
AOI31D0 U9071 ( .A1(n8800), .A2(n8799), .A3(n8798), .B(n8797)
   , .ZN(n8801) );
AOI221D0 U9072 ( .A1(n8804), .A2(n8803), .B1(n8802), .B2(
   n8803), .C(n8801),
      .ZN(n8912) );
NR4D0 U9073 ( .A1(n8808), .A2(n8807), .A3(n8806), .A4(n8805),
    .ZN(n8827) );
ND4D0 U9074 ( .A1(n8854), .A2(n8810), .A3(n8880), .A4(n8809),
    .ZN(n8927) );
IND4D0 U9075 ( .A1(n8927), .B1(n8856), .B2(n8812), .B3(n8811)
   , .ZN(n8818) );
ND4D0 U9076 ( .A1(n8816), .A2(n8815), .A3(n8814), .A4(n8813),
    .ZN(n8817) );
AOI22D0 U9077 ( .A1(n8899), .A2(n8818), .B1(n8697), .B2(n8817
   ), .ZN(n8826)
      );
ND4D0 U9078 ( .A1(n8870), .A2(n8821), .A3(n8820), .A4(n8819),
    .ZN(n8823) );
OAI21D0 U9079 ( .A1(n8824), .A2(n8823), .B(n8822), .ZN(n8825)
    );
OAI211D0 U9080 ( .A1(n8827), .A2(n8935), .B(n8826), .C(n8825)
   , .ZN(n8909) );
NR4D0 U9081 ( .A1(n8831), .A2(n8830), .A3(n8829), .A4(n8828),
    .ZN(n8832) );
ND4D0 U9082 ( .A1(n8835), .A2(n8834), .A3(n8833), .A4(n8832),
    .ZN(n8841) );
ND4D0 U9083 ( .A1(n8839), .A2(n8838), .A3(n8837), .A4(n8836),
    .ZN(n8840) );
AOI22D0 U9084 ( .A1(n8609), .A2(n8841), .B1(n8697), .B2(n8840
   ), .ZN(n8851)
      );
AO31D0 U9085 ( .A1(n8870), .A2(n8892), .A3(n8842), .B(n8935),
    .Z(n8850) );
IND4D0 U9086 ( .A1(n8933), .B1(n8845), .B2(n8844), .B3(n8843)
   , .ZN(n8846) );
OAI21D0 U9087 ( .A1(n8847), .A2(n8846), .B(n8899), .ZN(n8849)
    );
AOI31D0 U9088 ( .A1(n8851), .A2(n8850), .A3(n8849), .B(n8848)
   , .ZN(n8908) );
```

```
NR4D0 U9089 ( .A1(n8853), .A2(n8867), .A3(n8852), .A4(n8901),
    .ZN(n8921) );
ND4D0 U9090 ( .A1(n8856), .A2(n8921), .A3(n8855), .A4(n8854),
    .ZN(n8866) );
AOI31D0 U9091 ( .A1(n8858), .A2(n8922), .A3(n8857), .B(n8935)
   , .ZN(n8865) );
INR3D0 U9092 ( .A1(n8861), .B1(n8860), .B2(n8859), .ZN(n8890)
    );
AOI31D0 U9093 ( .A1(n8863), .A2(n8862), .A3(n8890), .B(n8919)
   , .ZN(n8864) );
AOI211D0 U9094 ( .A1(n8697), .A2(n8866), .B(n8865), .C(n8864)
   , .ZN(n8906) );
INR4D0 U9095 ( .A1(n8870), .B1(n8869), .B2(n8868), .B3(n8867)
   , .ZN(n8871) );
AOI31D0 U9096 ( .A1(n8873), .A2(n8872), .A3(n8871), .B(n8919)
   , .ZN(n8898) );
INVD0 U9097 ( .I(n8874), .ZN(n8881) );
NR4D0 U9098 ( .A1(n8878), .A2(n8877), .A3(n8876), .A4(n8875),
    .ZN(n8879) );
AOI31D0 U9099 ( .A1(n8881), .A2(n8880), .A3(n8879), .B(n8935)
   , .ZN(n8897) );
IND4D0 U9100 ( .A1(n8885), .B1(n8884), .B2(n8883), .B3(n8882)
   , .ZN(n8886) );
NR4D0 U9101 ( .A1(n8889), .A2(n8888), .A3(n8887), .A4(n8886),
    .ZN(n8894) );
AN4D0 U9102 ( .A1(n5576), .A2(n8892), .A3(n8891), .A4(n8890),
    .Z(n8893) );
OAI22D0 U9103 ( .A1(n8894), .A2(n8923), .B1(n8893), .B2(n8937
   ), .ZN(n8896)
       );
OAI31D0 U9104 ( .A1(n8898), .A2(n8897), .A3(n8896), .B(n8895)
   , .ZN(n8905) );
OAI31D0 U9105 ( .A1(n8902), .A2(n8901), .A3(n8900), .B(n8899)
   , .ZN(n8904) );
AOI32D0 U9106 ( .A1(n8906), .A2(n8905), .A3(n8904), .B1(n8903
   ), .B2(n8905),
      .ZN(n8907) );
AOI211D0 U9107 ( .A1(n8910), .A2(n8909), .B(n8908), .C(n8907)
   , .ZN(n8911) );
ND4D0 U9108 ( .A1(n8914), .A2(n8913), .A3(n8912), .A4(n8911),
    .ZN(n8915) );
AOI211D0 U9109 ( .A1(n8918), .A2(n8917), .B(n8916), .C(n8915)
   , .ZN(n8944) );
AOI31D0 U9110 ( .A1(n8922), .A2(n8921), .A3(n8920), .B(n8919)
   , .ZN(n8942) );
AOI31D0 U9111 ( .A1(n8926), .A2(n8925), .A3(n8924), .B(n8923)
   , .ZN(n8941) );
```

```
NR4D0 U9112 ( .A1(n8930), .A2(n8929), .A3(n8928), .A4(n8927),
    .ZN(n8938) );
NR4D0 U9113 ( .A1(n8934), .A2(n8933), .A3(n8932), .A4(n8931),
    .ZN(n8936) );
OAI22D0 U9114 ( .A1(n8938), .A2(n8937), .B1(n8936), .B2(n8935
   ), .ZN(n8940)
      );
OAI31D0 U9115 ( .A1(n8942), .A2(n8941), .A3(n8940), .B(n8939)
   , .ZN(n8943) );
OAI211D0 U9116 ( .A1(n8946), .A2(n8945), .B(n8944), .C(n8943)
   , .ZN(n8948) );
AOI22D0 U9117 ( .A1(n8949), .A2(n8948), .B1(n8947), .B2(q_out
   [6]), .ZN(n9331) );
INVD0 U9118 ( .I(n9072), .ZN(n8990) );
NR2D0 U9119 ( .A1(n9075), .A2(n8950), .ZN(n8984) );
INVD0 U9120 ( .I(n8984), .ZN(n9192) );
NR2D0 U9121 ( .A1(n9048), .A2(n9192), .ZN(n9169) );
IND2D0 U9122 ( .A1(n9169), .B1(n8999), .ZN(n9213) );
CKND2D0 U9123 ( .A1(counter[9]), .A2(n8951), .ZN(n9093) );
NR2D0 U9124 ( .A1(n9110), .A2(n9093), .ZN(n9220) );
OR3D0 U9125 ( .A1(n9213), .A2(n8987), .A3(n9220), .Z(n9040) )
   ;
NR2D0 U9126 ( .A1(n9048), .A2(n9093), .ZN(n8995) );
NR2D0 U9127 ( .A1(n9075), .A2(n9076), .ZN(n9122) );
CKND2D0 U9128 ( .A1(counter[9]), .A2(n8992), .ZN(n9197) );
INVD0 U9129 ( .I(n9197), .ZN(n9297) );
AOI211D0 U9130 ( .A1(n9025), .A2(n9173), .B(n9122), .C(n9297)
   , .ZN(n8952) );
CKND2D0 U9131 ( .A1(n8952), .A2(n9119), .ZN(n9177) );
NR2D0 U9132 ( .A1(n9075), .A2(n8953), .ZN(n9089) );
INVD0 U9133 ( .I(n9089), .ZN(n9209) );
AOI31D0 U9134 ( .A1(n8955), .A2(n9210), .A3(n9209), .B(n8954)
   , .ZN(n9269) );
NR2D0 U9135 ( .A1(n9075), .A2(n8957), .ZN(n9015) );
CKND2D0 U9136 ( .A1(n8971), .A2(n9015), .ZN(n9136) );
CKND2D0 U9137 ( .A1(n8956), .A2(n9136), .ZN(n9293) );
INVD0 U9138 ( .I(n9293), .ZN(n9035) );
CKND2D0 U9139 ( .A1(n8957), .A2(n9111), .ZN(n9128) );
NR2D0 U9140 ( .A1(n9311), .A2(n9192), .ZN(n9298) );
AOI211D0 U9141 ( .A1(n8979), .A2(n9128), .B(n9298), .C(n8958)
   , .ZN(n8960) );
CKND2D0 U9142 ( .A1(n8979), .A2(n8984), .ZN(n9208) );
INVD0 U9143 ( .I(n9208), .ZN(n9041) );
CKND2D0 U9144 ( .A1(n8964), .A2(n8984), .ZN(n9118) );
INVD0 U9145 ( .I(n9093), .ZN(n9059) );
CKND2D0 U9146 ( .A1(n8997), .A2(n9059), .ZN(n9120) );
CKND2D0 U9147 ( .A1(n9118), .A2(n9120), .ZN(n9012) );
CKND2D0 U9148 ( .A1(n8964), .A2(n9015), .ZN(n9034) );
```

```
  CKND2D0 U9149 ( .A1(n9198), .A2(n9034), .ZN(n9135) );
  CKND2D0 U9150 ( .A1(n8979), .A2(n9059), .ZN(n9083) );
  CKND2D0 U9151 ( .A1(n9172), .A2(n9089), .ZN(n9276) );
  CKND2D0 U9152 ( .A1(n9083), .A2(n9276), .ZN(n9162) );
  INVD0 U9153 ( .I(n9162), .ZN(n9131) );
  IND3D0 U9154 ( .A1(n9135), .B1(n9131), .B2(n9033), .ZN(n9080)
      );
  NR4D0 U9155 ( .A1(n9041), .A2(n9012), .A3(n9284), .A4(n9080),
      .ZN(n8959) );
  ND4D0 U9156 ( .A1(n8961), .A2(n9035), .A3(n8960), .A4(n8959),
      .ZN(n9175) );
  NR4D0 U9157 ( .A1(n8995), .A2(n9177), .A3(n9269), .A4(n9175),
      .ZN(n9247) );
  CKND2D0 U9158 ( .A1(n8964), .A2(n9059), .ZN(n9266) );
  CKND2D0 U9159 ( .A1(n9172), .A2(n9015), .ZN(n9183) );
  CKND2D0 U9160 ( .A1(n9129), .A2(n9059), .ZN(n9277) );
  ND4D0 U9161 ( .A1(n9247), .A2(n9266), .A3(n9183), .A4(n9277),
      .ZN(n9259) );
  NR2D0 U9162 ( .A1(n8962), .A2(n9192), .ZN(n9171) );
  CKND2D0 U9163 ( .A1(n9172), .A2(n8984), .ZN(n9288) );
  CKND2D0 U9164 ( .A1(n9200), .A2(n9288), .ZN(n9282) );
  CKND2D0 U9165 ( .A1(n9014), .A2(n9059), .ZN(n9301) );
  INVD0 U9166 ( .I(n9301), .ZN(n9042) );
  NR2D0 U9167 ( .A1(n8963), .A2(n9042), .ZN(n8993) );
  CKND2D0 U9168 ( .A1(n8964), .A2(n9089), .ZN(n9182) );
  ND4D0 U9169 ( .A1(n8993), .A2(n8975), .A3(n8996), .A4(n9182),
      .ZN(n9167) );
  INR4D0 U9170 ( .A1(n9267), .B1(n9171), .B2(n9282), .B3(n9167)
     , .ZN(n8967) );
  NR2D0 U9171 ( .A1(n9075), .A2(n9004), .ZN(n9179) );
  INVD0 U9172 ( .I(n9179), .ZN(n9156) );
  ND4D0 U9173 ( .A1(n8967), .A2(n8966), .A3(n8965), .A4(n9156),
      .ZN(n9315) );
  OAI31D0 U9174 ( .A1(n9040), .A2(n9259), .A3(n9315), .B(n9312)
     , .ZN(n8989) );
  OAI21D0 U9175 ( .A1(n9295), .A2(n9059), .B(n9090), .ZN(n8981)
      );
  CKND2D0 U9176 ( .A1(n9090), .A2(n9015), .ZN(n9308) );
  OAI211D0 U9177 ( .A1(n9029), .A2(n9192), .B(n9310), .C(n9183)
     , .ZN(n9117) );
  CKND2D0 U9178 ( .A1(n9174), .A2(n9301), .ZN(n8974) );
  AOI22D0 U9179 ( .A1(n8968), .A2(n9173), .B1(n9025), .B2(n9089
     ), .ZN(n8969)
        );
  ND3D0 U9180 ( .A1(n8969), .A2(n9119), .A3(n9277), .ZN(n9002)
     );
  OAI21D0 U9181 ( .A1(n8997), .A2(n9090), .B(n8984), .ZN(n9190)
      );
```

```
  NR2D0 U9182 ( .A1(n8970), .A2(n9209), .ZN(n9023) );
  INR4D0 U9183 ( .A1(n9190), .B1(n9254), .B2(n9009), .B3(n9023)
     , .ZN(n9086) );
  CKND2D0 U9184 ( .A1(n8971), .A2(n9059), .ZN(n9309) );
  INVD0 U9185 ( .I(n9309), .ZN(n9049) );
  INR2D0 U9186 ( .A1(n9120), .B1(n9049), .ZN(n9101) );
  ND4D0 U9187 ( .A1(n9086), .A2(n9101), .A3(n8999), .A4(n8972),
      .ZN(n8973) );
  NR4D0 U9188 ( .A1(n9117), .A2(n8974), .A3(n9002), .A4(n8973),
      .ZN(n9159) );
  INR4D0 U9189 ( .A1(n8975), .B1(n9043), .B2(n9113), .B3(n9220)
     , .ZN(n8976) );
  AN3D0 U9190 ( .A1(n9159), .A2(n8976), .A3(n9288), .Z(n9134) )
     ;
  AN4D0 U9191 ( .A1(n9279), .A2(n8977), .A3(n9308), .A4(n9134),
      .Z(n8980) );
  CKND2D0 U9192 ( .A1(n8978), .A2(n9089), .ZN(n9263) );
  CKND2D0 U9193 ( .A1(n8979), .A2(n9015), .ZN(n9027) );
  ND4D0 U9194 ( .A1(n8981), .A2(n8980), .A3(n9263), .A4(n9027),
      .ZN(n9203) );
  NR3D0 U9195 ( .A1(n9075), .A2(n8983), .A3(n8982), .ZN(n9001)
     );
  AOI211D0 U9196 ( .A1(n9060), .A2(n8984), .B(n9001), .C(n9162),
      .Z(n8985) );
  NR4D0 U9197 ( .A1(n8987), .A2(n8986), .A3(n9203), .A4(n8985),
      .ZN(n8988) );
  AOI32D0 U9198 ( .A1(n8990), .A2(n8989), .A3(n8988), .B1(n9272
     ), .B2(n8989),
        .ZN(n9071) );
  CKND2D0 U9199 ( .A1(n8997), .A2(n9089), .ZN(n9165) );
  INVD0 U9200 ( .I(n9165), .ZN(n9007) );
  NR2D0 U9201 ( .A1(n8991), .A2(n9192), .ZN(n9052) );
  NR2D0 U9202 ( .A1(n9043), .A2(n8992), .ZN(n9081) );
  IND4D0 U9203 ( .A1(n8994), .B1(n9081), .B2(n8993), .B3(n9289)
     , .ZN(n9019) );
  INR2D0 U9204 ( .A1(n8996), .B1(n8995), .ZN(n9078) );
  CKND2D0 U9205 ( .A1(n8997), .A2(n9015), .ZN(n9133) );
  INR2D0 U9206 ( .A1(n9133), .B1(n9220), .ZN(n9109) );
  CKND2D0 U9207 ( .A1(n8998), .A2(n9015), .ZN(n9039) );
  ND4D0 U9208 ( .A1(n9078), .A2(n9109), .A3(n8999), .A4(n9039),
      .ZN(n9000) );
  NR4D0 U9209 ( .A1(n9019), .A2(n9293), .A3(n9001), .A4(n9000),
      .ZN(n9082) );
  INVD0 U9210 ( .I(n9002), .ZN(n9045) );
  ND3D0 U9211 ( .A1(n9198), .A2(n9003), .A3(n9027), .ZN(n9281)
     );
  INVD0 U9212 ( .I(n9281), .ZN(n9223) );
```

```
ND3D0 U9213 ( .A1(n9004), .A2(n9267), .A3(n9118), .ZN(n9227)
   );
CKND2D0 U9214 ( .A1(n9200), .A2(n9183), .ZN(n9270) );
CKND2D0 U9215 ( .A1(n9129), .A2(n9089), .ZN(n9264) );
INR2D0 U9216 ( .A1(n9264), .B1(n9023), .ZN(n9230) );
OAI211D0 U9217 ( .A1(n9092), .A2(n9192), .B(n9230), .C(n9245)
   , .ZN(n9005) );
NR4D0 U9218 ( .A1(n9095), .A2(n9227), .A3(n9270), .A4(n9005),
    .ZN(n9006) );
CKND2D0 U9219 ( .A1(n9223), .A2(n9006), .ZN(n9032) );
OR2D0 U9220 ( .A1(n9254), .A2(n9127), .Z(n9051) );
NR4D0 U9221 ( .A1(n9073), .A2(n9162), .A3(n9032), .A4(n9051),
    .ZN(n9058) );
ND4D0 U9222 ( .A1(n9258), .A2(n9082), .A3(n9045), .A4(n9058),
    .ZN(n9098) );
NR4D0 U9223 ( .A1(n9009), .A2(n9007), .A3(n9052), .A4(n9098),
    .ZN(n9021) );
AN4D0 U9224 ( .A1(n9078), .A2(n9279), .A3(n9165), .A4(n9200),
    .Z(n9008) );
OAI31D0 U9225 ( .A1(n9348), .A2(n9143), .A3(n9210), .B(n9008)
   , .ZN(n9160) );
NR2D0 U9226 ( .A1(n9178), .A2(n9192), .ZN(n9262) );
INR4D0 U9227 ( .A1(n9277), .B1(n9009), .B2(n9049), .B3(n9262)
   , .ZN(n9010) );
CKND2D0 U9228 ( .A1(n9109), .A2(n9010), .ZN(n9166) );
OAI211D0 U9229 ( .A1(n9011), .A2(n9192), .B(n9083), .C(n9119)
   , .ZN(n9187) );
NR4D0 U9230 ( .A1(n9160), .A2(n9166), .A3(n9187), .A4(n9012),
    .ZN(n9013) );
ND4D0 U9231 ( .A1(n9196), .A2(n9013), .A3(n9156), .A4(n9266),
    .ZN(n9285) );
CKAN2D0 U9232 ( .A1(n9015), .A2(n9014), .Z(n9219) );
OR2D0 U9233 ( .A1(n9298), .A2(n9299), .Z(n9016) );
NR4D0 U9234 ( .A1(n9096), .A2(n9219), .A3(n9016), .A4(n9213),
    .ZN(n9031) );
INR2D0 U9235 ( .A1(n9182), .B1(n9041), .ZN(n9062) );
ND4D0 U9236 ( .A1(n9035), .A2(n9031), .A3(n9062), .A4(n9267),
    .ZN(n9244) );
NR2D0 U9237 ( .A1(n9114), .A2(n9113), .ZN(n9017) );
CKND2D0 U9238 ( .A1(n9276), .A2(n9027), .ZN(n9146) );
INVD0 U9239 ( .I(n9146), .ZN(n9091) );
INVD0 U9240 ( .I(n9073), .ZN(n9302) );
ND4D0 U9241 ( .A1(n9017), .A2(n9091), .A3(n9174), .A4(n9302),
    .ZN(n9018) );
NR4D0 U9242 ( .A1(n9019), .A2(n9285), .A3(n9244), .A4(n9018),
    .ZN(n9020) );
OAI22D0 U9243 ( .A1(n9021), .A2(n9300), .B1(n9020), .B2(n9318
   ), .ZN(n9070)
```

```
        );
INR4D0 U9244 ( .A1(n9034), .B1(n9038), .B2(n9285), .B3(n9022)
   , .ZN(n9246) );
IND4D0 U9245 ( .A1(n9023), .B1(n9099), .B2(n9288), .B3(n9308)
   , .ZN(n9024) );
INR2D0 U9246 ( .A1(n9246), .B1(n9024), .ZN(n9211) );
ND3D0 U9247 ( .A1(n9211), .A2(n9289), .A3(n9206), .ZN(n9145)
   );
CKND2D0 U9248 ( .A1(n9295), .A2(n9025), .ZN(n9026) );
CKND2D0 U9249 ( .A1(n9026), .A2(n9197), .ZN(n9207) );
OAI211D0 U9250 ( .A1(n9029), .A2(n9028), .B(n9301), .C(n9027)
   , .ZN(n9030) );
INR4D0 U9251 ( .A1(n9031), .B1(n9145), .B2(n9207), .B3(n9030)
   , .ZN(n9068) );
IND3D0 U9252 ( .A1(n9032), .B1(n9279), .B2(n9206), .ZN(n9057)
    );
ND4D0 U9253 ( .A1(n9076), .A2(n9047), .A3(n9033), .A4(n9133),
    .ZN(n9037) );
ND4D0 U9254 ( .A1(n9035), .A2(n9288), .A3(n9034), .A4(n9197),
    .ZN(n9036) );
NR4D0 U9255 ( .A1(n9262), .A2(n9038), .A3(n9037), .A4(n9036),
    .ZN(n9158) );
INVD0 U9256 ( .I(n9039), .ZN(n9170) );
NR3D0 U9257 ( .A1(n9040), .A2(n9249), .A3(n9170), .ZN(n9257)
   );
INVD0 U9258 ( .I(n9257), .ZN(n9065) );
NR4D0 U9259 ( .A1(n9043), .A2(n9042), .A3(n9041), .A4(n9065),
    .ZN(n9044) );
ND4D0 U9260 ( .A1(n9046), .A2(n9045), .A3(n9158), .A4(n9044),
    .ZN(n9056) );
OA21D0 U9261 ( .A1(counter[2]), .A2(n9048), .B(n9047), .Z(
   n9054) );
NR2D0 U9262 ( .A1(n9050), .A2(n9049), .ZN(n9112) );
ND4D0 U9263 ( .A1(n9082), .A2(n9112), .A3(n9302), .A4(n9165),
    .ZN(n9234) );
NR4D0 U9264 ( .A1(n9052), .A2(n9051), .A3(n9057), .A4(n9234),
    .ZN(n9053) );
AOI31D0 U9265 ( .A1(n9054), .A2(n9053), .A3(n9310), .B(n9272)
   , .ZN(n9055) );
AOI221D0 U9266 ( .A1(n9057), .A2(n9141), .B1(n9056), .B2(
   n9141), .C(n9055),
      .ZN(n9067) );
INVD0 U9267 ( .I(n9058), .ZN(n9064) );
OAI21D0 U9268 ( .A1(n9090), .A2(n9060), .B(n9059), .ZN(n9061)
    );
ND4D0 U9269 ( .A1(n9062), .A2(n9158), .A3(n9119), .A4(n9061),
    .ZN(n9063) );
```

```
OAI31D0 U9270 ( .A1(n9065), .A2(n9064), .A3(n9063), .B(n9226)
   , .ZN(n9066) );
OAI211D0 U9271 ( .A1(n9068), .A2(n9274), .B(n9067), .C(n9066)
   , .ZN(n9069) );
OAI32D0 U9272 ( .A1(n9342), .A2(n9071), .A3(n9070), .B1(
   counter[10]), .B2(
      n9069), .ZN(n9155) );
INR4D0 U9273 ( .A1(n9136), .B1(n9073), .B2(n9072), .B3(n9227)
   , .ZN(n9074) );
OAI211D0 U9274 ( .A1(n9076), .A2(n9075), .B(n9074), .C(n9266)
   , .ZN(n9202) );
NR2D0 U9275 ( .A1(n9077), .A2(n9219), .ZN(n9125) );
ND4D0 U9276 ( .A1(n9078), .A2(n9159), .A3(n9125), .A4(n9263),
    .ZN(n9079) );
NR3D0 U9277 ( .A1(n9202), .A2(n9080), .A3(n9079), .ZN(n9106)
   );
INVD0 U9278 ( .I(n9282), .ZN(n9097) );
AOI31D0 U9279 ( .A1(n9106), .A2(n9081), .A3(n9097), .B(n9272)
   , .ZN(n9152) );
NR2D0 U9280 ( .A1(n9221), .A2(n9127), .ZN(n9087) );
INVD0 U9281 ( .I(n9082), .ZN(n9195) );
ND4D0 U9282 ( .A1(n9302), .A2(n9263), .A3(n9083), .A4(n9182),
    .ZN(n9084) );
NR2D0 U9283 ( .A1(n9195), .A2(n9084), .ZN(n9085) );
ND4D0 U9284 ( .A1(n9087), .A2(n9086), .A3(n9085), .A4(n9099),
    .ZN(n9115) );
INVD0 U9285 ( .I(n9088), .ZN(n9191) );
CKND2D0 U9286 ( .A1(n9090), .A2(n9089), .ZN(n9265) );
OAI211D0 U9287 ( .A1(n9093), .A2(n9092), .B(n9091), .C(n9265)
   , .ZN(n9094) );
NR4D0 U9288 ( .A1(n9262), .A2(n9096), .A3(n9095), .A4(n9094),
    .ZN(n9184) );
INR2D0 U9289 ( .A1(n9278), .B1(n9179), .ZN(n9231) );
ND4D0 U9290 ( .A1(n9191), .A2(n9097), .A3(n9184), .A4(n9231),
    .ZN(n9103) );
INVD0 U9291 ( .I(n9098), .ZN(n9100) );
AOI31D0 U9292 ( .A1(n9101), .A2(n9100), .A3(n9099), .B(n9274)
   , .ZN(n9102) );
AOI221D0 U9293 ( .A1(n9115), .A2(n9226), .B1(n9103), .B2(
   n9226), .C(n9102),
      .ZN(n9108) );
INVD0 U9294 ( .I(n9104), .ZN(n9255) );
INR4D0 U9295 ( .A1(n9106), .B1(n9255), .B2(n9113), .B3(n9105)
   , .ZN(n9107) );
AOI32D0 U9296 ( .A1(n9109), .A2(n9108), .A3(n9107), .B1(n9300
   ), .B2(n9108),
      .ZN(n9151) );
NR2D0 U9297 ( .A1(n9111), .A2(n9110), .ZN(n9123) );
```

```
INVD0 U9298 ( .I(n9112), .ZN(n9253) );
OR4D0 U9299 ( .A1(n9115), .A2(n9253), .A3(n9114), .A4(n9113),
    .Z(n9116) );
IINR4D0 U9300 ( .A1(n9119), .A2(n9118), .B1(n9117), .B2(n9116
   ), .ZN(n9283)
       );
ND4D0 U9301 ( .A1(n9283), .A2(n9191), .A3(n9156), .A4(n9120),
    .ZN(n9121) );
NR4D0 U9302 ( .A1(n9124), .A2(n9123), .A3(n9122), .A4(n9121),
    .ZN(n9149) );
INVD0 U9303 ( .I(n9125), .ZN(n9126) );
AOI211D0 U9304 ( .A1(n9129), .A2(n9128), .B(n9127), .C(n9126)
   , .ZN(n9130) );
OAI211D0 U9305 ( .A1(n9132), .A2(n9200), .B(n9131), .C(n9130)
   , .ZN(n9142) );
IIND4D0 U9306 ( .A1(n9202), .A2(n9262), .B1(n9134), .B2(n9133
   ), .ZN(n9140)
       );
NR4D0 U9307 ( .A1(n9227), .A2(n9142), .A3(n9203), .A4(n9135),
    .ZN(n9137) );
AOI31D0 U9308 ( .A1(n9138), .A2(n9137), .A3(n9136), .B(n9274)
   , .ZN(n9139) );
AOI221D0 U9309 ( .A1(n9142), .A2(n9141), .B1(n9140), .B2(
   n9141), .C(n9139),
      .ZN(n9148) );
AO21D0 U9310 ( .A1(n9143), .A2(n9173), .B(n9244), .Z(n9144) )
   ;
OAI31D0 U9311 ( .A1(n9146), .A2(n9145), .A3(n9144), .B(n9307)
   , .ZN(n9147) );
OAI211D0 U9312 ( .A1(n9149), .A2(n9318), .B(n9148), .C(n9147)
   , .ZN(n9150) );
OAI32D0 U9313 ( .A1(counter[10]), .A2(n9152), .A3(n9151), .B1
   (n9342), .B2(
      n9150), .ZN(n9154) );
OAI221D0 U9314 ( .A1(counter[8]), .A2(n9155), .B1(n9338), .B2
   (n9154), .C(
      n9153), .ZN(n9327) );
CKND2D0 U9315 ( .A1(n9266), .A2(n9156), .ZN(n9161) );
CKND2D0 U9316 ( .A1(n9157), .A2(n9264), .ZN(n9248) );
IND4D0 U9317 ( .A1(n9160), .B1(n9159), .B2(n9158), .B3(n9267)
   , .ZN(n9218) );
NR4D0 U9318 ( .A1(n9162), .A2(n9161), .A3(n9248), .A4(n9218),
    .ZN(n9164) );
AOI31D0 U9319 ( .A1(n9164), .A2(n9245), .A3(n9163), .B(n9274)
   , .ZN(n9242) );
IIND4D0 U9320 ( .A1(n9167), .A2(n9166), .B1(n9258), .B2(n9165
   ), .ZN(n9168)
       );
```

```
NR3D0 U9321 ( .A1(n9170), .A2(n9169), .A3(n9168), .ZN(n9252)
   );
AOI221D0 U9322 ( .A1(n9294), .A2(n9173), .B1(n9172), .B2(
   n9173), .C(n9171),
      .ZN(n9176) );
CKND2D0 U9323 ( .A1(n9174), .A2(n9288), .ZN(n9261) );
IINR4D0 U9324 ( .A1(n9252), .A2(n9176), .B1(n9175), .B2(n9261
   ), .ZN(n9271)
       );
INVD0 U9325 ( .I(n9177), .ZN(n9181) );
NR2D0 U9326 ( .A1(n9178), .A2(n9209), .ZN(n9286) );
NR4D0 U9327 ( .A1(n9255), .A2(n9249), .A3(n9179), .A4(n9286),
    .ZN(n9180) );
AOI31D0 U9328 ( .A1(n9271), .A2(n9181), .A3(n9180), .B(n9300)
   , .ZN(n9241) );
ND4D0 U9329 ( .A1(n9185), .A2(n9184), .A3(n9183), .A4(n9182),
    .ZN(n9186) );
AOI211D0 U9330 ( .A1(n9189), .A2(n9188), .B(n9187), .C(n9186)
   , .ZN(n9228) );
ND3D0 U9331 ( .A1(n9190), .A2(n9228), .A3(n9279), .ZN(n9217)
   );
OAI211D0 U9332 ( .A1(n9193), .A2(n9192), .B(n9191), .C(n9264)
   , .ZN(n9194) );
INR4D0 U9333 ( .A1(n9196), .B1(n9195), .B2(n9217), .B3(n9194)
   , .ZN(n9205) );
ND4D0 U9334 ( .A1(n9200), .A2(n9199), .A3(n9198), .A4(n9197),
    .ZN(n9201) );
OAI31D0 U9335 ( .A1(n9203), .A2(n9202), .A3(n9201), .B(n9226)
   , .ZN(n9204) );
OAI211D0 U9336 ( .A1(n9205), .A2(n9272), .B(n9338), .C(n9204)
   , .ZN(n9240) );
IND4D0 U9337 ( .A1(n9207), .B1(n9206), .B2(n9278), .B3(n9301)
   , .ZN(n9243) );
INVD0 U9338 ( .I(n9248), .ZN(n9291) );
ND3D0 U9339 ( .A1(n9209), .A2(n9208), .A3(n9291), .ZN(n9212)
   );
IND3D0 U9340 ( .A1(n9212), .B1(n9211), .B2(n9210), .ZN(n9292)
    );
NR4D0 U9341 ( .A1(n9214), .A2(n9213), .A3(n9243), .A4(n9292),
    .ZN(n9238) );
OR4D0 U9342 ( .A1(n9217), .A2(n9234), .A3(n9216), .A4(n9215),
    .Z(n9225) );
NR4D0 U9343 ( .A1(n9221), .A2(n9220), .A3(n9219), .A4(n9218),
    .ZN(n9222) );
AOI32D0 U9344 ( .A1(n9223), .A2(counter[8]), .A3(n9222), .B1(
   n9272), .B2(
      counter[8]), .ZN(n9224) );
```

```
AOI221D0 U9345 ( .A1(n9227), .A2(n9226), .B1(n9225), .B2(
   n9226), .C(n9224),
      .ZN(n9237) );
INVD0 U9346 ( .I(n9228), .ZN(n9235) );
NR2D0 U9347 ( .A1(n9229), .A2(n9249), .ZN(n9232) );
ND4D0 U9348 ( .A1(n9232), .A2(n9258), .A3(n9231), .A4(n9230),
    .ZN(n9233) );
OAI31D0 U9349 ( .A1(n9235), .A2(n9234), .A3(n9233), .B(n9312)
   , .ZN(n9236) );
OAI211D0 U9350 ( .A1(n9238), .A2(n9300), .B(n9237), .C(n9236)
   , .ZN(n9239) );
OAI31D0 U9351 ( .A1(n9242), .A2(n9241), .A3(n9240), .B(n9239)
   , .ZN(n9325) );
INR3D0 U9352 ( .A1(n9265), .B1(n9244), .B2(n9243), .ZN(n9287)
    );
AOI31D0 U9353 ( .A1(n9246), .A2(n9287), .A3(n9245), .B(n9300)
   , .ZN(n9323) );
INVD0 U9354 ( .I(n9247), .ZN(n9314) );
NR4D0 U9355 ( .A1(n9249), .A2(n9270), .A3(n9314), .A4(n9248),
    .ZN(n9250) );
AOI32D0 U9356 ( .A1(n9252), .A2(n9251), .A3(n9250), .B1(n9318
   ), .B2(n9251),
      .ZN(n9322) );
NR4D0 U9357 ( .A1(n9255), .A2(n9254), .A3(n9286), .A4(n9253),
    .ZN(n9256) );
IND4D0 U9358 ( .A1(n9259), .B1(n9258), .B2(n9257), .B3(n9256)
   , .ZN(n9260) );
INR4D0 U9359 ( .A1(n9263), .B1(n9262), .B2(n9261), .B3(n9260)
   , .ZN(n9275) );
ND4D0 U9360 ( .A1(n9267), .A2(n9266), .A3(n9265), .A4(n9264),
    .ZN(n9268) );
INR4D0 U9361 ( .A1(n9271), .B1(n9270), .B2(n9269), .B3(n9268)
   , .ZN(n9273) );
OAI22D0 U9362 ( .A1(n9275), .A2(n9274), .B1(n9273), .B2(n9272
   ), .ZN(n9321)
       );
ND4D0 U9363 ( .A1(n9279), .A2(n9278), .A3(n9277), .A4(n9276),
    .ZN(n9280) );
INR4D0 U9364 ( .A1(n9283), .B1(n9282), .B2(n9281), .B3(n9280)
   , .ZN(n9319) );
INR4D0 U9365 ( .A1(n9287), .B1(n9286), .B2(n9285), .B3(n9284)
   , .ZN(n9290) );
ND4D0 U9366 ( .A1(n9291), .A2(n9290), .A3(n9289), .A4(n9288),
    .ZN(n9306) );
AO211D0 U9367 ( .A1(n9295), .A2(n9294), .B(n9293), .C(n9292),
    .Z(n9296) );
NR4D0 U9368 ( .A1(n9299), .A2(n9298), .A3(n9297), .A4(n9296),
    .ZN(n9303) );
```

```
  AOI31D0 U9369 ( .A1(n9303), .A2(n9302), .A3(n9301), .B(n9300)
     , .ZN(n9305) );
  AOI211D0 U9370 ( .A1(n9307), .A2(n9306), .B(n9305), .C(n9304)
     , .ZN(n9317) );
  ND4D0 U9371 ( .A1(n9311), .A2(n9310), .A3(n9309), .A4(n9308),
      .ZN(n9313) );
  OAI31D0 U9372 ( .A1(n9315), .A2(n9314), .A3(n9313), .B(n9312)
     , .ZN(n9316) );
  OAI211D0 U9373 ( .A1(n9319), .A2(n9318), .B(n9317), .C(n9316)
     , .ZN(n9320) );
  OAI31D0 U9374 ( .A1(n9323), .A2(n9322), .A3(n9321), .B(n9320)
     , .ZN(n9324) );
  OAI221D0 U9375 ( .A1(n9325), .A2(counter[10]), .B1(n9342), .
     B2(n9324), .C(
        counter[7]), .ZN(n9326) );
  ND4D0 U9376 ( .A1(n9329), .A2(n9328), .A3(n9327), .A4(n9326),
      .ZN(n9330) );
  OAI211D0 U9377 ( .A1(n9333), .A2(n9332), .B(n9331), .C(n9330)
     , .ZN(n4818) );
  NR2D0 U9378 ( .A1(counter[0]), .A2(n9349), .ZN(N4767) );
  NR2D0 U9379 ( .A1(n9335), .A2(n9334), .ZN(n9340) );
  IAO21D0 U9380 ( .A1(counter[11]), .A2(n9340), .B(n9349), .ZN(
     N4778) );
  NR2D0 U9381 ( .A1(n9349), .A2(n9336), .ZN(N4768) );
  CKND2D0 U9382 ( .A1(counter[7]), .A2(n9350), .ZN(n9337) );
  AOI211D0 U9383 ( .A1(n9338), .A2(n9337), .B(n9339), .C(n9349)
     , .ZN(N4775) );
  CKND2D0 U9384 ( .A1(counter[9]), .A2(n9339), .ZN(n9341) );
  AOI211D0 U9385 ( .A1(n9342), .A2(n9341), .B(n9340), .C(n9349)
     , .ZN(N4777) );
  AOI211D0 U9386 ( .A1(n9345), .A2(n9344), .B(n9349), .C(n9343)
     , .ZN(N4770) );
  AOI211D0 U9387 ( .A1(n9348), .A2(n9347), .B(n9349), .C(n9346)
     , .ZN(N4771) );
  AOI211D0 U9388 ( .A1(n9352), .A2(n9351), .B(n9350), .C(n9349)
     , .ZN(N4773) );
endmodule
```

18.2.2. Código del circuito con pads sintetizado

Cuadro 100. Archivo nanochip_mapped.v en el directorio `outputs_with_pads/`

```
////////////////////////////////////////////////////////////
// Created by: Synopsys DC Ultra(TM) in wire load mode
// Version   : W-2024.09-SP4
// Date      : Sun Nov  9 00:39:24 2025
////////////////////////////////////////////////////////////
```

```
module nanochip ( clk, reset_nanochip, select, reset_ring_osc,
   out_ring_osc,
        \q_out[7] , \q_out[6]_BAR , \q_out[5]_BAR , \q_out[4]
           _BAR ,
        \q_out[3]_BAR , \q_out[2]_BAR , \q_out[1]_BAR , \q_out
           [0]_BAR  );
  input [1:0] select;
  input clk, reset_nanochip, reset_ring_osc;
  output out_ring_osc, \q_out[7] , \q_out[6]_BAR , \q_out[5]
     _BAR ,
         \q_out[4]_BAR , \q_out[3]_BAR , \q_out[2]_BAR , \q_out
            [1]_BAR ,
         \q_out[0]_BAR ;
  wire   W_1, W_2, W_3, W_4, W_5, W_6, W_7, W_8, W_9, W_10,
     W_11, W_12, W_13,
         N4767, N4768, N4769, N4770, N4771, N4772, N4773, N4774
            , N4775, N4776,
         N4777, N4778, n4811, n4812, n4813, n4814, n4815, n4816
            , n4817, n4818,
         n1, n2, n3, n4, n5, n6, n7, n8, n9, n10, n11, n12, n13
            , n14, n15, n16,
         n17, n18, n19, n20, n21, n22, n23, n24, n25, n26, n27,
             n28, n29, n30,
         n31, n32, n33, n34, n35, n36, n37, n38, n39, n40, n41,
             n42, n43, n44,
         n45, n46, n47, n48, n49, n50, n51, n52, n53, n54, n55,
             n56, n57, n58,
         n59, n60, n61, n62, n63, n64, n65, n66, n67, n68, n69,
             n70, n71, n72,
         n73, n74, n75, n76, n77, n78, n79, n80, n81, n82, n83,
             n84, n85, n86,
         n87, n88, n89, n90, n91, n92, n93, n94, n95, n96, n97,
             n98, n99, n100,
         n101, n102, n103, n104, n105, n106, n107, n108, n109,
            n110, n111,
         n112, n113, n114, n115, n116, n117, n118, n119, n120,
            n121, n122,
         n123, n124, n125, n126, n127, n128, n129, n130, n131,
            n132, n133,
         n134, n135, n136, n137, n138, n139, n140, n141, n142,
            n143, n144,
         n145, n146, n147, n148, n149, n150, n151, n152, n153,
            n154, n155,
         n156, n157, n158, n159, n160, n161, n162, n163, n164,
            n165, n166,
```

```
        n167, n168, n169, n170, n171, n172, n173, n174, n175,
           n176, n177,
        n178, n179, n180, n181, n182, n183, n184, n185, n186,
           n187, n188,
        n189, n190, n191, n192, n193, n194, n195, n196, n197,
           n198, n199,
        n200, n201, n202, n203, n204, n205, n206, n207, n208,
           n209, n210,
        n211, n212, n213, n214, n215, n216, n217, n218, n219,
           n220, n221,
        n222, n223, n224, n225, n226, n227, n228, n229, n230,
           n231, n232,
        n233, n234, n235, n236, n237, n238, n239, n240, n241,
           n242, n243,
        n244, n245, n246, n247, n248, n249, n250, n251, n252,
           n253, n254,
        n255, n256, n257, n258, n259, n260, n261, n262, n263,
           n264, n265,
        n266, n267, n268, n269, n270, n271, n272, n273, n274,
           n275, n276,
        n277, n278, n279, n280, n281, n282, n283, n284, n285,
           n286, n287,
        n288, n289, n290, n291, n292, n293, n294, n295, n296,
           n297, n298,
        n299, n300, n301, n302, n303, n304, n305, n306, n307,
           n308, n309,
        n310, n311, n312, n313, n314, n315, n316, n317, n318,
           n319, n320,
        n321, n322, n323, n324, n325, n326, n327, n328, n329,
           n330, n331,
        n332, n333, n334, n335, n336, n337, n338, n339, n340,
           n341, n342,
        n343, n344, n345, n346, n347, n348, n349, n350, n351,
           n352, n353,
        n354, n355, n356, n357, n358, n359, n360, n361, n362,
           n363, n364,
        n365, n366, n367, n368, n369, n370, n371, n372, n373,
           n374, n375,
        n376, n377, n378, n379, n380, n381, n382, n383, n384,
           n385, n386,
        n387, n388, n389, n390, n391, n392, n393, n394, n395,
           n396, n397,
        n398, n399, n400, n401, n402, n403, n404, n405, n406,
           n407, n408,
        n409, n410, n411, n412, n413, n414, n415, n416, n417,
           n418, n419,
        n420, n421, n422, n423, n424, n425, n426, n427, n428,
           n429, n430,
```

```
        n431, n432, n433, n434, n435, n436, n437, n438, n439,
           n440, n441,
        n442, n443, n444, n445, n446, n447, n448, n449, n450,
           n451, n452,
        n453, n454, n455, n456, n457, n458, n459, n460, n461,
           n462, n463,
        n464, n465, n466, n467, n468, n469, n470, n471, n472,
           n473, n474,
        n475, n476, n477, n478, n479, n480, n481, n482, n483,
           n484, n485,
        n486, n487, n488, n489, n490, n491, n492, n493, n494,
           n495, n496,
        n497, n498, n499, n500, n501, n502, n503, n504, n505,
           n506, n507,
        n508, n509, n510, n511, n512, n513, n514, n515, n516,
           n517, n518,
        n519, n520, n521, n522, n523, n524, n525, n526, n527,
           n528, n529,
        n530, n531, n532, n533, n534, n535, n536, n537, n538,
           n539, n540,
        n541, n542, n543, n544, n545, n546, n547, n548, n549,
           n550, n551,
        n552, n553, n554, n555, n556, n557, n558, n559, n560,
           n561, n562,
        n563, n564, n565, n566, n567, n568, n569, n570, n571,
           n572, n573,
        n574, n575, n576, n577, n578, n579, n580, n581, n582,
           n583, n584,
        n585, n586, n587, n588, n589, n590, n591, n592, n593,
           n594, n595,
        n596, n597, n598, n599, n600, n601, n602, n603, n604,
           n605, n606,
        n607, n608, n609, n610, n611, n612, n613, n614, n615,
           n616, n617,
        n618, n619, n620, n621, n622, n623, n624, n625, n626,
           n627, n628,
        n629, n630, n631, n632, n633, n634, n635, n636, n637,
           n638, n639,
        n640, n641, n642, n643, n644, n645, n646, n647, n648,
           n649, n650,
        n651, n652, n653, n654, n655, n656, n657, n658, n659,
           n660, n661,
        n662, n663, n664, n665, n666, n667, n668, n669, n670,
           n671, n672,
        n673, n674, n675, n676, n677, n678, n679, n680, n681,
           n682, n683,
        n684, n685, n686, n687, n688, n689, n690, n691, n692,
           n693, n694,
```

```
        n695, n696, n697, n698, n699, n700, n701, n702, n703,
           n704, n705,
        n706, n707, n708, n709, n710, n711, n712, n713, n714,
           n715, n716,
        n717, n718, n719, n720, n721, n722, n723, n724, n725,
           n726, n727,
        n728, n729, n730, n731, n732, n733, n734, n735, n736,
           n737, n738,
        n739, n740, n741, n742, n743, n744, n745, n746, n747,
           n748, n749,
        n750, n751, n752, n753, n754, n755, n756, n757, n758,
           n759, n760,
        n761, n762, n763, n764, n765, n766, n767, n768, n769,
           n770, n771,
        n772, n773, n774, n775, n776, n777, n778, n779, n780,
           n781, n782,
        n783, n784, n785, n786, n787, n788, n789, n790, n791,
           n792, n793,
        n794, n795, n796, n797, n799, n800, n801, n802, n803,
           n804, n805,
        n806, n807, n808, n809, n810, n811, n812, n813, n814,
           n815, n816,
        n817, n818, n819, n820, n821, n822, n823, n824, n825,
           n826, n827,
        n828, n829, n830, n831, n832, n833, n834, n835, n836,
           n837, n838,
        n839, n840, n841, n842, n843, n844, n845, n846, n847,
           n848, n849,
        n850, n851, n852, n853, n854, n855, n856, n857, n858,
           n859, n860,
        n861, n862, n863, n864, n865, n866, n867, n868, n869,
           n870, n871,
        n872, n873, n874, n875, n876, n877, n878, n879, n880,
           n881, n882,
        n883, n884, n885, n886, n887, n888, n889, n890, n891,
           n892, n893,
        n894, n895, n896, n897, n898, n899, n900, n901, n902,
           n903, n904,
        n905, n906, n907, n908, n909, n910, n911, n912, n913,
           n914, n915,
        n916, n917, n918, n919, n920, n921, n922, n923, n924,
           n925, n926,
        n927, n928, n929, n930, n931, n932, n933, n934, n935,
           n936, n937,
        n938, n939, n940, n941, n942, n943, n944, n945, n946,
           n947, n948,
        n949, n950, n951, n952, n953, n954, n955, n956, n957,
           n958, n959,
```

```
        n960, n961, n962, n963, n964, n965, n966, n967, n968,
           n969, n970,
        n971, n972, n973, n974, n975, n976, n977, n978, n979,
           n980, n981,
        n982, n983, n984, n985, n986, n987, n988, n989, n990,
           n991, n992,
        n993, n994, n995, n996, n997, n998, n999, n1000, n1001
           , n1002, n1003,
        n1004, n1005, n1006, n1007, n1008, n1009, n1010, n1011
           , n1012, n1013,
        n1014, n1015, n1016, n1017, n1018, n1019, n1020, n1021
           , n1022, n1023,
        n1024, n1025, n1026, n1027, n1028, n1029, n1030, n1031
           , n1032, n1033,
        n1034, n1035, n1036, n1037, n1038, n1039, n1040, n1041
           , n1042, n1043,
        n1044, n1045, n1046, n1047, n1048, n1049, n1050, n1051
           , n1052, n1053,
        n1054, n1055, n1056, n1057, n1058, n1059, n1060, n1061
           , n1062, n1063,
        n1064, n1065, n1066, n1067, n1068, n1069, n1070, n1071
           , n1072, n1073,
        n1074, n1075, n1076, n1077, n1078, n1079, n1080, n1081
           , n1082, n1083,
        n1084, n1085, n1086, n1087, n1088, n1089, n1090, n1091
           , n1092, n1093,
        n1094, n1095, n1096, n1097, n1098, n1099, n1100, n1101
           , n1102, n1103,
        n1104, n1105, n1106, n1107, n1108, n1109, n1110, n1111
           , n1112, n1113,
        n1114, n1115, n1116, n1117, n1118, n1119, n1120, n1121
           , n1122, n1123,
        n1124, n1125, n1126, n1127, n1128, n1129, n1130, n1131
           , n1132, n1133,
        n1134, n1135, n1136, n1137, n1138, n1139, n1140, n1141
           , n1142, n1143,
        n1144, n1145, n1146, n1147, n1148, n1149, n1150, n1151
           , n1152, n1153,
        n1154, n1155, n1156, n1157, n1158, n1159, n1160, n1161
           , n1162, n1163,
        n1164, n1165, n1166, n1167, n1168, n1169, n1170, n1171
           , n1172, n1173,
        n1174, n1175, n1176, n1177, n1178, n1179, n1180, n1181
           , n1182, n1183,
        n1184, n1185, n1186, n1187, n1188, n1189, n1190, n1191
           , n1192, n1193,
        n1194, n1195, n1196, n1197, n1198, n1199, n1200, n1201
           , n1202, n1203,
```

```
        n1204, n1205, n1206, n1207, n1208, n1209, n1210, n1211
            , n1212, n1213,
        n1214, n1215, n1216, n1217, n1218, n1219, n1220, n1221
            , n1222, n1223,
        n1224, n1225, n1226, n1227, n1228, n1229, n1230, n1231
            , n1232, n1233,
        n1234, n1235, n1236, n1237, n1238, n1239, n1240, n1241
            , n1242, n1243,
        n1244, n1245, n1246, n1247, n1248, n1249, n1250, n1251
            , n1252, n1253,
        n1254, n1255, n1256, n1257, n1258, n1259, n1260, n1261
            , n1262, n1263,
        n1264, n1265, n1266, n1267, n1268, n1269, n1270, n1271
            , n1272, n1273,
        n1274, n1275, n1276, n1277, n1278, n1279, n1280, n1281
            , n1282, n1283,
        n1284, n1285, n1286, n1287, n1288, n1289, n1290, n1291
            , n1292, n1293,
        n1294, n1295, n1296, n1297, n1298, n1299, n1300, n1301
            , n1302, n1303,
        n1304, n1305, n1306, n1307, n1308, n1309, n1310, n1311
            , n1312, n1313,
        n1314, n1315, n1316, n1317, n1318, n1319, n1320, n1321
            , n1322, n1323,
        n1324, n1325, n1326, n1327, n1328, n1329, n1330, n1331
            , n1332, n1333,
        n1334, n1335, n1336, n1337, n1338, n1339, n1340, n1341
            , n1342, n1343,
        n1345, n1346, n1347, n1348, n1349, n1350, n1351, n1352
            , n1353, n1354,
        n1355, n1356, n1357, n1358, n1359, n1360, n1361, n1362
            , n1363, n1364,
        n1365, n1366, n1367, n1368, n1369, n1370, n1371, n1372
            , n1373, n1374,
        n1375, n1376, n1377, n1378, n1379, n1380, n1381, n1382
            , n1383, n1384,
        n1385, n1386, n1387, n1388, n1389, n1390, n1391, n1392
            , n1393, n1394,
        n1395, n1396, n1397, n1398, n1399, n1400, n1401, n1402
            , n1403, n1404,
        n1405, n1406, n1407, n1408, n1409, n1410, n1411, n1412
            , n1413, n1414,
        n1415, n1416, n1417, n1418, n1419, n1420, n1421, n1422
            , n1423, n1424,
        n1425, n1426, n1427, n1428, n1429, n1430, n1431, n1432
            , n1433, n1434,
        n1435, n1436, n1437, n1438, n1439, n1440, n1441, n1442
            , n1443, n1444,
```

```
        n1445, n1446, n1447, n1448, n1449, n1450, n1451, n1452
            , n1453, n1454,
        n1455, n1456, n1457, n1458, n1459, n1460, n1461, n1462
            , n1463, n1464,
        n1465, n1466, n1467, n1468, n1469, n1470, n1471, n1472
            , n1473, n1474,
        n1475, n1476, n1477, n1478, n1479, n1480, n1481, n1482
            , n1483, n1484,
        n1485, n1486, n1487, n1488, n1489, n1490, n1491, n1492
            , n1493, n1494,
        n1495, n1496, n1497, n1498, n1499, n1500, n1501, n1502
            , n1503, n1504,
        n1505, n1506, n1507, n1508, n1509, n1510, n1511, n1512
            , n1513, n1514,
        n1515, n1516, n1517, n1518, n1519, n1520, n1521, n1522
            , n1523, n1524,
        n1525, n1526, n1527, n1528, n1529, n1530, n1531, n1532
            , n1533, n1534,
        n1535, n1536, n1537, n1538, n1539, n1540, n1541, n1542
            , n1543, n1544,
        n1545, n1546, n1547, n1548, n1549, n1550, n1551, n1552
            , n1553, n1554,
        n1555, n1556, n1557, n1558, n1559, n1560, n1561, n1562
            , n1563, n1564,
        n1565, n1566, n1567, n1568, n1569, n1570, n1571, n1572
            , n1573, n1574,
        n1575, n1576, n1577, n1578, n1579, n1580, n1581, n1582
            , n1583, n1584,
        n1585, n1586, n1587, n1588, n1589, n1590, n1591, n1592
            , n1593, n1594,
        n1595, n1596, n1597, n1598, n1599, n1600, n1601, n1602
            , n1603, n1604,
        n1605, n1606, n1607, n1608, n1609, n1610, n1611, n1612
            , n1613, n1614,
        n1615, n1616, n1617, n1618, n1619, n1620, n1621, n1622
            , n1623, n1624,
        n1625, n1626, n1627, n1628, n1629, n1630, n1631, n1632
            , n1633, n1634,
        n1635, n1636, n1637, n1638, n1639, n1640, n1641, n1642
            , n1643, n1644,
        n1645, n1646, n1647, n1648, n1649, n1650, n1651, n1652
            , n1653, n1654,
        n1655, n1656, n1657, n1658, n1659, n1660, n1661, n1662
            , n1663, n1664,
        n1665, n1666, n1667, n1668, n1669, n1670, n1671, n1672
            , n1673, n1674,
        n1675, n1676, n1677, n1678, n1679, n1680, n1681, n1682
            , n1683, n1684,
```

```
        n1685, n1686, n1687, n1688, n1689, n1690, n1691, n1692
           , n1693, n1694,
        n1695, n1696, n1697, n1698, n1699, n1700, n1701, n1702
           , n1703, n1704,
        n1705, n1706, n1707, n1708, n1709, n1710, n1711, n1712
           , n1713, n1714,
        n1715, n1716, n1717, n1718, n1719, n1720, n1721, n1722
           , n1723, n1724,
        n1725, n1726, n1727, n1728, n1729, n1730, n1731, n1732
           , n1733, n1734,
        n1735, n1736, n1737, n1738, n1739, n1740, n1741, n1742
           , n1743, n1744,
        n1745, n1746, n1747, n1748, n1749, n1750, n1751, n1752
           , n1753, n1754,
        n1755, n1756, n1757, n1758, n1759, n1760, n1761, n1762
           , n1763, n1764,
        n1765, n1766, n1767, n1768, n1769, n1770, n1771, n1772
           , n1773, n1774,
        n1775, n1776, n1777, n1778, n1779, n1780, n1781, n1782
           , n1783, n1784,
        n1785, n1786, n1787, n1788, n1789, n1790, n1791, n1792
           , n1793, n1794,
        n1795, n1796, n1797, n1798, n1799, n1800, n1801, n1802
           , n1803, n1804,
        n1805, n1806, n1807, n1808, n1809, n1810, n1811, n1812
           , n1813, n1814,
        n1815, n1816, n1817, n1818, n1819, n1820, n1821, n1822
           , n1823, n1824,
        n1825, n1826, n1827, n1828, n1829, n1830, n1831, n1832
           , n1833, n1834,
        n1835, n1836, n1837, n1838, n1839, n1840, n1841, n1842
           , n1843, n1844,
        n1845, n1846, n1847, n1848, n1849, n1850, n1851, n1852
           , n1853, n1854,
        n1855, n1856, n1857, n1858, n1859, n1860, n1861, n1862
           , n1863, n1864,
        n1865, n1866, n1867, n1868, n1869, n1870, n1871, n1872
           , n1873, n1874,
        n1875, n1876, n1877, n1878, n1879, n1880, n1881, n1882
           , n1883, n1884,
        n1885, n1886, n1887, n1888, n1889, n1890, n1891, n1892
           , n1893, n1894,
        n1895, n1896, n1897, n1898, n1899, n1900, n1901, n1902
           , n1903, n1904,
        n1905, n1906, n1907, n1908, n1909, n1910, n1911, n1912
           , n1913, n1914,
        n1915, n1916, n1917, n1918, n1919, n1920, n1921, n1922
           , n1923, n1924,
```

```
        n1925, n1926, n1927, n1928, n1929, n1930, n1931, n1932
            , n1933, n1934,
        n1935, n1936, n1937, n1938, n1939, n1940, n1941, n1942
            , n1943, n1944,
        n1945, n1946, n1947, n1948, n1949, n1950, n1951, n1952
            , n1953, n1954,
        n1955, n1956, n1957, n1958, n1959, n1960, n1961, n1962
            , n1963, n1964,
        n1965, n1966, n1967, n1968, n1969, n1970, n1971, n1972
            , n1973, n1974,
        n1975, n1976, n1977, n1978, n1979, n1980, n1981, n1982
            , n1983, n1984,
        n1985, n1986, n1987, n1988, n1989, n1990, n1991, n1992
            , n1993, n1994,
        n1995, n1996, n1997, n1998, n1999, n2000, n2001, n2002
            , n2003, n2004,
        n2005, n2006, n2007, n2008, n2009, n2010, n2011, n2012
            , n2014, n2015,
        n2016, n2017, n2018, n2019, n2020, n2021, n2022, n2023
            , n2024, n2025,
        n2026, n2027, n2028, n2029, n2030, n2031, n2032, n2033
            , n2034, n2035,
        n2036, n2037, n2038, n2039, n2040, n2041, n2042, n2043
            , n2044, n2045,
        n2046, n2047, n2048, n2049, n2050, n2051, n2052, n2053
            , n2054, n2055,
        n2056, n2057, n2058, n2059, n2060, n2061, n2062, n2063
            , n2064, n2065,
        n2066, n2067, n2068, n2069, n2070, n2071, n2072, n2073
            , n2074, n2075,
        n2076, n2077, n2078, n2079, n2080, n2081, n2082, n2083
            , n2084, n2085,
        n2086, n2087, n2088, n2089, n2090, n2091, n2092, n2093
            , n2094, n2095,
        n2096, n2097, n2098, n2099, n2100, n2101, n2102, n2103
            , n2104, n2105,
        n2106, n2107, n2108, n2109, n2110, n2111, n2112, n2113
            , n2114, n2115,
        n2116, n2117, n2118, n2119, n2120, n2121, n2122, n2123
            , n2124, n2125,
        n2126, n2127, n2128, n2129, n2130, n2131, n2132, n2133
            , n2134, n2135,
        n2136, n2137, n2138, n2139, n2140, n2141, n2142, n2143
            , n2144, n2145,
        n2146, n2147, n2148, n2149, n2150, n2151, n2152, n2153
            , n2154, n2155,
        n2156, n2157, n2158, n2159, n2160, n2161, n2162, n2163
            , n2164, n2165,
```

```
        n2166, n2167, n2168, n2169, n2170, n2171, n2172, n2173
            , n2174, n2175,
        n2176, n2177, n2178, n2179, n2180, n2181, n2182, n2183
            , n2184, n2185,
        n2186, n2187, n2188, n2189, n2190, n2191, n2192, n2193
            , n2194, n2195,
        n2196, n2197, n2198, n2199, n2200, n2201, n2202, n2203
            , n2204, n2205,
        n2206, n2207, n2208, n2209, n2210, n2211, n2212, n2213
            , n2214, n2215,
        n2216, n2217, n2218, n2219, n2220, n2221, n2222, n2223
            , n2224, n2225,
        n2226, n2227, n2228, n2229, n2230, n2231, n2232, n2233
            , n2234, n2235,
        n2236, n2237, n2238, n2239, n2240, n2241, n2242, n2243
            , n2244, n2245,
        n2246, n2247, n2248, n2249, n2250, n2251, n2252, n2253
            , n2254, n2255,
        n2256, n2257, n2258, n2259, n2260, n2261, n2262, n2263
            , n2264, n2265,
        n2266, n2267, n2268, n2269, n2270, n2271, n2272, n2273
            , n2274, n2275,
        n2276, n2277, n2278, n2279, n2280, n2281, n2282, n2283
            , n2284, n2285,
        n2286, n2287, n2288, n2289, n2290, n2291, n2292, n2293
            , n2294, n2295,
        n2296, n2297, n2298, n2299, n2300, n2301, n2302, n2303
            , n2304, n2305,
        n2306, n2307, n2308, n2309, n2310, n2311, n2312, n2313
            , n2314, n2315,
        n2316, n2317, n2318, n2319, n2320, n2321, n2322, n2323
            , n2324, n2325,
        n2326, n2327, n2328, n2329, n2330, n2331, n2332, n2333
            , n2334, n2335,
        n2336, n2337, n2338, n2339, n2340, n2341, n2342, n2343
            , n2344, n2345,
        n2346, n2347, n2348, n2349, n2350, n2351, n2352, n2353
            , n2354, n2355,
        n2356, n2357, n2358, n2359, n2360, n2361, n2362, n2363
            , n2364, n2365,
        n2366, n2367, n2368, n2369, n2370, n2371, n2372, n2373
            , n2374, n2375,
        n2376, n2377, n2378, n2379, n2380, n2381, n2382, n2383
            , n2384, n2385,
        n2386, n2387, n2388, n2389, n2390, n2391, n2392, n2393
            , n2394, n2395,
        n2396, n2397, n2398, n2399, n2400, n2401, n2402, n2403
            , n2404, n2405,
```

```
        n2406, n2407, n2408, n2409, n2410, n2411, n2412, n2413
            , n2414, n2415,
        n2416, n2417, n2418, n2419, n2420, n2421, n2422, n2423
            , n2424, n2425,
        n2426, n2427, n2428, n2429, n2430, n2431, n2432, n2433
            , n2434, n2435,
        n2436, n2437, n2438, n2439, n2440, n2441, n2442, n2443
            , n2444, n2445,
        n2446, n2447, n2448, n2449, n2450, n2451, n2452, n2453
            , n2454, n2455,
        n2456, n2457, n2458, n2459, n2460, n2461, n2462, n2463
            , n2464, n2465,
        n2466, n2467, n2468, n2469, n2470, n2471, n2472, n2473
            , n2474, n2475,
        n2476, n2477, n2478, n2479, n2480, n2481, n2482, n2483
            , n2484, n2485,
        n2486, n2487, n2488, n2489, n2490, n2491, n2492, n2493
            , n2494, n2495,
        n2496, n2497, n2498, n2499, n2500, n2501, n2502, n2503
            , n2504, n2505,
        n2506, n2507, n2508, n2509, n2510, n2511, n2512, n2513
            , n2514, n2515,
        n2516, n2517, n2518, n2519, n2520, n2521, n2522, n2523
            , n2524, n2525,
        n2526, n2527, n2528, n2529, n2530, n2531, n2532, n2533
            , n2534, n2535,
        n2536, n2537, n2538, n2539, n2540, n2541, n2542, n2543
            , n2544, n2545,
        n2546, n2547, n2548, n2549, n2550, n2551, n2552, n2553
            , n2554, n2555,
        n2556, n2557, n2558, n2559, n2560, n2561, n2562, n2563
            , n2564, n2565,
        n2566, n2567, n2568, n2569, n2570, n2571, n2572, n2573
            , n2574, n2575,
        n2576, n2577, n2578, n2579, n2580, n2581, n2582, n2583
            , n2584, n2585,
        n2586, n2587, n2588, n2589, n2590, n2591, n2592, n2593
            , n2594, n2595,
        n2596, n2597, n2598, n2599, n2600, n2601, n2602, n2603
            , n2604, n2605,
        n2606, n2607, n2608, n2609, n2610, n2611, n2612, n2613
            , n2614, n2615,
        n2616, n2617, n2618, n2619, n2620, n2621, n2622, n2623
            , n2624, n2625,
        n2626, n2627, n2628, n2629, n2630, n2631, n2632, n2633
            , n2634, n2635,
        n2636, n2637, n2638, n2639, n2640, n2641, n2642, n2643
            , n2644, n2645,
```

```
        n2646, n2647, n2648, n2649, n2650, n2651, n2652, n2653
            , n2654, n2655,
        n2656, n2657, n2658, n2659, n2660, n2661, n2662, n2663
            , n2664, n2665,
        n2666, n2667, n2668, n2669, n2670, n2671, n2672, n2673
            , n2674, n2675,
        n2676, n2677, n2678, n2679, n2680, n2681, n2682, n2683
            , n2684, n2685,
        n2686, n2687, n2688, n2689, n2690, n2691, n2692, n2693
            , n2694, n2695,
        n2696, n2697, n2698, n2699, n2700, n2701, n2702, n2703
            , n2704, n2705,
        n2706, n2707, n2708, n2709, n2710, n2711, n2712, n2713
            , n2714, n2715,
        n2716, n2717, n2718, n2719, n2720, n2721, n2722, n2723
            , n2724, n2725,
        n2726, n2727, n2728, n2729, n2731, n2732, n2733, n2734
            , n2735, n2736,
        n2737, n2738, n2739, n2740, n2741, n2742, n2743, n2744
            , n2745, n2746,
        n2747, n2748, n2749, n2750, n2751, n2752, n2753, n2754
            , n2755, n2756,
        n2757, n2758, n2759, n2760, n2761, n2762, n2763, n2764
            , n2765, n2766,
        n2767, n2768, n2769, n2770, n2771, n2772, n2773, n2774
            , n2775, n2776,
        n2777, n2778, n2779, n2780, n2781, n2782, n2783, n2784
            , n2785, n2786,
        n2787, n2788, n2789, n2790, n2791, n2792, n2793, n2794
            , n2795, n2796,
        n2797, n2798, n2799, n2800, n2801, n2802, n2803, n2804
            , n2805, n2806,
        n2807, n2808, n2809, n2810, n2811, n2812, n2813, n2814
            , n2815, n2816,
        n2817, n2818, n2819, n2820, n2821, n2822, n2823, n2824
            , n2825, n2826,
        n2827, n2828, n2829, n2830, n2831, n2832, n2833, n2834
            , n2835, n2836,
        n2837, n2838, n2839, n2840, n2841, n2842, n2843, n2844
            , n2845, n2846,
        n2847, n2848, n2849, n2850, n2851, n2852, n2853, n2854
            , n2855, n2856,
        n2857, n2858, n2859, n2860, n2861, n2862, n2863, n2864
            , n2865, n2866,
        n2867, n2868, n2869, n2870, n2871, n2872, n2873, n2874
            , n2875, n2876,
        n2877, n2878, n2879, n2880, n2881, n2882, n2883, n2884
            , n2885, n2886,
```

```
        n2887, n2888, n2889, n2890, n2891, n2892, n2893, n2894
            , n2895, n2896,
        n2897, n2898, n2899, n2900, n2901, n2902, n2903, n2904
            , n2905, n2906,
        n2907, n2908, n2909, n2910, n2911, n2912, n2913, n2914
            , n2915, n2916,
        n2917, n2918, n2919, n2920, n2921, n2922, n2923, n2924
            , n2925, n2926,
        n2927, n2928, n2929, n2930, n2931, n2932, n2933, n2934
            , n2935, n2936,
        n2937, n2938, n2939, n2940, n2941, n2942, n2943, n2944
            , n2945, n2946,
        n2947, n2948, n2949, n2950, n2951, n2952, n2953, n2954
            , n2955, n2956,
        n2957, n2958, n2959, n2960, n2961, n2962, n2963, n2964
            , n2965, n2966,
        n2967, n2968, n2969, n2970, n2971, n2972, n2973, n2974
            , n2975, n2976,
        n2977, n2978, n2979, n2980, n2981, n2982, n2983, n2984
            , n2985, n2986,
        n2987, n2988, n2989, n2990, n2991, n2992, n2993, n2994
            , n2995, n2996,
        n2997, n2998, n2999, n3000, n3001, n3002, n3003, n3004
            , n3005, n3006,
        n3007, n3008, n3009, n3010, n3011, n3012, n3013, n3014
            , n3015, n3016,
        n3017, n3018, n3019, n3020, n3021, n3022, n3023, n3024
            , n3025, n3026,
        n3027, n3028, n3029, n3030, n3031, n3032, n3033, n3034
            , n3035, n3036,
        n3037, n3038, n3039, n3040, n3041, n3042, n3043, n3044
            , n3045, n3046,
        n3047, n3048, n3049, n3050, n3051, n3052, n3053, n3054
            , n3055, n3056,
        n3057, n3058, n3059, n3060, n3061, n3062, n3063, n3064
            , n3065, n3066,
        n3067, n3068, n3069, n3070, n3071, n3072, n3073, n3074
            , n3075, n3076,
        n3077, n3078, n3079, n3080, n3081, n3082, n3083, n3084
            , n3085, n3086,
        n3087, n3088, n3089, n3090, n3091, n3092, n3093, n3094
            , n3095, n3096,
        n3097, n3098, n3099, n3100, n3101, n3102, n3103, n3104
            , n3105, n3106,
        n3107, n3108, n3109, n3110, n3111, n3112, n3113, n3114
            , n3115, n3116,
        n3117, n3118, n3119, n3120, n3121, n3122, n3123, n3124
            , n3125, n3126,
```

```
        n3127, n3128, n3129, n3130, n3131, n3132, n3133, n3134
           , n3135, n3136,
        n3137, n3138, n3139, n3140, n3141, n3142, n3143, n3144
           , n3145, n3146,
        n3147, n3148, n3149, n3150, n3151, n3152, n3153, n3154
           , n3155, n3156,
        n3157, n3158, n3159, n3160, n3161, n3162, n3163, n3164
           , n3165, n3166,
        n3167, n3168, n3169, n3170, n3171, n3172, n3173, n3174
           , n3175, n3176,
        n3177, n3178, n3179, n3180, n3181, n3182, n3183, n3184
           , n3185, n3186,
        n3187, n3188, n3189, n3190, n3191, n3192, n3193, n3194
           , n3195, n3196,
        n3197, n3198, n3199, n3200, n3201, n3202, n3203, n3204
           , n3205, n3206,
        n3207, n3208, n3209, n3210, n3211, n3212, n3213, n3214
           , n3215, n3216,
        n3217, n3218, n3219, n3220, n3221, n3222, n3223, n3224
           , n3225, n3226,
        n3227, n3228, n3229, n3230, n3231, n3232, n3233, n3234
           , n3235, n3236,
        n3237, n3238, n3239, n3240, n3241, n3242, n3243, n3244
           , n3245, n3246,
        n3247, n3248, n3249, n3250, n3251, n3252, n3253, n3254
           , n3255, n3256,
        n3257, n3258, n3259, n3260, n3261, n3262, n3263, n3264
           , n3265, n3266,
        n3267, n3268, n3269, n3270, n3271, n3272, n3273, n3274
           , n3275, n3276,
        n3277, n3278, n3279, n3280, n3281, n3282, n3283, n3284
           , n3285, n3286,
        n3287, n3288, n3289, n3290, n3291, n3292, n3293, n3294
           , n3295, n3296,
        n3297, n3298, n3299, n3300, n3301, n3302, n3303, n3304
           , n3305, n3306,
        n3307, n3308, n3309, n3310, n3311, n3312, n3313, n3314
           , n3315, n3316,
        n3317, n3318, n3319, n3320, n3321, n3322, n3323, n3324
           , n3325, n3326,
        n3327, n3328, n3329, n3330, n3331, n3332, n3333, n3334
           , n3335, n3336,
        n3337, n3338, n3339, n3340, n3341, n3342, n3343, n3344
           , n3345, n3346,
        n3347, n3348, n3349, n3350, n3351, n3352, n3353, n3354
           , n3355, n3356,
        n3357, n3358, n3359, n3360, n3361, n3362, n3363, n3365
           , n3366, n3367,
```

```
        n3368, n3369, n3370, n3371, n3372, n3373, n3374, n3375
           , n3376, n3377,
        n3378, n3379, n3380, n3381, n3382, n3383, n3384, n3385
           , n3386, n3387,
        n3388, n3389, n3390, n3391, n3392, n3393, n3394, n3395
           , n3396, n3397,
        n3398, n3399, n3400, n3401, n3402, n3403, n3404, n3405
           , n3406, n3407,
        n3408, n3409, n3410, n3411, n3412, n3413, n3414, n3415
           , n3416, n3417,
        n3418, n3419, n3420, n3421, n3422, n3423, n3424, n3425
           , n3426, n3427,
        n3428, n3429, n3430, n3431, n3432, n3433, n3434, n3435
           , n3436, n3437,
        n3438, n3439, n3440, n3441, n3442, n3443, n3444, n3445
           , n3446, n3447,
        n3448, n3449, n3450, n3451, n3452, n3453, n3454, n3455
           , n3456, n3457,
        n3458, n3459, n3460, n3461, n3462, n3463, n3464, n3465
           , n3466, n3467,
        n3468, n3469, n3470, n3471, n3472, n3473, n3474, n3475
           , n3476, n3477,
        n3478, n3479, n3480, n3481, n3482, n3483, n3484, n3485
           , n3486, n3487,
        n3488, n3489, n3490, n3491, n3492, n3493, n3494, n3495
           , n3496, n3497,
        n3498, n3499, n3500, n3501, n3502, n3503, n3504, n3505
           , n3506, n3507,
        n3508, n3509, n3510, n3511, n3512, n3513, n3514, n3515
           , n3516, n3517,
        n3518, n3519, n3520, n3521, n3522, n3523, n3524, n3525
           , n3526, n3527,
        n3528, n3529, n3530, n3531, n3532, n3533, n3534, n3535
           , n3536, n3537,
        n3538, n3539, n3540, n3541, n3542, n3543, n3544, n3545
           , n3546, n3547,
        n3548, n3549, n3550, n3551, n3552, n3553, n3554, n3555
           , n3556, n3557,
        n3558, n3559, n3560, n3561, n3562, n3563, n3564, n3565
           , n3566, n3567,
        n3568, n3569, n3570, n3571, n3572, n3573, n3574, n3575
           , n3576, n3577,
        n3578, n3579, n3580, n3581, n3582, n3583, n3584, n3585
           , n3586, n3587,
        n3588, n3589, n3590, n3591, n3592, n3593, n3594, n3595
           , n3596, n3597,
        n3598, n3599, n3600, n3601, n3602, n3603, n3604, n3605
           , n3606, n3607,
```

```
        n3608, n3609, n3610, n3611, n3612, n3613, n3614, n3615
            , n3616, n3617,
        n3618, n3619, n3620, n3621, n3622, n3623, n3624, n3625
            , n3626, n3627,
        n3628, n3629, n3630, n3631, n3632, n3633, n3634, n3635
            , n3636, n3637,
        n3638, n3639, n3640, n3641, n3642, n3643, n3644, n3645
            , n3646, n3647,
        n3648, n3649, n3650, n3651, n3652, n3653, n3654, n3655
            , n3656, n3657,
        n3658, n3659, n3660, n3661, n3662, n3663, n3664, n3665
            , n3666, n3667,
        n3668, n3669, n3670, n3671, n3672, n3673, n3674, n3675
            , n3676, n3677,
        n3678, n3679, n3680, n3681, n3682, n3683, n3684, n3685
            , n3686, n3687,
        n3688, n3689, n3690, n3691, n3692, n3693, n3694, n3695
            , n3696, n3697,
        n3698, n3699, n3700, n3701, n3702, n3703, n3704, n3705
            , n3706, n3707,
        n3708, n3709, n3710, n3711, n3712, n3713, n3714, n3715
            , n3716, n3717,
        n3718, n3719, n3720, n3721, n3722, n3723, n3724, n3725
            , n3726, n3727,
        n3728, n3729, n3730, n3731, n3732, n3733, n3734, n3735
            , n3736, n3737,
        n3738, n3739, n3740, n3741, n3742, n3743, n3744, n3745
            , n3746, n3747,
        n3748, n3749, n3750, n3751, n3752, n3753, n3754, n3755
            , n3756, n3757,
        n3758, n3759, n3760, n3761, n3762, n3763, n3764, n3765
            , n3766, n3767,
        n3768, n3769, n3770, n3771, n3772, n3773, n3774, n3775
            , n3776, n3777,
        n3778, n3779, n3780, n3781, n3782, n3783, n3784, n3785
            , n3786, n3787,
        n3788, n3789, n3790, n3791, n3792, n3793, n3794, n3795
            , n3796, n3797,
        n3798, n3799, n3800, n3801, n3802, n3803, n3804, n3805
            , n3806, n3807,
        n3808, n3809, n3810, n3811, n3812, n3813, n3814, n3815
            , n3816, n3817,
        n3818, n3819, n3820, n3821, n3822, n3823, n3824, n3825
            , n3826, n3827,
        n3828, n3829, n3830, n3831, n3832, n3833, n3834, n3835
            , n3836, n3837,
        n3838, n3839, n3840, n3841, n3842, n3843, n3844, n3845
            , n3846, n3847,
```

```
        n3848, n3849, n3850, n3851, n3852, n3853, n3854, n3855
            , n3856, n3857,
        n3858, n3859, n3860, n3861, n3862, n3863, n3864, n3865
            , n3866, n3867,
        n3868, n3869, n3870, n3871, n3872, n3873, n3874, n3875
            , n3876, n3877,
        n3878, n3879, n3880, n3881, n3882, n3883, n3884, n3885
            , n3886, n3887,
        n3888, n3889, n3890, n3891, n3892, n3893, n3894, n3895
            , n3896, n3897,
        n3898, n3899, n3900, n3901, n3902, n3903, n3904, n3905
            , n3906, n3907,
        n3908, n3909, n3910, n3911, n3912, n3913, n3914, n3915
            , n3916, n3917,
        n3918, n3919, n3920, n3921, n3922, n3923, n3924, n3925
            , n3927, n3928,
        n3929, n3930, n3931, n3932, n3933, n3934, n3935, n3936
            , n3937, n3938,
        n3939, n3940, n3941, n3942, n3943, n3944, n3945, n3946
            , n3947, n3948,
        n3949, n3950, n3951, n3952, n3953, n3954, n3955, n3956
            , n3957, n3958,
        n3959, n3960, n3961, n3962, n3963, n3964, n3965, n3966
            , n3967, n3968,
        n3969, n3970, n3971, n3972, n3973, n3974, n3975, n3976
            , n3977, n3978,
        n3979, n3980, n3981, n3982, n3983, n3984, n3985, n3986
            , n3987, n3988,
        n3989, n3990, n3991, n3992, n3993, n3994, n3995, n3996
            , n3997, n3998,
        n3999, n4000, n4001, n4002, n4003, n4004, n4005, n4006
            , n4007, n4008,
        n4009, n4010, n4011, n4012, n4013, n4014, n4015, n4016
            , n4017, n4018,
        n4019, n4020, n4021, n4022, n4023, n4024, n4025, n4026
            , n4027, n4028,
        n4029, n4030, n4031, n4032, n4033, n4034, n4035, n4036
            , n4037, n4038,
        n4039, n4040, n4041, n4042, n4043, n4044, n4045, n4046
            , n4047, n4048,
        n4049, n4050, n4051, n4052, n4053, n4054, n4055, n4056
            , n4057, n4058,
        n4059, n4060, n4061, n4062, n4063, n4064, n4065, n4066
            , n4067, n4068,
        n4069, n4070, n4071, n4072, n4073, n4074, n4075, n4076
            , n4077, n4078,
        n4079, n4080, n4081, n4082, n4083, n4084, n4085, n4086
            , n4087, n4088,
```

```
        n4089, n4090, n4091, n4092, n4093, n4094, n4095, n4096
            , n4097, n4098,
        n4099, n4100, n4101, n4102, n4103, n4104, n4105, n4106
            , n4107, n4108,
        n4109, n4110, n4111, n4112, n4113, n4114, n4115, n4116
            , n4117, n4118,
        n4119, n4120, n4121, n4122, n4123, n4124, n4125, n4126
            , n4127, n4128,
        n4129, n4130, n4131, n4132, n4133, n4134, n4135, n4136
            , n4137, n4138,
        n4139, n4140, n4141, n4142, n4143, n4144, n4145, n4146
            , n4147, n4148,
        n4149, n4150, n4151, n4152, n4153, n4154, n4155, n4156
            , n4157, n4158,
        n4159, n4160, n4161, n4162, n4163, n4164, n4165, n4166
            , n4167, n4168,
        n4169, n4170, n4171, n4172, n4173, n4174, n4175, n4176
            , n4177, n4178,
        n4179, n4180, n4181, n4182, n4183, n4184, n4185, n4186
            , n4187, n4188,
        n4189, n4190, n4191, n4192, n4193, n4194, n4195, n4196
            , n4197, n4198,
        n4199, n4200, n4201, n4202, n4203, n4204, n4205, n4206
            , n4207, n4208,
        n4209, n4210, n4211, n4212, n4213, n4214, n4215, n4216
            , n4217, n4218,
        n4219, n4220, n4221, n4222, n4223, n4224, n4225, n4226
            , n4227, n4228,
        n4229, n4230, n4231, n4232, n4233, n4234, n4235, n4236
            , n4237, n4238,
        n4239, n4240, n4241, n4242, n4243, n4244, n4245, n4246
            , n4247, n4248,
        n4249, n4250, n4251, n4252, n4253, n4254, n4255, n4256
            , n4257, n4258,
        n4259, n4260, n4261, n4262, n4263, n4264, n4265, n4266
            , n4267, n4268,
        n4269, n4270, n4271, n4272, n4273, n4274, n4275, n4276
            , n4277, n4278,
        n4279, n4280, n4281, n4282, n4283, n4284, n4285, n4286
            , n4287, n4288,
        n4289, n4290, n4291, n4292, n4293, n4294, n4295, n4296
            , n4297, n4298,
        n4299, n4300, n4301, n4302, n4303, n4304, n4305, n4306
            , n4307, n4308,
        n4309, n4310, n4311, n4312, n4313, n4314, n4315, n4316
            , n4317, n4318,
        n4319, n4320, n4321, n4322, n4323, n4324, n4325, n4326
            , n4327, n4328,
```

```
        n4329, n4330, n4331, n4332, n4333, n4334, n4335, n4336
            , n4337, n4338,
        n4339, n4340, n4341, n4342, n4343, n4344, n4345, n4346
            , n4347, n4348,
        n4349, n4350, n4351, n4352, n4353, n4354, n4355, n4356
            , n4357, n4358,
        n4359, n4360, n4361, n4362, n4363, n4364, n4365, n4366
            , n4367, n4368,
        n4369, n4370, n4371, n4372, n4373, n4374, n4375, n4376
            , n4377, n4378,
        n4379, n4380, n4381, n4382, n4383, n4384, n4385, n4386
            , n4387, n4388,
        n4389, n4390, n4391, n4392, n4393, n4394, n4395, n4396
            , n4397, n4398,
        n4399, n4400, n4401, n4402, n4403, n4404, n4405, n4406
            , n4407, n4408,
        n4409, n4410, n4411, n4412, n4413, n4414, n4415, n4416
            , n4417, n4418,
        n4419, n4420, n4421, n4422, n4423, n4424, n4425, n4426
            , n4427, n4428,
        n4429, n4430, n4431, n4432, n4433, n4434, n4435, n4436
            , n4437, n4438,
        n4439, n4440, n4441, n4442, n4443, n4444, n4445, n4446
            , n4447, n4448,
        n4449, n4450, n4451, n4452, n4453, n4454, n4455, n4456
            , n4457, n4458,
        n4459, n4460, n4461, n4462, n4463, n4464, n4465, n4466
            , n4467, n4468,
        n4469, n4470, n4471, n4472, n4473, n4474, n4475, n4476
            , n4477, n4478,
        n4479, n4480, n4481, n4482, n4483, n4484, n4485, n4486
            , n4487, n4488,
        n4489, n4490, n4491, n4492, n4493, n4494, n4495, n4496
            , n4497, n4498,
        n4499, n4500, n4501, n4502, n4503, n4504, n4505, n4506
            , n4507, n4508,
        n4509, n4510, n4511, n4512, n4513, n4514, n4515, n4516
            , n4517, n4518,
        n4519, n4520, n4521, n4522, n4523, n4524, n4525, n4526
            , n4527, n4528,
        n4529, n4530, n4531, n4532, n4533, n4534, n4535, n4536
            , n4537, n4538,
        n4539, n4540, n4541, n4542, n4543, n4544, n4545;
wire    [7:0] q_out;
wire    [11:0] counter;

CKND2 I_4734 ( .I(W_13), .ZN(out_ring_osc) );
CKND2 I_4733 ( .I(W_12), .ZN(W_13) );
```

```
OR2D1 C91623 ( .A1(reset_ring_osc), .A2(out_ring_osc), .Z(W_1
   ) );
DFCNQD1 \counter_reg[11]  ( .D(N4778), .CP(clk), .CDN(n4811),
    .Q(counter[11]) );
DFCNQD1 \counter_reg[10]  ( .D(N4777), .CP(clk), .CDN(n4811),
    .Q(counter[10]) );
DFCNQD1 \counter_reg[9]  ( .D(N4776), .CP(clk), .CDN(n4811),
   .Q(counter[9])
       );
DFCNQD1 \counter_reg[8]  ( .D(N4775), .CP(clk), .CDN(n4811),
   .Q(counter[8])
       );
DFCNQD1 \counter_reg[7]  ( .D(N4774), .CP(clk), .CDN(n4811),
   .Q(counter[7])
       );
DFCNQD1 \counter_reg[6]  ( .D(N4773), .CP(clk), .CDN(n4811),
   .Q(counter[6])
       );
DFCNQD1 \counter_reg[5]  ( .D(N4772), .CP(clk), .CDN(n4811),
   .Q(counter[5])
       );
DFCNQD1 \counter_reg[4]  ( .D(N4771), .CP(clk), .CDN(n4811),
   .Q(counter[4])
       );
DFCNQD1 \counter_reg[3]  ( .D(N4770), .CP(clk), .CDN(n4811),
   .Q(counter[3])
       );
DFCNQD1 \counter_reg[2]  ( .D(N4769), .CP(clk), .CDN(n4811),
   .Q(counter[2])
       );
DFCNQD1 \counter_reg[1]  ( .D(N4768), .CP(clk), .CDN(n4811),
   .Q(counter[1])
       );
DFCNQD1 \counter_reg[0]  ( .D(N4767), .CP(clk), .CDN(n4811),
   .Q(counter[0])
       );
DFQD1 \q_reg[6]  ( .D(n4818), .CP(clk), .Q(q_out[6]) );
DFQD1 \q_reg[5]  ( .D(n4817), .CP(clk), .Q(q_out[5]) );
DFQD1 \q_reg[4]  ( .D(n4816), .CP(clk), .Q(q_out[4]) );
DFQD1 \q_reg[3]  ( .D(n4815), .CP(clk), .Q(q_out[3]) );
DFQD1 \q_reg[2]  ( .D(n4814), .CP(clk), .Q(q_out[2]) );
DFQD1 \q_reg[1]  ( .D(n4813), .CP(clk), .Q(q_out[1]) );
DFQD1 \q_reg[0]  ( .D(n4812), .CP(clk), .Q(q_out[0]) );
INVD1 I_4722 ( .I(W_1), .ZN(W_2) );
INVD1 I_4723 ( .I(W_2), .ZN(W_3) );
INVD1 I_4724 ( .I(W_3), .ZN(W_4) );
INVD1 I_4725 ( .I(W_4), .ZN(W_5) );
INVD1 I_4726 ( .I(W_5), .ZN(W_6) );
```

```
  INVD1 I_4727 ( .I(W_6), .ZN(W_7) );
  INVD1 I_4728 ( .I(W_7), .ZN(W_8) );
  INVD1 I_4729 ( .I(W_8), .ZN(W_9) );
  INVD1 I_4730 ( .I(W_9), .ZN(W_10) );
  INVD1 I_4731 ( .I(W_10), .ZN(W_11) );
  INVD1 I_4732 ( .I(W_11), .ZN(W_12) );
  CKND2D1 U3 ( .A1(n3757), .A2(n2274), .ZN(n3243) );
  CKND2D0 U4 ( .A1(counter[7]), .A2(counter[8]), .ZN(n101) );
  INVD0 U5 ( .I(n101), .ZN(n546) );
  INVD0 U6 ( .I(counter[4]), .ZN(n3757) );
  CKND2D0 U7 ( .A1(counter[3]), .A2(counter[2]), .ZN(n3472) );
  CKND2D0 U8 ( .A1(counter[0]), .A2(counter[1]), .ZN(n924) );
  NR2D0 U9 ( .A1(n3472), .A2(n924), .ZN(n1878) );
  INVD0 U10 ( .I(n1878), .ZN(n1888) );
  NR2D0 U11 ( .A1(n3757), .A2(n1888), .ZN(n1189) );
  CKND2D0 U12 ( .A1(counter[5]), .A2(n1189), .ZN(n6) );
  NR2D0 U13 ( .A1(n1513), .A2(n6), .ZN(n1174) );
  CKND2D0 U14 ( .A1(n546), .A2(n1174), .ZN(n7) );
  INVD0 U15 ( .I(n7), .ZN(n1175) );
  INVD0 U16 ( .I(counter[9]), .ZN(n1153) );
  INVD0 U17 ( .I(counter[3]), .ZN(n1859) );
  INVD0 U18 ( .I(counter[2]), .ZN(n3737) );
  CKND2D0 U19 ( .A1(n1859), .A2(n3737), .ZN(n3821) );
  INVD0 U20 ( .I(n3821), .ZN(n4079) );
  INVD0 U21 ( .I(counter[5]), .ZN(n2274) );
  CKND1 U22 ( .I(n3243), .ZN(n4480) );
  AOI21D0 U23 ( .A1(n4079), .A2(n4480), .B(n1513), .ZN(n8) );
  CKXOR2D0 U24 ( .A1(select[0]), .A2(select[1]), .Z(n633) );
  NR2D0 U25 ( .A1(n8), .A2(n633), .ZN(n2) );
  INVD0 U26 ( .I(counter[1]), .ZN(n1865) );
  NR2D0 U27 ( .A1(n1865), .A2(counter[2]), .ZN(n2654) );
  INVD0 U28 ( .I(counter[0]), .ZN(n1362) );
  CKND2D0 U29 ( .A1(n1859), .A2(counter[6]), .ZN(n3349) );
  NR2D0 U30 ( .A1(n1362), .A2(n3349), .ZN(n1416) );
  CKND2D0 U31 ( .A1(n2654), .A2(n1416), .ZN(n942) );
  CKND2D0 U32 ( .A1(n1513), .A2(n2274), .ZN(n4038) );
  CKND1 U33 ( .I(n4038), .ZN(n3933) );
  INVD0 U34 ( .I(n3472), .ZN(n1) );
  CKND2D0 U35 ( .A1(counter[4]), .A2(n1), .ZN(n10) );
  AOI22D0 U36 ( .A1(n2), .A2(n942), .B1(n3933), .B2(n10), .ZN(
     n3) );
  NR2D0 U37 ( .A1(counter[9]), .A2(counter[10]), .ZN(n1042) );
  INVD0 U38 ( .I(n1042), .ZN(n87) );
  AOI221D0 U39 ( .A1(counter[7]), .A2(counter[8]), .B1(n3), .B2
     (counter[8]),
        .C(n87), .ZN(n4) );
  NR2D0 U40 ( .A1(n4), .A2(n3956), .ZN(n1179) );
```

```
AOI221D0 U41 ( .A1(counter[9]), .A2(n1175), .B1(n1153), .B2(
   n7), .C(n1179),
      .ZN(N4776) );
INVD0 U42 ( .I(n924), .ZN(n2315) );
CKND2D0 U43 ( .A1(n3737), .A2(n2315), .ZN(n2271) );
AOI221D0 U44 ( .A1(n2315), .A2(n2271), .B1(n3737), .B2(n2271)
   , .C(n1179),
      .ZN(N4769) );
INVD0 U45 ( .I(counter[7]), .ZN(n1075) );
INVD0 U46 ( .I(n1174), .ZN(n5) );
AOI221D0 U47 ( .A1(counter[7]), .A2(n1174), .B1(n1075), .B2(
   n5), .C(n1179),
      .ZN(N4774) );
INVD0 U48 ( .I(n1189), .ZN(n2627) );
AOI221D0 U49 ( .A1(n1189), .A2(counter[5]), .B1(n2627), .B2(
   n2274), .C(n1179), .ZN(N4772) );
INVD0 U50 ( .I(q_out[1]), .ZN(\q_out[1]_BAR ) );
INVD0 U51 ( .I(q_out[3]), .ZN(\q_out[3]_BAR ) );
INVD0 U52 ( .I(q_out[2]), .ZN(\q_out[2]_BAR ) );
INVD0 U53 ( .I(q_out[6]), .ZN(\q_out[6]_BAR ) );
INVD0 U54 ( .I(q_out[5]), .ZN(\q_out[5]_BAR ) );
INVD0 U55 ( .I(q_out[4]), .ZN(\q_out[4]_BAR ) );
INVD0 U56 ( .I(q_out[0]), .ZN(\q_out[0]_BAR ) );
INVD0 U57 ( .I(reset_nanochip), .ZN(n4811) );
AOI211D0 U58 ( .A1(n3757), .A2(n1888), .B(n1179), .C(n1189),
   .ZN(N4771) );
AOI211D0 U59 ( .A1(n1513), .A2(n6), .B(n1174), .C(n1179), .ZN
   (N4773) );
INVD0 U60 ( .I(counter[10]), .ZN(n1157) );
NR2D0 U61 ( .A1(n1153), .A2(n1157), .ZN(n1010) );
INVD0 U62 ( .I(n1010), .ZN(n49) );
NR2D0 U63 ( .A1(n49), .A2(n7), .ZN(n638) );
IAO21D0 U64 ( .A1(counter[11]), .A2(n638), .B(n1179), .ZN(
   N4778) );
AOI221D0 U65 ( .A1(counter[7]), .A2(counter[8]), .B1(n8), .B2
   (counter[8]),
      .C(n87), .ZN(n9) );
IAO21D0 U66 ( .A1(n9), .A2(n3956), .B(n633), .ZN(n916) );
INVD0 U67 ( .I(n916), .ZN(n4544) );
CKND2D0 U68 ( .A1(n1362), .A2(n1865), .ZN(n2341) );
INVD0 U69 ( .I(n2341), .ZN(n562) );
CKND2D0 U70 ( .A1(n1075), .A2(n2274), .ZN(n1163) );
INVD0 U71 ( .I(n1163), .ZN(n1135) );
OAI211D0 U72 ( .A1(n562), .A2(n10), .B(n1135), .C(n1513), .ZN
   (n11) );
AOI32D0 U73 ( .A1(counter[8]), .A2(counter[11]), .A3(n11), .
   B1(n87), .B2(
      counter[11]), .ZN(n12) );
```

```
  CKND2D0 U74 ( .A1(n633), .A2(n12), .ZN(n4542) );
  CKND2D0 U75 ( .A1(n4544), .A2(n4542), .ZN(n4545) );
  INVD0 U76 ( .I(n4542), .ZN(n1169) );
  CKND2D0 U77 ( .A1(counter[4]), .A2(counter[5]), .ZN(n4476) );
  NR2D0 U78 ( .A1(n3737), .A2(counter[1]), .ZN(n1872) );
  INVD0 U79 ( .I(n1872), .ZN(n2265) );
  NR2D0 U80 ( .A1(n2265), .A2(counter[11]), .ZN(n1284) );
  CKND2D0 U81 ( .A1(n1513), .A2(n1859), .ZN(n2987) );
  NR2D0 U82 ( .A1(n1362), .A2(n2987), .ZN(n3049) );
  CKND2D0 U83 ( .A1(n1284), .A2(n3049), .ZN(n4316) );
  CKND2D0 U84 ( .A1(n1362), .A2(n1284), .ZN(n3837) );
  NR2D0 U85 ( .A1(n2987), .A2(n3837), .ZN(n3109) );
  INVD0 U86 ( .I(n1416), .ZN(n2996) );
  INVD0 U87 ( .I(n1284), .ZN(n2424) );
  NR2D0 U88 ( .A1(n2996), .A2(n2424), .ZN(n3319) );
  NR2D0 U89 ( .A1(n3109), .A2(n3319), .ZN(n4153) );
  NR2D0 U90 ( .A1(counter[11]), .A2(counter[0]), .ZN(n3176) );
  CKND2D0 U91 ( .A1(n1865), .A2(n3176), .ZN(n3822) );
  INVD0 U92 ( .I(n3349), .ZN(n3443) );
  CKND2D0 U93 ( .A1(n3737), .A2(n3443), .ZN(n3474) );
  NR2D0 U94 ( .A1(n3822), .A2(n3474), .ZN(n4161) );
  CKND2D0 U95 ( .A1(n1513), .A2(counter[3]), .ZN(n2596) );
  NR2D0 U96 ( .A1(n2596), .A2(n3822), .ZN(n3457) );
  NR2D0 U97 ( .A1(n4161), .A2(n3457), .ZN(n3572) );
  CKND2D0 U98 ( .A1(counter[6]), .A2(counter[3]), .ZN(n3439) );
  NR2D0 U99 ( .A1(n1362), .A2(n3439), .ZN(n13) );
  NR2D0 U100 ( .A1(counter[2]), .A2(counter[1]), .ZN(n2978) );
  INVD0 U101 ( .I(n2978), .ZN(n2269) );
  NR2D0 U102 ( .A1(n2269), .A2(counter[11]), .ZN(n3833) );
  CKND2D0 U103 ( .A1(n13), .A2(n3833), .ZN(n4320) );
  INVD0 U104 ( .I(counter[11]), .ZN(n3956) );
  CKND2D0 U105 ( .A1(n3956), .A2(n2654), .ZN(n1556) );
  INVD0 U106 ( .I(n1556), .ZN(n1937) );
  NR2D0 U107 ( .A1(n3439), .A2(counter[0]), .ZN(n1436) );
  CKND2D0 U108 ( .A1(n1937), .A2(n1436), .ZN(n4133) );
  CKND2D0 U109 ( .A1(n4320), .A2(n4133), .ZN(n167) );
  CKND2D0 U110 ( .A1(n1436), .A2(n3833), .ZN(n4418) );
  INVD0 U111 ( .I(n4418), .ZN(n3162) );
  INVD0 U112 ( .I(n2596), .ZN(n2508) );
  CKND2D0 U113 ( .A1(counter[2]), .A2(n2508), .ZN(n4223) );
  NR2D0 U114 ( .A1(n1362), .A2(counter[11]), .ZN(n4098) );
  CKND2D0 U115 ( .A1(counter[1]), .A2(n4098), .ZN(n3477) );
  NR2D0 U116 ( .A1(n4223), .A2(n3477), .ZN(n4307) );
  INVD0 U117 ( .I(n4307), .ZN(n4317) );
  NR2D0 U118 ( .A1(counter[3]), .A2(counter[1]), .ZN(n3750) );
  INVD0 U119 ( .I(n3176), .ZN(n4018) );
  NR2D0 U120 ( .A1(n4018), .A2(counter[6]), .ZN(n4144) );
  INVD0 U121 ( .I(n4144), .ZN(n1367) );
```

```
  INR2D0 U122 ( .A1(n3750), .B1(n1367), .ZN(n4437) );
  CKND2D0 U123 ( .A1(n4437), .A2(n3737), .ZN(n4362) );
  CKND2D0 U124 ( .A1(n4317), .A2(n4362), .ZN(n3368) );
  NR2D0 U125 ( .A1(n3162), .A2(n3368), .ZN(n137) );
  INVD0 U126 ( .I(n137), .ZN(n3495) );
  CKND2D0 U127 ( .A1(counter[2]), .A2(counter[1]), .ZN(n4351) )
     ;
  NR2D0 U128 ( .A1(n1859), .A2(n4351), .ZN(n1022) );
  INVD0 U129 ( .I(n1022), .ZN(n1259) );
  NR2D0 U130 ( .A1(n1259), .A2(n1367), .ZN(n4124) );
  CKND2D0 U131 ( .A1(n1513), .A2(n4098), .ZN(n152) );
  NR2D0 U132 ( .A1(n2269), .A2(n152), .ZN(n1363) );
  NR4D0 U133 ( .A1(n167), .A2(n3495), .A3(n4124), .A4(n1363), .
     ZN(n155) );
  ND4D0 U134 ( .A1(n4316), .A2(n4153), .A3(n3572), .A4(n155), .
     ZN(n177) );
  INVD0 U135 ( .I(n4351), .ZN(n1243) );
  CKND2D0 U136 ( .A1(n1859), .A2(n1243), .ZN(n1881) );
  NR2D0 U137 ( .A1(n1881), .A2(n152), .ZN(n3574) );
  CKND2D0 U138 ( .A1(n1865), .A2(n4098), .ZN(n3595) );
  NR2D0 U139 ( .A1(n3595), .A2(n4223), .ZN(n4170) );
  NR2D0 U140 ( .A1(n3574), .A2(n4170), .ZN(n3486) );
  INVD0 U141 ( .I(n3486), .ZN(n15) );
  CKND2D0 U142 ( .A1(n1284), .A2(n13), .ZN(n4334) );
  INVD0 U143 ( .I(n4334), .ZN(n4311) );
  CKND2D0 U144 ( .A1(n1513), .A2(n1243), .ZN(n1799) );
  INVD0 U145 ( .I(n1799), .ZN(n4099) );
  CKND2D0 U146 ( .A1(n4099), .A2(n3176), .ZN(n3324) );
  NR2D0 U147 ( .A1(counter[3]), .A2(n3324), .ZN(n4438) );
  NR2D0 U148 ( .A1(n4311), .A2(n4438), .ZN(n4447) );
  NR2D0 U149 ( .A1(n1513), .A2(n4351), .ZN(n1526) );
  INVD0 U150 ( .I(n1526), .ZN(n1506) );
  NR2D0 U151 ( .A1(n1506), .A2(n4018), .ZN(n3541) );
  INVD0 U152 ( .I(n3541), .ZN(n3343) );
  CKND2D0 U153 ( .A1(n4447), .A2(n3343), .ZN(n191) );
  NR2D0 U154 ( .A1(n15), .A2(n191), .ZN(n206) );
  INVD0 U155 ( .I(n2654), .ZN(n3390) );
  NR2D0 U156 ( .A1(n1513), .A2(n3390), .ZN(n4257) );
  INVD0 U157 ( .I(n4257), .ZN(n1524) );
  INVD0 U158 ( .I(n4098), .ZN(n1188) );
  NR2D0 U159 ( .A1(n1524), .A2(n1188), .ZN(n4134) );
  INVD0 U160 ( .I(n4134), .ZN(n1574) );
  NR2D0 U161 ( .A1(n1513), .A2(n4018), .ZN(n4261) );
  CKND2D0 U162 ( .A1(n1872), .A2(n4261), .ZN(n4504) );
  IND4D0 U163 ( .A1(n177), .B1(n206), .B2(n1574), .B3(n4504), .
     ZN(n20) );
  CKND2D0 U164 ( .A1(n1859), .A2(counter[1]), .ZN(n3529) );
  CKND2D0 U165 ( .A1(n3737), .A2(n4098), .ZN(n1722) );
```

```
  NR2D0 U166 ( .A1(n3529), .A2(n1722), .ZN(n4439) );
  CKND2D0 U167 ( .A1(n1513), .A2(n4439), .ZN(n4389) );
  INVD0 U168 ( .I(n4389), .ZN(n4495) );
  CKND2D0 U169 ( .A1(n1937), .A2(n1416), .ZN(n4345) );
  NR2D0 U170 ( .A1(n3349), .A2(counter[0]), .ZN(n1446) );
  CKND2D0 U171 ( .A1(n1284), .A2(n1446), .ZN(n4157) );
  CKND2D0 U172 ( .A1(n4345), .A2(n4157), .ZN(n256) );
  NR2D0 U173 ( .A1(n1556), .A2(n2596), .ZN(n3098) );
  INVD0 U174 ( .I(n3098), .ZN(n1478) );
  IND2D0 U175 ( .A1(n256), .B1(n1478), .ZN(n212) );
  INVD0 U176 ( .I(n1436), .ZN(n2977) );
  CKND2D0 U177 ( .A1(n3956), .A2(n1243), .ZN(n3009) );
  NR2D0 U178 ( .A1(n2977), .A2(n3009), .ZN(n4232) );
  NR2D0 U179 ( .A1(n4311), .A2(n4232), .ZN(n142) );
  INVD0 U180 ( .I(n142), .ZN(n188) );
  CKND2D0 U181 ( .A1(n3737), .A2(counter[3]), .ZN(n2088) );
  NR2D0 U182 ( .A1(n3595), .A2(n2088), .ZN(n1303) );
  INVD0 U183 ( .I(n1303), .ZN(n4109) );
  CKND2D0 U184 ( .A1(counter[6]), .A2(n4098), .ZN(n3357) );
  NR2D0 U185 ( .A1(n1259), .A2(n3357), .ZN(n4373) );
  CKND2D0 U186 ( .A1(n3833), .A2(n3049), .ZN(n4194) );
  INVD0 U187 ( .I(n4194), .ZN(n4250) );
  NR2D0 U188 ( .A1(n4373), .A2(n4250), .ZN(n4296) );
  INVD0 U189 ( .I(n1446), .ZN(n1412) );
  NR2D0 U190 ( .A1(n1556), .A2(n1412), .ZN(n4172) );
  INVD0 U191 ( .I(n4172), .ZN(n4137) );
  NR2D0 U192 ( .A1(n2424), .A2(n2977), .ZN(n4426) );
  NR2D0 U193 ( .A1(n3109), .A2(n4426), .ZN(n3201) );
  CKND2D0 U194 ( .A1(n4137), .A2(n3201), .ZN(n202) );
  NR2D0 U195 ( .A1(n202), .A2(n3162), .ZN(n97) );
  INVD0 U196 ( .I(n3477), .ZN(n1719) );
  NR2D0 U197 ( .A1(n3737), .A2(n3349), .ZN(n3442) );
  CKND2D0 U198 ( .A1(n1719), .A2(n3442), .ZN(n3530) );
  INVD0 U199 ( .I(n3530), .ZN(n3540) );
  NR2D0 U200 ( .A1(n4161), .A2(n3540), .ZN(n3427) );
  CKND2D0 U201 ( .A1(n97), .A2(n3427), .ZN(n235) );
  INR2D0 U202 ( .A1(n4296), .B1(n235), .ZN(n169) );
  CKAN2D0 U203 ( .A1(n169), .A2(n4317), .Z(n160) );
  INVD0 U204 ( .I(n4133), .ZN(n4225) );
  INVD0 U205 ( .I(n4362), .ZN(n4374) );
  CKND2D0 U206 ( .A1(n1416), .A2(n3833), .ZN(n4335) );
  INVD0 U207 ( .I(n4335), .ZN(n4090) );
  NR2D0 U208 ( .A1(n4124), .A2(n4090), .ZN(n4488) );
  NR2D0 U209 ( .A1(n1859), .A2(n2269), .ZN(n2125) );
  CKND2D0 U210 ( .A1(n2125), .A2(n4144), .ZN(n4333) );
  CKND2D0 U211 ( .A1(n4333), .A2(n4316), .ZN(n1407) );
  INVD0 U212 ( .I(n1407), .ZN(n44) );
  CKND2D0 U213 ( .A1(n4488), .A2(n44), .ZN(n198) );
```

```
  NR3D0 U214 ( .A1(n4225), .A2(n4374), .A3(n198), .ZN(n225) );
  INVD0 U215 ( .I(n225), .ZN(n174) );
  INR2D0 U216 ( .A1(n160), .B1(n174), .ZN(n190) );
  CKND2D0 U217 ( .A1(n4109), .A2(n190), .ZN(n100) );
  NR2D0 U218 ( .A1(n100), .A2(n4495), .ZN(n254) );
  INVD0 U219 ( .I(n254), .ZN(n221) );
  INVD0 U220 ( .I(n3574), .ZN(n4201) );
  NR2D0 U221 ( .A1(n3529), .A2(counter[2]), .ZN(n4118) );
  INVD0 U222 ( .I(n4118), .ZN(n1242) );
  NR2D0 U223 ( .A1(n1242), .A2(n4018), .ZN(n3542) );
  INVD0 U224 ( .I(n3542), .ZN(n4330) );
  NR2D0 U225 ( .A1(n4330), .A2(counter[6]), .ZN(n4413) );
  INVD0 U226 ( .I(n4413), .ZN(n4363) );
  CKND2D0 U227 ( .A1(n4201), .A2(n4363), .ZN(n57) );
  NR4D0 U228 ( .A1(n212), .A2(n188), .A3(n221), .A4(n57), .ZN(
     n18) );
  NR2D0 U229 ( .A1(n3757), .A2(counter[5]), .ZN(n4289) );
  CKND1 U230 ( .I(n4289), .ZN(n4517) );
  INVD0 U231 ( .I(n4517), .ZN(n4342) );
  INVD0 U232 ( .I(n4316), .ZN(n3353) );
  NR2D0 U233 ( .A1(n3353), .A2(n4124), .ZN(n205) );
  NR2D0 U234 ( .A1(n2596), .A2(counter[0]), .ZN(n1445) );
  CKND2D0 U235 ( .A1(n1284), .A2(n1445), .ZN(n4294) );
  INVD0 U236 ( .I(n4294), .ZN(n4323) );
  NR2D0 U237 ( .A1(n3319), .A2(n4413), .ZN(n172) );
  INVD0 U238 ( .I(n172), .ZN(n62) );
  INVD0 U239 ( .I(n4333), .ZN(n4310) );
  NR2D0 U240 ( .A1(n4090), .A2(n4310), .ZN(n1392) );
  CKND2D0 U241 ( .A1(n1392), .A2(n4389), .ZN(n161) );
  NR4D0 U242 ( .A1(n1303), .A2(n4323), .A3(n62), .A4(n161), .ZN
     (n113) );
  CKND2D0 U243 ( .A1(n113), .A2(n4133), .ZN(n123) );
  NR2D0 U244 ( .A1(n4373), .A2(n123), .ZN(n107) );
  CKND2D0 U245 ( .A1(n205), .A2(n107), .ZN(n121) );
  INVD0 U246 ( .I(n13), .ZN(n14) );
  NR2D0 U247 ( .A1(n1556), .A2(n14), .ZN(n4181) );
  INVD0 U248 ( .I(n4181), .ZN(n4368) );
  CKND2D0 U249 ( .A1(n4368), .A2(n1478), .ZN(n3045) );
  NR2D0 U250 ( .A1(n3045), .A2(n256), .ZN(n175) );
  NR2D0 U251 ( .A1(n3737), .A2(counter[3]), .ZN(n3335) );
  INVD0 U252 ( .I(n3335), .ZN(n3441) );
  CKND2D0 U253 ( .A1(counter[1]), .A2(n3176), .ZN(n4025) );
  NR2D0 U254 ( .A1(n3441), .A2(n4025), .ZN(n3604) );
  CKND2D0 U255 ( .A1(counter[6]), .A2(n3604), .ZN(n3505) );
  INVD0 U256 ( .I(n3505), .ZN(n4422) );
  NR2D0 U257 ( .A1(n4422), .A2(n15), .ZN(n262) );
  ND3D0 U258 ( .A1(n175), .A2(n97), .A3(n262), .ZN(n105) );
```

```
OAI31D0 U259 ( .A1(n4232), .A2(n121), .A3(n105), .B(n4480), .
   ZN(n17) );
INVD0 U260 ( .I(n206), .ZN(n227) );
IND2D0 U261 ( .A1(n3368), .B1(n169), .ZN(n144) );
INVD0 U262 ( .I(n4320), .ZN(n4156) );
NR2D0 U263 ( .A1(n4495), .A2(n4156), .ZN(n3356) );
NR2D0 U264 ( .A1(n4090), .A2(n4323), .ZN(n3235) );
INVD0 U265 ( .I(n3235), .ZN(n3203) );
CKND2D0 U266 ( .A1(n205), .A2(n175), .ZN(n128) );
NR3D0 U267 ( .A1(n4310), .A2(n3203), .A3(n128), .ZN(n53) );
CKND2D0 U268 ( .A1(n3356), .A2(n53), .ZN(n208) );
CKND2D0 U269 ( .A1(n3757), .A2(counter[5]), .ZN(n54) );
CKND1 U270 ( .I(n54), .ZN(n4452) );
OAI31D0 U271 ( .A1(n227), .A2(n144), .A3(n208), .B(n4452), .
   ZN(n16) );
OAI211D0 U272 ( .A1(n18), .A2(n4517), .B(n17), .C(n16), .ZN(
   n19) );
AOI211D0 U273 ( .A1(n4291), .A2(n20), .B(n4495), .C(n19), .ZN
   (n271) );
INVD0 U274 ( .I(counter[8]), .ZN(n1177) );
CKND2D0 U275 ( .A1(n1177), .A2(counter[7]), .ZN(n486) );
CKND2D0 U276 ( .A1(n1157), .A2(counter[9]), .ZN(n216) );
NR2D0 U277 ( .A1(n486), .A2(n216), .ZN(n3437) );
INVD0 U278 ( .I(n3437), .ZN(n4410) );
CKND1 U279 ( .I(n4476), .ZN(n4291) );
NR2D0 U280 ( .A1(n4161), .A2(n4307), .ZN(n4364) );
ND4D0 U281 ( .A1(n206), .A2(n4364), .A3(n3530), .A4(n4418), .
   ZN(n213) );
NR2D0 U282 ( .A1(n4374), .A2(n213), .ZN(n207) );
NR2D0 U283 ( .A1(n3956), .A2(counter[0]), .ZN(n1522) );
INVD0 U284 ( .I(n1522), .ZN(n4222) );
NR2D0 U285 ( .A1(n3737), .A2(n4222), .ZN(n3907) );
CKND2D0 U286 ( .A1(n3750), .A2(n3907), .ZN(n3453) );
NR2D0 U287 ( .A1(counter[6]), .A2(n3453), .ZN(n2980) );
NR2D0 U288 ( .A1(n1859), .A2(n1865), .ZN(n3749) );
INVD0 U289 ( .I(n3749), .ZN(n1368) );
NR2D0 U290 ( .A1(n3956), .A2(n1362), .ZN(n4256) );
CKND2D0 U291 ( .A1(counter[6]), .A2(n4256), .ZN(n23) );
NR2D0 U292 ( .A1(n1368), .A2(n23), .ZN(n1429) );
CKND2D0 U293 ( .A1(n3737), .A2(n1429), .ZN(n3460) );
INVD0 U294 ( .I(n3460), .ZN(n3060) );
CKND2D0 U295 ( .A1(counter[11]), .A2(n2978), .ZN(n3735) );
NR2D0 U296 ( .A1(n2977), .A2(n3735), .ZN(n3459) );
INVD0 U297 ( .I(n4256), .ZN(n3462) );
NR2D0 U298 ( .A1(n3737), .A2(n3462), .ZN(n3990) );
INVD0 U299 ( .I(n3990), .ZN(n3748) );
NR2D0 U300 ( .A1(n3529), .A2(n3748), .ZN(n4110) );
INVD0 U301 ( .I(counter[6]), .ZN(n1513) );
```

```
  CKND2D0 U302 ( .A1(n4110), .A2(n1513), .ZN(n3082) );
  CKND2D0 U303 ( .A1(n1522), .A2(n3737), .ZN(n2538) );
  NR2D0 U304 ( .A1(n2596), .A2(n2538), .ZN(n3537) );
  CKND2D0 U305 ( .A1(counter[1]), .A2(n3537), .ZN(n3054) );
  CKND2D0 U306 ( .A1(n3082), .A2(n3054), .ZN(n129) );
  NR2D0 U307 ( .A1(n3459), .A2(n129), .ZN(n63) );
  AOI22D0 U308 ( .A1(counter[1]), .A2(n3439), .B1(n2596), .B2(
     n1865), .ZN(
        n3414) );
  INVD0 U309 ( .I(n2987), .ZN(n2577) );
  CKND2D0 U310 ( .A1(n2577), .A2(n3737), .ZN(n3333) );
  NR2D0 U311 ( .A1(n4222), .A2(counter[1]), .ZN(n3901) );
  INVD0 U312 ( .I(n3901), .ZN(n3753) );
  NR2D0 U313 ( .A1(n3333), .A2(n3753), .ZN(n4075) );
  AOI21D0 U314 ( .A1(n3990), .A2(n3414), .B(n4075), .ZN(n80) );
  INVD0 U315 ( .I(n3735), .ZN(n3732) );
  CKND2D0 U316 ( .A1(n1446), .A2(n3732), .ZN(n3053) );
  INVD0 U317 ( .I(n3053), .ZN(n4089) );
  NR2D0 U318 ( .A1(n1513), .A2(n4222), .ZN(n4117) );
  CKND2D0 U319 ( .A1(n2654), .A2(n4117), .ZN(n4244) );
  NR2D0 U320 ( .A1(n1859), .A2(n4244), .ZN(n3524) );
  NR2D0 U321 ( .A1(n4089), .A2(n3524), .ZN(n75) );
  NR2D0 U322 ( .A1(n1865), .A2(counter[0]), .ZN(n3527) );
  CKND2D0 U323 ( .A1(counter[11]), .A2(n3527), .ZN(n2351) );
  INVD0 U324 ( .I(n2351), .ZN(n1945) );
  CKND2D0 U325 ( .A1(n1865), .A2(n4256), .ZN(n1862) );
  INVD0 U326 ( .I(n1862), .ZN(n3544) );
  NR2D0 U327 ( .A1(n1945), .A2(n3544), .ZN(n3531) );
  NR2D0 U328 ( .A1(n3531), .A2(n3474), .ZN(n3452) );
  NR2D0 U329 ( .A1(n4222), .A2(counter[6]), .ZN(n4113) );
  INR2D0 U330 ( .A1(n4113), .B1(n1242), .ZN(n3444) );
  CKND2D0 U331 ( .A1(counter[11]), .A2(counter[1]), .ZN(n3332)
     );
  NR2D0 U332 ( .A1(n3737), .A2(n3332), .ZN(n3743) );
  CKND2D0 U333 ( .A1(n1445), .A2(n3743), .ZN(n4240) );
  INVD0 U334 ( .I(n4240), .ZN(n3065) );
  NR2D0 U335 ( .A1(n3444), .A2(n3065), .ZN(n3553) );
  NR2D0 U336 ( .A1(n1859), .A2(n2265), .ZN(n1214) );
  CKND2D0 U337 ( .A1(n1214), .A2(n4117), .ZN(n4108) );
  CKND2D0 U338 ( .A1(n3442), .A2(n1945), .ZN(n4241) );
  ND3D0 U339 ( .A1(n3553), .A2(n4108), .A3(n4241), .ZN(n64) );
  CKND2D0 U340 ( .A1(n1416), .A2(n3743), .ZN(n4091) );
  NR2D0 U341 ( .A1(counter[6]), .A2(n3462), .ZN(n4081) );
  CKND2D0 U342 ( .A1(n4118), .A2(n4081), .ZN(n4107) );
  CKND2D0 U343 ( .A1(n4091), .A2(n4107), .ZN(n21) );
  INR4D0 U344 ( .A1(n205), .B1(n3452), .B2(n64), .B3(n21), .ZN(
     n22) );
```

```
  ND4D0 U345 ( .A1(n63), .A2(n80), .A3(n75), .A4(n22), .ZN(n26)
      );
  NR3D0 U346 ( .A1(n2980), .A2(n3060), .A3(n26), .ZN(n35) );
  CKND2D0 U347 ( .A1(n3737), .A2(n2508), .ZN(n3461) );
  INVD0 U348 ( .I(n3461), .ZN(n94) );
  NR2D0 U349 ( .A1(n3737), .A2(n3439), .ZN(n3406) );
  NR2D0 U350 ( .A1(n94), .A2(n3406), .ZN(n3528) );
  NR2D0 U351 ( .A1(n3528), .A2(n1862), .ZN(n3523) );
  INVD0 U352 ( .I(n23), .ZN(n3470) );
  CKND2D0 U353 ( .A1(n2125), .A2(n3470), .ZN(n1434) );
  INVD0 U354 ( .I(n1434), .ZN(n4074) );
  CKND2D0 U355 ( .A1(counter[1]), .A2(n4256), .ZN(n3449) );
  OAI22D0 U356 ( .A1(n1513), .A2(n3453), .B1(n2596), .B2(n3449)
     , .ZN(n34) );
  CKND2D0 U357 ( .A1(n1362), .A2(n2577), .ZN(n1414) );
  INVD0 U358 ( .I(n1414), .ZN(n1418) );
  CKND2D0 U359 ( .A1(n3743), .A2(n1418), .ZN(n4078) );
  CKND2D0 U360 ( .A1(n3049), .A2(n3732), .ZN(n4265) );
  CKND2D0 U361 ( .A1(n4078), .A2(n4265), .ZN(n4103) );
  NR3D0 U362 ( .A1(n4074), .A2(n34), .A3(n4103), .ZN(n36) );
  CKND2D0 U363 ( .A1(n1022), .A2(n4117), .ZN(n4266) );
  CKND2D0 U364 ( .A1(n36), .A2(n4266), .ZN(n65) );
  NR4D0 U365 ( .A1(n4426), .A2(n3523), .A3(n212), .A4(n65), .ZN
     (n24) );
  ND4D0 U366 ( .A1(n107), .A2(n207), .A3(n35), .A4(n24), .ZN(
     n29) );
  INVD0 U367 ( .I(n3201), .ZN(n3276) );
  INVD0 U368 ( .I(n4373), .ZN(n4212) );
  CKND2D0 U369 ( .A1(n3427), .A2(n4212), .ZN(n3458) );
  NR2D0 U370 ( .A1(n3276), .A2(n3458), .ZN(n255) );
  NR2D0 U371 ( .A1(n4134), .A2(n123), .ZN(n203) );
  NR2D0 U372 ( .A1(n4250), .A2(n4172), .ZN(n3392) );
  NR2D0 U373 ( .A1(n3441), .A2(n1862), .ZN(n2803) );
  AOI211D0 U374 ( .A1(n1719), .A2(n2508), .B(n2803), .C(n3523),
      .ZN(n25) );
  IND4D0 U375 ( .A1(n26), .B1(n203), .B2(n3392), .B3(n25), .ZN(
     n27) );
  NR2D0 U376 ( .A1(n191), .A2(n27), .ZN(n32) );
  INVD0 U377 ( .I(n4170), .ZN(n4367) );
  NR2D0 U378 ( .A1(n2996), .A2(n3009), .ZN(n4171) );
  INVD0 U379 ( .I(n4171), .ZN(n4506) );
  CKND2D0 U380 ( .A1(n4367), .A2(n4506), .ZN(n1380) );
  NR2D0 U381 ( .A1(n4374), .A2(n1380), .ZN(n3246) );
  IND4D0 U382 ( .A1(n65), .B1(n255), .B2(n32), .B3(n3246), .ZN(
     n28) );
  AOI22D0 U383 ( .A1(n4291), .A2(n29), .B1(n4342), .B2(n28), .
     ZN(n40) );
  NR2D0 U384 ( .A1(n3162), .A2(n3540), .ZN(n30) );
```

```
  CKND2D0 U385 ( .A1(n30), .A2(n3460), .ZN(n69) );
  CKND2D0 U386 ( .A1(n1445), .A2(n3732), .ZN(n4209) );
  INVD0 U387 ( .I(n4209), .ZN(n3037) );
  NR2D0 U388 ( .A1(n3956), .A2(n2265), .ZN(n2636) );
  CKND2D0 U389 ( .A1(n1445), .A2(n2636), .ZN(n4262) );
  INVD0 U390 ( .I(n4262), .ZN(n3338) );
  CKND2D0 U391 ( .A1(counter[11]), .A2(n2654), .ZN(n3897) );
  NR2D0 U392 ( .A1(n2996), .A2(n3897), .ZN(n71) );
  NR4D0 U393 ( .A1(n3037), .A2(n3338), .A3(n2980), .A4(n71), .
     ZN(n67) );
  NR2D0 U394 ( .A1(n3574), .A2(n4373), .ZN(n31) );
  ND4D0 U395 ( .A1(n67), .A2(n32), .A3(n31), .A4(n4157), .ZN(
     n33) );
  OAI31D0 U396 ( .A1(n69), .A2(n34), .A3(n33), .B(n4480), .ZN(
     n39) );
  CKND2D0 U397 ( .A1(n4389), .A2(n4109), .ZN(n4173) );
  CKND2D0 U398 ( .A1(counter[6]), .A2(n2803), .ZN(n4121) );
  IND4D0 U399 ( .A1(n4173), .B1(n35), .B2(n4262), .B3(n4121), .
     ZN(n73) );
  INVD0 U400 ( .I(n144), .ZN(n133) );
  CKND2D0 U401 ( .A1(n4447), .A2(n175), .ZN(n106) );
  INVD0 U402 ( .I(n106), .ZN(n154) );
  CKND2D0 U403 ( .A1(n133), .A2(n154), .ZN(n47) );
  INVD0 U404 ( .I(n47), .ZN(n43) );
  INVD0 U405 ( .I(n1214), .ZN(n1211) );
  CKND2D0 U406 ( .A1(n1211), .A2(n1242), .ZN(n2798) );
  CKND2D0 U407 ( .A1(n4144), .A2(n2798), .ZN(n4503) );
  CKND2D0 U408 ( .A1(n262), .A2(n4503), .ZN(n220) );
  NR2D0 U409 ( .A1(n3319), .A2(n220), .ZN(n189) );
  ND4D0 U410 ( .A1(n36), .A2(n43), .A3(n189), .A4(n4209), .ZN(
     n37) );
  OAI21D0 U411 ( .A1(n73), .A2(n37), .B(n4452), .ZN(n38) );
  NR2D0 U412 ( .A1(n87), .A2(n486), .ZN(n3567) );
  INVD0 U413 ( .I(n3567), .ZN(n4129) );
  AOI31D0 U414 ( .A1(n40), .A2(n39), .A3(n38), .B(n4129), .ZN(
     n250) );
  NR2D0 U415 ( .A1(n2088), .A2(n3477), .ZN(n4236) );
  NR2D0 U416 ( .A1(n4438), .A2(n4232), .ZN(n3094) );
  NR3D0 U417 ( .A1(n4495), .A2(n4310), .A3(n202), .ZN(n252) );
  CKND2D0 U418 ( .A1(n4133), .A2(n4335), .ZN(n4384) );
  INVD0 U419 ( .I(n4384), .ZN(n4502) );
  CKND2D0 U420 ( .A1(n2508), .A2(n1865), .ZN(n41) );
  NR2D0 U421 ( .A1(n1722), .A2(n41), .ZN(n4306) );
  NR2D0 U422 ( .A1(n3353), .A2(n4306), .ZN(n3273) );
  ND4D0 U423 ( .A1(n252), .A2(n189), .A3(n4502), .A4(n3273), .
     ZN(n42) );
  NR2D0 U424 ( .A1(n3458), .A2(n42), .ZN(n153) );
```

```
IND4D0 U425 ( .A1(n4236), .B1(n3094), .B2(n137), .B3(n153), .
   ZN(n46) );
INVD0 U426 ( .I(n167), .ZN(n3351) );
IND4D0 U427 ( .A1(n220), .B1(n3351), .B2(n44), .B3(n43), .ZN(
   n45) );
AOI22D0 U428 ( .A1(n4452), .A2(n46), .B1(n4342), .B2(n45), .
   ZN(n52) );
CKND2D0 U429 ( .A1(n3486), .A2(n203), .ZN(n93) );
CKND2D0 U430 ( .A1(counter[3]), .A2(n2654), .ZN(n4100) );
INVD0 U431 ( .I(n4100), .ZN(n1269) );
CKND2D0 U432 ( .A1(n1269), .A2(n4144), .ZN(n4507) );
CKND2D0 U433 ( .A1(n4157), .A2(n4507), .ZN(n134) );
INVD0 U434 ( .I(n134), .ZN(n1454) );
CKND2D0 U435 ( .A1(n133), .A2(n1454), .ZN(n114) );
OAI31D0 U436 ( .A1(n93), .A2(n188), .A3(n114), .B(n4480), .ZN
   (n51) );
INVD0 U437 ( .I(n4124), .ZN(n4388) );
CKND2D0 U438 ( .A1(n4388), .A2(n3505), .ZN(n112) );
INVD0 U439 ( .I(n112), .ZN(n4464) );
CKND2D0 U440 ( .A1(n4464), .A2(n4201), .ZN(n48) );
INVD0 U441 ( .I(n113), .ZN(n257) );
OAI31D0 U442 ( .A1(n48), .A2(n257), .A3(n47), .B(n4291), .ZN(
   n50) );
NR2D0 U443 ( .A1(n49), .A2(n101), .ZN(n4521) );
INVD0 U444 ( .I(n4521), .ZN(n3265) );
AOI31D0 U445 ( .A1(n52), .A2(n51), .A3(n50), .B(n3265), .ZN(
   n249) );
INVD0 U446 ( .I(n3319), .ZN(n3320) );
CKND2D0 U447 ( .A1(n3320), .A2(n4367), .ZN(n4246) );
NR2D0 U448 ( .A1(n191), .A2(n4246), .ZN(n253) );
NR2D0 U449 ( .A1(n3495), .A2(n4173), .ZN(n3573) );
ND4D0 U450 ( .A1(n53), .A2(n255), .A3(n253), .A4(n3573), .ZN(
   n61) );
INVD0 U451 ( .I(n3094), .ZN(n4183) );
NR4D0 U452 ( .A1(n4170), .A2(n123), .A3(n128), .A4(n4183), .
   ZN(n55) );
AOI21D0 U453 ( .A1(n169), .A2(n55), .B(n54), .ZN(n60) );
NR2D0 U454 ( .A1(n3319), .A2(n4323), .ZN(n3097) );
NR2D0 U455 ( .A1(n227), .A2(n4413), .ZN(n158) );
CKND2D0 U456 ( .A1(n3097), .A2(n158), .ZN(n99) );
INVD0 U457 ( .I(n99), .ZN(n159) );
INVD0 U458 ( .I(n1722), .ZN(n4445) );
CKND2D0 U459 ( .A1(n2577), .A2(n4445), .ZN(n3350) );
INVD0 U460 ( .I(n4306), .ZN(n4465) );
IND4D0 U461 ( .A1(n128), .B1(n159), .B2(n3350), .B3(n4465), .
   ZN(n56) );
NR2D0 U462 ( .A1(n235), .A2(n56), .ZN(n228) );
INVD0 U463 ( .I(n175), .ZN(n222) );
```

```
NR4D0 U464 ( .A1(n222), .A2(n100), .A3(n57), .A4(n4183), .ZN(
   n58) );
INVD0 U465 ( .I(n4291), .ZN(n4428) );
OAI22D0 U466 ( .A1(n228), .A2(n4517), .B1(n58), .B2(n4428), .
   ZN(n59) );
AOI211D0 U467 ( .A1(n4480), .A2(n61), .B(n60), .C(n59), .ZN(
   n89) );
NR2D0 U468 ( .A1(n87), .A2(n101), .ZN(n3664) );
INVD0 U469 ( .I(n3664), .ZN(n4408) );
NR2D0 U470 ( .A1(n4156), .A2(n4323), .ZN(n4460) );
NR2D0 U471 ( .A1(n3109), .A2(n62), .ZN(n4217) );
ND4D0 U472 ( .A1(n4296), .A2(n225), .A3(n4460), .A4(n4217), .
   ZN(n211) );
INVD0 U473 ( .I(n63), .ZN(n3469) );
INVD0 U474 ( .I(n64), .ZN(n3548) );
CKND2D0 U475 ( .A1(counter[0]), .A2(n2508), .ZN(n1385) );
NR2D0 U476 ( .A1(n1385), .A2(n3735), .ZN(n3064) );
NR4D0 U477 ( .A1(n3452), .A2(n2803), .A3(n3064), .A4(n65), .
   ZN(n66) );
ND3D0 U478 ( .A1(n67), .A2(n3548), .A3(n66), .ZN(n74) );
CKND2D0 U479 ( .A1(n4137), .A2(n4465), .ZN(n3509) );
NR4D0 U480 ( .A1(n256), .A2(n3469), .A3(n74), .A4(n3509), .ZN
   (n68) );
IND4D0 U481 ( .A1(n69), .B1(n206), .B2(n80), .B3(n68), .ZN(
   n86) );
NR2D0 U482 ( .A1(n1214), .A2(n1269), .ZN(n857) );
INVD0 U483 ( .I(n857), .ZN(n4083) );
CKND2D0 U484 ( .A1(n4083), .A2(n4144), .ZN(n2968) );
INVD0 U485 ( .I(n2968), .ZN(n3233) );
NR2D0 U486 ( .A1(n3319), .A2(n4225), .ZN(n4467) );
INVD0 U487 ( .I(n3523), .ZN(n1431) );
CKND2D0 U488 ( .A1(n2803), .A2(n1513), .ZN(n2982) );
IND4D0 U489 ( .A1(n4103), .B1(n4467), .B2(n1431), .B3(n2982),
    .ZN(n70) );
NR4D0 U490 ( .A1(n4134), .A2(n71), .A3(n3233), .A4(n70), .ZN(
   n72) );
IND4D0 U491 ( .A1(n73), .B1(n133), .B2(n158), .B3(n72), .ZN(
   n79) );
INR2D0 U492 ( .A1(n3406), .B1(n1862), .ZN(n4093) );
INR3D0 U493 ( .A1(n75), .B1(n4093), .B2(n74), .ZN(n82) );
CKND2D0 U494 ( .A1(n4079), .A2(n4261), .ZN(n3485) );
CKND2D0 U495 ( .A1(n3201), .A2(n3485), .ZN(n91) );
NR4D0 U496 ( .A1(n4250), .A2(n3459), .A3(n222), .A4(n91), .ZN
   (n77) );
INR4D0 U497 ( .A1(n3054), .B1(n3060), .B2(n3495), .B3(n227),
   .ZN(n76) );
ND4D0 U498 ( .A1(n107), .A2(n82), .A3(n77), .A4(n76), .ZN(n78
   ) );
```

```
AOI22D0 U499 ( .A1(n4291), .A2(n79), .B1(n4452), .B2(n78), .
   ZN(n84) );
ND4D0 U500 ( .A1(n80), .A2(n4502), .A3(n4334), .A4(n4107), .
   ZN(n81) );
INR4D0 U501 ( .A1(n82), .B1(n114), .B2(n4173), .B3(n81), .ZN(
   n83) );
AOI32D0 U502 ( .A1(n189), .A2(n84), .A3(n83), .B1(n3243), .B2
   (n84), .ZN(n85)
       );
AOI221D0 U503 ( .A1(n211), .A2(n4289), .B1(n86), .B2(n4342),
   .C(n85), .ZN(
      n88) );
NR2D0 U504 ( .A1(counter[7]), .A2(n87), .ZN(n145) );
CKND2D0 U505 ( .A1(n145), .A2(n1177), .ZN(n4277) );
OAI22D0 U506 ( .A1(n89), .A2(n4408), .B1(n88), .B2(n4277), .
   ZN(n248) );
NR2D0 U507 ( .A1(n1157), .A2(counter[9]), .ZN(n178) );
CKND2D0 U508 ( .A1(n546), .A2(n178), .ZN(n4325) );
INVD0 U509 ( .I(n4325), .ZN(n3709) );
ND3D0 U510 ( .A1(n205), .A2(n160), .A3(n1454), .ZN(n122) );
NR3D0 U511 ( .A1(n4183), .A2(n93), .A3(n122), .ZN(n92) );
INVD0 U512 ( .I(n4223), .ZN(n3175) );
CKND2D0 U513 ( .A1(n4098), .A2(n3175), .ZN(n4146) );
IND4D0 U514 ( .A1(n121), .B1(n154), .B2(n4418), .B3(n4146), .
   ZN(n90) );
NR2D0 U515 ( .A1(n91), .A2(n90), .ZN(n229) );
INVD0 U516 ( .I(n4157), .ZN(n1457) );
NR2D0 U517 ( .A1(n4495), .A2(n1457), .ZN(n4176) );
CKND2D0 U518 ( .A1(n4176), .A2(n4507), .ZN(n4378) );
INVD0 U519 ( .I(n3097), .ZN(n3215) );
NR4D0 U520 ( .A1(n4307), .A2(n4236), .A3(n174), .A4(n3215), .
   ZN(n214) );
IND2D0 U521 ( .A1(n91), .B1(n214), .ZN(n226) );
NR4D0 U522 ( .A1(n1303), .A2(n191), .A3(n4378), .A4(n226), .
   ZN(n187) );
OAI222D0 U523 ( .A1(n54), .A2(n92), .B1(n4517), .B2(n229), .
   C1(n4428), .C2(
      n187), .ZN(n120) );
NR2D0 U524 ( .A1(n4311), .A2(n1457), .ZN(n3193) );
IND4D0 U525 ( .A1(n93), .B1(n169), .B2(n3193), .B3(n1367), .
   ZN(n96) );
CKND2D0 U526 ( .A1(n1719), .A2(n94), .ZN(n3323) );
INVD0 U527 ( .I(n3323), .ZN(n4403) );
NR2D0 U528 ( .A1(n1457), .A2(n4403), .ZN(n4336) );
ND4D0 U529 ( .A1(n206), .A2(n254), .A3(n3097), .A4(n4336), .
   ZN(n95) );
AOI22D0 U530 ( .A1(n4291), .A2(n96), .B1(n4289), .B2(n95), .
   ZN(n104) );
```

```
  ND3D0 U531 ( .A1(n1574), .A2(n225), .A3(n3094), .ZN(n259) );
  IND4D0 U532 ( .A1(n259), .B1(n3356), .B2(n97), .B3(n3323), .
     ZN(n98) );
  OAI31D0 U533 ( .A1(n220), .A2(n3458), .A3(n98), .B(n4452), .
     ZN(n103) );
  OAI21D0 U534 ( .A1(n100), .A2(n99), .B(n4480), .ZN(n102) );
  NR2D0 U535 ( .A1(n216), .A2(n101), .ZN(n3852) );
  INVD0 U536 ( .I(n3852), .ZN(n4190) );
  AOI31D0 U537 ( .A1(n104), .A2(n103), .A3(n102), .B(n4190), .
     ZN(n119) );
  NR2D0 U538 ( .A1(n4373), .A2(n3540), .ZN(n3499) );
  INVD0 U539 ( .I(n3499), .ZN(n3395) );
  NR2D0 U540 ( .A1(n3353), .A2(n3395), .ZN(n3569) );
  IND4D0 U541 ( .A1(n105), .B1(n142), .B2(n3569), .B3(n4317), .
     ZN(n110) );
  INVD0 U542 ( .I(n152), .ZN(n4082) );
  CKND2D0 U543 ( .A1(n3335), .A2(n4082), .ZN(n3420) );
  INVD0 U544 ( .I(n3420), .ZN(n3030) );
  NR2D0 U545 ( .A1(n4426), .A2(n3030), .ZN(n3270) );
  NR2D0 U546 ( .A1(n4374), .A2(n4422), .ZN(n3570) );
  INVD0 U547 ( .I(n3570), .ZN(n4179) );
  INVD0 U548 ( .I(n3392), .ZN(n3447) );
  INR4D0 U549 ( .A1(n107), .B1(n106), .B2(n4179), .B3(n3447), .
     ZN(n108) );
  AOI31D0 U550 ( .A1(n3270), .A2(n108), .A3(n4367), .B(n54), .
     ZN(n109) );
  AOI211D0 U551 ( .A1(n4342), .A2(n110), .B(n257), .C(n109), .
     ZN(n117) );
  ND4D0 U552 ( .A1(n133), .A2(n3094), .A3(n203), .A4(n1478), .
     ZN(n111) );
  OAI31D0 U553 ( .A1(n3574), .A2(n112), .A3(n111), .B(n4291), .
     ZN(n116) );
  NR2D0 U554 ( .A1(n4124), .A2(n4403), .ZN(n1337) );
  INVD0 U555 ( .I(n1337), .ZN(n3455) );
  CKND2D0 U556 ( .A1(n206), .A2(n113), .ZN(n138) );
  OAI31D0 U557 ( .A1(n114), .A2(n3455), .A3(n138), .B(n4480), .
     ZN(n115) );
  INVD0 U558 ( .I(n486), .ZN(n265) );
  CKND2D0 U559 ( .A1(n265), .A2(n178), .ZN(n4148) );
  AOI31D0 U560 ( .A1(n117), .A2(n116), .A3(n115), .B(n4148), .
     ZN(n118) );
  AOI211D0 U561 ( .A1(n3709), .A2(n120), .B(n119), .C(n118), .
     ZN(n246) );
  CKND2D0 U562 ( .A1(n1367), .A2(n3357), .ZN(n4349) );
  CKND2D0 U563 ( .A1(n4349), .A2(n1269), .ZN(n3521) );
  INVD0 U564 ( .I(n3521), .ZN(n3425) );
  INR4D0 U565 ( .A1(n207), .B1(n3425), .B2(n3447), .B3(n121), .
     ZN(n125) );
```

```
NR4D0 U566 ( .A1(n4422), .A2(n4236), .A3(n123), .A4(n122), .
   ZN(n124) );
INVD0 U567 ( .I(n4452), .ZN(n4474) );
OAI22D0 U568 ( .A1(n125), .A2(n4517), .B1(n124), .B2(n4474),
   .ZN(n151) );
NR2D0 U569 ( .A1(n1177), .A2(counter[7]), .ZN(n186) );
INVD0 U570 ( .I(n186), .ZN(n179) );
NR2D0 U571 ( .A1(n216), .A2(n179), .ZN(n3850) );
CKND2D0 U572 ( .A1(n255), .A2(n1454), .ZN(n261) );
IND2D0 U573 ( .A1(n161), .B1(n158), .ZN(n166) );
INR4D0 U574 ( .A1(n155), .B1(n3319), .B2(n261), .B3(n166), .
   ZN(n127) );
CKND2D0 U575 ( .A1(n4212), .A2(n4137), .ZN(n4511) );
ND3D0 U576 ( .A1(n155), .A2(n175), .A3(n3427), .ZN(n156) );
NR4D0 U577 ( .A1(n3109), .A2(n4511), .A3(n166), .A4(n156), .
   ZN(n126) );
OAI22D0 U578 ( .A1(n127), .A2(n3243), .B1(n126), .B2(n4428),
   .ZN(n150) );
CKND2D0 U579 ( .A1(n3462), .A2(n2351), .ZN(n3538) );
AOI211D0 U580 ( .A1(n3175), .A2(n3538), .B(n2980), .C(n128),
   .ZN(n132) );
CKND2D0 U581 ( .A1(n4389), .A2(n4133), .ZN(n4182) );
INVD0 U582 ( .I(n129), .ZN(n3526) );
NR2D0 U583 ( .A1(n1385), .A2(n3897), .ZN(n1430) );
AOI211D0 U584 ( .A1(n3750), .A2(n4081), .B(n3037), .C(n1430),
    .ZN(n130) );
OAI211D0 U585 ( .A1(n3333), .A2(n3332), .B(n3526), .C(n130),
   .ZN(n135) );
NR4D0 U586 ( .A1(n3338), .A2(n4306), .A3(n4182), .A4(n135), .
   ZN(n131) );
ND4D0 U587 ( .A1(n133), .A2(n253), .A3(n132), .A4(n131), .ZN(
   n140) );
NR3D0 U588 ( .A1(n4426), .A2(n134), .A3(n4511), .ZN(n195) );
CKND2D0 U589 ( .A1(n4316), .A2(n4345), .ZN(n3451) );
NR2D0 U590 ( .A1(n4075), .A2(n3064), .ZN(n4259) );
CKND2D0 U591 ( .A1(n4259), .A2(n4078), .ZN(n4215) );
NR4D0 U592 ( .A1(n3540), .A2(n135), .A3(n3451), .A4(n4215), .
   ZN(n136) );
IND4D0 U593 ( .A1(n138), .B1(n137), .B2(n195), .B3(n136), .ZN
   (n139) );
AOI22D0 U594 ( .A1(n4480), .A2(n140), .B1(n4289), .B2(n139),
   .ZN(n148) );
NR2D0 U595 ( .A1(n2987), .A2(n2424), .ZN(n3174) );
INVD0 U596 ( .I(n3174), .ZN(n1336) );
ND4D0 U597 ( .A1(n203), .A2(n207), .A3(n3323), .A4(n1336), .
   ZN(n141) );
OAI21D0 U598 ( .A1(n4172), .A2(n141), .B(n4452), .ZN(n147) );
OR2D0 U599 ( .A1(n3357), .A2(n2088), .Z(n4344) );
```

```
ND4D0 U600 ( .A1(n4336), .A2(n142), .A3(n189), .A4(n4344), .
   ZN(n143) );
OAI31D0 U601 ( .A1(n198), .A2(n144), .A3(n143), .B(n4291), .
   ZN(n146) );
CKND2D0 U602 ( .A1(n145), .A2(counter[8]), .ZN(n3777) );
AOI31D0 U603 ( .A1(n148), .A2(n147), .A3(n146), .B(n3777), .
   ZN(n149) );
AOI221D0 U604 ( .A1(n151), .A2(n3850), .B1(n150), .B2(n3850),
    .C(n149), .ZN(
      n245) );
AOI31D0 U605 ( .A1(n154), .A2(n153), .A3(n152), .B(n54), .ZN(
   n185) );
NR2D0 U606 ( .A1(counter[7]), .A2(counter[8]), .ZN(n301) );
CKND2D0 U607 ( .A1(n1010), .A2(n301), .ZN(n4527) );
INVD0 U608 ( .I(n4527), .ZN(n3707) );
NR2D0 U609 ( .A1(n4232), .A2(n4323), .ZN(n3422) );
AN4D0 U610 ( .A1(n155), .A2(n172), .A3(n3422), .A4(n154), .Z(
   n251) );
NR2D0 U611 ( .A1(n4090), .A2(n4426), .ZN(n4420) );
AN4D0 U612 ( .A1(n3420), .A2(n3427), .A3(n251), .A4(n4420), .
   Z(n164) );
INVD0 U613 ( .I(n156), .ZN(n157) );
AO31D0 U614 ( .A1(n158), .A2(n252), .A3(n157), .B(n3243), .Z(
   n163) );
ND3D0 U615 ( .A1(n160), .A2(n159), .A3(n4133), .ZN(n165) );
OAI31D0 U616 ( .A1(n222), .A2(n161), .A3(n165), .B(n4291), .
   ZN(n162) );
OAI211D0 U617 ( .A1(n164), .A2(n4517), .B(n163), .C(n162), .
   ZN(n184) );
OR4D0 U618 ( .A1(n165), .A2(n212), .A3(n4173), .A4(n3353), .Z
   (n171) );
INVD0 U619 ( .I(n4345), .ZN(n4076) );
INVD0 U620 ( .I(n4507), .ZN(n4473) );
NR2D0 U621 ( .A1(n4076), .A2(n4473), .ZN(n3321) );
NR4D0 U622 ( .A1(n4323), .A2(n4374), .A3(n167), .A4(n166), .
   ZN(n168) );
ND4D0 U623 ( .A1(n169), .A2(n1337), .A3(n3321), .A4(n168), .
   ZN(n170) );
AOI22D0 U624 ( .A1(n4452), .A2(n171), .B1(n4289), .B2(n170),
   .ZN(n182) );
INVD0 U625 ( .I(n213), .ZN(n196) );
INVD0 U626 ( .I(n1363), .ZN(n1720) );
ND4D0 U627 ( .A1(n172), .A2(n196), .A3(n1720), .A4(n3521), .
   ZN(n173) );
OAI31D0 U628 ( .A1(n174), .A2(n202), .A3(n173), .B(n4291), .
   ZN(n181) );
ND4D0 U629 ( .A1(n175), .A2(n4201), .A3(n4212), .A4(n4137), .
   ZN(n176) );
```

```
OAI31D0 U630 ( .A1(n177), .A2(n191), .A3(n176), .B(n4480), .
   ZN(n180) );
INVD0 U631 ( .I(n178), .ZN(n237) );
NR2D0 U632 ( .A1(n237), .A2(n179), .ZN(n3386) );
INVD0 U633 ( .I(n3386), .ZN(n4298) );
AOI31D0 U634 ( .A1(n182), .A2(n181), .A3(n180), .B(n4298), .
   ZN(n183) );
AOI221D0 U635 ( .A1(n185), .A2(n3707), .B1(n184), .B2(n3707),
    .C(n183), .ZN(
      n244) );
CKND2D0 U636 ( .A1(n1010), .A2(n186), .ZN(n4355) );
INVD0 U637 ( .I(n4355), .ZN(n4063) );
NR2D0 U638 ( .A1(n3162), .A2(n4413), .ZN(n3305) );
AN4D0 U639 ( .A1(n187), .A2(n3595), .A3(n4345), .A4(n3305), .
   Z(n201) );
NR2D0 U640 ( .A1(n222), .A2(n188), .ZN(n223) );
ND4D0 U641 ( .A1(n190), .A2(n189), .A3(n223), .A4(n4389), .ZN
   (n193) );
NR4D0 U642 ( .A1(n3030), .A2(n3495), .A3(n191), .A4(n3455), .
   ZN(n258) );
ND4D0 U643 ( .A1(n3427), .A2(n258), .A3(n203), .A4(n1367), .
   ZN(n192) );
AOI22D0 U644 ( .A1(n4480), .A2(n193), .B1(n4452), .B2(n192),
   .ZN(n200) );
NR2D0 U645 ( .A1(n1303), .A2(n4323), .ZN(n194) );
CKND2D0 U646 ( .A1(n194), .A2(n3350), .ZN(n260) );
NR2D0 U647 ( .A1(n4413), .A2(n260), .ZN(n224) );
ND4D0 U648 ( .A1(n196), .A2(n195), .A3(n224), .A4(n1574), .ZN
   (n197) );
OAI21D0 U649 ( .A1(n198), .A2(n197), .B(n4291), .ZN(n199) );
OAI211D0 U650 ( .A1(n201), .A2(n4517), .B(n200), .C(n199), .
   ZN(n242) );
NR2D0 U651 ( .A1(n4307), .A2(n1457), .ZN(n4126) );
IINR4D0 U652 ( .A1(n4126), .A2(n203), .B1(n4403), .B2(n202),
   .ZN(n204) );
ND4D0 U653 ( .A1(n206), .A2(n205), .A3(n3499), .A4(n204), .ZN
   (n210) );
INVD0 U654 ( .I(n4426), .ZN(n3535) );
IND4D0 U655 ( .A1(n208), .B1(n4153), .B2(n207), .B3(n3535), .
   ZN(n209) );
AOI22D0 U656 ( .A1(n4291), .A2(n210), .B1(n4289), .B2(n209),
   .ZN(n219) );
OAI31D0 U657 ( .A1(n213), .A2(n212), .A3(n211), .B(n4480), .
   ZN(n218) );
IND4D0 U658 ( .A1(n261), .B1(n214), .B2(n4418), .B3(n3350), .
   ZN(n215) );
OAI21D0 U659 ( .A1(n227), .A2(n215), .B(n4452), .ZN(n217) );
INVD0 U660 ( .I(n301), .ZN(n236) );
```

```
  NR2D0 U661 ( .A1(n216), .A2(n236), .ZN(n4490) );
  INVD0 U662 ( .I(n4490), .ZN(n3379) );
  AOI31D0 U663 ( .A1(n219), .A2(n218), .A3(n217), .B(n3379), .
     ZN(n241) );
  NR4D0 U664 ( .A1(n4311), .A2(n222), .A3(n221), .A4(n220), .ZN
     (n239) );
  CKND2D0 U665 ( .A1(n4480), .A2(n3709), .ZN(n3012) );
  ND4D0 U666 ( .A1(n225), .A2(n224), .A3(n223), .A4(n4317), .ZN
     (n234) );
  CKND2D0 U667 ( .A1(n4320), .A2(n3350), .ZN(n4450) );
  NR4D0 U668 ( .A1(n227), .A2(n256), .A3(n226), .A4(n4450), .ZN
     (n232) );
  AO31D0 U669 ( .A1(n4212), .A2(n4502), .A3(n228), .B(n4476), .
     Z(n231) );
  INVD0 U670 ( .I(n3833), .ZN(n2788) );
  AO31D0 U671 ( .A1(n2788), .A2(n3505), .A3(n229), .B(n4517), .
     Z(n230) );
  OAI211D0 U672 ( .A1(n232), .A2(n54), .B(n231), .C(n230), .ZN(
     n233) );
  AOI221D0 U673 ( .A1(n235), .A2(n4480), .B1(n234), .B2(n4480),
      .C(n233), .ZN(
        n238) );
  NR2D0 U674 ( .A1(n237), .A2(n236), .ZN(n3311) );
  INVD0 U675 ( .I(n3311), .ZN(n4192) );
  OAI22D0 U676 ( .A1(n239), .A2(n3012), .B1(n238), .B2(n4192),
     .ZN(n240) );
  AOI211D0 U677 ( .A1(n4063), .A2(n242), .B(n241), .C(n240), .
     ZN(n243) );
  ND4D0 U678 ( .A1(n246), .A2(n245), .A3(n244), .A4(n243), .ZN(
     n247) );
  NR4D0 U679 ( .A1(n250), .A2(n249), .A3(n248), .A4(n247), .ZN(
     n270) );
  AOI31D0 U680 ( .A1(n252), .A2(n251), .A3(n3530), .B(n4428), .
     ZN(n268) );
  AOI31D0 U681 ( .A1(n254), .A2(n253), .A3(n1556), .B(n4517), .
     ZN(n267) );
  INVD0 U682 ( .I(n255), .ZN(n3512) );
  INR4D0 U683 ( .A1(n258), .B1(n257), .B2(n256), .B3(n3512), .
     ZN(n264) );
  INR4D0 U684 ( .A1(n262), .B1(n261), .B2(n260), .B3(n259), .ZN
     (n263) );
  OAI22D0 U685 ( .A1(n264), .A2(n54), .B1(n263), .B2(n3243), .
     ZN(n266) );
  CKND2D0 U686 ( .A1(n1010), .A2(n265), .ZN(n3381) );
  INVD0 U687 ( .I(n3381), .ZN(n4492) );
  OAI31D0 U688 ( .A1(n268), .A2(n267), .A3(n266), .B(n4492), .
     ZN(n269) );
```

```
OAI211D0 U689 ( .A1(n271), .A2(n4410), .B(n270), .C(n269), .
   ZN(n636) );
NR2D0 U690 ( .A1(n4351), .A2(counter[9]), .ZN(n645) );
CKND2D0 U691 ( .A1(n3757), .A2(counter[0]), .ZN(n3977) );
INVD0 U692 ( .I(n3977), .ZN(n1208) );
CKND2D0 U693 ( .A1(counter[3]), .A2(n1208), .ZN(n2777) );
INVD0 U694 ( .I(n2777), .ZN(n934) );
CKND2D0 U695 ( .A1(n645), .A2(n934), .ZN(n579) );
NR2D0 U696 ( .A1(n2269), .A2(counter[9]), .ZN(n641) );
CKND2D0 U697 ( .A1(n1362), .A2(counter[4]), .ZN(n1183) );
INVD0 U698 ( .I(n1183), .ZN(n906) );
CKND2D0 U699 ( .A1(counter[3]), .A2(n906), .ZN(n3736) );
INVD0 U700 ( .I(n3736), .ZN(n2638) );
CKND2D0 U701 ( .A1(n641), .A2(n2638), .ZN(n361) );
CKND2D0 U702 ( .A1(n1153), .A2(n1872), .ZN(n735) );
INVD0 U703 ( .I(n735), .ZN(n654) );
NR2D0 U704 ( .A1(counter[4]), .A2(counter[0]), .ZN(n1285) );
INVD0 U705 ( .I(n1285), .ZN(n1213) );
NR2D0 U706 ( .A1(n1213), .A2(counter[3]), .ZN(n946) );
CKND2D0 U707 ( .A1(n654), .A2(n946), .ZN(n349) );
CKND2D0 U708 ( .A1(n645), .A2(n2638), .ZN(n592) );
INVD0 U709 ( .I(n592), .ZN(n611) );
NR2D0 U710 ( .A1(n1859), .A2(counter[4]), .ZN(n3949) );
CKND2D0 U711 ( .A1(n1362), .A2(n3949), .ZN(n1099) );
INVD0 U712 ( .I(n1099), .ZN(n1209) );
CKND2D0 U713 ( .A1(n645), .A2(n1209), .ZN(n526) );
INVD0 U714 ( .I(n526), .ZN(n610) );
NR2D0 U715 ( .A1(n1859), .A2(n3757), .ZN(n2760) );
CKND2D0 U716 ( .A1(counter[0]), .A2(n2760), .ZN(n1194) );
NR2D0 U717 ( .A1(n735), .A2(n1194), .ZN(n500) );
CKND2D0 U718 ( .A1(n2978), .A2(n1209), .ZN(n1139) );
NR2D0 U719 ( .A1(counter[9]), .A2(n1139), .ZN(n314) );
NR4D0 U720 ( .A1(n611), .A2(n610), .A3(n500), .A4(n314), .ZN(
   n570) );
CKND2D0 U721 ( .A1(n641), .A2(n934), .ZN(n431) );
NR2D0 U722 ( .A1(n3757), .A2(n1362), .ZN(n1234) );
CKND2D0 U723 ( .A1(n1859), .A2(n1234), .ZN(n1270) );
INVD0 U724 ( .I(n1270), .ZN(n1043) );
CKND2D0 U725 ( .A1(n654), .A2(n1043), .ZN(n331) );
INVD0 U726 ( .I(n331), .ZN(n386) );
INVD0 U727 ( .I(n1194), .ZN(n1182) );
CKND2D0 U728 ( .A1(n645), .A2(n1182), .ZN(n432) );
NR2D0 U729 ( .A1(n3757), .A2(counter[3]), .ZN(n929) );
NR2D0 U730 ( .A1(n2269), .A2(counter[0]), .ZN(n2287) );
CKND2D0 U731 ( .A1(n929), .A2(n2287), .ZN(n3759) );
NR2D0 U732 ( .A1(n3759), .A2(counter[9]), .ZN(n299) );
INVD0 U733 ( .I(n299), .ZN(n488) );
CKND2D0 U734 ( .A1(n432), .A2(n488), .ZN(n326) );
```

```
  NR2D0 U735 ( .A1(n3977), .A2(counter[3]), .ZN(n2637) );
  INVD0 U736 ( .I(n2637), .ZN(n2617) );
  CKND2D0 U737 ( .A1(n1153), .A2(n2654), .ZN(n741) );
  AOI21D0 U738 ( .A1(n1183), .A2(n2617), .B(n741), .ZN(n272) );
  INR4D0 U739 ( .A1(n431), .B1(n386), .B2(n326), .B3(n272), .ZN
     (n273) );
  CKND2D0 U740 ( .A1(n645), .A2(n1043), .ZN(n591) );
  INVD0 U741 ( .I(n741), .ZN(n642) );
  CKND2D0 U742 ( .A1(n642), .A2(n1043), .ZN(n529) );
  CKND2D0 U743 ( .A1(n591), .A2(n529), .ZN(n340) );
  INVD0 U744 ( .I(n641), .ZN(n380) );
  CKND2D0 U745 ( .A1(n1859), .A2(n3757), .ZN(n2732) );
  NR2D0 U746 ( .A1(n380), .A2(n2732), .ZN(n573) );
  AOI21D0 U747 ( .A1(n946), .A2(n642), .B(n573), .ZN(n561) );
  INVD0 U748 ( .I(n561), .ZN(n596) );
  IINR4D0 U749 ( .A1(n570), .A2(n273), .B1(n340), .B2(n596), .
     ZN(n283) );
  ND4D0 U750 ( .A1(n579), .A2(n361), .A3(n349), .A4(n283), .ZN(
     n274) );
  CKND2D0 U751 ( .A1(n2978), .A2(n1182), .ZN(n941) );
  NR2D0 U752 ( .A1(counter[9]), .A2(n941), .ZN(n408) );
  NR2D0 U753 ( .A1(n741), .A2(n1194), .ZN(n515) );
  INVD0 U754 ( .I(n515), .ZN(n586) );
  INVD0 U755 ( .I(n2732), .ZN(n3731) );
  CKND2D0 U756 ( .A1(n645), .A2(n3731), .ZN(n455) );
  CKND2D0 U757 ( .A1(n586), .A2(n455), .ZN(n533) );
  NR2D0 U758 ( .A1(n408), .A2(n533), .ZN(n558) );
  NR2D0 U759 ( .A1(n1183), .A2(counter[3]), .ZN(n1018) );
  CKND2D0 U760 ( .A1(n645), .A2(n1018), .ZN(n338) );
  INVD0 U761 ( .I(n338), .ZN(n294) );
  CKND2D0 U762 ( .A1(n642), .A2(n1209), .ZN(n286) );
  INVD0 U763 ( .I(n286), .ZN(n365) );
  NR2D0 U764 ( .A1(n294), .A2(n365), .ZN(n439) );
  CKND2D0 U765 ( .A1(n642), .A2(n934), .ZN(n316) );
  IND4D0 U766 ( .A1(n274), .B1(n558), .B2(n439), .B3(n316), .ZN
     (n277) );
  CKND2D0 U767 ( .A1(n654), .A2(n2638), .ZN(n330) );
  INVD0 U768 ( .I(n330), .ZN(n600) );
  NR2D0 U769 ( .A1(n408), .A2(n600), .ZN(n447) );
  CKND2D0 U770 ( .A1(n641), .A2(n1043), .ZN(n585) );
  INVD0 U771 ( .I(n585), .ZN(n444) );
  CKND2D0 U772 ( .A1(n654), .A2(n2637), .ZN(n385) );
  INVD0 U773 ( .I(n385), .ZN(n511) );
  NR2D0 U774 ( .A1(n515), .A2(n511), .ZN(n371) );
  INVD0 U775 ( .I(n1018), .ZN(n945) );
  NR2D0 U776 ( .A1(n735), .A2(n945), .ZN(n322) );
  INVD0 U777 ( .I(n322), .ZN(n387) );
  INVD0 U778 ( .I(n645), .ZN(n644) );
```

```
INVD0 U779 ( .I(n946), .ZN(n982) );
NR2D0 U780 ( .A1(n644), .A2(n982), .ZN(n328) );
INVD0 U781 ( .I(n328), .ZN(n382) );
ND4D0 U782 ( .A1(n371), .A2(n316), .A3(n387), .A4(n382), .ZN(
   n336) );
NR3D0 U783 ( .A1(n444), .A2(n274), .A3(n336), .ZN(n285) );
NR2D0 U784 ( .A1(n735), .A2(n1099), .ZN(n517) );
INVD0 U785 ( .I(n517), .ZN(n275) );
OR2D0 U786 ( .A1(n2274), .A2(counter[6]), .Z(n4010) );
AOI31D0 U787 ( .A1(n447), .A2(n285), .A3(n275), .B(n4010), .
   ZN(n276) );
AOI211D0 U788 ( .A1(n3933), .A2(n277), .B(n276), .C(n596), .
   ZN(n313) );
AOI21D0 U789 ( .A1(n1270), .A2(n3736), .B(n741), .ZN(n278) );
OR3D0 U790 ( .A1(n946), .A2(n294), .A3(n278), .Z(n419) );
NR2D0 U791 ( .A1(n419), .A2(n299), .ZN(n306) );
CKND2D0 U792 ( .A1(n642), .A2(n2637), .ZN(n464) );
CKND2D0 U793 ( .A1(n464), .A2(n361), .ZN(n298) );
IND2D0 U794 ( .A1(n298), .B1(n387), .ZN(n282) );
CKND2D0 U795 ( .A1(n642), .A2(n1018), .ZN(n438) );
CKND2D0 U796 ( .A1(n330), .A2(n438), .ZN(n315) );
AOI21D0 U797 ( .A1(n1865), .A2(n934), .B(n315), .ZN(n279) );
OAI211D0 U798 ( .A1(n1213), .A2(n735), .B(n558), .C(n279), .
   ZN(n281) );
INR3D0 U799 ( .A1(n306), .B1(n282), .B2(n281), .ZN(n293) );
ND4D0 U800 ( .A1(n570), .A2(n293), .A3(n316), .A4(n591), .ZN(
   n291) );
NR2D0 U801 ( .A1(n741), .A2(n3736), .ZN(n544) );
NR2D0 U802 ( .A1(n444), .A2(n544), .ZN(n426) );
INVD0 U803 ( .I(n1234), .ZN(n1651) );
OAI21D0 U804 ( .A1(n934), .A2(n2760), .B(n645), .ZN(n280) );
OAI211D0 U805 ( .A1(n735), .A2(n1651), .B(n286), .C(n280), .
   ZN(n292) );
INR4D0 U806 ( .A1(n316), .B1(n610), .B2(n281), .B3(n292), .ZN
   (n296) );
INVD0 U807 ( .I(n314), .ZN(n605) );
ND4D0 U808 ( .A1(n296), .A2(n605), .A3(n591), .A4(n385), .ZN(
   n303) );
INR4D0 U809 ( .A1(n426), .B1(n573), .B2(n282), .B3(n303), .ZN
   (n289) );
NR2D0 U810 ( .A1(n294), .A2(n511), .ZN(n614) );
CKND2D0 U811 ( .A1(n614), .A2(n579), .ZN(n565) );
INVD0 U812 ( .I(n3949), .ZN(n3754) );
OAI211D0 U813 ( .A1(n3754), .A2(n735), .B(n283), .C(n286), .
   ZN(n305) );
NR2D0 U814 ( .A1(n644), .A2(n2617), .ZN(n409) );
INVD0 U815 ( .I(n409), .ZN(n359) );
```

```
ND4D0 U816 ( .A1(n447), .A2(n361), .A3(n316), .A4(n359), .ZN(
   n284) );
CKND2D1 U817 ( .A1(n2274), .A2(counter[6]), .ZN(n4059) );
INVD0 U818 ( .I(n4059), .ZN(n4000) );
OAI31D0 U819 ( .A1(n565), .A2(n305), .A3(n284), .B(n4000), .
   ZN(n288) );
CKND2D0 U820 ( .A1(n654), .A2(n934), .ZN(n479) );
CKAN2D0 U821 ( .A1(counter[6]), .A2(counter[5]), .Z(n3844) );
CKND1 U822 ( .I(n3844), .ZN(n4045) );
AO31D0 U823 ( .A1(n286), .A2(n479), .A3(n285), .B(n4045), .Z(
   n287) );
OAI211D0 U824 ( .A1(n289), .A2(n4010), .B(n288), .C(n287), .
   ZN(n290) );
AOI211D0 U825 ( .A1(n3933), .A2(n291), .B(n1075), .C(n290), .
   ZN(n312) );
INR4D0 U826 ( .A1(n293), .B1(n444), .B2(n511), .B3(n292), .ZN
   (n302) );
NR2D0 U827 ( .A1(n294), .A2(n322), .ZN(n295) );
ND4D0 U828 ( .A1(n561), .A2(n296), .A3(n295), .A4(n529), .ZN(
   n297) );
INVD0 U829 ( .I(n4010), .ZN(n3769) );
OAI31D0 U830 ( .A1(n299), .A2(n298), .A3(n297), .B(n3769), .
   ZN(n300) );
OAI211D0 U831 ( .A1(n302), .A2(n4045), .B(n301), .C(n300), .
   ZN(n310) );
INVD0 U832 ( .I(n303), .ZN(n308) );
ND4D0 U833 ( .A1(n349), .A2(n338), .A3(n585), .A4(n387), .ZN(
   n304) );
OAI31D0 U834 ( .A1(n533), .A2(n305), .A3(n304), .B(n4000), .
   ZN(n307) );
AOI32D0 U835 ( .A1(n308), .A2(n307), .A3(n306), .B1(n4038), .
   B2(n307), .ZN(
      n309) );
OAI211D0 U836 ( .A1(n310), .A2(n309), .B(n916), .C(counter
   [11]), .ZN(n311)
       );
AOI211D0 U837 ( .A1(n313), .A2(counter[8]), .B(n312), .C(n311
   ), .ZN(n635) );
INVD0 U838 ( .I(n591), .ZN(n357) );
NR2D0 U839 ( .A1(n314), .A2(n357), .ZN(n320) );
NR2D0 U840 ( .A1(n1153), .A2(n2269), .ZN(n535) );
INVD0 U841 ( .I(n535), .ZN(n498) );
NR2D0 U842 ( .A1(n1651), .A2(n498), .ZN(n383) );
NR2D0 U843 ( .A1(n1153), .A2(n3390), .ZN(n567) );
INVD0 U844 ( .I(n567), .ZN(n406) );
AOI21D0 U845 ( .A1(n2617), .A2(n945), .B(n406), .ZN(n563) );
NR4D0 U846 ( .A1(n500), .A2(n408), .A3(n383), .A4(n563), .ZN(
   n332) );
```

```
  CKND2D0 U847 ( .A1(counter[9]), .A2(n1243), .ZN(n363) );
  NR2D0 U848 ( .A1(n1270), .A2(n363), .ZN(n543) );
  NR2D0 U849 ( .A1(n365), .A2(n543), .ZN(n433) );
  CKND2D0 U850 ( .A1(n1872), .A2(n934), .ZN(n2773) );
  CKND2D0 U851 ( .A1(n585), .A2(n2773), .ZN(n346) );
  IINR4D0 U852 ( .A1(n529), .A2(n433), .B1(n315), .B2(n346), .
     ZN(n413) );
  NR2D0 U853 ( .A1(n945), .A2(n363), .ZN(n429) );
  IND2D0 U854 ( .A1(n429), .B1(n431), .ZN(n377) );
  NR2D0 U855 ( .A1(n1153), .A2(n2265), .ZN(n362) );
  CKND2D0 U856 ( .A1(n2637), .A2(n362), .ZN(n437) );
  INVD0 U857 ( .I(n363), .ZN(n462) );
  CKND2D0 U858 ( .A1(n946), .A2(n462), .ZN(n604) );
  CKND2D0 U859 ( .A1(n437), .A2(n604), .ZN(n323) );
  CKND2D0 U860 ( .A1(n1182), .A2(n362), .ZN(n369) );
  CKND2D0 U861 ( .A1(n316), .A2(n369), .ZN(n420) );
  NR3D0 U862 ( .A1(n1153), .A2(n3737), .A3(n3736), .ZN(n403) );
  NR4D0 U863 ( .A1(n377), .A2(n323), .A3(n420), .A4(n403), .ZN(
     n317) );
  CKND2D0 U864 ( .A1(n929), .A2(n362), .ZN(n485) );
  ND4D0 U865 ( .A1(n413), .A2(n317), .A3(n331), .A4(n485), .ZN(
     n454) );
  INVD0 U866 ( .I(n544), .ZN(n410) );
  NR2D0 U867 ( .A1(n3977), .A2(n498), .ZN(n504) );
  INVD0 U868 ( .I(n504), .ZN(n366) );
  CKND2D0 U869 ( .A1(n1209), .A2(n462), .ZN(n374) );
  NR2D0 U870 ( .A1(n3736), .A2(n498), .ZN(n556) );
  INVD0 U871 ( .I(n556), .ZN(n430) );
  ND4D0 U872 ( .A1(n410), .A2(n366), .A3(n374), .A4(n430), .ZN(
     n318) );
  NR2D0 U873 ( .A1(n454), .A2(n318), .ZN(n319) );
  ND4D0 U874 ( .A1(n320), .A2(n332), .A3(n319), .A4(n382), .ZN(
     n352) );
  INVD0 U875 ( .I(n565), .ZN(n355) );
  CKND2D0 U876 ( .A1(n946), .A2(n567), .ZN(n528) );
  CKND2D0 U877 ( .A1(n528), .A2(n561), .ZN(n345) );
  INVD0 U878 ( .I(n345), .ZN(n520) );
  NR2D0 U879 ( .A1(n1194), .A2(n406), .ZN(n503) );
  CKND2D0 U880 ( .A1(n946), .A2(n535), .ZN(n518) );
  CKND2D0 U881 ( .A1(n1209), .A2(n362), .ZN(n449) );
  CKND2D0 U882 ( .A1(n518), .A2(n449), .ZN(n390) );
  INVD0 U883 ( .I(n390), .ZN(n411) );
  CKND2D0 U884 ( .A1(n1018), .A2(n535), .ZN(n512) );
  OAI211D0 U885 ( .A1(n363), .A2(n3977), .B(n411), .C(n512), .
     ZN(n321) );
  NR4D0 U886 ( .A1(n503), .A2(n322), .A3(n326), .A4(n321), .ZN(
     n560) );
  NR2D0 U887 ( .A1(n1153), .A2(n1139), .ZN(n572) );
```

```
  NR2D0 U888 ( .A1(n517), .A2(n572), .ZN(n613) );
  ND4D0 U889 ( .A1(n355), .A2(n520), .A3(n560), .A4(n613), .ZN(
     n344) );
  INVD0 U890 ( .I(n323), .ZN(n373) );
  NR2D0 U891 ( .A1(n1194), .A2(n363), .ZN(n525) );
  INVD0 U892 ( .I(n525), .ZN(n348) );
  CKND2D0 U893 ( .A1(n2637), .A2(n462), .ZN(n375) );
  CKND2D0 U894 ( .A1(n348), .A2(n375), .ZN(n333) );
  INVD0 U895 ( .I(n454), .ZN(n569) );
  NR2D0 U896 ( .A1(n2777), .A2(n363), .ZN(n516) );
  OAI22D0 U897 ( .A1(n1213), .A2(n735), .B1(n1183), .B2(n498),
     .ZN(n324) );
  INR3D0 U898 ( .A1(n464), .B1(n516), .B2(n324), .ZN(n446) );
  CKND2D0 U899 ( .A1(n374), .A2(n518), .ZN(n402) );
  AN3D0 U900 ( .A1(n449), .A2(n592), .A3(n387), .Z(n606) );
  CKND2D0 U901 ( .A1(n946), .A2(n362), .ZN(n329) );
  INVD0 U902 ( .I(n329), .ZN(n557) );
  NR2D0 U903 ( .A1(n596), .A2(n557), .ZN(n497) );
  CKND2D0 U904 ( .A1(n2638), .A2(n567), .ZN(n376) );
  ND3D0 U905 ( .A1(n1139), .A2(n579), .A3(n376), .ZN(n624) );
  NR2D0 U906 ( .A1(n2777), .A2(n498), .ZN(n566) );
  NR2D0 U907 ( .A1(n383), .A2(n566), .ZN(n612) );
  OAI211D0 U908 ( .A1(n3977), .A2(n406), .B(n612), .C(n586), .
     ZN(n325) );
  INR4D0 U909 ( .A1(n497), .B1(n326), .B2(n624), .B3(n325), .ZN
     (n327) );
  CKND2D0 U910 ( .A1(n606), .A2(n327), .ZN(n436) );
  OR2D0 U911 ( .A1(n500), .A2(n357), .Z(n458) );
  NR4D0 U912 ( .A1(n544), .A2(n402), .A3(n436), .A4(n458), .ZN(
     n461) );
  ND4D0 U913 ( .A1(n614), .A2(n569), .A3(n446), .A4(n461), .ZN(
     n407) );
  OAI31D0 U914 ( .A1(n328), .A2(n333), .A3(n407), .B(n4000), .
     ZN(n342) );
  OAI211D0 U915 ( .A1(n3754), .A2(n406), .B(n526), .C(n329), .
     ZN(n350) );
  INVD0 U916 ( .I(n543), .ZN(n450) );
  CKND2D0 U917 ( .A1(n488), .A2(n450), .ZN(n335) );
  IND4D0 U918 ( .A1(n333), .B1(n332), .B2(n331), .B3(n330), .ZN
     (n334) );
  INR4D0 U919 ( .A1(n446), .B1(n350), .B2(n335), .B3(n334), .ZN
     (n582) );
  CKND2D0 U920 ( .A1(n2654), .A2(n1043), .ZN(n1096) );
  INVD0 U921 ( .I(n336), .ZN(n405) );
  NR2D0 U922 ( .A1(n3736), .A2(n363), .ZN(n493) );
  INVD0 U923 ( .I(n493), .ZN(n584) );
  OAI211D0 U924 ( .A1(n1096), .A2(n1153), .B(n405), .C(n584), .
     ZN(n337) );
```

```
INR4D0 U925 ( .A1(n369), .B1(n544), .B2(n624), .B3(n337), .ZN
   (n594) );
CKND2D0 U926 ( .A1(n1043), .A2(n362), .ZN(n603) );
CKND2D0 U927 ( .A1(n338), .A2(n603), .ZN(n356) );
CKND2D0 U928 ( .A1(n2638), .A2(n362), .ZN(n442) );
CKND2D0 U929 ( .A1(n592), .A2(n442), .ZN(n368) );
INVD0 U930 ( .I(n402), .ZN(n588) );
IND3D0 U931 ( .A1(n368), .B1(n588), .B2(n438), .ZN(n422) );
NR4D0 U932 ( .A1(n504), .A2(n377), .A3(n356), .A4(n422), .ZN(
   n339) );
ND3D0 U933 ( .A1(n582), .A2(n594), .A3(n339), .ZN(n347) );
NR4D0 U934 ( .A1(n573), .A2(n409), .A3(n340), .A4(n347), .ZN(
   n341) );
AOI32D0 U935 ( .A1(n373), .A2(n342), .A3(n341), .B1(n4045), .
   B2(n342), .ZN(
      n343) );
AOI221D0 U936 ( .A1(n352), .A2(n3933), .B1(n344), .B2(n3933),
    .C(n343), .ZN(
      n400) );
OAI31D0 U937 ( .A1(n347), .A2(n346), .A3(n345), .B(n3769), .
   ZN(n399) );
NR2D0 U938 ( .A1(n380), .A2(n982), .ZN(n353) );
NR2D0 U939 ( .A1(n1153), .A2(n1096), .ZN(n427) );
CKND2D0 U940 ( .A1(n349), .A2(n348), .ZN(n499) );
NR4D0 U941 ( .A1(n515), .A2(n409), .A3(n350), .A4(n499), .ZN(
   n351) );
IND4D0 U942 ( .A1(n352), .B1(n351), .B2(n464), .B3(n376), .ZN
   (n521) );
INR4D0 U943 ( .A1(n3759), .B1(n353), .B2(n427), .B3(n521), .
   ZN(n354) );
INVD0 U944 ( .I(n572), .ZN(n583) );
ND4D0 U945 ( .A1(n355), .A2(n354), .A3(n375), .A4(n583), .ZN(
   n397) );
CKND2D0 U946 ( .A1(n2265), .A2(n380), .ZN(n421) );
AOI211D0 U947 ( .A1(n934), .A2(n421), .B(n357), .C(n356), .ZN
   (n358) );
OAI211D0 U948 ( .A1(n562), .A2(n561), .B(n588), .C(n358), .ZN
   (n391) );
ND4D0 U949 ( .A1(n582), .A2(n359), .A3(n604), .A4(n528), .ZN(
   n360) );
INR3D0 U950 ( .A1(n432), .B1(n444), .B2(n360), .ZN(n394) );
INVD0 U951 ( .I(n361), .ZN(n564) );
CKND2D0 U952 ( .A1(n1018), .A2(n362), .ZN(n381) );
INVD0 U953 ( .I(n381), .ZN(n524) );
AOI21D0 U954 ( .A1(n741), .A2(n363), .B(n945), .ZN(n364) );
NR4D0 U955 ( .A1(n564), .A2(n365), .A3(n524), .A4(n364), .ZN(
   n367) );
```

```
  ND4D0 U956 ( .A1(n394), .A2(n367), .A3(n366), .A4(n449), .ZN(
     n401) );
  NR4D0 U957 ( .A1(n624), .A2(n391), .A3(n401), .A4(n368), .ZN(
     n370) );
  AOI31D0 U958 ( .A1(n371), .A2(n370), .A3(n369), .B(n4059), .
     ZN(n396) );
  NR4D0 U959 ( .A1(n408), .A2(n525), .A3(n516), .A4(n503), .ZN(
     n372) );
  CKND2D0 U960 ( .A1(n373), .A2(n372), .ZN(n482) );
  NR2D0 U961 ( .A1(n1043), .A2(n934), .ZN(n3896) );
  OAI211D0 U962 ( .A1(n3896), .A2(n406), .B(n374), .C(n464), .
     ZN(n555) );
  CKND2D0 U963 ( .A1(n376), .A2(n375), .ZN(n423) );
  NR2D0 U964 ( .A1(n2617), .A2(n498), .ZN(n481) );
  AO31D0 U965 ( .A1(counter[4]), .A2(counter[0]), .A3(n645), .B
     (n377), .Z(n378) );
  NR4D0 U966 ( .A1(n481), .A2(n564), .A3(n596), .A4(n378), .ZN(
     n581) );
  ND4D0 U967 ( .A1(n570), .A2(n581), .A3(n583), .A4(n584), .ZN(
     n379) );
  NR4D0 U968 ( .A1(n482), .A2(n555), .A3(n423), .A4(n379), .ZN(
     n530) );
  ND4D0 U969 ( .A1(n439), .A2(n530), .A3(n380), .A4(n442), .ZN(
     n514) );
  IND4D0 U970 ( .A1(n383), .B1(n382), .B2(n528), .B3(n381), .ZN
     (n384) );
  NR2D0 U971 ( .A1(n514), .A2(n384), .ZN(n538) );
  ND3D0 U972 ( .A1(n538), .A2(n529), .A3(n385), .ZN(n453) );
  NR2D0 U973 ( .A1(n1099), .A2(n406), .ZN(n534) );
  NR2D0 U974 ( .A1(n556), .A2(n534), .ZN(n465) );
  INVD0 U975 ( .I(n579), .ZN(n494) );
  NR2D0 U976 ( .A1(n2265), .A2(n982), .ZN(n2642) );
  NR2D0 U977 ( .A1(n2617), .A2(n406), .ZN(n539) );
  NR2D0 U978 ( .A1(n2642), .A2(n539), .ZN(n388) );
  NR2D0 U979 ( .A1(n945), .A2(n406), .ZN(n484) );
  NR2D0 U980 ( .A1(n386), .A2(n484), .ZN(n601) );
  ND4D0 U981 ( .A1(n388), .A2(n601), .A3(n387), .A4(n603), .ZN(
     n452) );
  INR4D0 U982 ( .A1(n465), .B1(n494), .B2(n420), .B3(n452), .ZN
     (n513) );
  OAI21D0 U983 ( .A1(counter[0]), .A2(n735), .B(n513), .ZN(n389
     ) );
  OAI31D0 U984 ( .A1(n390), .A2(n453), .A3(n389), .B(n3769), .
     ZN(n393) );
  IINR4D0 U985 ( .A1(n594), .A2(n437), .B1(n503), .B2(n391), .
     ZN(n392) );
  AOI32D0 U986 ( .A1(n394), .A2(n393), .A3(n392), .B1(n4045), .
     B2(n393), .ZN(
```

```
      n395) );
AOI211D0 U987 ( .A1(n3933), .A2(n397), .B(n396), .C(n395), .
   ZN(n398) );
AOI32D0 U988 ( .A1(n400), .A2(n1157), .A3(n399), .B1(counter
   [10]), .B2(n398),
      .ZN(n478) );
INVD0 U989 ( .I(n401), .ZN(n593) );
AOI211D0 U990 ( .A1(n2760), .A2(n567), .B(n403), .C(n402), .
   ZN(n404) );
ND4D0 U991 ( .A1(n405), .A2(n593), .A3(n404), .A4(n479), .ZN(
   n418) );
NR2D0 U992 ( .A1(n906), .A2(n1208), .ZN(n1694) );
INVD0 U993 ( .I(n1694), .ZN(n2776) );
NR2D0 U994 ( .A1(n2776), .A2(n406), .ZN(n460) );
NR4D0 U995 ( .A1(n408), .A2(n481), .A3(n460), .A4(n407), .ZN(
   n416) );
NR2D0 U996 ( .A1(n515), .A2(n409), .ZN(n412) );
ND4D0 U997 ( .A1(n412), .A2(n411), .A3(n488), .A4(n410), .ZN(
   n414) );
IND4D0 U998 ( .A1(n414), .B1(n413), .B2(n530), .B3(n513), .ZN
   (n415) );
MOAI22D0 U999 ( .A1(n416), .A2(n4045), .B1(n415), .B2(n3933),
    .ZN(n417) );
AOI221D0 U1000 ( .A1(n419), .A2(n3769), .B1(n418), .B2(n3769)
   , .C(n417),
      .ZN(n476) );
ND3D0 U1001 ( .A1(n601), .A2(n479), .A3(n604), .ZN(n443) );
INVD0 U1002 ( .I(n420), .ZN(n541) );
AOI211D0 U1003 ( .A1(n1209), .A2(n421), .B(n539), .C(n564), .
   ZN(n425) );
NR4D0 U1004 ( .A1(n534), .A2(n423), .A3(n533), .A4(n422), .ZN
   (n424) );
ND4D0 U1005 ( .A1(n426), .A2(n541), .A3(n425), .A4(n424), .ZN
   (n491) );
OAI31D0 U1006 ( .A1(n642), .A2(n645), .A3(n535), .B(n1043), .
   ZN(n496) );
NR2D0 U1007 ( .A1(n1153), .A2(n2773), .ZN(n597) );
AOI211D0 U1008 ( .A1(n906), .A2(n654), .B(n427), .C(n597), .
   ZN(n428) );
CKAN2D0 U1009 ( .A1(n428), .A2(n464), .Z(n575) );
IIND4D0 U1010 ( .A1(n491), .A2(n429), .B1(n496), .B2(n575), .
   ZN(n523) );
NR4D0 U1011 ( .A1(n523), .A2(n493), .A3(n557), .A4(n516), .ZN
   (n507) );
INVD0 U1012 ( .I(n507), .ZN(n435) );
CKND2D0 U1013 ( .A1(n934), .A2(n567), .ZN(n489) );
AN4D0 U1014 ( .A1(n433), .A2(n432), .A3(n431), .A4(n430), .Z(
   n480) );
```

```
AN4D0 U1015 ( .A1(n579), .A2(n489), .A3(n520), .A4(n480), .Z(
   n434) );
ND4D0 U1016 ( .A1(n434), .A2(n3759), .A3(n941), .A4(n583), .
   ZN(n522) );
OAI31D0 U1017 ( .A1(n443), .A2(n435), .A3(n522), .B(n4000), .
   ZN(n475) );
NR3D0 U1018 ( .A1(n436), .A2(n564), .A3(n511), .ZN(n457) );
INVD0 U1019 ( .I(n457), .ZN(n473) );
ND4D0 U1020 ( .A1(n1096), .A2(n455), .A3(n438), .A4(n437), .
   ZN(n441) );
INVD0 U1021 ( .I(n597), .ZN(n448) );
IND4D0 U1022 ( .A1(n503), .B1(n541), .B2(n439), .B3(n448), .
   ZN(n440) );
IINR4D0 U1023 ( .A1(n442), .A2(n528), .B1(n441), .B2(n440), .
   ZN(n580) );
IND3D0 U1024 ( .A1(n443), .B1(n526), .B2(n485), .ZN(n502) );
NR4D0 U1025 ( .A1(n444), .A2(n543), .A3(n534), .A4(n502), .ZN
   (n445) );
ND4D0 U1026 ( .A1(n447), .A2(n446), .A3(n580), .A4(n445), .ZN
   (n472) );
OAI21D0 U1027 ( .A1(n1183), .A2(n741), .B(n448), .ZN(n510) );
OAI211D0 U1028 ( .A1(n3754), .A2(n735), .B(n450), .C(n449), .
   ZN(n451) );
NR4D0 U1029 ( .A1(n453), .A2(n452), .A3(n510), .A4(n451), .ZN
   (n470) );
NR4D0 U1030 ( .A1(n544), .A2(n481), .A3(n454), .A4(n499), .ZN
   (n609) );
OA21D0 U1031 ( .A1(n945), .A2(n1153), .B(n455), .Z(n456) );
ND4D0 U1032 ( .A1(n457), .A2(n609), .A3(n456), .A4(n526), .ZN
   (n459) );
OAI31D0 U1033 ( .A1(n460), .A2(n459), .A3(n458), .B(n3769), .
   ZN(n469) );
INVD0 U1034 ( .I(n461), .ZN(n467) );
OAI21D0 U1035 ( .A1(n1018), .A2(n2760), .B(n462), .ZN(n463) )
   ;
ND4D0 U1036 ( .A1(n465), .A2(n580), .A3(n464), .A4(n463), .ZN
   (n466) );
OAI31D0 U1037 ( .A1(n502), .A2(n467), .A3(n466), .B(n3933), .
   ZN(n468) );
OAI211D0 U1038 ( .A1(n470), .A2(n4059), .B(n469), .C(n468), .
   ZN(n471) );
AOI221D0 U1039 ( .A1(n473), .A2(n3844), .B1(n472), .B2(n3844)
   , .C(n471),
      .ZN(n474) );
AOI32D0 U1040 ( .A1(n476), .A2(counter[10]), .A3(n475), .B1(
   n474), .B2(n1157), .ZN(n477) );
AOI221D0 U1041 ( .A1(counter[8]), .A2(n478), .B1(n1177), .B2(
   n477), .C(
```

```
      counter[7]), .ZN(n632) );
INR2D0 U1042 ( .A1(n479), .B1(n566), .ZN(n587) );
IIND4D0 U1043 ( .A1(n482), .A2(n481), .B1(n614), .B2(n480), .
   ZN(n483) );
INR3D0 U1044 ( .A1(n485), .B1(n484), .B2(n483), .ZN(n492) );
ND4D0 U1045 ( .A1(n497), .A2(n587), .A3(n492), .A4(n526), .ZN
   (n487) );
AOI221D0 U1046 ( .A1(n523), .A2(n3933), .B1(n487), .B2(n3933)
   , .C(n486),
      .ZN(n554) );
CKND2D0 U1047 ( .A1(n488), .A2(n528), .ZN(n501) );
AOI32D0 U1048 ( .A1(n1651), .A2(n489), .A3(n982), .B1(n735),
   .B2(n489), .ZN(
      n490) );
INR4D0 U1049 ( .A1(n492), .B1(n491), .B2(n501), .B3(n490), .
   ZN(n576) );
INR4D0 U1050 ( .A1(n512), .B1(n494), .B2(n493), .B3(n566), .
   ZN(n495) );
ND4D0 U1051 ( .A1(n497), .A2(n496), .A3(n576), .A4(n495), .ZN
   (n509) );
NR2D0 U1052 ( .A1(n1194), .A2(n498), .ZN(n571) );
NR4D0 U1053 ( .A1(n573), .A2(n500), .A3(n571), .A4(n499), .ZN
   (n506) );
NR4D0 U1054 ( .A1(n504), .A2(n503), .A3(n502), .A4(n501), .ZN
   (n505) );
ND4D0 U1055 ( .A1(n614), .A2(n507), .A3(n506), .A4(n505), .ZN
   (n508) );
AOI22D0 U1056 ( .A1(n3769), .A2(n509), .B1(n4000), .B2(n508),
    .ZN(n553) );
NR4D0 U1057 ( .A1(n511), .A2(n517), .A3(n543), .A4(n510), .ZN
   (n602) );
ND3D0 U1058 ( .A1(n513), .A2(n602), .A3(n512), .ZN(n532) );
OAI31D0 U1059 ( .A1(n515), .A2(n514), .A3(n532), .B(n3844), .
   ZN(n552) );
INR4D0 U1060 ( .A1(n518), .B1(n564), .B2(n517), .B3(n516), .
   ZN(n519) );
IND4D0 U1061 ( .A1(n521), .B1(n520), .B2(n606), .B3(n519), .
   ZN(n550) );
NR4D0 U1062 ( .A1(n525), .A2(n524), .A3(n523), .A4(n522), .ZN
   (n527) );
AOI31D0 U1063 ( .A1(n527), .A2(n2617), .A3(n526), .B(n4059),
   .ZN(n549) );
ND4D0 U1064 ( .A1(n530), .A2(n587), .A3(n529), .A4(n528), .ZN
   (n531) );
NR4D0 U1065 ( .A1(n571), .A2(n533), .A3(n532), .A4(n531), .ZN
   (n547) );
INVD0 U1066 ( .I(n587), .ZN(n536) );
NR3D0 U1067 ( .A1(n536), .A2(n535), .A3(n534), .ZN(n537) );
```

```
ND3D0 U1068 ( .A1(n538), .A2(n644), .A3(n537), .ZN(n599) );
NR4D0 U1069 ( .A1(n2642), .A2(n539), .A3(n597), .A4(n599), .
   ZN(n540) );
OAI211D0 U1070 ( .A1(n741), .A2(n1651), .B(n541), .C(n540), .
   ZN(n542) );
OAI31D0 U1071 ( .A1(n544), .A2(n543), .A3(n542), .B(n3844), .
   ZN(n545) );
OAI211D0 U1072 ( .A1(n547), .A2(n4010), .B(n546), .C(n545), .
   ZN(n548) );
AOI211D0 U1073 ( .A1(n3933), .A2(n550), .B(n549), .C(n548), .
   ZN(n551) );
AOI31D0 U1074 ( .A1(n554), .A2(n553), .A3(n552), .B(n551), .
   ZN(n630) );
INR4D0 U1075 ( .A1(n558), .B1(n557), .B2(n556), .B3(n555), .
   ZN(n559) );
OAI211D0 U1076 ( .A1(n562), .A2(n561), .B(n560), .C(n559), .
   ZN(n618) );
NR3D0 U1077 ( .A1(n564), .A2(n563), .A3(n618), .ZN(n598) );
AOI211D0 U1078 ( .A1(n1285), .A2(n567), .B(n566), .C(n565), .
   ZN(n568) );
ND4D0 U1079 ( .A1(n570), .A2(n569), .A3(n598), .A4(n568), .ZN
   (n578) );
NR4D0 U1080 ( .A1(n573), .A2(n610), .A3(n572), .A4(n571), .ZN
   (n574) );
AOI31D0 U1081 ( .A1(n576), .A2(n575), .A3(n574), .B(n4045), .
   ZN(n577) );
AOI211D0 U1082 ( .A1(n3769), .A2(n578), .B(counter[8]), .C(
   n577), .ZN(n628)
       );
ND4D0 U1083 ( .A1(n582), .A2(n581), .A3(n580), .A4(n579), .ZN
   (n608) );
CKND2D0 U1084 ( .A1(n584), .A2(n583), .ZN(n590) );
ND4D0 U1085 ( .A1(n588), .A2(n587), .A3(n586), .A4(n585), .ZN
   (n589) );
OAI31D0 U1086 ( .A1(n608), .A2(n590), .A3(n589), .B(n4000), .
   ZN(n627) );
ND4D0 U1087 ( .A1(n594), .A2(n593), .A3(n592), .A4(n591), .ZN
   (n595) );
OAI31D0 U1088 ( .A1(n597), .A2(n596), .A3(n595), .B(n3933), .
   ZN(n626) );
NR2D0 U1089 ( .A1(n3737), .A2(n2341), .ZN(n2544) );
INVD0 U1090 ( .I(n2544), .ZN(n2423) );
NR2D0 U1091 ( .A1(n1362), .A2(n4351), .ZN(n2320) );
INVD0 U1092 ( .I(n2320), .ZN(n2281) );
ND4D0 U1093 ( .A1(n609), .A2(n598), .A3(n2423), .A4(n2281), .
   ZN(n623) );
IINR4D0 U1094 ( .A1(n602), .A2(n601), .B1(n600), .B2(n599), .
   ZN(n621) );
```

```
ND4D0 U1095 ( .A1(n606), .A2(n605), .A3(n604), .A4(n603), .ZN
   (n607) );
AOI221D0 U1096 ( .A1(n608), .A2(n3769), .B1(n607), .B2(n3769)
   , .C(n1177),
      .ZN(n620) );
INVD0 U1097 ( .I(n609), .ZN(n617) );
NR2D0 U1098 ( .A1(n611), .A2(n610), .ZN(n615) );
ND4D0 U1099 ( .A1(n615), .A2(n614), .A3(n613), .A4(n612), .ZN
   (n616) );
OAI31D0 U1100 ( .A1(n618), .A2(n617), .A3(n616), .B(n4000), .
   ZN(n619) );
OAI211D0 U1101 ( .A1(n621), .A2(n4045), .B(n620), .C(n619), .
   ZN(n622) );
AOI221D0 U1102 ( .A1(n624), .A2(n3933), .B1(n623), .B2(n3933)
   , .C(n622),
      .ZN(n625) );
AOI31D0 U1103 ( .A1(n628), .A2(n627), .A3(n626), .B(n625), .
   ZN(n629) );
AOI221D0 U1104 ( .A1(counter[10]), .A2(n630), .B1(n1157), .B2
   (n629), .C(
      n1075), .ZN(n631) );
NR4D0 U1105 ( .A1(counter[11]), .A2(n633), .A3(n632), .A4(
   n631), .ZN(n634)
       );
AOI211D0 U1106 ( .A1(n1169), .A2(n636), .B(n635), .C(n634), .
   ZN(n637) );
OAI21D0 U1107 ( .A1(n4545), .A2(\q_out[6]_BAR ), .B(n637), .
   ZN(n4818) );
CKND2D0 U1108 ( .A1(counter[9]), .A2(n1175), .ZN(n639) );
AOI211D0 U1109 ( .A1(n1157), .A2(n639), .B(n638), .C(n1179),
   .ZN(N4777) );
CKND2D0 U1110 ( .A1(n3956), .A2(n2274), .ZN(n2422) );
NR2D0 U1111 ( .A1(n1153), .A2(n2422), .ZN(n649) );
INVD0 U1112 ( .I(n649), .ZN(n795) );
NR2D0 U1113 ( .A1(n795), .A2(n1259), .ZN(n752) );
CKND2D0 U1114 ( .A1(n3956), .A2(counter[5]), .ZN(n2193) );
NR2D0 U1115 ( .A1(n1153), .A2(n2193), .ZN(n840) );
INVD0 U1116 ( .I(n840), .ZN(n640) );
NR2D0 U1117 ( .A1(n640), .A2(n1242), .ZN(n648) );
NR2D0 U1118 ( .A1(n1859), .A2(n2422), .ZN(n1904) );
INVD0 U1119 ( .I(n1904), .ZN(n1905) );
NR2D0 U1120 ( .A1(n1905), .A2(n644), .ZN(n778) );
NR2D0 U1121 ( .A1(n648), .A2(n778), .ZN(n793) );
INVD0 U1122 ( .I(n1881), .ZN(n4084) );
CKND2D0 U1123 ( .A1(n649), .A2(n4084), .ZN(n713) );
CKND2D0 U1124 ( .A1(n793), .A2(n713), .ZN(n762) );
INVD0 U1125 ( .I(n2193), .ZN(n2124) );
CKND2D0 U1126 ( .A1(n1859), .A2(n2978), .ZN(n3976) );
```

```
  INVD0 U1127 ( .I(n3976), .ZN(n1987) );
  CKND2D0 U1128 ( .A1(n2124), .A2(n1987), .ZN(n2039) );
  NR2D0 U1129 ( .A1(n1153), .A2(n2039), .ZN(n899) );
  INVD0 U1130 ( .I(n899), .ZN(n855) );
  CKND2D0 U1131 ( .A1(n649), .A2(n2654), .ZN(n871) );
  CKND2D0 U1132 ( .A1(n855), .A2(n871), .ZN(n791) );
  CKND2D0 U1133 ( .A1(n1859), .A2(n1872), .ZN(n1260) );
  NR2D0 U1134 ( .A1(n2193), .A2(counter[9]), .ZN(n753) );
  CKND2D0 U1135 ( .A1(n753), .A2(n2125), .ZN(n847) );
  CKND2D0 U1136 ( .A1(n840), .A2(n2125), .ZN(n740) );
  NR2D0 U1137 ( .A1(n2422), .A2(n3976), .ZN(n1915) );
  CKND2D0 U1138 ( .A1(counter[9]), .A2(n1915), .ZN(n727) );
  CKND2D0 U1139 ( .A1(n740), .A2(n727), .ZN(n790) );
  NR2D0 U1140 ( .A1(n1905), .A2(n735), .ZN(n682) );
  NR2D0 U1141 ( .A1(n790), .A2(n682), .ZN(n699) );
  OAI211D0 U1142 ( .A1(n1260), .A2(n640), .B(n847), .C(n699), .
     ZN(n665) );
  NR2D0 U1143 ( .A1(n791), .A2(n665), .ZN(n775) );
  INVD0 U1144 ( .I(n775), .ZN(n738) );
  CKND2D0 U1145 ( .A1(n1214), .A2(n753), .ZN(n872) );
  NR2D0 U1146 ( .A1(n4351), .A2(n640), .ZN(n779) );
  INR2D0 U1147 ( .A1(n872), .B1(n779), .ZN(n651) );
  CKND2D0 U1148 ( .A1(n840), .A2(n3749), .ZN(n787) );
  IND2D0 U1149 ( .A1(n651), .B1(n787), .ZN(n750) );
  CKND2D0 U1150 ( .A1(n1904), .A2(n641), .ZN(n845) );
  INVD0 U1151 ( .I(n1260), .ZN(n1198) );
  CKND2D0 U1152 ( .A1(n753), .A2(n1198), .ZN(n697) );
  ND3D0 U1153 ( .A1(n750), .A2(n845), .A3(n697), .ZN(n716) );
  CKND2D0 U1154 ( .A1(n2125), .A2(n649), .ZN(n706) );
  NR3D0 U1155 ( .A1(n1556), .A2(counter[9]), .A3(counter[3]), .
     ZN(n800) );
  INVD0 U1156 ( .I(n2422), .ZN(n2032) );
  CKND2D0 U1157 ( .A1(n2032), .A2(n1198), .ZN(n1856) );
  CKND2D0 U1158 ( .A1(n1904), .A2(n642), .ZN(n848) );
  CKND2D0 U1159 ( .A1(n1214), .A2(n840), .ZN(n794) );
  IND4D0 U1160 ( .A1(n800), .B1(n1856), .B2(n848), .B3(n794), .
     ZN(n816) );
  INVD0 U1161 ( .I(n816), .ZN(n743) );
  CKND2D0 U1162 ( .A1(n706), .A2(n743), .ZN(n755) );
  NR2D0 U1163 ( .A1(n716), .A2(n755), .ZN(n732) );
  INVD0 U1164 ( .I(n732), .ZN(n739) );
  CKND2D0 U1165 ( .A1(n840), .A2(n1022), .ZN(n733) );
  INVD0 U1166 ( .I(n1915), .ZN(n882) );
  NR2D0 U1167 ( .A1(n882), .A2(counter[9]), .ZN(n881) );
  INVD0 U1168 ( .I(n881), .ZN(n730) );
  CKND2D0 U1169 ( .A1(n733), .A2(n730), .ZN(n707) );
  NR2D0 U1170 ( .A1(n739), .A2(n707), .ZN(n858) );
  CKND2D0 U1171 ( .A1(n1214), .A2(n649), .ZN(n786) );
```

```
  NR2D0 U1172 ( .A1(counter[9]), .A2(n2039), .ZN(n643) );
  INR2D0 U1173 ( .A1(n786), .B1(n643), .ZN(n797) );
  CKND2D0 U1174 ( .A1(n753), .A2(n1269), .ZN(n878) );
  ND3D0 U1175 ( .A1(n858), .A2(n797), .A3(n878), .ZN(n852) );
  NR4D0 U1176 ( .A1(n752), .A2(n762), .A3(n738), .A4(n852), .ZN
     (n865) );
  CKND2D0 U1177 ( .A1(n1859), .A2(n2032), .ZN(n1887) );
  NR2D0 U1178 ( .A1(n644), .A2(n1887), .ZN(n678) );
  NR2D0 U1179 ( .A1(n2193), .A2(n4100), .ZN(n676) );
  NR2D0 U1180 ( .A1(n678), .A2(n676), .ZN(n825) );
  INVD0 U1181 ( .I(n825), .ZN(n668) );
  NR2D0 U1182 ( .A1(counter[3]), .A2(n2193), .ZN(n1925) );
  CKND2D0 U1183 ( .A1(n1925), .A2(n645), .ZN(n662) );
  CKND2D0 U1184 ( .A1(n840), .A2(n1198), .ZN(n646) );
  CKND2D0 U1185 ( .A1(n662), .A2(n646), .ZN(n669) );
  NR2D0 U1186 ( .A1(n752), .A2(n669), .ZN(n796) );
  CKND2D0 U1187 ( .A1(n753), .A2(n1022), .ZN(n856) );
  INVD0 U1188 ( .I(n856), .ZN(n650) );
  INVD0 U1189 ( .I(n847), .ZN(n663) );
  NR2D0 U1190 ( .A1(n650), .A2(n663), .ZN(n704) );
  INVD0 U1191 ( .I(n762), .ZN(n828) );
  CKND2D0 U1192 ( .A1(n704), .A2(n828), .ZN(n671) );
  INVD0 U1193 ( .I(n671), .ZN(n789) );
  OAI211D0 U1194 ( .A1(counter[9]), .A2(n2039), .B(n796), .C(
     n789), .ZN(n675)
       );
  NR2D0 U1195 ( .A1(n668), .A2(n675), .ZN(n844) );
  INVD0 U1196 ( .I(n699), .ZN(n860) );
  INVD0 U1197 ( .I(n871), .ZN(n647) );
  NR2D0 U1198 ( .A1(n860), .A2(n647), .ZN(n811) );
  CKAN2D0 U1199 ( .A1(n811), .A2(n3956), .Z(n680) );
  AOI31D0 U1200 ( .A1(n858), .A2(n844), .A3(n680), .B(n1213), .
     ZN(n661) );
  CKND2D0 U1201 ( .A1(n706), .A2(n848), .ZN(n652) );
  CKND2D0 U1202 ( .A1(n2274), .A2(n800), .ZN(n889) );
  CKND2D0 U1203 ( .A1(n855), .A2(n889), .ZN(n681) );
  IND3D0 U1204 ( .A1(n681), .B1(n1856), .B2(n699), .ZN(n714) );
  AOI211D0 U1205 ( .A1(n647), .A2(n1859), .B(n652), .C(n714), .
     ZN(n869) );
  INVD0 U1206 ( .I(n797), .ZN(n677) );
  AOI211D0 U1207 ( .A1(n649), .A2(n1243), .B(n648), .C(n677), .
     ZN(n824) );
  NR2D0 U1208 ( .A1(n650), .A2(n676), .ZN(n827) );
  CKND2D0 U1209 ( .A1(n824), .A2(n827), .ZN(n776) );
  NR2D0 U1210 ( .A1(n678), .A2(n776), .ZN(n774) );
  CKND2D0 U1211 ( .A1(n774), .A2(n662), .ZN(n702) );
  NR2D0 U1212 ( .A1(n778), .A2(n702), .ZN(n709) );
  ND3D0 U1213 ( .A1(n651), .A2(n869), .A3(n709), .ZN(n694) );
```

```
  CKND2D0 U1214 ( .A1(n858), .A2(n3956), .ZN(n674) );
  NR2D0 U1215 ( .A1(n694), .A2(n674), .ZN(n659) );
  INVD0 U1216 ( .I(n790), .ZN(n902) );
  NR2D0 U1217 ( .A1(n791), .A2(n682), .ZN(n885) );
  CKND2D0 U1218 ( .A1(n902), .A2(n885), .ZN(n785) );
  INVD0 U1219 ( .I(n785), .ZN(n837) );
  INVD0 U1220 ( .I(n652), .ZN(n892) );
  CKND2D0 U1221 ( .A1(n837), .A2(n892), .ZN(n653) );
  NR2D0 U1222 ( .A1(n800), .A2(n653), .ZN(n656) );
  INVD0 U1223 ( .I(n1887), .ZN(n1932) );
  CKND2D0 U1224 ( .A1(n1932), .A2(n654), .ZN(n708) );
  CKND2D0 U1225 ( .A1(n656), .A2(n708), .ZN(n804) );
  CKND2D0 U1226 ( .A1(n786), .A2(n844), .ZN(n698) );
  INVD0 U1227 ( .I(n698), .ZN(n731) );
  CKND2D0 U1228 ( .A1(n731), .A2(n845), .ZN(n664) );
  NR3D0 U1229 ( .A1(n664), .A2(n753), .A3(n881), .ZN(n742) );
  CKND2D0 U1230 ( .A1(n742), .A2(n3956), .ZN(n655) );
  OAI31D0 U1231 ( .A1(n779), .A2(n804), .A3(n655), .B(n906), .
     ZN(n658) );
  INVD0 U1232 ( .I(n655), .ZN(n756) );
  AO31D0 U1233 ( .A1(n733), .A2(n656), .A3(n756), .B(n3977), .Z
     (n657) );
  OAI211D0 U1234 ( .A1(n659), .A2(n1651), .B(n658), .C(n657), .
     ZN(n660) );
  OAI31D0 U1235 ( .A1(counter[7]), .A2(n661), .A3(n660), .B(
     counter[8]), .ZN(
        n773) );
  NR2D0 U1236 ( .A1(n1213), .A2(counter[6]), .ZN(n973) );
  INVD0 U1237 ( .I(n973), .ZN(n772) );
  CKND2D0 U1238 ( .A1(n732), .A2(n837), .ZN(n822) );
  CKND2D0 U1239 ( .A1(n662), .A2(n730), .ZN(n782) );
  OR4D0 U1240 ( .A1(n822), .A2(n776), .A3(n782), .A4(n663), .Z(
     n841) );
  CKND2D0 U1241 ( .A1(n697), .A2(n730), .ZN(n888) );
  NR2D0 U1242 ( .A1(n779), .A2(n664), .ZN(n884) );
  IND2D0 U1243 ( .A1(n888), .B1(n884), .ZN(n815) );
  CKND2D0 U1244 ( .A1(n892), .A2(n794), .ZN(n715) );
  NR2D0 U1245 ( .A1(n815), .A2(n715), .ZN(n783) );
  INVD0 U1246 ( .I(n783), .ZN(n898) );
  OR2D0 U1247 ( .A1(n714), .A2(n898), .Z(n877) );
  AOI22D0 U1248 ( .A1(n1234), .A2(n841), .B1(n906), .B2(n877),
     .ZN(n667) );
  OAI31D0 U1249 ( .A1(n739), .A2(n665), .A3(n702), .B(n1208), .
     ZN(n666) );
  AOI31D0 U1250 ( .A1(counter[7]), .A2(n667), .A3(n666), .B(
     n773), .ZN(n770)
         );
  INVD0 U1251 ( .I(n885), .ZN(n726) );
```

```
NR3D0 U1252 ( .A1(n669), .A2(n677), .A3(n668), .ZN(n736) );
CKND2D0 U1253 ( .A1(n858), .A2(n736), .ZN(n670) );
NR4D0 U1254 ( .A1(counter[11]), .A2(n726), .A3(n671), .A4(
   n670), .ZN(n672)
      );
OAI22D0 U1255 ( .A1(counter[0]), .A2(n762), .B1(n672), .B2(
   n3977), .ZN(n673)
      );
AOI211D0 U1256 ( .A1(n906), .A2(n778), .B(n1075), .C(n673), .
   ZN(n693) );
OAI31D0 U1257 ( .A1(n822), .A2(n675), .A3(n674), .B(n1234), .
   ZN(n692) );
AOI21D0 U1258 ( .A1(counter[9]), .A2(n676), .B(n816), .ZN(
   n873) );
INVD0 U1259 ( .I(n796), .ZN(n788) );
NR3D0 U1260 ( .A1(n788), .A2(n677), .A3(n707), .ZN(n823) );
CKND2D0 U1261 ( .A1(n823), .A2(n789), .ZN(n853) );
NR2D0 U1262 ( .A1(n853), .A2(n678), .ZN(n874) );
INVD0 U1263 ( .I(n716), .ZN(n679) );
CKAN2D0 U1264 ( .A1(n874), .A2(n679), .Z(n867) );
OAI211D0 U1265 ( .A1(n873), .A2(n681), .B(n867), .C(n680), .
   ZN(n690) );
INR4D0 U1266 ( .A1(n697), .B1(counter[11]), .B2(n682), .B3(
   n681), .ZN(n683)
      );
CKND2D0 U1267 ( .A1(counter[3]), .A2(counter[5]), .ZN(n1772)
   );
INVD0 U1268 ( .I(n1772), .ZN(n1931) );
CKND2D0 U1269 ( .A1(n1931), .A2(n3956), .ZN(n2092) );
OAI21D0 U1270 ( .A1(n2265), .A2(n2092), .B(n884), .ZN(n805) )
   ;
CKND2D0 U1271 ( .A1(counter[5]), .A2(n800), .ZN(n695) );
CKND2D0 U1272 ( .A1(n1856), .A2(n695), .ZN(n897) );
NR2D0 U1273 ( .A1(n805), .A2(n897), .ZN(n812) );
OAI211D0 U1274 ( .A1(n1987), .A2(n1269), .B(n2032), .C(
   counter[9]), .ZN(n854) );
AOI31D0 U1275 ( .A1(n683), .A2(n812), .A3(n854), .B(n1213), .
   ZN(n689) );
INVD0 U1276 ( .I(n858), .ZN(n686) );
CKND2D0 U1277 ( .A1(n3737), .A2(n1368), .ZN(n2614) );
NR2D0 U1278 ( .A1(n3750), .A2(n2614), .ZN(n684) );
NR2D0 U1279 ( .A1(n3956), .A2(n2274), .ZN(n1860) );
INVD0 U1280 ( .I(n1860), .ZN(n2270) );
NR2D0 U1281 ( .A1(n3956), .A2(counter[5]), .ZN(n2263) );
INVD0 U1282 ( .I(n2263), .ZN(n2260) );
OAI22D0 U1283 ( .A1(n684), .A2(n2270), .B1(n4083), .B2(n2260)
   , .ZN(n725) );
NR2D0 U1284 ( .A1(n4118), .A2(n725), .ZN(n734) );
```

```
ND4D0 U1285 ( .A1(n902), .A2(n734), .A3(n871), .A4(n2260), .
   ZN(n685) );
OAI31D0 U1286 ( .A1(n698), .A2(n686), .A3(n685), .B(n1208), .
   ZN(n687) );
OAI211D0 U1287 ( .A1(n790), .A2(n1651), .B(n1075), .C(n687),
   .ZN(n688) );
AOI211D0 U1288 ( .A1(n906), .A2(n690), .B(n689), .C(n688), .
   ZN(n691) );
AOI211D0 U1289 ( .A1(n693), .A2(n692), .B(counter[8]), .C(
   n691), .ZN(n769)
       );
INVD0 U1290 ( .I(n694), .ZN(n696) );
AOI31D0 U1291 ( .A1(n696), .A2(n847), .A3(n695), .B(n3977), .
   ZN(n721) );
INR2D0 U1292 ( .A1(n697), .B1(n755), .ZN(n883) );
ND3D0 U1293 ( .A1(n845), .A2(n775), .A3(n883), .ZN(n777) );
INR2D0 U1294 ( .A1(n750), .B1(n698), .ZN(n705) );
CKND2D0 U1295 ( .A1(n699), .A2(n705), .ZN(n887) );
INVD0 U1296 ( .I(n887), .ZN(n700) );
NR2D0 U1297 ( .A1(n755), .A2(n707), .ZN(n751) );
AOI31D0 U1298 ( .A1(n700), .A2(n751), .A3(n855), .B(n1651), .
   ZN(n701) );
AO221D0 U1299 ( .A1(n804), .A2(n1285), .B1(n898), .B2(n1285),
    .C(n701), .Z(
      n703) );
OA32D0 U1300 ( .A1(n777), .A2(n703), .A3(n702), .B1(n906), .
   B2(n703), .Z(
      n720) );
AN3D0 U1301 ( .A1(n823), .A2(n704), .A3(n825), .Z(n724) );
ND4D0 U1302 ( .A1(n883), .A2(n837), .A3(n705), .A4(n2260), .
   ZN(n712) );
NR2D0 U1303 ( .A1(n716), .A2(n785), .ZN(n875) );
INVD0 U1304 ( .I(n875), .ZN(n821) );
CKND2D0 U1305 ( .A1(n706), .A2(n889), .ZN(n810) );
NR3D0 U1306 ( .A1(n707), .A2(n821), .A3(n810), .ZN(n801) );
AOI31D0 U1307 ( .A1(n801), .A2(n844), .A3(n708), .B(n1183), .
   ZN(n711) );
AOI31D0 U1308 ( .A1(n751), .A2(n875), .A3(n709), .B(n1651), .
   ZN(n710) );
AOI211D0 U1309 ( .A1(n1285), .A2(n712), .B(n711), .C(n710), .
   ZN(n718) );
CKND2D0 U1310 ( .A1(n713), .A2(n2260), .ZN(n722) );
NR4D0 U1311 ( .A1(n716), .A2(n715), .A3(n714), .A4(n722), .ZN
   (n717) );
AOI32D0 U1312 ( .A1(n724), .A2(n718), .A3(n717), .B1(n3977),
   .B2(n718), .ZN(
      n719) );
```

```
OAI32D0 U1313 ( .A1(n1075), .A2(n721), .A3(n720), .B1(counter
   [7]), .B2(n719),
      .ZN(n767) );
INVD0 U1314 ( .I(n722), .ZN(n729) );
CKND2D0 U1315 ( .A1(n2125), .A2(n1860), .ZN(n723) );
INR2D0 U1316 ( .A1(n723), .B1(n778), .ZN(n760) );
CKND2D0 U1317 ( .A1(n732), .A2(n724), .ZN(n792) );
INR4D0 U1318 ( .A1(n760), .B1(n792), .B2(n726), .B3(n725), .
   ZN(n728) );
AOI31D0 U1319 ( .A1(n729), .A2(n728), .A3(n727), .B(n1213), .
   ZN(n765) );
CKND2D0 U1320 ( .A1(n885), .A2(n740), .ZN(n814) );
CKND2D0 U1321 ( .A1(n731), .A2(n730), .ZN(n820) );
CKND2D0 U1322 ( .A1(n732), .A2(n3956), .ZN(n749) );
AOI211D0 U1323 ( .A1(n814), .A2(n854), .B(n820), .C(n749), .
   ZN(n748) );
ND3D0 U1324 ( .A1(n2270), .A2(n734), .A3(n733), .ZN(n745) );
IND4D0 U1325 ( .A1(n745), .B1(n828), .B2(n736), .B3(n735), .
   ZN(n737) );
OAI31D0 U1326 ( .A1(n739), .A2(n738), .A3(n737), .B(n906), .
   ZN(n747) );
ND4D0 U1327 ( .A1(n743), .A2(n742), .A3(n741), .A4(n740), .ZN
   (n744) );
OAI31D0 U1328 ( .A1(n791), .A2(n745), .A3(n744), .B(n1234), .
   ZN(n746) );
OAI211D0 U1329 ( .A1(n748), .A2(n3977), .B(n747), .C(n746), .
   ZN(n764) );
IIND4D0 U1330 ( .A1(n814), .A2(n749), .B1(n874), .B2(n878), .
   ZN(n759) );
ND3D0 U1331 ( .A1(n837), .A2(n751), .A3(n750), .ZN(n842) );
NR4D0 U1332 ( .A1(counter[11]), .A2(n753), .A3(n752), .A4(
   n842), .ZN(n754)
       );
AOI31D0 U1333 ( .A1(n825), .A2(n793), .A3(n754), .B(n3977), .
   ZN(n758) );
INVD0 U1334 ( .I(n755), .ZN(n838) );
AOI31D0 U1335 ( .A1(n838), .A2(n869), .A3(n756), .B(n1651), .
   ZN(n757) );
AOI211D0 U1336 ( .A1(n1285), .A2(n759), .B(n758), .C(n757), .
   ZN(n761) );
AOI32D0 U1337 ( .A1(n762), .A2(n761), .A3(n760), .B1(n1183),
   .B2(n761), .ZN(
      n763) );
OAI32D0 U1338 ( .A1(counter[7]), .A2(n765), .A3(n764), .B1(
   n1075), .B2(n763),
      .ZN(n766) );
AOI22D0 U1339 ( .A1(counter[8]), .A2(n767), .B1(n766), .B2(
   n1177), .ZN(n768)
```

```
        );
OAI32D0 U1340 ( .A1(counter[6]), .A2(n770), .A3(n769), .B1(
   n1513), .B2(n768),
      .ZN(n771) );
OAI31D0 U1341 ( .A1(n865), .A2(n773), .A3(n772), .B(n771), .
   ZN(n918) );
AOI31D0 U1342 ( .A1(n775), .A2(n858), .A3(n774), .B(counter
   [0]), .ZN(n781)
        );
NR4D0 U1343 ( .A1(n779), .A2(n778), .A3(n777), .A4(n776), .ZN
   (n780) );
OAI21D0 U1344 ( .A1(n780), .A2(n1651), .B(counter[7]), .ZN(
   n870) );
AOI211D0 U1345 ( .A1(n1234), .A2(n782), .B(n781), .C(n870), .
   ZN(n809) );
CKND2D0 U1346 ( .A1(n783), .A2(n795), .ZN(n784) );
OAI31D0 U1347 ( .A1(n800), .A2(n785), .A3(n784), .B(n1208), .
   ZN(n808) );
CKND2D0 U1348 ( .A1(n787), .A2(n786), .ZN(n894) );
NR2D0 U1349 ( .A1(n788), .A2(n894), .ZN(n866) );
IND3D0 U1350 ( .A1(n822), .B1(n789), .B2(n866), .ZN(n879) );
INR4D0 U1351 ( .A1(n793), .B1(n792), .B2(n791), .B3(n790), .
   ZN(n802) );
CKND2D0 U1352 ( .A1(n795), .A2(n794), .ZN(n886) );
ND3D0 U1353 ( .A1(n796), .A2(n825), .A3(n828), .ZN(n861) );
IND2D0 U1354 ( .A1(n861), .B1(n797), .ZN(n799) );
INR4D0 U1355 ( .A1(n801), .B1(n800), .B2(n886), .B3(n799), .
   ZN(n846) );
OAI22D0 U1356 ( .A1(n802), .A2(n1213), .B1(n846), .B2(n1651),
    .ZN(n803) );
AOI211D0 U1357 ( .A1(n1208), .A2(n879), .B(counter[7]), .C(
   n803), .ZN(n807)
        );
OAI31D0 U1358 ( .A1(n881), .A2(n805), .A3(n804), .B(n906), .
   ZN(n806) );
AOI22D0 U1359 ( .A1(n809), .A2(n808), .B1(n807), .B2(n806), .
   ZN(n836) );
INVD0 U1360 ( .I(n810), .ZN(n813) );
AOI31D0 U1361 ( .A1(n813), .A2(n812), .A3(n811), .B(n1183), .
   ZN(n819) );
NR3D0 U1362 ( .A1(n881), .A2(n861), .A3(n822), .ZN(n893) );
OAI31D0 U1363 ( .A1(n816), .A2(n815), .A3(n814), .B(n1234), .
   ZN(n817) );
OAI211D0 U1364 ( .A1(n893), .A2(n1213), .B(counter[7]), .C(
   n817), .ZN(n818)
        );
NR2D0 U1365 ( .A1(n819), .A2(n818), .ZN(n834) );
```

```
OAI31D0 U1366 ( .A1(n821), .A2(n897), .A3(n820), .B(n1208), .
   ZN(n833) );
INR2D0 U1367 ( .A1(n823), .B1(n822), .ZN(n826) );
ND4D0 U1368 ( .A1(n825), .A2(n824), .A3(n826), .A4(n847), .ZN
   (n831) );
CKND2D0 U1369 ( .A1(n906), .A2(n856), .ZN(n830) );
ND4D0 U1370 ( .A1(n828), .A2(n827), .A3(n826), .A4(n1183), .
   ZN(n829) );
OAI222D0 U1371 ( .A1(n831), .A2(n830), .B1(n831), .B2(n3977),
    .C1(n1208),
      .C2(n829), .ZN(n832) );
AOI22D0 U1372 ( .A1(n834), .A2(n833), .B1(n1075), .B2(n832),
   .ZN(n835) );
AOI22D0 U1373 ( .A1(counter[8]), .A2(n836), .B1(n835), .B2(
   n1177), .ZN(n915)
      );
AOI31D0 U1374 ( .A1(n838), .A2(n837), .A3(n884), .B(n1651), .
   ZN(n839) );
AOI211D0 U1375 ( .A1(n906), .A2(n841), .B(n840), .C(n839), .
   ZN(n851) );
INVD0 U1376 ( .I(n842), .ZN(n843) );
AO31D0 U1377 ( .A1(n845), .A2(n844), .A3(n843), .B(n1213), .Z
   (n850) );
AO31D0 U1378 ( .A1(n848), .A2(n847), .A3(n846), .B(n3977), .Z
   (n849) );
ND4D0 U1379 ( .A1(counter[7]), .A2(n851), .A3(n850), .A4(n849
   ), .ZN(n913) );
INR4D0 U1380 ( .A1(n854), .B1(n899), .B2(n853), .B3(n852), .
   ZN(n863) );
ND4D0 U1381 ( .A1(n858), .A2(n857), .A3(n856), .A4(n855), .ZN
   (n859) );
OAI31D0 U1382 ( .A1(n861), .A2(n860), .A3(n859), .B(n906), .
   ZN(n862) );
OA211D0 U1383 ( .A1(n863), .A2(n3977), .B(n1075), .C(n862), .
   Z(n864) );
OA221D0 U1384 ( .A1(n866), .A2(n1651), .B1(n865), .B2(n1651),
    .C(n864), .Z(
      n868) );
AOI32D0 U1385 ( .A1(n869), .A2(n868), .A3(n867), .B1(n1213),
   .B2(n868), .ZN(
      n912) );
INVD0 U1386 ( .I(n870), .ZN(n910) );
CKND2D0 U1387 ( .A1(n872), .A2(n871), .ZN(n896) );
AOI31D0 U1388 ( .A1(n875), .A2(n874), .A3(n873), .B(n3977), .
   ZN(n876) );
AOI221D0 U1389 ( .A1(n896), .A2(n906), .B1(n877), .B2(n906),
   .C(n876), .ZN(
      n909) );
```

```
INVD0 U1390 ( .I(n878), .ZN(n880) );
OAI31D0 U1391 ( .A1(n881), .A2(n880), .A3(n879), .B(n1285), .
   ZN(n908) );
ND4D0 U1392 ( .A1(n885), .A2(n884), .A3(n883), .A4(n882), .ZN
   (n905) );
INVD0 U1393 ( .I(n886), .ZN(n891) );
INR3D0 U1394 ( .A1(n889), .B1(n888), .B2(n887), .ZN(n890) );
AOI31D0 U1395 ( .A1(n892), .A2(n891), .A3(n890), .B(n1651), .
   ZN(n904) );
INVD0 U1396 ( .I(n893), .ZN(n895) );
AOI211D0 U1397 ( .A1(n1285), .A2(n895), .B(n894), .C(counter
   [7]), .ZN(n901)
       );
NR4D0 U1398 ( .A1(n899), .A2(n898), .A3(n897), .A4(n896), .ZN
   (n900) );
AOI32D0 U1399 ( .A1(n902), .A2(n901), .A3(n900), .B1(n3977),
   .B2(n901), .ZN(
      n903) );
AOI211D0 U1400 ( .A1(n906), .A2(n905), .B(n904), .C(n903), .
   ZN(n907) );
AOI31D0 U1401 ( .A1(n910), .A2(n909), .A3(n908), .B(n907), .
   ZN(n911) );
AOI32D0 U1402 ( .A1(n913), .A2(n1177), .A3(n912), .B1(counter
   [8]), .B2(n911),
      .ZN(n914) );
AOI22D0 U1403 ( .A1(counter[6]), .A2(n915), .B1(n914), .B2(
   n1513), .ZN(n917)
       );
OAI221D0 U1404 ( .A1(counter[10]), .A2(n918), .B1(n1157), .B2
   (n917), .C(n916), .ZN(n1173) );
NR2D0 U1405 ( .A1(n1513), .A2(n3977), .ZN(n1021) );
AOI22D0 U1406 ( .A1(n1285), .A2(n1987), .B1(n4084), .B2(n1021
   ), .ZN(n919) );
CKND2D0 U1407 ( .A1(n1513), .A2(n1872), .ZN(n1557) );
NR2D0 U1408 ( .A1(n1099), .A2(n1557), .ZN(n972) );
INR2D0 U1409 ( .A1(n919), .B1(n972), .ZN(n931) );
NR2D0 U1410 ( .A1(n1513), .A2(n1213), .ZN(n926) );
INVD0 U1411 ( .I(n926), .ZN(n1555) );
NR2D0 U1412 ( .A1(n1242), .A2(n1555), .ZN(n1000) );
INVD0 U1413 ( .I(n1021), .ZN(n1800) );
CKND2D0 U1414 ( .A1(n1513), .A2(n2654), .ZN(n1098) );
INVD0 U1415 ( .I(n1098), .ZN(n1530) );
AOI22D0 U1416 ( .A1(n1530), .A2(n2638), .B1(n2642), .B2(n1513
   ), .ZN(n1073)
       );
OAI21D0 U1417 ( .A1(n1209), .A2(n1018), .B(n4099), .ZN(n920)
   );
```

```
OAI211D0 U1418 ( .A1(n3472), .A2(n1800), .B(n1073), .C(n920),
    .ZN(n960) );
NR2D0 U1419 ( .A1(n3896), .A2(n1557), .ZN(n1011) );
INVD0 U1420 ( .I(n1011), .ZN(n1054) );
CKND2D0 U1421 ( .A1(n4084), .A2(n973), .ZN(n1087) );
OA211D0 U1422 ( .A1(n1557), .A2(n3736), .B(n1054), .C(n1087),
    .Z(n1019) );
CKND2D0 U1423 ( .A1(n1260), .A2(n3390), .ZN(n922) );
CKND2D0 U1424 ( .A1(n1526), .A2(n1182), .ZN(n1122) );
INVD0 U1425 ( .I(n1122), .ZN(n939) );
NR2D0 U1426 ( .A1(n1513), .A2(n2269), .ZN(n1536) );
INVD0 U1427 ( .I(n1536), .ZN(n4444) );
CKND2D0 U1428 ( .A1(n4099), .A2(n1182), .ZN(n1086) );
CKND2D0 U1429 ( .A1(n1018), .A2(n1536), .ZN(n984) );
CKND2D0 U1430 ( .A1(counter[6]), .A2(counter[4]), .ZN(n2120)
   );
NR2D0 U1431 ( .A1(n4100), .A2(n2120), .ZN(n1164) );
CKND2D0 U1432 ( .A1(counter[0]), .A2(n1164), .ZN(n1059) );
AN3D0 U1433 ( .A1(n1086), .A2(n984), .A3(n1059), .Z(n958) );
CKND2D0 U1434 ( .A1(n1530), .A2(n1209), .ZN(n921) );
OAI211D0 U1435 ( .A1(n4444), .A2(n1651), .B(n958), .C(n921),
   .ZN(n1080) );
AOI211D0 U1436 ( .A1(n1021), .A2(n922), .B(n939), .C(n1080),
   .ZN(n1025) );
CKND2D0 U1437 ( .A1(n3737), .A2(n3527), .ZN(n2526) );
INVD0 U1438 ( .I(n2526), .ZN(n2565) );
CKND2D0 U1439 ( .A1(n3757), .A2(counter[6]), .ZN(n2126) );
INVD0 U1440 ( .I(n2126), .ZN(n2165) );
CKND2D0 U1441 ( .A1(n2565), .A2(n2165), .ZN(n987) );
NR2D0 U1442 ( .A1(counter[0]), .A2(n4444), .ZN(n923) );
CKND2D0 U1443 ( .A1(n923), .A2(n3754), .ZN(n1124) );
CKND2D0 U1444 ( .A1(n987), .A2(n1124), .ZN(n1145) );
CKND2D0 U1445 ( .A1(counter[3]), .A2(n1145), .ZN(n1028) );
NR2D0 U1446 ( .A1(counter[0]), .A2(n3737), .ZN(n2316) );
CKND2D0 U1447 ( .A1(n3750), .A2(n2316), .ZN(n4272) );
AOI32D0 U1448 ( .A1(n1242), .A2(n1028), .A3(n4272), .B1(n2120
   ), .B2(n1028),
      .ZN(n1033) );
CKND2D0 U1449 ( .A1(n2341), .A2(n924), .ZN(n1178) );
CKND2D0 U1450 ( .A1(n1513), .A2(counter[4]), .ZN(n1988) );
NR2D0 U1451 ( .A1(n1988), .A2(counter[2]), .ZN(n1811) );
CKND2D0 U1452 ( .A1(n1530), .A2(n934), .ZN(n1120) );
CKND2D0 U1453 ( .A1(n1530), .A2(n1018), .ZN(n1088) );
CKND2D0 U1454 ( .A1(n1120), .A2(n1088), .ZN(n961) );
NR2D0 U1455 ( .A1(counter[6]), .A2(counter[4]), .ZN(n2164) );
CKND2D0 U1456 ( .A1(counter[0]), .A2(n2164), .ZN(n1682) );
NR2D0 U1457 ( .A1(n1242), .A2(n1682), .ZN(n1014) );
CKND2D0 U1458 ( .A1(n1513), .A2(n2978), .ZN(n1543) );
```

```
  NR2D0 U1459 ( .A1(n1651), .A2(n1543), .ZN(n963) );
  INVD0 U1460 ( .I(n963), .ZN(n1692) );
  IND2D0 U1461 ( .A1(n1014), .B1(n1692), .ZN(n983) );
  NR2D0 U1462 ( .A1(n961), .A2(n983), .ZN(n928) );
  INVD0 U1463 ( .I(n928), .ZN(n1082) );
  AOI31D0 U1464 ( .A1(n3759), .A2(n1178), .A3(n1811), .B(n1082)
     , .ZN(n1020) );
  CKND2D0 U1465 ( .A1(n4084), .A2(n926), .ZN(n1097) );
  IND3D0 U1466 ( .A1(n1033), .B1(n1020), .B2(n1097), .ZN(n925)
     );
  INVD0 U1467 ( .I(n925), .ZN(n1044) );
  ND3D0 U1468 ( .A1(n1019), .A2(n1025), .A3(n1044), .ZN(n1067)
     );
  NR2D0 U1469 ( .A1(n960), .A2(n1067), .ZN(n1052) );
  CKND2D0 U1470 ( .A1(n1198), .A2(n926), .ZN(n1130) );
  OAI211D0 U1471 ( .A1(n1506), .A2(n945), .B(n1052), .C(n1130),
      .ZN(n1143) );
  NR2D0 U1472 ( .A1(n1099), .A2(n1543), .ZN(n962) );
  NR2D0 U1473 ( .A1(n2617), .A2(n1557), .ZN(n1016) );
  CKND2D0 U1474 ( .A1(counter[6]), .A2(n1872), .ZN(n1537) );
  INVD0 U1475 ( .I(n1537), .ZN(n1529) );
  CKND2D0 U1476 ( .A1(n1043), .A2(n1529), .ZN(n1107) );
  CKND2D0 U1477 ( .A1(n2638), .A2(n1526), .ZN(n1137) );
  OAI211D0 U1478 ( .A1(n2777), .A2(n1799), .B(n1107), .C(n1137)
     , .ZN(n1072) );
  INVD0 U1479 ( .I(n2164), .ZN(n2062) );
  CKND2D0 U1480 ( .A1(n2638), .A2(n4257), .ZN(n1002) );
  CKND2D0 U1481 ( .A1(n2125), .A2(n1021), .ZN(n1026) );
  CKND2D0 U1482 ( .A1(n1002), .A2(n1026), .ZN(n1079) );
  AO31D0 U1483 ( .A1(n2062), .A2(n1214), .A3(n1694), .B(n1079),
      .Z(n927) );
  NR3D0 U1484 ( .A1(n1016), .A2(n1072), .A3(n927), .ZN(n933) );
  NR2D0 U1485 ( .A1(n1557), .A2(n945), .ZN(n1069) );
  INVD0 U1486 ( .I(n1069), .ZN(n985) );
  CKND2D0 U1487 ( .A1(counter[0]), .A2(n2978), .ZN(n1941) );
  INVD0 U1488 ( .I(n1941), .ZN(n2321) );
  CKND2D0 U1489 ( .A1(n2164), .A2(n2321), .ZN(n1783) );
  IND4D0 U1490 ( .A1(n962), .B1(n933), .B2(n985), .B3(n1783), .
     ZN(n1128) );
  INR4D0 U1491 ( .A1(n931), .B1(n1000), .B2(n1143), .B3(n1128),
      .ZN(n970) );
  NR2D0 U1492 ( .A1(n1075), .A2(counter[5]), .ZN(n1101) );
  INVD0 U1493 ( .I(n1101), .ZN(n1149) );
  INVD0 U1494 ( .I(n1557), .ZN(n1518) );
  NR2D0 U1495 ( .A1(n1518), .A2(n1536), .ZN(n1789) );
  NR2D0 U1496 ( .A1(n1789), .A2(n3736), .ZN(n1104) );
  OAI21D0 U1497 ( .A1(n2777), .A2(n1524), .B(n945), .ZN(n944) )
     ;
```

```
  CKND2D0 U1498 ( .A1(n1022), .A2(n1021), .ZN(n947) );
  ND3D0 U1499 ( .A1(n947), .A2(n1097), .A3(n928), .ZN(n975) );
  NR2D0 U1500 ( .A1(n1194), .A2(n4444), .ZN(n971) );
  INVD0 U1501 ( .I(n929), .ZN(n4024) );
  NR2D0 U1502 ( .A1(n1865), .A2(n4024), .ZN(n3880) );
  INVD0 U1503 ( .I(n3880), .ZN(n3946) );
  CKND2D0 U1504 ( .A1(counter[2]), .A2(counter[0]), .ZN(n1939)
     );
  NR2D0 U1505 ( .A1(n3946), .A2(n1939), .ZN(n2650) );
  CKND2D0 U1506 ( .A1(n973), .A2(n2798), .ZN(n1108) );
  CKND2D0 U1507 ( .A1(n1108), .A2(n3759), .ZN(n930) );
  AOI22D0 U1508 ( .A1(counter[6]), .A2(n2650), .B1(n2577), .B2(
     n930), .ZN(
        n1049) );
  CKND2D0 U1509 ( .A1(n1209), .A2(n1526), .ZN(n1119) );
  NR2D0 U1510 ( .A1(n3736), .A2(n1799), .ZN(n949) );
  INR2D0 U1511 ( .A1(n1119), .B1(n949), .ZN(n977) );
  CKND2D0 U1512 ( .A1(n4099), .A2(n2637), .ZN(n1148) );
  ND4D0 U1513 ( .A1(n1049), .A2(n931), .A3(n977), .A4(n1148), .
     ZN(n1144) );
  NR2D0 U1514 ( .A1(n2638), .A2(n2637), .ZN(n937) );
  NR2D0 U1515 ( .A1(n1099), .A2(n4444), .ZN(n1105) );
  INVD0 U1516 ( .I(n1105), .ZN(n932) );
  OAI211D0 U1517 ( .A1(n1506), .A2(n945), .B(n933), .C(n932), .
     ZN(n936) );
  CKND2D0 U1518 ( .A1(n2978), .A2(n1416), .ZN(n1089) );
  NR2D0 U1519 ( .A1(n1799), .A2(n1270), .ZN(n1129) );
  INVD0 U1520 ( .I(n1129), .ZN(n952) );
  CKND2D0 U1521 ( .A1(n952), .A2(n985), .ZN(n995) );
  INVD0 U1522 ( .I(n1543), .ZN(n1523) );
  CKND2D0 U1523 ( .A1(n1523), .A2(n934), .ZN(n954) );
  ND4D0 U1524 ( .A1(n1087), .A2(n987), .A3(n1130), .A4(n954), .
     ZN(n935) );
  OR2D0 U1525 ( .A1(n995), .A2(n935), .Z(n1081) );
  NR2D0 U1526 ( .A1(n1513), .A2(n1694), .ZN(n964) );
  CKAN2D0 U1527 ( .A1(n964), .A2(n1198), .Z(n957) );
  AO21D0 U1528 ( .A1(n1182), .A2(n1530), .B(n957), .Z(n986) );
  NR2D0 U1529 ( .A1(n3736), .A2(n1537), .ZN(n1001) );
  INR4D0 U1530 ( .A1(n1086), .B1(n1081), .B2(n986), .B3(n1001),
      .ZN(n992) );
  IND4D0 U1531 ( .A1(n936), .B1(n1054), .B2(n1089), .B3(n992),
     .ZN(n976) );
  INVD0 U1532 ( .I(n976), .ZN(n1060) );
  OAI211D0 U1533 ( .A1(n937), .A2(n1543), .B(n1060), .C(n984),
     .ZN(n938) );
  OR4D0 U1534 ( .A1(n939), .A2(n971), .A3(n1144), .A4(n938), .Z
     (n959) );
  CKND2D0 U1535 ( .A1(n1543), .A2(n1799), .ZN(n1017) );
```

```
CKND2D0 U1536 ( .A1(n1209), .A2(n1017), .ZN(n1121) );
INVD0 U1537 ( .I(n1121), .ZN(n996) );
OR4D0 U1538 ( .A1(n944), .A2(n975), .A3(n959), .A4(n996), .Z(
   n940) );
AOI221D0 U1539 ( .A1(n1104), .A2(n1135), .B1(n940), .B2(n1135
   ), .C(
      counter[10]), .ZN(n969) );
CKND2D0 U1540 ( .A1(n1075), .A2(counter[5]), .ZN(n1053) );
INVD0 U1541 ( .I(n1053), .ZN(n1141) );
CKND2D0 U1542 ( .A1(n2316), .A2(n3757), .ZN(n2192) );
AOI21D0 U1543 ( .A1(n2192), .A2(n941), .B(counter[6]), .ZN(
   n956) );
NR2D0 U1544 ( .A1(n1211), .A2(n1800), .ZN(n1070) );
AOI32D0 U1545 ( .A1(n1098), .A2(n942), .A3(n1557), .B1(n3736)
   , .B2(n942),
      .ZN(n943) );
AOI211D0 U1546 ( .A1(n4257), .A2(n944), .B(n1070), .C(n943),
   .ZN(n1066) );
OAI211D0 U1547 ( .A1(n1799), .A2(n945), .B(n1066), .C(n1049),
    .ZN(n1117) );
CKND2D0 U1548 ( .A1(n946), .A2(n1523), .ZN(n1063) );
CKND2D0 U1549 ( .A1(n1182), .A2(n1529), .ZN(n1138) );
OAI211D0 U1550 ( .A1(n2777), .A2(n1799), .B(n1063), .C(n1138)
   , .ZN(n1106) );
OAI211D0 U1551 ( .A1(n3736), .A2(n2269), .B(n1148), .C(n947),
    .ZN(n1064) );
CKND2D0 U1552 ( .A1(counter[1]), .A2(n2316), .ZN(n2540) );
NR2D0 U1553 ( .A1(n2540), .A2(n2120), .ZN(n1515) );
CKND2D0 U1554 ( .A1(n1518), .A2(n1182), .ZN(n948) );
CKND2D0 U1555 ( .A1(n1119), .A2(n948), .ZN(n999) );
AOI221D0 U1556 ( .A1(n1523), .A2(n2637), .B1(n1526), .B2(
   n2637), .C(n949),
      .ZN(n1062) );
CKND2D0 U1557 ( .A1(n1062), .A2(n1139), .ZN(n950) );
INR4D0 U1558 ( .A1(n1107), .B1(n1515), .B2(n999), .B3(n950),
   .ZN(n1051) );
ND4D0 U1559 ( .A1(n1051), .A2(n1054), .A3(n1122), .A4(n1097),
    .ZN(n951) );
NR4D0 U1560 ( .A1(n1117), .A2(n1106), .A3(n1064), .A4(n951),
   .ZN(n979) );
NR2D0 U1561 ( .A1(n1099), .A2(n1537), .ZN(n955) );
AOI211D0 U1562 ( .A1(n2978), .A2(n1021), .B(n1016), .C(n1001)
   , .ZN(n953) );
CKND2D0 U1563 ( .A1(n953), .A2(n952), .ZN(n1140) );
CKND2D0 U1564 ( .A1(n954), .A2(n1002), .ZN(n988) );
NR4D0 U1565 ( .A1(n1000), .A2(n955), .A3(n1140), .A4(n988), .
   ZN(n1068) );
CKND2D0 U1566 ( .A1(n979), .A2(n1068), .ZN(n1090) );
```

```
INR4D0 U1567 ( .A1(n1120), .B1(n957), .B2(n956), .B3(n1090),
    .ZN(n1012) );
OAI211D0 U1568 ( .A1(n1651), .A2(n3333), .B(n958), .C(n1012),
    .ZN(n967) );
NR3D0 U1569 ( .A1(n961), .A2(n960), .A3(n959), .ZN(n1100) );
AOI211D0 U1570 ( .A1(n4118), .A2(n964), .B(n963), .C(n962), .
    ZN(n965) );
NR2D0 U1571 ( .A1(n1075), .A2(n2274), .ZN(n1110) );
INVD0 U1572 ( .I(n1110), .ZN(n1041) );
AOI31D0 U1573 ( .A1(n1100), .A2(n965), .A3(n1059), .B(n1041),
    .ZN(n966) );
AOI211D0 U1574 ( .A1(n1141), .A2(n967), .B(n1069), .C(n966),
    .ZN(n968) );
OAI211D0 U1575 ( .A1(n970), .A2(n1149), .B(n969), .C(n968), .
    ZN(n1040) );
AOI211D0 U1576 ( .A1(n1043), .A2(n1530), .B(n972), .C(n971),
    .ZN(n981) );
NR2D0 U1577 ( .A1(n1198), .A2(n1269), .ZN(n1212) );
INVD0 U1578 ( .I(n1212), .ZN(n4112) );
CKND2D0 U1579 ( .A1(n973), .A2(n4112), .ZN(n980) );
ND4D0 U1580 ( .A1(n1148), .A2(n1063), .A3(n980), .A4(n1059),
    .ZN(n974) );
NR4D0 U1581 ( .A1(n1117), .A2(n976), .A3(n975), .A4(n974), .
    ZN(n1123) );
ND4D0 U1582 ( .A1(n977), .A2(n981), .A3(n1123), .A4(n1941), .
    ZN(n990) );
NR2D0 U1583 ( .A1(n3976), .A2(n1800), .ZN(n978) );
AOI211D0 U1584 ( .A1(n4099), .A2(n1209), .B(n1016), .C(n978),
    .ZN(n1058) );
CKND2D0 U1585 ( .A1(n979), .A2(n1058), .ZN(n991) );
OAI211D0 U1586 ( .A1(n4444), .A2(n982), .B(n981), .C(n980), .
    ZN(n994) );
INR3D0 U1587 ( .A1(n984), .B1(n994), .B2(n983), .ZN(n1165) );
ND3D0 U1588 ( .A1(n1165), .A2(n985), .A3(n1120), .ZN(n1092) )
    ;
INR4D0 U1589 ( .A1(n1026), .B1(n986), .B2(n991), .B3(n1092),
    .ZN(n1094) );
IIND4D0 U1590 ( .A1(n988), .A2(n1001), .B1(n1094), .B2(n987),
    .ZN(n989) );
AOI22D0 U1591 ( .A1(n1135), .A2(n990), .B1(n1101), .B2(n989),
    .ZN(n1009) );
INVD0 U1592 ( .I(n991), .ZN(n1085) );
ND3D0 U1593 ( .A1(n1085), .A2(n992), .A3(n1088), .ZN(n1162) )
    ;
ND3D0 U1594 ( .A1(n1099), .A2(n1089), .A3(n1059), .ZN(n993) )
    ;
OAI31D0 U1595 ( .A1(n1162), .A2(n994), .A3(n993), .B(n1141),
    .ZN(n1008) );
```

```
  NR2D0 U1596 ( .A1(n2978), .A2(n1022), .ZN(n998) );
  IINR4D0 U1597 ( .A1(n1108), .A2(n1148), .B1(n996), .B2(n995),
      .ZN(n997) );
  OAI211D0 U1598 ( .A1(n998), .A2(n1682), .B(n1073), .C(n997),
     .ZN(n1015) );
  INVD0 U1599 ( .I(n999), .ZN(n1027) );
  INR4D0 U1600 ( .A1(n1002), .B1(n1070), .B2(n1001), .B3(n1000)
     , .ZN(n1003) );
  OAI21D0 U1601 ( .A1(counter[0]), .A2(n4444), .B(n1003), .ZN(
     n1004) );
  AOI31D0 U1602 ( .A1(n1526), .A2(n1859), .A3(n1213), .B(n1004)
     , .ZN(n1023) );
  NR2D0 U1603 ( .A1(n2126), .A2(n2423), .ZN(n1005) );
  INR2D0 U1604 ( .A1(n1138), .B1(n1005), .ZN(n1048) );
  ND4D0 U1605 ( .A1(n1027), .A2(n1023), .A3(n1048), .A4(n1063),
      .ZN(n1006) );
  OAI31D0 U1606 ( .A1(n1015), .A2(n1067), .A3(n1006), .B(n1110)
     , .ZN(n1007) );
  ND4D0 U1607 ( .A1(n1009), .A2(n1153), .A3(n1008), .A4(n1007),
      .ZN(n1039) );
  CKND2D0 U1608 ( .A1(n1063), .A2(n1087), .ZN(n1047) );
  OA31D0 U1609 ( .A1(n1163), .A2(n1047), .A3(n1011), .B(n1010),
      .Z(n1037) );
  INVD0 U1610 ( .I(n1012), .ZN(n1013) );
  OAI31D0 U1611 ( .A1(n1014), .A2(n2987), .A3(n1013), .B(n1141)
     , .ZN(n1036) );
  AOI211D0 U1612 ( .A1(n1018), .A2(n1017), .B(n1016), .C(n1015)
     , .ZN(n1045) );
  ND4D0 U1613 ( .A1(n1020), .A2(n1045), .A3(n1019), .A4(n1063),
      .ZN(n1030) );
  OAI21D0 U1614 ( .A1(n1022), .A2(n1987), .B(n1021), .ZN(n1131)
      );
  OAI211D0 U1615 ( .A1(n4351), .A2(n3736), .B(n1023), .C(n1131)
     , .ZN(n1024) );
  INR2D0 U1616 ( .A1(n1025), .B1(n1024), .ZN(n1031) );
  ND3D0 U1617 ( .A1(n1031), .A2(n1027), .A3(n1026), .ZN(n1046)
     );
  OAI211D0 U1618 ( .A1(n1555), .A2(n3441), .B(n1138), .C(n1028)
     , .ZN(n1029) );
  OAI31D0 U1619 ( .A1(n1030), .A2(n1046), .A3(n1029), .B(n1110)
     , .ZN(n1035) );
  INVD0 U1620 ( .I(n1030), .ZN(n1168) );
  ND4D0 U1621 ( .A1(n1168), .A2(n1031), .A3(n1048), .A4(n1107),
      .ZN(n1032) );
  OAI21D0 U1622 ( .A1(n1033), .A2(n1032), .B(n1101), .ZN(n1034)
      );
  ND4D0 U1623 ( .A1(n1037), .A2(n1036), .A3(n1035), .A4(n1034),
      .ZN(n1038) );
```

```
AOI31D0 U1624 ( .A1(n1040), .A2(n1039), .A3(n1038), .B(n1042)
   , .ZN(n1161) );
AOI32D0 U1625 ( .A1(n1043), .A2(n1042), .A3(n1518), .B1(n1041
   ), .B2(n1042),
      .ZN(n1057) );
IND3D0 U1626 ( .A1(n1046), .B1(n1045), .B2(n1044), .ZN(n1078)
    );
INR3D0 U1627 ( .A1(n1048), .B1(n1078), .B2(n1047), .ZN(n1147)
    );
AN3D0 U1628 ( .A1(n4444), .A2(n1068), .A3(n1049), .Z(n1050) )
   ;
AO31D0 U1629 ( .A1(n1052), .A2(n1051), .A3(n1050), .B(n1163),
    .Z(n1055) );
AOI32D0 U1630 ( .A1(n1147), .A2(n1055), .A3(n1054), .B1(n1053
   ), .B2(n1055),
      .ZN(n1056) );
AOI211D0 U1631 ( .A1(n1058), .A2(n1101), .B(n1057), .C(n1056)
   , .ZN(n1160) );
ND4D0 U1632 ( .A1(n1165), .A2(n1060), .A3(n1121), .A4(n1059),
    .ZN(n1061) );
INR2D0 U1633 ( .A1(n1062), .B1(n1061), .ZN(n1118) );
IND4D0 U1634 ( .A1(n1064), .B1(n1118), .B2(n1088), .B3(n1063)
   , .ZN(n1065) );
INR2D0 U1635 ( .A1(n1066), .B1(n1065), .ZN(n1076) );
NR2D0 U1636 ( .A1(n1067), .A2(n1144), .ZN(n1132) );
INVD0 U1637 ( .I(n1068), .ZN(n1112) );
OR4D0 U1638 ( .A1(n1070), .A2(n1112), .A3(n1069), .A4(n1105),
    .Z(n1071) );
IINR4D0 U1639 ( .A1(n1073), .A2(n1132), .B1(n1072), .B2(n1071
   ), .ZN(n1074)
      );
AOI221D0 U1640 ( .A1(counter[7]), .A2(n1076), .B1(n1075), .B2
   (n1074), .C(
      n2274), .ZN(n1077) );
AOI211D0 U1641 ( .A1(n1135), .A2(n1078), .B(n1153), .C(n1077)
   , .ZN(n1084) );
NR4D0 U1642 ( .A1(n1082), .A2(n1081), .A3(n1080), .A4(n1079),
    .ZN(n1083) );
AOI32D0 U1643 ( .A1(n1085), .A2(n1084), .A3(n1083), .B1(n1149
   ), .B2(n1084),
      .ZN(n1158) );
AN3D0 U1644 ( .A1(n1088), .A2(n1087), .A3(n1086), .Z(n1093) )
   ;
IND4D0 U1645 ( .A1(n1090), .B1(n1093), .B2(n1213), .B3(n1089)
   , .ZN(n1091) );
AOI221D0 U1646 ( .A1(n1092), .A2(n1141), .B1(n1091), .B2(
   n1141), .C(
      counter[9]), .ZN(n1116) );
```

```
OAI211D0 U1647 ( .A1(n2126), .A2(n2423), .B(n1094), .C(n1093)
   , .ZN(n1095) );
OAI31D0 U1648 ( .A1(n1129), .A2(n1164), .A3(n1095), .B(n1135)
   , .ZN(n1115) );
OAI211D0 U1649 ( .A1(n1099), .A2(n1098), .B(n1097), .C(n1096)
   , .ZN(n1103) );
INVD0 U1650 ( .I(n1100), .ZN(n1102) );
OAI31D0 U1651 ( .A1(n1104), .A2(n1103), .A3(n1102), .B(n1101)
   , .ZN(n1114) );
NR3D0 U1652 ( .A1(n1106), .A2(n1182), .A3(n1105), .ZN(n1109)
   );
ND4D0 U1653 ( .A1(n1109), .A2(n1148), .A3(n1108), .A4(n1107),
    .ZN(n1111) );
OAI31D0 U1654 ( .A1(n1143), .A2(n1112), .A3(n1111), .B(n1110)
   , .ZN(n1113) );
ND4D0 U1655 ( .A1(n1116), .A2(n1115), .A3(n1114), .A4(n1113),
    .ZN(n1156) );
INVD0 U1656 ( .I(n1117), .ZN(n1127) );
AN4D0 U1657 ( .A1(n2274), .A2(n1120), .A3(n1119), .A4(n1118),
    .Z(n1126) );
AN4D0 U1658 ( .A1(n1123), .A2(n1122), .A3(n1121), .A4(counter
   [5]), .Z(n1125)
       );
AOI22D0 U1659 ( .A1(n1127), .A2(n1126), .B1(n1125), .B2(n1124
   ), .ZN(n1136)
       );
IINR4D0 U1660 ( .A1(n1131), .A2(n1130), .B1(n1129), .B2(n1128
   ), .ZN(n1133)
       );
CKND2D0 U1661 ( .A1(n1133), .A2(n1132), .ZN(n1134) );
AOI22D0 U1662 ( .A1(counter[7]), .A2(n1136), .B1(n1135), .B2(
   n1134), .ZN(
      n1154) );
IND4D0 U1663 ( .A1(n1140), .B1(n1139), .B2(n1138), .B3(n1137)
   , .ZN(n1142) );
OAI31D0 U1664 ( .A1(n1144), .A2(n1143), .A3(n1142), .B(n1141)
   , .ZN(n1152) );
INVD0 U1665 ( .I(n1988), .ZN(n2173) );
NR2D0 U1666 ( .A1(n1859), .A2(counter[5]), .ZN(n1577) );
CKAN2D0 U1667 ( .A1(n1145), .A2(n1577), .Z(n1146) );
AOI31D0 U1668 ( .A1(n2173), .A2(counter[5]), .A3(n1147), .B(
   n1146), .ZN(
      n1150) );
OAI22D0 U1669 ( .A1(counter[7]), .A2(n1150), .B1(n1149), .B2(
   n1148), .ZN(
      n1151) );
AOI32D0 U1670 ( .A1(n1154), .A2(n1153), .A3(n1152), .B1(
   counter[9]), .B2(
```

```
      n1151), .ZN(n1155) );
AOI32D0 U1671 ( .A1(n1158), .A2(n1157), .A3(n1156), .B1(
   counter[10]), .B2(
      n1155), .ZN(n1159) );
OAI32D0 U1672 ( .A1(counter[8]), .A2(n1161), .A3(n1160), .B1(
   n1177), .B2(
      n1159), .ZN(n1171) );
INVD0 U1673 ( .I(n1162), .ZN(n1167) );
INR4D0 U1674 ( .A1(n1165), .B1(counter[8]), .B2(n1164), .B3(
   n1163), .ZN(
      n1166) );
AOI22D0 U1675 ( .A1(counter[8]), .A2(n1168), .B1(n1167), .B2(
   n1166), .ZN(
      n1170) );
OAI221D0 U1676 ( .A1(counter[11]), .A2(n1171), .B1(n3956), .
   B2(n1170), .C(
      n1169), .ZN(n1172) );
OAI211D0 U1677 ( .A1(n4545), .A2(\q_out[5]_BAR ), .B(n1173),
   .C(n1172), .ZN(
      n4817) );
NR2D0 U1678 ( .A1(counter[0]), .A2(n1179), .ZN(N4767) );
AOI211D0 U1679 ( .A1(n1859), .A2(n2281), .B(n1179), .C(n1878)
   , .ZN(N4770) );
CKND2D0 U1680 ( .A1(counter[7]), .A2(n1174), .ZN(n1176) );
AOI211D0 U1681 ( .A1(n1177), .A2(n1176), .B(n1175), .C(n1179)
   , .ZN(N4775) );
NR2D0 U1682 ( .A1(n1179), .A2(n1178), .ZN(N4768) );
CKND2D0 U1683 ( .A1(n3757), .A2(n4098), .ZN(n2639) );
NR2D0 U1684 ( .A1(n4351), .A2(n2639), .ZN(n1528) );
INVD0 U1685 ( .I(n1528), .ZN(n3661) );
NR2D0 U1686 ( .A1(n3661), .A2(counter[3]), .ZN(n3919) );
INVD0 U1687 ( .I(n3919), .ZN(n3788) );
NR2D0 U1688 ( .A1(n4025), .A2(n2732), .ZN(n3787) );
CKND2D0 U1689 ( .A1(n3737), .A2(n3787), .ZN(n3913) );
CKND2D0 U1690 ( .A1(n3788), .A2(n3913), .ZN(n1286) );
INVD0 U1691 ( .I(n3595), .ZN(n3271) );
CKND2D0 U1692 ( .A1(n3949), .A2(n3271), .ZN(n3633) );
NR2D0 U1693 ( .A1(n3633), .A2(counter[2]), .ZN(n1288) );
CKND2D0 U1694 ( .A1(n1198), .A2(n3176), .ZN(n1466) );
NR2D0 U1695 ( .A1(counter[4]), .A2(n1466), .ZN(n3854) );
NR2D0 U1696 ( .A1(n1288), .A2(n3854), .ZN(n3713) );
CKND2D0 U1697 ( .A1(counter[4]), .A2(n3176), .ZN(n1505) );
NR2D0 U1698 ( .A1(n2088), .A2(n1505), .ZN(n2671) );
INVD0 U1699 ( .I(n2671), .ZN(n2697) );
INVD0 U1700 ( .I(n2639), .ZN(n1820) );
CKND2D0 U1701 ( .A1(n1269), .A2(n1820), .ZN(n3676) );
CKND2D0 U1702 ( .A1(n2697), .A2(n3676), .ZN(n1321) );
INVD0 U1703 ( .I(n1321), .ZN(n2870) );
```

```
  NR2D0 U1704 ( .A1(counter[2]), .A2(n4018), .ZN(n1527) );
  CKND2D0 U1705 ( .A1(n3949), .A2(n1527), .ZN(n3700) );
  NR2D0 U1706 ( .A1(n3700), .A2(counter[1]), .ZN(n3860) );
  INVD0 U1707 ( .I(n3860), .ZN(n2711) );
  IND4D0 U1708 ( .A1(n1286), .B1(n3713), .B2(n2870), .B3(n2711)
     , .ZN(n1181) );
  CKND2D0 U1709 ( .A1(n1198), .A2(n1820), .ZN(n3813) );
  CKND2D0 U1710 ( .A1(counter[4]), .A2(n3749), .ZN(n4017) );
  INVD0 U1711 ( .I(n4017), .ZN(n2714) );
  CKND2D0 U1712 ( .A1(n4445), .A2(n2714), .ZN(n3609) );
  CKND2D0 U1713 ( .A1(n3813), .A2(n3609), .ZN(n2819) );
  INVD0 U1714 ( .I(n2819), .ZN(n2628) );
  NR2D0 U1715 ( .A1(n1556), .A2(n1270), .ZN(n2762) );
  INVD0 U1716 ( .I(n3676), .ZN(n3751) );
  NR2D0 U1717 ( .A1(n2762), .A2(n3751), .ZN(n3807) );
  INVD0 U1718 ( .I(n3807), .ZN(n1322) );
  NR2D0 U1719 ( .A1(n3737), .A2(n3633), .ZN(n2919) );
  NR2D0 U1720 ( .A1(n1322), .A2(n2919), .ZN(n2821) );
  NR2D0 U1721 ( .A1(n3441), .A2(n1188), .ZN(n1193) );
  CKND2D0 U1722 ( .A1(n3880), .A2(n1193), .ZN(n3927) );
  INVD0 U1723 ( .I(n3927), .ZN(n3742) );
  NR2D0 U1724 ( .A1(n3860), .A2(n3742), .ZN(n1300) );
  INVD0 U1725 ( .I(n3787), .ZN(n1187) );
  ND4D0 U1726 ( .A1(n2628), .A2(n2821), .A3(n1300), .A4(n1187),
      .ZN(n1180) );
  AOI22D0 U1727 ( .A1(n3933), .A2(n1181), .B1(n3844), .B2(n1180
     ), .ZN(n1186)
        );
  INVD0 U1728 ( .I(n3913), .ZN(n3783) );
  CKND2D0 U1729 ( .A1(n2638), .A2(n3833), .ZN(n3677) );
  INVD0 U1730 ( .I(n3677), .ZN(n3786) );
  NR2D0 U1731 ( .A1(n3737), .A2(n4018), .ZN(n3950) );
  CKND2D0 U1732 ( .A1(n3749), .A2(n3950), .ZN(n3249) );
  INVD0 U1733 ( .I(n3249), .ZN(n4143) );
  CKND2D0 U1734 ( .A1(n3757), .A2(n4143), .ZN(n3835) );
  INVD0 U1735 ( .I(n3835), .ZN(n3614) );
  NR2D0 U1736 ( .A1(n3786), .A2(n3614), .ZN(n2765) );
  INVD0 U1737 ( .I(n2765), .ZN(n2862) );
  INVD0 U1738 ( .I(n2762), .ZN(n3631) );
  CKND2D0 U1739 ( .A1(n2711), .A2(n3631), .ZN(n2905) );
  OAI31D0 U1740 ( .A1(n3783), .A2(n2862), .A3(n2905), .B(n4000)
     , .ZN(n1185) );
  NR2D0 U1741 ( .A1(n3759), .A2(counter[11]), .ZN(n3870) );
  CKND2D0 U1742 ( .A1(n1182), .A2(n1284), .ZN(n2748) );
  INVD0 U1743 ( .I(n2748), .ZN(n2820) );
  NR2D0 U1744 ( .A1(n3870), .A2(n2820), .ZN(n3658) );
  INVD0 U1745 ( .I(n3658), .ZN(n3775) );
  NR2D0 U1746 ( .A1(n1183), .A2(n2424), .ZN(n2706) );
```

```
  INVD0 U1747 ( .I(n2706), .ZN(n3617) );
  NR2D0 U1748 ( .A1(n3617), .A2(counter[3]), .ZN(n3636) );
  INVD0 U1749 ( .I(n3636), .ZN(n4050) );
  NR2D0 U1750 ( .A1(n3754), .A2(n3837), .ZN(n3890) );
  INVD0 U1751 ( .I(n3890), .ZN(n3816) );
  CKND2D0 U1752 ( .A1(n4050), .A2(n3816), .ZN(n3605) );
  OAI31D0 U1753 ( .A1(n1193), .A2(n3775), .A3(n3605), .B(n3769)
     , .ZN(n1184) );
  AOI31D0 U1754 ( .A1(n1186), .A2(n1185), .A3(n1184), .B(n4410)
     , .ZN(n1335) );
  CKND2D0 U1755 ( .A1(n1214), .A2(n3176), .ZN(n4387) );
  NR2D0 U1756 ( .A1(n3757), .A2(n4387), .ZN(n3608) );
  INVD0 U1757 ( .I(n3608), .ZN(n3619) );
  CKND2D0 U1758 ( .A1(n3813), .A2(n3619), .ZN(n2908) );
  INVD0 U1759 ( .I(n2908), .ZN(n2616) );
  INVD0 U1760 ( .I(n3950), .ZN(n4321) );
  NR2D0 U1761 ( .A1(n4321), .A2(n4017), .ZN(n3892) );
  NR2D0 U1762 ( .A1(n3892), .A2(n2820), .ZN(n3683) );
  NR2D0 U1763 ( .A1(n3737), .A2(n1187), .ZN(n3891) );
  CKND2D0 U1764 ( .A1(n2978), .A2(n1820), .ZN(n3715) );
  INVD0 U1765 ( .I(n3715), .ZN(n2796) );
  CKND2D0 U1766 ( .A1(n1859), .A2(n2796), .ZN(n2938) );
  INVD0 U1767 ( .I(n2938), .ZN(n3650) );
  NR2D0 U1768 ( .A1(n3891), .A2(n3650), .ZN(n2835) );
  NR2D0 U1769 ( .A1(n1259), .A2(n1188), .ZN(n3638) );
  INVD0 U1770 ( .I(n3638), .ZN(n2873) );
  NR2D0 U1771 ( .A1(n2873), .A2(counter[4]), .ZN(n3716) );
  NR2D0 U1772 ( .A1(n1270), .A2(n2424), .ZN(n3843) );
  NR2D0 U1773 ( .A1(n3716), .A2(n3843), .ZN(n3931) );
  ND4D0 U1774 ( .A1(n2616), .A2(n3683), .A3(n2835), .A4(n3931),
      .ZN(n1192) );
  NR2D0 U1775 ( .A1(n3892), .A2(n3870), .ZN(n1263) );
  CKND2D0 U1776 ( .A1(n3813), .A2(n3631), .ZN(n2897) );
  NR2D0 U1777 ( .A1(n2897), .A2(n3786), .ZN(n2941) );
  AOI31D0 U1778 ( .A1(n1263), .A2(n2941), .A3(n2870), .B(n4045)
     , .ZN(n1191) );
  CKND2D0 U1779 ( .A1(n3956), .A2(n1189), .ZN(n3987) );
  CKND2D0 U1780 ( .A1(n3788), .A2(n3987), .ZN(n3810) );
  INVD0 U1781 ( .I(n3810), .ZN(n2899) );
  NR3D0 U1782 ( .A1(n3737), .A2(n4024), .A3(n4025), .ZN(n3674)
     );
  NR2D0 U1783 ( .A1(n2265), .A2(n2639), .ZN(n1314) );
  NR2D0 U1784 ( .A1(n3674), .A2(n1314), .ZN(n3861) );
  AOI31D0 U1785 ( .A1(n2899), .A2(n1263), .A3(n3861), .B(n4059)
     , .ZN(n1190) );
  AOI211D0 U1786 ( .A1(n3933), .A2(n1192), .B(n1191), .C(n1190)
     , .ZN(n1207) );
  CKND2D0 U1787 ( .A1(n1193), .A2(n1865), .ZN(n4167) );
```

```
  CKND2D0 U1788 ( .A1(n3816), .A2(n3631), .ZN(n3598) );
  INVD0 U1789 ( .I(n3674), .ZN(n3621) );
  NR2D0 U1790 ( .A1(n1194), .A2(n2788), .ZN(n3656) );
  INVD0 U1791 ( .I(n3656), .ZN(n3637) );
  CKND2D0 U1792 ( .A1(n3621), .A2(n3637), .ZN(n1230) );
  INR4D0 U1793 ( .A1(n4167), .B1(n3860), .B2(n3598), .B3(n1230)
     , .ZN(n1197) );
  NR2D0 U1794 ( .A1(n1270), .A2(n2788), .ZN(n2882) );
  OR2D0 U1795 ( .A1(n4025), .A2(n2088), .Z(n3497) );
  NR2D0 U1796 ( .A1(n3497), .A2(counter[4]), .ZN(n3723) );
  NR2D0 U1797 ( .A1(n2882), .A2(n3723), .ZN(n2736) );
  INVD0 U1798 ( .I(n3854), .ZN(n4011) );
  ND3D0 U1799 ( .A1(n2736), .A2(n1300), .A3(n4011), .ZN(n2742)
     );
  INVD0 U1800 ( .I(n2742), .ZN(n1196) );
  INVD0 U1801 ( .I(n3723), .ZN(n4022) );
  CKND2D0 U1802 ( .A1(n3631), .A2(n4022), .ZN(n3967) );
  NR3D0 U1803 ( .A1(n1303), .A2(n3608), .A3(n3891), .ZN(n2678)
     );
  NR2D0 U1804 ( .A1(n3742), .A2(n3854), .ZN(n2929) );
  NR2D0 U1805 ( .A1(n3757), .A2(n4330), .ZN(n3971) );
  INVD0 U1806 ( .I(n3971), .ZN(n3745) );
  ND3D0 U1807 ( .A1(n2678), .A2(n2929), .A3(n3745), .ZN(n2917)
     );
  NR2D0 U1808 ( .A1(n3967), .A2(n2917), .ZN(n1195) );
  OAI222D0 U1809 ( .A1(n4059), .A2(n1197), .B1(n4045), .B2(
     n1196), .C1(n4010),
        .C2(n1195), .ZN(n1204) );
  INVD0 U1810 ( .I(n3529), .ZN(n1438) );
  CKND2D0 U1811 ( .A1(n1438), .A2(n1820), .ZN(n2877) );
  INVD0 U1812 ( .I(n2877), .ZN(n3782) );
  NR2D0 U1813 ( .A1(n3472), .A2(n2639), .ZN(n4043) );
  NR4D0 U1814 ( .A1(n3542), .A2(n3843), .A3(n3782), .A4(n4043),
      .ZN(n2827) );
  NR2D0 U1815 ( .A1(n1198), .A2(n2125), .ZN(n4114) );
  INVD0 U1816 ( .I(n2882), .ZN(n1279) );
  CKND2D0 U1817 ( .A1(n1279), .A2(n3816), .ZN(n2759) );
  INVD0 U1818 ( .I(n2759), .ZN(n2754) );
  OAI211D0 U1819 ( .A1(n4114), .A2(n4018), .B(n2754), .C(n4022)
     , .ZN(n1199) );
  NR4D0 U1820 ( .A1(n3860), .A2(n3892), .A3(n3891), .A4(n1322),
      .ZN(n2757) );
  MAOI22D0 U1821 ( .A1(n3769), .A2(n1199), .B1(n2757), .B2(
     n4038), .ZN(n1202)
        );
  NR2D0 U1822 ( .A1(n4100), .A2(n1505), .ZN(n4044) );
  INVD0 U1823 ( .I(n4044), .ZN(n3672) );
  CKND2D0 U1824 ( .A1(n3633), .A2(n3672), .ZN(n1320) );
```

```
CKND2D0 U1825 ( .A1(n3757), .A2(n3176), .ZN(n1501) );
OAI211D0 U1826 ( .A1(n1501), .A2(n3821), .B(n2929), .C(n2873)
   , .ZN(n1200) );
OAI31D0 U1827 ( .A1(n3614), .A2(n1320), .A3(n1200), .B(n4000)
   , .ZN(n1201) );
OAI211D0 U1828 ( .A1(n2827), .A2(n4045), .B(n1202), .C(n1201)
   , .ZN(n1203) );
AOI22D0 U1829 ( .A1(n4490), .A2(n1204), .B1(n4063), .B2(n1203
   ), .ZN(n1206)
      );
CKND2D0 U1830 ( .A1(n3987), .A2(n3835), .ZN(n4058) );
NR2D0 U1831 ( .A1(n3783), .A2(n3843), .ZN(n1306) );
NR2D0 U1832 ( .A1(n3742), .A2(n3892), .ZN(n3616) );
CKND2D0 U1833 ( .A1(n1306), .A2(n3616), .ZN(n2825) );
OAI31D0 U1834 ( .A1(n1321), .A2(n4058), .A3(n2825), .B(n3769)
   , .ZN(n1205) );
INVD0 U1835 ( .I(n3850), .ZN(n4535) );
AOI32D0 U1836 ( .A1(n1207), .A2(n1206), .A3(n1205), .B1(n4535
   ), .B2(n1206),
      .ZN(n1334) );
CKND2D0 U1837 ( .A1(n3621), .A2(n3676), .ZN(n3690) );
CKND2D0 U1838 ( .A1(counter[11]), .A2(n1208), .ZN(n2789) );
NR2D0 U1839 ( .A1(n4351), .A2(n2789), .ZN(n1272) );
NR3D0 U1840 ( .A1(n3757), .A2(n2088), .A3(n1862), .ZN(n2797)
   );
NR2D0 U1841 ( .A1(n3757), .A2(n4222), .ZN(n1268) );
AOI22D0 U1842 ( .A1(counter[11]), .A2(n2642), .B1(n1214), .B2
   (n1268), .ZN(
      n3964) );
CKND2D0 U1843 ( .A1(n1209), .A2(n3732), .ZN(n3985) );
IND3D0 U1844 ( .A1(n2797), .B1(n3964), .B2(n3985), .ZN(n2652)
    );
CKND2D0 U1845 ( .A1(n2938), .A2(n3835), .ZN(n4028) );
NR4D0 U1846 ( .A1(n1272), .A2(n3782), .A3(n2652), .A4(n4028),
    .ZN(n1210) );
NR2D0 U1847 ( .A1(n2192), .A2(n3332), .ZN(n2615) );
CKND2D0 U1848 ( .A1(n2615), .A2(n1859), .ZN(n3739) );
NR2D0 U1849 ( .A1(n3462), .A2(counter[2]), .ZN(n3881) );
INVD0 U1850 ( .I(n3881), .ZN(n3955) );
NR2D0 U1851 ( .A1(n3955), .A2(n4017), .ZN(n3889) );
INVD0 U1852 ( .I(n3889), .ZN(n3925) );
ND4D0 U1853 ( .A1(n1210), .A2(n1279), .A3(n3739), .A4(n3925),
    .ZN(n1225) );
INVD0 U1854 ( .I(n1268), .ZN(n1798) );
NR2D0 U1855 ( .A1(n1211), .A2(n1798), .ZN(n2644) );
INVD0 U1856 ( .I(n2125), .ZN(n4263) );
CKND2D0 U1857 ( .A1(counter[11]), .A2(n2776), .ZN(n1785) );
```

```
OAI22D0 U1858 ( .A1(n1212), .A2(n2789), .B1(n4263), .B2(n1785
   ), .ZN(n1216)
      );
NR2D0 U1859 ( .A1(n3956), .A2(n1213), .ZN(n2799) );
CKND2D0 U1860 ( .A1(n1214), .A2(n2799), .ZN(n3962) );
INVD0 U1861 ( .I(n2919), .ZN(n3986) );
CKND2D0 U1862 ( .A1(n3962), .A2(n3986), .ZN(n2779) );
NR2D0 U1863 ( .A1(n3742), .A2(n2820), .ZN(n2875) );
NR2D0 U1864 ( .A1(n1269), .A2(n4084), .ZN(n2623) );
OR2D0 U1865 ( .A1(n2623), .A2(n1798), .Z(n2655) );
IND4D0 U1866 ( .A1(n2779), .B1(n2765), .B2(n2875), .B3(n2655)
   , .ZN(n1215) );
NR4D0 U1867 ( .A1(n3783), .A2(n2644), .A3(n1216), .A4(n1215),
    .ZN(n1223) );
CKND2D0 U1868 ( .A1(n4118), .A2(n1820), .ZN(n3657) );
CKND2D0 U1869 ( .A1(n2697), .A2(n3657), .ZN(n3632) );
CKND2D0 U1870 ( .A1(n2938), .A2(n4011), .ZN(n2705) );
INVD0 U1871 ( .I(n2705), .ZN(n2817) );
OAI211D0 U1872 ( .A1(n2773), .A2(n3956), .B(n2817), .C(n3985)
   , .ZN(n2619) );
OR2D0 U1873 ( .A1(n2538), .A2(n3529), .Z(n1217) );
OAI211D0 U1874 ( .A1(n2789), .A2(n3390), .B(n1217), .C(n3835)
   , .ZN(n1218) );
OAI31D0 U1875 ( .A1(n3632), .A2(n2619), .A3(n1218), .B(n3933)
   , .ZN(n1222) );
CKND2D0 U1876 ( .A1(n3745), .A2(n3835), .ZN(n2750) );
NR2D0 U1877 ( .A1(n2750), .A2(n3608), .ZN(n2837) );
INVD0 U1878 ( .I(n2837), .ZN(n2913) );
CKND2D0 U1879 ( .A1(n1269), .A2(n2799), .ZN(n3916) );
CKND2D0 U1880 ( .A1(n3916), .A2(n3739), .ZN(n2630) );
NR2D0 U1881 ( .A1(n2088), .A2(n2639), .ZN(n3655) );
NR2D0 U1882 ( .A1(n3956), .A2(n1651), .ZN(n1535) );
CKND2D0 U1883 ( .A1(n1872), .A2(n1535), .ZN(n1219) );
NR2D0 U1884 ( .A1(n1859), .A2(n1219), .ZN(n2780) );
INVD0 U1885 ( .I(n2780), .ZN(n3905) );
IND4D0 U1886 ( .A1(n3655), .B1(n3677), .B2(n3985), .B3(n3905)
   , .ZN(n1220) );
OAI31D0 U1887 ( .A1(n2913), .A2(n2630), .A3(n1220), .B(n3844)
   , .ZN(n1221) );
OAI211D0 U1888 ( .A1(n1223), .A2(n4059), .B(n1222), .C(n1221)
   , .ZN(n1224) );
AOI221D0 U1889 ( .A1(n3690), .A2(n3769), .B1(n1225), .B2(
   n3769), .C(n1224),
      .ZN(n1253) );
INVD0 U1890 ( .I(n3657), .ZN(n3710) );
NR2D0 U1891 ( .A1(n3860), .A2(n3710), .ZN(n2859) );
INVD0 U1892 ( .I(n2859), .ZN(n4021) );
INVD0 U1893 ( .I(n1288), .ZN(n3789) );
```

```
  NR2D0 U1894 ( .A1(n3891), .A2(n2882), .ZN(n2860) );
  CKND2D0 U1895 ( .A1(n3789), .A2(n2860), .ZN(n2684) );
  CKND2D0 U1896 ( .A1(n3619), .A2(n3621), .ZN(n3958) );
  NR4D0 U1897 ( .A1(n3783), .A2(n3890), .A3(n2684), .A4(n3958),
      .ZN(n2839) );
  AOI21D0 U1898 ( .A1(n2839), .A2(n3637), .B(n4045), .ZN(n1227)
      );
  NR2D0 U1899 ( .A1(n3971), .A2(n3892), .ZN(n3856) );
  INVD0 U1900 ( .I(n3856), .ZN(n3867) );
  INVD0 U1901 ( .I(n3870), .ZN(n1301) );
  CKND2D0 U1902 ( .A1(n1301), .A2(n4011), .ZN(n3811) );
  NR2D0 U1903 ( .A1(n3867), .A2(n3811), .ZN(n2651) );
  INVD0 U1904 ( .I(n3891), .ZN(n3692) );
  OAI211D0 U1905 ( .A1(n2732), .A2(n3822), .B(n4387), .C(n3692)
     , .ZN(n2782) );
  INVD0 U1906 ( .I(n2782), .ZN(n2826) );
  OAI22D0 U1907 ( .A1(n2651), .A2(n4059), .B1(n2826), .B2(n4010
     ), .ZN(n1226)
         );
  AO211D0 U1908 ( .A1(n3933), .A2(n4021), .B(n1227), .C(n1226),
      .Z(n1239) );
  CKND2D0 U1909 ( .A1(n3788), .A2(n3835), .ZN(n4042) );
  INVD0 U1910 ( .I(n4042), .ZN(n3606) );
  ND3D0 U1911 ( .A1(n3658), .A2(n3606), .A3(n2736), .ZN(n2824)
     );
  CKND2D0 U1912 ( .A1(n3927), .A2(n3856), .ZN(n2740) );
  NR2D0 U1913 ( .A1(n3674), .A2(n4021), .ZN(n2912) );
  NR2D0 U1914 ( .A1(n3843), .A2(n3854), .ZN(n1254) );
  IND4D0 U1915 ( .A1(n2740), .B1(n2912), .B2(n1254), .B3(n3692)
     , .ZN(n1228) );
  AOI22D0 U1916 ( .A1(n3769), .A2(n2824), .B1(n3844), .B2(n1228
     ), .ZN(n1233)
         );
  INVD0 U1917 ( .I(n1254), .ZN(n2901) );
  ND4D0 U1918 ( .A1(n2860), .A2(n2837), .A3(n3986), .A4(n3672),
      .ZN(n1229) );
  OAI21D0 U1919 ( .A1(n2901), .A2(n1229), .B(n3933), .ZN(n1232)
      );
  OAI31D0 U1920 ( .A1(n3636), .A2(n2919), .A3(n1230), .B(n4000)
     , .ZN(n1231) );
  AOI31D0 U1921 ( .A1(n1233), .A2(n1232), .A3(n1231), .B(n3381)
     , .ZN(n1238) );
  CKND2D0 U1922 ( .A1(n1937), .A2(n1234), .ZN(n3671) );
  CKND2D0 U1923 ( .A1(n3671), .A2(n3676), .ZN(n2775) );
  CKND2D0 U1924 ( .A1(n4050), .A2(n4022), .ZN(n3740) );
  NR4D0 U1925 ( .A1(n3614), .A2(n4021), .A3(n2775), .A4(n3740),
      .ZN(n1236) );
  CKND2D0 U1926 ( .A1(n4490), .A2(n3933), .ZN(n1235) );
```

```
AOI31D0 U1927 ( .A1(n1236), .A2(n3816), .A3(n3913), .B(n1235)
   , .ZN(n1237) );
AOI211D0 U1928 ( .A1(n3707), .A2(n1239), .B(n1238), .C(n1237)
   , .ZN(n1252) );
NR2D0 U1929 ( .A1(n3956), .A2(n2627), .ZN(n3944) );
NR4D0 U1930 ( .A1(n1288), .A2(n3723), .A3(n3944), .A4(n3598),
    .ZN(n1240) );
CKND2D0 U1931 ( .A1(n2654), .A2(n1268), .ZN(n1539) );
CKND2D0 U1932 ( .A1(n2760), .A2(n1865), .ZN(n3760) );
INVD0 U1933 ( .I(n3760), .ZN(n3730) );
CKND2D0 U1934 ( .A1(n4256), .A2(n3730), .ZN(n3961) );
AOI31D0 U1935 ( .A1(n1240), .A2(n1539), .A3(n3961), .B(n4038)
   , .ZN(n1250) );
INVD0 U1936 ( .I(n3616), .ZN(n2733) );
INVD0 U1937 ( .I(n3713), .ZN(n3781) );
NR4D0 U1938 ( .A1(n3751), .A2(n1314), .A3(n2733), .A4(n3781),
    .ZN(n1241) );
AOI21D0 U1939 ( .A1(n2736), .A2(n1241), .B(n4059), .ZN(n1249)
    );
INVD0 U1940 ( .I(n3843), .ZN(n3883) );
CKND2D0 U1941 ( .A1(n2711), .A2(n3883), .ZN(n3660) );
NR4D0 U1942 ( .A1(n3870), .A2(n3919), .A3(n2740), .A4(n3660),
    .ZN(n1247) );
CKND2D0 U1943 ( .A1(n2711), .A2(n4050), .ZN(n2918) );
INVD0 U1944 ( .I(n1501), .ZN(n1667) );
CKND2D0 U1945 ( .A1(n1987), .A2(n1667), .ZN(n4023) );
INVD0 U1946 ( .I(n4023), .ZN(n3832) );
NR2D0 U1947 ( .A1(n2918), .A2(n3832), .ZN(n2822) );
INVD0 U1948 ( .I(n2822), .ZN(n2672) );
CKND2D0 U1949 ( .A1(n3609), .A2(n3616), .ZN(n2830) );
NR2D0 U1950 ( .A1(n2830), .A2(n3870), .ZN(n2840) );
NR2D0 U1951 ( .A1(n1242), .A2(n2789), .ZN(n3903) );
AOI221D0 U1952 ( .A1(n1243), .A2(n2799), .B1(n1987), .B2(
   n2799), .C(n3903),
      .ZN(n1244) );
OAI211D0 U1953 ( .A1(n2088), .A2(n1798), .B(n2840), .C(n1244)
   , .ZN(n1245) );
NR4D0 U1954 ( .A1(n3674), .A2(n3944), .A3(n2672), .A4(n1245),
    .ZN(n1246) );
OAI22D0 U1955 ( .A1(n1247), .A2(n4045), .B1(n1246), .B2(n4010
   ), .ZN(n1248)
       );
INVD0 U1956 ( .I(n3777), .ZN(n4305) );
OAI31D0 U1957 ( .A1(n1250), .A2(n1249), .A3(n1248), .B(n4305)
   , .ZN(n1251) );
OAI211D0 U1958 ( .A1(n1253), .A2(n4129), .B(n1252), .C(n1251)
   , .ZN(n1333) );
INVD0 U1959 ( .I(n3716), .ZN(n3917) );
```

```
CKND2D0 U1960 ( .A1(n3917), .A2(n3677), .ZN(n2895) );
CKND2D0 U1961 ( .A1(n3637), .A2(n4011), .ZN(n2894) );
AOI211D0 U1962 ( .A1(n4079), .A2(n1820), .B(n2895), .C(n2894)
   , .ZN(n2721) );
INVD0 U1963 ( .I(n2684), .ZN(n2712) );
NR2D0 U1964 ( .A1(n3892), .A2(n3710), .ZN(n2708) );
ND4D0 U1965 ( .A1(n2712), .A2(n1254), .A3(n2708), .A4(n3617),
    .ZN(n1255) );
CKND2D0 U1966 ( .A1(n4023), .A2(n3816), .ZN(n2872) );
AOI22D0 U1967 ( .A1(n3933), .A2(n1255), .B1(n4000), .B2(n2872
   ), .ZN(n1257)
       );
INVD0 U1968 ( .I(n1320), .ZN(n2935) );
ND3D0 U1969 ( .A1(n2935), .A2(n3676), .A3(n2877), .ZN(n2852)
   );
CKND2D0 U1970 ( .A1(n1306), .A2(n3621), .ZN(n2703) );
OAI21D0 U1971 ( .A1(n2852), .A2(n2703), .B(n3769), .ZN(n1256)
    );
OAI211D0 U1972 ( .A1(n2721), .A2(n4045), .B(n1257), .C(n1256)
   , .ZN(n1299) );
AOI21D0 U1973 ( .A1(n1260), .A2(n1259), .B(n1798), .ZN(n2649)
    );
NR2D0 U1974 ( .A1(n3946), .A2(n4222), .ZN(n3738) );
CKND2D0 U1975 ( .A1(n3987), .A2(n4022), .ZN(n2844) );
NR4D0 U1976 ( .A1(n3843), .A2(n3738), .A3(n2844), .A4(n3598),
    .ZN(n1258) );
IND4D0 U1977 ( .A1(n2649), .B1(n2912), .B2(n1258), .B3(n1301)
   , .ZN(n1267) );
CKND2D0 U1978 ( .A1(n1260), .A2(n1259), .ZN(n1262) );
INVD0 U1979 ( .I(n3916), .ZN(n1261) );
AOI211D0 U1980 ( .A1(n1535), .A2(n1262), .B(n1261), .C(n2779)
   , .ZN(n2635) );
INVD0 U1981 ( .I(n2636), .ZN(n2618) );
NR2D0 U1982 ( .A1(n2617), .A2(n2618), .ZN(n1264) );
INVD0 U1983 ( .I(n1306), .ZN(n2914) );
INVD0 U1984 ( .I(n1263), .ZN(n3649) );
NR4D0 U1985 ( .A1(n1272), .A2(n1264), .A3(n2914), .A4(n3649),
    .ZN(n1265) );
CKND2D0 U1986 ( .A1(n4079), .A2(n1268), .ZN(n2657) );
ND4D0 U1987 ( .A1(n2635), .A2(n1265), .A3(n2938), .A4(n2657),
    .ZN(n1266) );
AOI22D0 U1988 ( .A1(n4000), .A2(n1267), .B1(n3844), .B2(n1266
   ), .ZN(n1278)
       );
INVD0 U1989 ( .I(n3743), .ZN(n2995) );
CKND2D0 U1990 ( .A1(n1269), .A2(n1268), .ZN(n3912) );
CKND2D0 U1991 ( .A1(n4118), .A2(n2799), .ZN(n3984) );
```

```
OAI211D0 U1992 ( .A1(n2995), .A2(n1270), .B(n3912), .C(n3984)
   , .ZN(n2641) );
OAI211D0 U1993 ( .A1(n2623), .A2(n2789), .B(n3692), .C(n3835)
   , .ZN(n2855) );
IND4D0 U1994 ( .A1(n2855), .B1(n3683), .B2(n1300), .B3(n3739)
   , .ZN(n1271) );
OAI31D0 U1995 ( .A1(n3660), .A2(n2641), .A3(n1271), .B(n3933)
   , .ZN(n1277) );
CKND2D0 U1996 ( .A1(counter[3]), .A2(n1272), .ZN(n2626) );
INVD0 U1997 ( .I(n2626), .ZN(n2802) );
NR2D0 U1998 ( .A1(n3472), .A2(n2351), .ZN(n4088) );
CKND2D0 U1999 ( .A1(n4088), .A2(n3757), .ZN(n2854) );
OAI21D0 U2000 ( .A1(n3759), .A2(n3956), .B(n2854), .ZN(n2643)
    );
INVD0 U2001 ( .I(n3683), .ZN(n3756) );
AOI211D0 U2002 ( .A1(n1535), .A2(n4112), .B(n2643), .C(n3756)
   , .ZN(n1273) );
CKND2D0 U2003 ( .A1(n1273), .A2(n3913), .ZN(n2808) );
AOI211D0 U2004 ( .A1(n2799), .A2(n4112), .B(n2802), .C(n2808)
   , .ZN(n1274) );
CKND2D0 U2005 ( .A1(n3949), .A2(n3544), .ZN(n3882) );
ND3D0 U2006 ( .A1(n1274), .A2(n3676), .A3(n3882), .ZN(n1275)
   );
OAI31D0 U2007 ( .A1(n3971), .A2(n2649), .A3(n1275), .B(n3769)
   , .ZN(n1276) );
AOI31D0 U2008 ( .A1(n1278), .A2(n1277), .A3(n1276), .B(n4277)
   , .ZN(n1298) );
INVD0 U2009 ( .I(n3605), .ZN(n3864) );
NR2D0 U2010 ( .A1(n3390), .A2(n1505), .ZN(n2725) );
INVD0 U2011 ( .I(n2725), .ZN(n2299) );
ND4D0 U2012 ( .A1(n2912), .A2(n3864), .A3(n2299), .A4(n3788),
    .ZN(n1283) );
ND3D0 U2013 ( .A1(n2711), .A2(n3816), .A3(n1301), .ZN(n3654)
   );
CKND2D0 U2014 ( .A1(n3637), .A2(n4022), .ZN(n3695) );
INVD0 U2015 ( .I(n3695), .ZN(n2786) );
INVD0 U2016 ( .I(n3987), .ZN(n3898) );
NR2D0 U2017 ( .A1(n3783), .A2(n3898), .ZN(n2940) );
CKND2D0 U2018 ( .A1(n2786), .A2(n2940), .ZN(n2950) );
NR2D0 U2019 ( .A1(n3654), .A2(n2950), .ZN(n2646) );
AOI21D0 U2020 ( .A1(n2646), .A2(n3713), .B(n4010), .ZN(n1282)
    );
NR4D0 U2021 ( .A1(n3786), .A2(n3674), .A3(n2705), .A4(n4058),
    .ZN(n2709) );
CKND2D0 U2022 ( .A1(n1279), .A2(n3609), .ZN(n2690) );
CKND2D0 U2023 ( .A1(n2817), .A2(n3788), .ZN(n2880) );
NR4D0 U2024 ( .A1(n3674), .A2(n4044), .A3(n2690), .A4(n2880),
    .ZN(n1280) );
```

```
OAI22D0 U2025 ( .A1(n2709), .A2(n4045), .B1(n1280), .B2(n4059
   ), .ZN(n1281)
      );
AOI211D0 U2026 ( .A1(n3933), .A2(n1283), .B(n1282), .C(n1281)
   , .ZN(n1296) );
INVD0 U2027 ( .I(n2750), .ZN(n2753) );
CKND2D0 U2028 ( .A1(n1285), .A2(n1284), .ZN(n3758) );
INVD0 U2029 ( .I(n3758), .ZN(n3651) );
NR2D0 U2030 ( .A1(n3651), .A2(n2762), .ZN(n3681) );
ND4D0 U2031 ( .A1(n2678), .A2(n2753), .A3(n3681), .A4(n2697),
    .ZN(n1294) );
CKND2D0 U2032 ( .A1(n4050), .A2(n3676), .ZN(n2896) );
NR2D0 U2033 ( .A1(n2896), .A2(n3870), .ZN(n2930) );
CKND2D0 U2034 ( .A1(n3672), .A2(n2930), .ZN(n2871) );
INVD0 U2035 ( .I(n2871), .ZN(n2722) );
NR4D0 U2036 ( .A1(n3860), .A2(n2882), .A3(n3832), .A4(n1286),
    .ZN(n1287) );
AOI31D0 U2037 ( .A1(n2722), .A2(n2753), .A3(n1287), .B(n4059)
   , .ZN(n1293) );
NR3D0 U2038 ( .A1(n3898), .A2(n4043), .A3(n1322), .ZN(n2836)
   );
INR4D0 U2039 ( .A1(n2836), .B1(n1288), .B2(n3782), .B3(n2825)
   , .ZN(n1291) );
NR2D0 U2040 ( .A1(n3783), .A2(n2862), .ZN(n1289) );
IND2D0 U2041 ( .A1(n2690), .B1(n1289), .ZN(n2906) );
NR4D0 U2042 ( .A1(n4439), .A2(n3716), .A3(n3892), .A4(n2906),
    .ZN(n1290) );
OAI22D0 U2043 ( .A1(n1291), .A2(n4010), .B1(n1290), .B2(n4045
   ), .ZN(n1292)
      );
AOI211D0 U2044 ( .A1(n3933), .A2(n1294), .B(n1293), .C(n1292)
   , .ZN(n1295) );
OAI22D0 U2045 ( .A1(n1296), .A2(n4148), .B1(n1295), .B2(n4325
   ), .ZN(n1297)
      );
AOI211D0 U2046 ( .A1(n4521), .A2(n1299), .B(n1298), .C(n1297)
   , .ZN(n1331) );
ND3D0 U2047 ( .A1(n2616), .A2(n1300), .A3(n3745), .ZN(n2931)
   );
NR2D0 U2048 ( .A1(n2706), .A2(n3919), .ZN(n2939) );
NR2D0 U2049 ( .A1(n4044), .A2(n3919), .ZN(n3653) );
CKND2D0 U2050 ( .A1(n2616), .A2(n3987), .ZN(n3627) );
CKND2D0 U2051 ( .A1(n3631), .A2(n1301), .ZN(n1313) );
INR4D0 U2052 ( .A1(n3653), .B1(n2894), .B2(n3627), .B3(n1313)
   , .ZN(n2795) );
OAI22D0 U2053 ( .A1(n2939), .A2(n4038), .B1(n2795), .B2(n4010
   ), .ZN(n1302)
      );
```

```
AOI221D0 U2054 ( .A1(n1303), .A2(n3844), .B1(n2931), .B2(
   n3844), .C(n1302),
      .ZN(n1305) );
AOI211D0 U2055 ( .A1(n1820), .A2(n3737), .B(n3660), .C(n3810)
   , .ZN(n1304) );
AOI32D0 U2056 ( .A1(n3856), .A2(n1305), .A3(n1304), .B1(n4059
   ), .B2(n1305),
      .ZN(n1312) );
INR4D0 U2057 ( .A1(n2875), .B1(n2796), .B2(n2882), .B3(n4043)
   , .ZN(n1310) );
NR2D0 U2058 ( .A1(n3716), .A2(n2820), .ZN(n4016) );
CKND2D0 U2059 ( .A1(n4016), .A2(n3621), .ZN(n2791) );
INVD0 U2060 ( .I(n2791), .ZN(n3601) );
CKND2D0 U2061 ( .A1(n2786), .A2(n3601), .ZN(n2943) );
INVD0 U2062 ( .I(n2943), .ZN(n2663) );
CKND2D0 U2063 ( .A1(n2860), .A2(n1306), .ZN(n2625) );
NR2D0 U2064 ( .A1(n3614), .A2(n2625), .ZN(n2869) );
INVD0 U2065 ( .I(n4439), .ZN(n4419) );
ND4D0 U2066 ( .A1(n2663), .A2(n2869), .A3(n4419), .A4(n3927),
    .ZN(n1307) );
CKND2D0 U2067 ( .A1(n3789), .A2(n3657), .ZN(n3646) );
INVD0 U2068 ( .I(n3646), .ZN(n3865) );
INVD0 U2069 ( .I(n3609), .ZN(n3918) );
NR2D0 U2070 ( .A1(n3918), .A2(n3614), .ZN(n3808) );
ND4D0 U2071 ( .A1(n3865), .A2(n3808), .A3(n2754), .A4(n3621),
    .ZN(n2673) );
AOI22D0 U2072 ( .A1(n3769), .A2(n1307), .B1(n3844), .B2(n2673
   ), .ZN(n1309)
      );
NR2D0 U2073 ( .A1(n1651), .A2(n2788), .ZN(n2664) );
CKND2D0 U2074 ( .A1(n3813), .A2(n2753), .ZN(n2900) );
OAI31D0 U2075 ( .A1(n2664), .A2(n3660), .A3(n2900), .B(n3933)
   , .ZN(n1308) );
OAI211D0 U2076 ( .A1(n1310), .A2(n4059), .B(n1309), .C(n1308)
   , .ZN(n1311) );
AOI22D0 U2077 ( .A1(n3311), .A2(n1312), .B1(n3386), .B2(n1311
   ), .ZN(n1330)
      );
NR2D0 U2078 ( .A1(n3891), .A2(n3918), .ZN(n2720) );
INVD0 U2079 ( .I(n3813), .ZN(n3904) );
NR2D0 U2080 ( .A1(n3904), .A2(n1313), .ZN(n3697) );
AOI31D0 U2081 ( .A1(n2720), .A2(n2859), .A3(n3697), .B(n4038)
   , .ZN(n1318) );
INVD0 U2082 ( .I(n3604), .ZN(n2056) );
CKND2D0 U2083 ( .A1(n2056), .A2(n4050), .ZN(n3943) );
OA31D0 U2084 ( .A1(n3892), .A2(n1314), .A3(n3943), .B(n4000),
    .Z(n1317) );
```

```
NR3D0 U2085 ( .A1(n3782), .A2(n2759), .A3(n2901), .ZN(n2752)
   );
NR2D0 U2086 ( .A1(n3870), .A2(n2913), .ZN(n4062) );
CKND2D0 U2087 ( .A1(n3789), .A2(n3677), .ZN(n3685) );
INR4D0 U2088 ( .A1(n4062), .B1(n3716), .B2(n3783), .B3(n3685)
   , .ZN(n1315) );
OAI22D0 U2089 ( .A1(n2752), .A2(n4010), .B1(n1315), .B2(n4045
   ), .ZN(n1316)
       );
OAI31D0 U2090 ( .A1(n1318), .A2(n1317), .A3(n1316), .B(n3852)
   , .ZN(n1329) );
NR2D0 U2091 ( .A1(n2796), .A2(n3786), .ZN(n2687) );
NR4D0 U2092 ( .A1(n2872), .A2(n3649), .A3(n2750), .A4(n2703),
    .ZN(n1319) );
AOI31D0 U2093 ( .A1(n2687), .A2(n1319), .A3(n3917), .B(n4045)
   , .ZN(n1327) );
INVD0 U2094 ( .I(n2860), .ZN(n3639) );
NR3D0 U2095 ( .A1(n3639), .A2(n1320), .A3(n3627), .ZN(n2910)
   );
AOI31D0 U2096 ( .A1(n3807), .A2(n2910), .A3(n3745), .B(n4059)
   , .ZN(n1326) );
NR2D0 U2097 ( .A1(n3608), .A2(n3854), .ZN(n3600) );
INR4D0 U2098 ( .A1(n3600), .B1(n2914), .B2(n1321), .B3(n3710)
   , .ZN(n1324) );
CKND2D0 U2099 ( .A1(n2711), .A2(n2748), .ZN(n3797) );
NR4D0 U2100 ( .A1(n3614), .A2(n2950), .A3(n1322), .A4(n3797),
    .ZN(n1323) );
OAI22D0 U2101 ( .A1(n1324), .A2(n4038), .B1(n1323), .B2(n4010
   ), .ZN(n1325)
       );
OAI31D0 U2102 ( .A1(n1327), .A2(n1326), .A3(n1325), .B(n3664)
   , .ZN(n1328) );
ND4D0 U2103 ( .A1(n1331), .A2(n1330), .A3(n1329), .A4(n1328),
    .ZN(n1332) );
NR4D0 U2104 ( .A1(n1335), .A2(n1334), .A3(n1333), .A4(n1332),
    .ZN(n1500) );
NR2D0 U2105 ( .A1(n4225), .A2(n4143), .ZN(n3200) );
ND4D0 U2106 ( .A1(n3246), .A2(n3200), .A3(n1336), .A4(n2968),
    .ZN(n1339) );
ND4D0 U2107 ( .A1(n1337), .A2(n4109), .A3(n4345), .A4(n1336),
    .ZN(n1338) );
AOI22D0 U2108 ( .A1(n4480), .A2(n1339), .B1(n4291), .B2(n1338
   ), .ZN(n1342)
       );
CKAN2D0 U2109 ( .A1(n4316), .A2(n4368), .Z(n3036) );
CKND2D0 U2110 ( .A1(n3246), .A2(n3036), .ZN(n3079) );
NR2D0 U2111 ( .A1(n3319), .A2(n3162), .ZN(n4234) );
INVD0 U2112 ( .I(n4438), .ZN(n3397) );
```

```
  CKND2D0 U2113 ( .A1(n4234), .A2(n3397), .ZN(n3303) );
  OAI21D0 U2114 ( .A1(n3079), .A2(n3303), .B(n4342), .ZN(n1341)
      );
  CKND2D0 U2115 ( .A1(n4133), .A2(n3323), .ZN(n3011) );
  NR2D0 U2116 ( .A1(n4438), .A2(n4426), .ZN(n4520) );
  INVD0 U2117 ( .I(n4520), .ZN(n4219) );
  INVD0 U2118 ( .I(n3457), .ZN(n3396) );
  CKND2D0 U2119 ( .A1(n3396), .A2(n3505), .ZN(n3473) );
  OAI31D0 U2120 ( .A1(n3011), .A2(n4219), .A3(n3473), .B(n4452)
     , .ZN(n1340) );
  AOI31D0 U2121 ( .A1(n1342), .A2(n1341), .A3(n1340), .B(n3379)
     , .ZN(n1498) );
  NR2D0 U2122 ( .A1(n4156), .A2(n1407), .ZN(n3068) );
  CKND2D0 U2123 ( .A1(n4419), .A2(n4212), .ZN(n3132) );
  INVD0 U2124 ( .I(n3132), .ZN(n3157) );
  ND4D0 U2125 ( .A1(n3068), .A2(n3157), .A3(n2968), .A4(n4465),
      .ZN(n1343) );
  CKND2D0 U2126 ( .A1(n4157), .A2(n4320), .ZN(n3126) );
  INVD0 U2127 ( .I(n3126), .ZN(n3199) );
  CKND2D0 U2128 ( .A1(n4445), .A2(n2508), .ZN(n4390) );
  CKND2D0 U2129 ( .A1(n3199), .A2(n4390), .ZN(n3078) );
  AOI22D0 U2130 ( .A1(n4480), .A2(n1343), .B1(n4289), .B2(n3078
     ), .ZN(n1347)
        );
  NR2D0 U2131 ( .A1(n1436), .A2(n3049), .ZN(n3056) );
  NR2D0 U2132 ( .A1(n3056), .A2(n2788), .ZN(n3234) );
  CKND2D0 U2133 ( .A1(n3324), .A2(n3535), .ZN(n3307) );
  NR2D0 U2134 ( .A1(n1407), .A2(n1457), .ZN(n1349) );
  INVD0 U2135 ( .I(n1349), .ZN(n3129) );
  OAI31D0 U2136 ( .A1(n3234), .A2(n3307), .A3(n3129), .B(n4291)
     , .ZN(n1346) );
  CKND2D0 U2137 ( .A1(n4506), .A2(n4320), .ZN(n4513) );
  CKND2D0 U2138 ( .A1(n4345), .A2(n4367), .ZN(n4393) );
  ND4D0 U2139 ( .A1(n3094), .A2(n4333), .A3(n3535), .A4(n3505),
      .ZN(n2965) );
  OAI31D0 U2140 ( .A1(n4513), .A2(n4393), .A3(n2965), .B(n4452)
     , .ZN(n1345) );
  AOI31D0 U2141 ( .A1(n1347), .A2(n1346), .A3(n1345), .B(n4410)
     , .ZN(n1497) );
  INVD0 U2142 ( .I(n3368), .ZN(n3024) );
  CKND2D0 U2143 ( .A1(n3024), .A2(n4133), .ZN(n3147) );
  INVD0 U2144 ( .I(n3109), .ZN(n4136) );
  CKND2D0 U2145 ( .A1(n4136), .A2(n4368), .ZN(n2967) );
  INVD0 U2146 ( .I(n2967), .ZN(n3373) );
  ND4D0 U2147 ( .A1(n3373), .A2(n3320), .A3(n3505), .A4(n4507),
      .ZN(n1355) );
  NR2D0 U2148 ( .A1(n4232), .A2(n4393), .ZN(n1453) );
  CKND2D0 U2149 ( .A1(n4334), .A2(n4418), .ZN(n3248) );
```

```
NR2D0 U2150 ( .A1(n3574), .A2(n4422), .ZN(n4141) );
CKND2D0 U2151 ( .A1(n1518), .A2(n4098), .ZN(n4233) );
IND4D0 U2152 ( .A1(n3248), .B1(n4141), .B2(n4233), .B3(n1466)
   , .ZN(n1351) );
CKND2D0 U2153 ( .A1(n2978), .A2(n4261), .ZN(n3140) );
INVD0 U2154 ( .I(n3140), .ZN(n4494) );
NR2D0 U2155 ( .A1(n4494), .A2(n3447), .ZN(n1348) );
ND4D0 U2156 ( .A1(n1349), .A2(n1348), .A3(n4294), .A4(n4345),
    .ZN(n1350) );
AOI22D0 U2157 ( .A1(n4480), .A2(n1351), .B1(n4291), .B2(n1350
   ), .ZN(n1353)
      );
CKND2D0 U2158 ( .A1(n1937), .A2(n2577), .ZN(n4343) );
INVD0 U2159 ( .I(n4343), .ZN(n4203) );
NR4D0 U2160 ( .A1(n4090), .A2(n4236), .A3(n4203), .A4(n3447),
    .ZN(n1352) );
AOI32D0 U2161 ( .A1(n1453), .A2(n1353), .A3(n1352), .B1(n4517
   ), .B2(n1353),
      .ZN(n1354) );
AOI221D0 U2162 ( .A1(n3147), .A2(n4452), .B1(n1355), .B2(
   n4452), .C(n1354),
      .ZN(n1361) );
CKND2D0 U2163 ( .A1(n3535), .A2(n4137), .ZN(n3080) );
NR2D0 U2164 ( .A1(n3574), .A2(n4171), .ZN(n3142) );
CKND2D0 U2165 ( .A1(n2654), .A2(n4082), .ZN(n4094) );
IND4D0 U2166 ( .A1(n3080), .B1(n3036), .B2(n3142), .B3(n4094)
   , .ZN(n1359) );
CKND2D0 U2167 ( .A1(n4389), .A2(n3320), .ZN(n3308) );
NR2D0 U2168 ( .A1(n3509), .A2(n3308), .ZN(n1465) );
CKND2D0 U2169 ( .A1(n4212), .A2(n4316), .ZN(n3240) );
NR2D0 U2170 ( .A1(n4170), .A2(n3240), .ZN(n1477) );
AOI31D0 U2171 ( .A1(n1465), .A2(n1477), .A3(n1478), .B(n4474)
   , .ZN(n1358) );
NR3D0 U2172 ( .A1(n4250), .A2(n4473), .A3(n4513), .ZN(n3016)
   );
NR2D0 U2173 ( .A1(n3109), .A2(n4306), .ZN(n1399) );
INVD0 U2174 ( .I(n1399), .ZN(n1449) );
CKND2D0 U2175 ( .A1(n3140), .A2(n4137), .ZN(n1486) );
NR4D0 U2176 ( .A1(n4310), .A2(n4246), .A3(n1449), .A4(n1486),
    .ZN(n1356) );
OAI22D0 U2177 ( .A1(n3016), .A2(n4428), .B1(n1356), .B2(n3243
   ), .ZN(n1357)
      );
AOI211D0 U2178 ( .A1(n4342), .A2(n1359), .B(n1358), .C(n1357)
   , .ZN(n1360) );
OAI22D0 U2179 ( .A1(n1361), .A2(n4355), .B1(n1360), .B2(n4148
   ), .ZN(n1496)
      );
```

```
  NR2D0 U2180 ( .A1(n4250), .A2(n4161), .ZN(n1372) );
  CKND2D0 U2181 ( .A1(n4362), .A2(n4316), .ZN(n3195) );
  INR4D0 U2182 ( .A1(n1372), .B1(n4225), .B2(n3203), .B3(n3195)
     , .ZN(n1371) );
  CKND2D0 U2183 ( .A1(n1362), .A2(n4203), .ZN(n3192) );
  CKND2D0 U2184 ( .A1(n3192), .A2(n3094), .ZN(n1485) );
  IND4D0 U2185 ( .A1(n1485), .B1(n3199), .B2(n4345), .B3(n4094)
     , .ZN(n1366) );
  CKND2D0 U2186 ( .A1(n4389), .A2(n3535), .ZN(n3196) );
  INVD0 U2187 ( .I(n3196), .ZN(n3507) );
  CKND2D0 U2188 ( .A1(n4334), .A2(n4316), .ZN(n3214) );
  NR3D0 U2189 ( .A1(n4124), .A2(n3162), .A3(n3214), .ZN(n4295)
     );
  NR2D0 U2190 ( .A1(n3349), .A2(n1722), .ZN(n4115) );
  NR4D0 U2191 ( .A1(n1363), .A2(n4115), .A3(n3368), .A4(n1485),
      .ZN(n1364) );
  AOI31D0 U2192 ( .A1(n3507), .A2(n4295), .A3(n1364), .B(n4517)
     , .ZN(n1365) );
  AOI221D0 U2193 ( .A1(n3162), .A2(n4452), .B1(n1366), .B2(
     n4452), .C(n1365),
        .ZN(n1370) );
  NR2D0 U2194 ( .A1(n1368), .A2(n1367), .ZN(n4195) );
  CKND2D0 U2195 ( .A1(n4133), .A2(n4506), .ZN(n3197) );
  OAI31D0 U2196 ( .A1(n3162), .A2(n4195), .A3(n3197), .B(n4291)
     , .ZN(n1369) );
  OAI211D0 U2197 ( .A1(n1371), .A2(n3243), .B(n1370), .C(n1369)
     , .ZN(n1379) );
  CKND2D0 U2198 ( .A1(n4367), .A2(n4320), .ZN(n3158) );
  CKND2D0 U2199 ( .A1(n3157), .A2(n1720), .ZN(n3205) );
  INR4D0 U2200 ( .A1(n3142), .B1(n2967), .B2(n3158), .B3(n3205)
     , .ZN(n1377) );
  INVD0 U2201 ( .I(n4141), .ZN(n3085) );
  CKND2D0 U2202 ( .A1(n3397), .A2(n4212), .ZN(n3148) );
  INVD0 U2203 ( .I(n3148), .ZN(n1393) );
  ND4D0 U2204 ( .A1(n3235), .A2(n1393), .A3(n1372), .A4(n4094),
      .ZN(n1374) );
  NR2D0 U2205 ( .A1(n4181), .A2(n4307), .ZN(n3149) );
  INVD0 U2206 ( .I(n3149), .ZN(n3161) );
  NR2D0 U2207 ( .A1(n4422), .A2(n3161), .ZN(n3184) );
  AOI31D0 U2208 ( .A1(n3392), .A2(n3184), .A3(n3140), .B(n4517)
     , .ZN(n1373) );
  AOI221D0 U2209 ( .A1(n3085), .A2(n4291), .B1(n1374), .B2(
     n4291), .C(n1373),
        .ZN(n1376) );
  CKND2D0 U2210 ( .A1(n4333), .A2(n3505), .ZN(n3110) );
  CKND2D0 U2211 ( .A1(n4133), .A2(n4094), .ZN(n2966) );
  OAI31D0 U2212 ( .A1(n3109), .A2(n3110), .A3(n2966), .B(n4452)
     , .ZN(n1375) );
```

```
OAI211D0 U2213 ( .A1(n1377), .A2(n3243), .B(n1376), .C(n1375)
   , .ZN(n1378) );
AOI22D0 U2214 ( .A1(n4492), .A2(n1379), .B1(n4521), .B2(n1378
   ), .ZN(n1494)
      );
NR2D0 U2215 ( .A1(n4311), .A2(n4250), .ZN(n3018) );
NR2D0 U2216 ( .A1(n4310), .A2(n4076), .ZN(n4286) );
OAI211D0 U2217 ( .A1(n3897), .A2(n2596), .B(n4286), .C(n4320)
   , .ZN(n1384) );
INVD0 U2218 ( .I(n2982), .ZN(n4243) );
NR4D0 U2219 ( .A1(n3174), .A2(n1380), .A3(n3509), .A4(n3126),
    .ZN(n1382) );
NR2D0 U2220 ( .A1(n4438), .A2(n4161), .ZN(n1398) );
INVD0 U2221 ( .I(n1398), .ZN(n3086) );
NR4D0 U2222 ( .A1(n4374), .A2(n2966), .A3(n3086), .A4(n3158),
    .ZN(n1381) );
OAI22D0 U2223 ( .A1(n1382), .A2(n4428), .B1(n1381), .B2(n4474
   ), .ZN(n1383)
      );
AOI211D0 U2224 ( .A1(n4342), .A2(n1384), .B(n4243), .C(n1383)
   , .ZN(n1387) );
NR2D0 U2225 ( .A1(n1385), .A2(n2618), .ZN(n4119) );
INVD0 U2226 ( .I(n4161), .ZN(n4369) );
CKND2D0 U2227 ( .A1(n3505), .A2(n4369), .ZN(n1391) );
NR4D0 U2228 ( .A1(n4426), .A2(n4119), .A3(n4306), .A4(n1391),
    .ZN(n1440) );
INVD0 U2229 ( .I(n1430), .ZN(n1411) );
OAI211D0 U2230 ( .A1(n1414), .A2(n2618), .B(n1440), .C(n1411)
   , .ZN(n1444) );
NR3D0 U2231 ( .A1(n1444), .A2(n4373), .A3(n3079), .ZN(n1386)
   );
AOI32D0 U2232 ( .A1(n3018), .A2(n1387), .A3(n1386), .B1(n3243
   ), .B2(n1387),
      .ZN(n1476) );
NR2D0 U2233 ( .A1(n4311), .A2(n4076), .ZN(n3095) );
CKND2D0 U2234 ( .A1(n1536), .A2(n4098), .ZN(n3295) );
ND4D0 U2235 ( .A1(n1398), .A2(n3095), .A3(n3295), .A4(n3192),
    .ZN(n1390) );
NR2D0 U2236 ( .A1(n3085), .A2(n1449), .ZN(n3251) );
AOI31D0 U2237 ( .A1(n3094), .A2(n3251), .A3(n4506), .B(n3243)
   , .ZN(n1389) );
NR2D0 U2238 ( .A1(n4250), .A2(n4171), .ZN(n4459) );
CKND2D0 U2239 ( .A1(n4136), .A2(n4212), .ZN(n3237) );
INVD0 U2240 ( .I(n3237), .ZN(n3220) );
AOI31D0 U2241 ( .A1(n4459), .A2(n3097), .A3(n3220), .B(n4517)
   , .ZN(n1388) );
AOI211D0 U2242 ( .A1(n4291), .A2(n1390), .B(n1389), .C(n1388)
   , .ZN(n1410) );
```

```
NR4D0 U2243 ( .A1(n4323), .A2(n4307), .A3(n1391), .A4(n3196),
    .ZN(n1397) );
IND4D0 U2244 ( .A1(n3078), .B1(n1392), .B2(n1393), .B3(n3149)
   , .ZN(n1394) );
ND4D0 U2245 ( .A1(n3246), .A2(n4488), .A3(n1393), .A4(n4317),
    .ZN(n3092) );
AOI22D0 U2246 ( .A1(n4291), .A2(n1394), .B1(n4342), .B2(n3092
   ), .ZN(n1396)
       );
OAI31D0 U2247 ( .A1(n4494), .A2(n3085), .A3(n1485), .B(n4480)
   , .ZN(n1395) );
OAI211D0 U2248 ( .A1(n1397), .A2(n54), .B(n1396), .C(n1395),
   .ZN(n1406) );
NR2D0 U2249 ( .A1(n4225), .A2(n4076), .ZN(n4463) );
ND4D0 U2250 ( .A1(n4463), .A2(n3036), .A3(n4157), .A4(n4506),
    .ZN(n1402) );
AOI31D0 U2251 ( .A1(n3392), .A2(n1398), .A3(n3323), .B(n3243)
   , .ZN(n1401) );
NR2D0 U2252 ( .A1(n4225), .A2(n4310), .ZN(n3141) );
AOI31D0 U2253 ( .A1(n1399), .A2(n1454), .A3(n3141), .B(n54),
   .ZN(n1400) );
AOI211D0 U2254 ( .A1(n4289), .A2(n1402), .B(n1401), .C(n1400)
   , .ZN(n1404) );
CKND2D0 U2255 ( .A1(n3097), .A2(n4333), .ZN(n3131) );
CKND2D0 U2256 ( .A1(n4345), .A2(n3535), .ZN(n3128) );
NR4D0 U2257 ( .A1(n4225), .A2(n3086), .A3(n3131), .A4(n3128),
    .ZN(n1403) );
AOI32D0 U2258 ( .A1(n3373), .A2(n1404), .A3(n1403), .B1(n4476
   ), .B2(n1404),
      .ZN(n1405) );
AOI22D0 U2259 ( .A1(n3850), .A2(n1406), .B1(n3707), .B2(n1405
   ), .ZN(n1409)
       );
CKND2D0 U2260 ( .A1(n4463), .A2(n4367), .ZN(n2986) );
OAI31D0 U2261 ( .A1(n3237), .A2(n1407), .A3(n2986), .B(n4452)
   , .ZN(n1408) );
AOI32D0 U2262 ( .A1(n1410), .A2(n1409), .A3(n1408), .B1(n4190
   ), .B2(n1409),
      .ZN(n1475) );
INVD0 U2263 ( .I(n4117), .ZN(n3440) );
CKND2D0 U2264 ( .A1(n3732), .A2(n1418), .ZN(n3081) );
OAI211D0 U2265 ( .A1(n3440), .A2(n3441), .B(n1411), .C(n3081)
   , .ZN(n3008) );
NR2D0 U2266 ( .A1(n4223), .A2(n3449), .ZN(n4111) );
AOI21D0 U2267 ( .A1(n2788), .A2(n3897), .B(n1412), .ZN(n1423)
    );
CKND2D0 U2268 ( .A1(n4334), .A2(n4506), .ZN(n4376) );
```

```
NR4D0 U2269 ( .A1(n4093), .A2(n4111), .A3(n1423), .A4(n4376),
    .ZN(n1413) );
IND4D0 U2270 ( .A1(n3008), .B1(n1413), .B2(n4107), .B3(n4262)
   , .ZN(n1428) );
OAI211D0 U2271 ( .A1(n1414), .A2(n2618), .B(n4463), .C(n4265)
   , .ZN(n2994) );
IND4D0 U2272 ( .A1(n1429), .B1(n3320), .B2(n4316), .B3(n3053)
   , .ZN(n1415) );
INR4D0 U2273 ( .A1(n4262), .B1(n3234), .B2(n2994), .B3(n1415)
   , .ZN(n1417) );
CKND2D0 U2274 ( .A1(n3897), .A2(n2618), .ZN(n3048) );
CKND2D0 U2275 ( .A1(n1416), .A2(n3048), .ZN(n3038) );
AOI31D0 U2276 ( .A1(n1417), .A2(n3149), .A3(n3038), .B(n4428)
   , .ZN(n1427) );
INVD0 U2277 ( .I(n4111), .ZN(n3044) );
CKND2D0 U2278 ( .A1(n1434), .A2(n3044), .ZN(n4092) );
CKND2D0 U2279 ( .A1(n3199), .A2(n4362), .ZN(n3071) );
NR4D0 U2280 ( .A1(n3203), .A2(n3197), .A3(n3071), .A4(n3008),
    .ZN(n1419) );
INVD0 U2281 ( .I(n3332), .ZN(n1686) );
CKND2D0 U2282 ( .A1(n1686), .A2(n1418), .ZN(n3066) );
ND4D0 U2283 ( .A1(n4141), .A2(n1419), .A3(n3066), .A4(n4465),
    .ZN(n1420) );
AOI211D0 U2284 ( .A1(n3750), .A2(n4081), .B(n4092), .C(n1420)
   , .ZN(n1425) );
CKND2D0 U2285 ( .A1(n4367), .A2(n4343), .ZN(n3043) );
INVD0 U2286 ( .I(n3439), .ZN(n3114) );
AOI211D0 U2287 ( .A1(n3732), .A2(n3114), .B(n4088), .C(n4232)
   , .ZN(n1421) );
CKND2D0 U2288 ( .A1(n3527), .A2(n3442), .ZN(n4125) );
ND4D0 U2289 ( .A1(n1421), .A2(n4136), .A3(n4121), .A4(n4125),
    .ZN(n1422) );
NR4D0 U2290 ( .A1(n1423), .A2(n2994), .A3(n3043), .A4(n1422),
    .ZN(n1424) );
OAI22D0 U2291 ( .A1(n1425), .A2(n4517), .B1(n1424), .B2(n4474
   ), .ZN(n1426)
       );
AOI211D0 U2292 ( .A1(n4480), .A2(n1428), .B(n1427), .C(n1426)
   , .ZN(n1473) );
CKND2D0 U2293 ( .A1(counter[2]), .A2(n1429), .ZN(n4120) );
INVD0 U2294 ( .I(n3459), .ZN(n2998) );
ND3D0 U2295 ( .A1(n4209), .A2(n4120), .A3(n2998), .ZN(n3058)
   );
NR2D0 U2296 ( .A1(n3444), .A2(n1430), .ZN(n4221) );
CKND2D0 U2297 ( .A1(n1431), .A2(n4221), .ZN(n3042) );
INVD0 U2298 ( .I(n3042), .ZN(n4252) );
CKND2D0 U2299 ( .A1(n4317), .A2(n3081), .ZN(n1437) );
```

```
AOI211D0 U2300 ( .A1(n1518), .A2(counter[11]), .B(n3162), .C(
   n1437), .ZN(
      n1432) );
CKND2D0 U2301 ( .A1(n4252), .A2(n1432), .ZN(n1433) );
NR4D0 U2302 ( .A1(n3524), .A2(n3240), .A3(n3058), .A4(n1433),
    .ZN(n1452) );
ND4D0 U2303 ( .A1(n4108), .A2(n4506), .A3(n1434), .A4(n3081),
    .ZN(n3052) );
OAI211D0 U2304 ( .A1(n3735), .A2(n2996), .B(n4120), .C(n4317)
   , .ZN(n3041) );
NR4D0 U2305 ( .A1(n4374), .A2(n3459), .A3(n3052), .A4(n3041),
    .ZN(n1435) );
ND4D0 U2306 ( .A1(n3095), .A2(n1435), .A3(n4507), .A4(n4241),
    .ZN(n1443) );
AOI211D0 U2307 ( .A1(n1436), .A2(n3048), .B(n3064), .C(n4403)
   , .ZN(n1441) );
ND4D0 U2308 ( .A1(n3018), .A2(n3320), .A3(n4320), .A4(n3044),
    .ZN(n2985) );
AOI211D0 U2309 ( .A1(n1438), .A2(n4081), .B(n1437), .C(n2985)
   , .ZN(n1439) );
AOI31D0 U2310 ( .A1(n1441), .A2(n1440), .A3(n1439), .B(n4517)
   , .ZN(n1442) );
AOI221D0 U2311 ( .A1(n1444), .A2(n4291), .B1(n1443), .B2(
   n4291), .C(n1442),
      .ZN(n1451) );
AO22D0 U2312 ( .A1(n3048), .A2(n1446), .B1(n1445), .B2(n2636)
   , .Z(n2989) );
NR4D0 U2313 ( .A1(n3444), .A2(n4074), .A3(n3148), .A4(n2989),
    .ZN(n1447) );
ND4D0 U2314 ( .A1(n1447), .A2(n1478), .A3(n4194), .A4(n4125),
    .ZN(n1448) );
OAI31D0 U2315 ( .A1(n3041), .A2(n1449), .A3(n1448), .B(n4452)
   , .ZN(n1450) );
OAI211D0 U2316 ( .A1(n1452), .A2(n3243), .B(n1451), .C(n1450)
   , .ZN(n1463) );
NR2D0 U2317 ( .A1(n2967), .A2(n4172), .ZN(n2988) );
AN4D0 U2318 ( .A1(n1453), .A2(n3235), .A3(n1454), .A4(n2988),
    .Z(n1461) );
INVD0 U2319 ( .I(n3214), .ZN(n4324) );
NR2D0 U2320 ( .A1(n4090), .A2(n2967), .ZN(n3074) );
NR2D0 U2321 ( .A1(n3162), .A2(n3148), .ZN(n3160) );
ND4D0 U2322 ( .A1(n3246), .A2(n4324), .A3(n3074), .A4(n3160),
    .ZN(n1456) );
NR2D0 U2323 ( .A1(n3574), .A2(n3128), .ZN(n3241) );
ND4D0 U2324 ( .A1(n1454), .A2(n3097), .A3(n3241), .A4(n3249),
    .ZN(n1455) );
AOI22D0 U2325 ( .A1(n4480), .A2(n1456), .B1(n4342), .B2(n1455
   ), .ZN(n1460)
```

```
        );
INVD0 U2326 ( .I(n3011), .ZN(n4177) );
CKND2D0 U2327 ( .A1(n4177), .A2(n4335), .ZN(n3367) );
OR4D0 U2328 ( .A1(n3195), .A2(n1457), .A3(n4171), .A4(n4307),
    .Z(n1458) );
OAI31D0 U2329 ( .A1(n3367), .A2(n1486), .A3(n1458), .B(n4291)
   , .ZN(n1459) );
OAI211D0 U2330 ( .A1(n1461), .A2(n54), .B(n1460), .C(n1459),
   .ZN(n1462) );
AOI22D0 U2331 ( .A1(n3567), .A2(n1463), .B1(n3664), .B2(n1462
   ), .ZN(n1472)
        );
CKND2D0 U2332 ( .A1(n3241), .A2(n3140), .ZN(n3025) );
INR4D0 U2333 ( .A1(n1477), .B1(n4090), .B2(n4473), .B3(n3025)
   , .ZN(n1464) );
AOI31D0 U2334 ( .A1(n1464), .A2(n4320), .A3(n4317), .B(n3243)
   , .ZN(n1470) );
NR2D0 U2335 ( .A1(n4310), .A2(n4161), .ZN(n3178) );
NR2D0 U2336 ( .A1(n3368), .A2(n4183), .ZN(n3026) );
AOI31D0 U2337 ( .A1(n3178), .A2(n2988), .A3(n3026), .B(n4476)
   , .ZN(n1469) );
CKND2D0 U2338 ( .A1(counter[3]), .A2(n4144), .ZN(n3145) );
INVD0 U2339 ( .I(n3145), .ZN(n3391) );
NR2D0 U2340 ( .A1(n3391), .A2(n3132), .ZN(n3017) );
INVD0 U2341 ( .I(n1465), .ZN(n3225) );
CKND2D0 U2342 ( .A1(n3074), .A2(n1478), .ZN(n2973) );
OAI31D0 U2343 ( .A1(n3225), .A2(n4393), .A3(n2973), .B(n4342)
   , .ZN(n1467) );
AOI32D0 U2344 ( .A1(n3017), .A2(n1467), .A3(n1466), .B1(n4474
   ), .B2(n1467),
      .ZN(n1468) );
OAI31D0 U2345 ( .A1(n1470), .A2(n1469), .A3(n1468), .B(n3709)
   , .ZN(n1471) );
OAI211D0 U2346 ( .A1(n1473), .A2(n4277), .B(n1472), .C(n1471)
   , .ZN(n1474) );
AOI211D0 U2347 ( .A1(n4305), .A2(n1476), .B(n1475), .C(n1474)
   , .ZN(n1493) );
NR2D0 U2348 ( .A1(n4494), .A2(n3080), .ZN(n3182) );
AOI31D0 U2349 ( .A1(n3141), .A2(n3199), .A3(n3182), .B(n3243)
   , .ZN(n1484) );
NR2D0 U2350 ( .A1(n4076), .A2(n4422), .ZN(n3218) );
AOI21D0 U2351 ( .A1(n1477), .A2(n3218), .B(n4517), .ZN(n1483)
    );
INR4D0 U2352 ( .A1(n3068), .B1(n4225), .B2(n4422), .B3(n3086)
   , .ZN(n1481) );
CKND2D0 U2353 ( .A1(n1478), .A2(n4194), .ZN(n1479) );
NR4D0 U2354 ( .A1(n3109), .A2(n3248), .A3(n2986), .A4(n1479),
    .ZN(n1480) );
```

```
OAI22D0 U2355 ( .A1(n1481), .A2(n54), .B1(n1480), .B2(n4428),
    .ZN(n1482) );
OAI31D0 U2356 ( .A1(n1484), .A2(n1483), .A3(n1482), .B(n3311)
   , .ZN(n1492) );
CKND2D0 U2357 ( .A1(n4389), .A2(n4507), .ZN(n4359) );
OA31D0 U2358 ( .A1(n1486), .A2(n4359), .A3(n1485), .B(n4342),
    .Z(n1490) );
CKND2D0 U2359 ( .A1(n4362), .A2(n3192), .ZN(n3125) );
INVD0 U2360 ( .I(n3125), .ZN(n3198) );
CKND2D0 U2361 ( .A1(n2265), .A2(n3391), .ZN(n3111) );
INVD0 U2362 ( .I(n3111), .ZN(n3123) );
NR4D0 U2363 ( .A1(n4438), .A2(n3109), .A3(n3123), .A4(n3080),
    .ZN(n3156) );
AOI31D0 U2364 ( .A1(n3198), .A2(n3142), .A3(n3156), .B(n4476)
   , .ZN(n1489) );
NR3D0 U2365 ( .A1(n3147), .A2(n3509), .A3(n4183), .ZN(n3172)
   );
CKND2D0 U2366 ( .A1(n4126), .A2(n4334), .ZN(n3239) );
NR4D0 U2367 ( .A1(n3574), .A2(n2966), .A3(n3237), .A4(n3239),
    .ZN(n1487) );
OAI22D0 U2368 ( .A1(n3172), .A2(n3243), .B1(n1487), .B2(n4474
   ), .ZN(n1488)
       );
OAI31D0 U2369 ( .A1(n1490), .A2(n1489), .A3(n1488), .B(n3386)
   , .ZN(n1491) );
ND4D0 U2370 ( .A1(n1494), .A2(n1493), .A3(n1492), .A4(n1491),
    .ZN(n1495) );
NR4D0 U2371 ( .A1(n1498), .A2(n1497), .A3(n1496), .A4(n1495),
    .ZN(n1499) );
OAI222D0 U2372 ( .A1(\q_out[4]_BAR ), .A2(n4545), .B1(n4544),
    .B2(n1500),
      .C1(n4542), .C2(n1499), .ZN(n4816) );
INVD0 U2373 ( .I(n1811), .ZN(n1697) );
NR2D0 U2374 ( .A1(n1697), .A2(n3822), .ZN(n1803) );
NR2D0 U2375 ( .A1(n4444), .A2(n1501), .ZN(n1560) );
INVD0 U2376 ( .I(n1560), .ZN(n1558) );
INVD0 U2377 ( .I(n4233), .ZN(n4472) );
CKND2D0 U2378 ( .A1(counter[4]), .A2(n4472), .ZN(n1704) );
CKND2D0 U2379 ( .A1(n1558), .A2(n1704), .ZN(n1616) );
CKND2D0 U2380 ( .A1(counter[2]), .A2(n2173), .ZN(n1502) );
NR2D0 U2381 ( .A1(n3477), .A2(n1502), .ZN(n1600) );
NR2D0 U2382 ( .A1(counter[6]), .A2(n3617), .ZN(n1688) );
OR2D0 U2383 ( .A1(n1600), .A2(n1688), .Z(n1723) );
NR2D0 U2384 ( .A1(n1697), .A2(n4025), .ZN(n1718) );
INVD0 U2385 ( .I(n1718), .ZN(n1631) );
OAI21D0 U2386 ( .A1(n3595), .A2(n2120), .B(n1631), .ZN(n1835)
    );
INVD0 U2387 ( .I(n1835), .ZN(n1581) );
```

```
  CKND2D0 U2388 ( .A1(n2664), .A2(n1513), .ZN(n1665) );
  IND3D0 U2389 ( .A1(n1723), .B1(n1581), .B2(n1665), .ZN(n1615)
      );
  NR2D0 U2390 ( .A1(n1543), .A2(n2639), .ZN(n1759) );
  NR2D0 U2391 ( .A1(n1524), .A2(n1505), .ZN(n1538) );
  NR2D0 U2392 ( .A1(n1759), .A2(n1538), .ZN(n1733) );
  INVD0 U2393 ( .I(n1733), .ZN(n1591) );
  NR4D0 U2394 ( .A1(n1803), .A2(n1616), .A3(n1615), .A4(n1591),
      .ZN(n1510) );
  INVD0 U2395 ( .I(n1577), .ZN(n1926) );
  INVD0 U2396 ( .I(n1616), .ZN(n1823) );
  NR2D0 U2397 ( .A1(n1697), .A2(n3477), .ZN(n1749) );
  NR2D0 U2398 ( .A1(n1799), .A2(n1505), .ZN(n1597) );
  NR2D0 U2399 ( .A1(n1749), .A2(n1597), .ZN(n1578) );
  NR2D0 U2400 ( .A1(n1556), .A2(n1682), .ZN(n1586) );
  INVD0 U2401 ( .I(n1586), .ZN(n1614) );
  NR2D0 U2402 ( .A1(n4444), .A2(n1505), .ZN(n1656) );
  INVD0 U2403 ( .I(n1656), .ZN(n1762) );
  CKND2D0 U2404 ( .A1(n1614), .A2(n1762), .ZN(n1559) );
  NR2D0 U2405 ( .A1(n2062), .A2(n4025), .ZN(n1657) );
  CKND2D0 U2406 ( .A1(counter[2]), .A2(n1657), .ZN(n1611) );
  OAI21D0 U2407 ( .A1(n1722), .A2(n2120), .B(n1611), .ZN(n1662)
      );
  NR3D0 U2408 ( .A1(n1559), .A2(n1835), .A3(n1662), .ZN(n1604)
     );
  NR3D0 U2409 ( .A1(n3757), .A2(n4351), .A3(n3357), .ZN(n1797)
     );
  INVD0 U2410 ( .I(n1797), .ZN(n1763) );
  ND4D0 U2411 ( .A1(n1823), .A2(n1578), .A3(n1604), .A4(n1763),
      .ZN(n1504) );
  CKND2D0 U2412 ( .A1(n1859), .A2(counter[5]), .ZN(n1792) );
  INVD0 U2413 ( .I(n1792), .ZN(n1936) );
  NR2D0 U2414 ( .A1(n2265), .A2(n3357), .ZN(n3456) );
  CKND2D0 U2415 ( .A1(counter[4]), .A2(n3456), .ZN(n1756) );
  INVD0 U2416 ( .I(n1756), .ZN(n1612) );
  NR2D0 U2417 ( .A1(n1612), .A2(n1749), .ZN(n1639) );
  INR2D0 U2418 ( .A1(n1639), .B1(n1560), .ZN(n1784) );
  NR2D0 U2419 ( .A1(counter[4]), .A2(n4504), .ZN(n1691) );
  NR2D0 U2420 ( .A1(n1691), .A2(n1759), .ZN(n1832) );
  NR2D0 U2421 ( .A1(n3757), .A2(n3295), .ZN(n1766) );
  NR2D0 U2422 ( .A1(n1766), .A2(n1597), .ZN(n1619) );
  ND4D0 U2423 ( .A1(n1784), .A2(n1832), .A3(n1619), .A4(n1614),
      .ZN(n1503) );
  AOI22D0 U2424 ( .A1(n1931), .A2(n1504), .B1(n1936), .B2(n1503
     ), .ZN(n1509)
        );
  INVD0 U2425 ( .I(n1559), .ZN(n1507) );
  CKND2D0 U2426 ( .A1(n1526), .A2(n1667), .ZN(n1764) );
```

```
INVD0 U2427 ( .I(n1764), .ZN(n1687) );
NR2D0 U2428 ( .A1(n1524), .A2(n2639), .ZN(n1802) );
NR2D0 U2429 ( .A1(n1687), .A2(n1802), .ZN(n1602) );
INVD0 U2430 ( .I(n1602), .ZN(n1642) );
NR2D0 U2431 ( .A1(n1557), .A2(n4018), .ZN(n4309) );
CKND2D0 U2432 ( .A1(n4309), .A2(n3757), .ZN(n1741) );
INVD0 U2433 ( .I(n1741), .ZN(n1636) );
NR2D0 U2434 ( .A1(n1506), .A2(n1505), .ZN(n1673) );
NR4D0 U2435 ( .A1(n1642), .A2(n1766), .A3(n1636), .A4(n1673),
    .ZN(n1595) );
NR2D0 U2436 ( .A1(n1688), .A2(n1612), .ZN(n1732) );
CKND2D0 U2437 ( .A1(n1859), .A2(n2274), .ZN(n1738) );
AO31D0 U2438 ( .A1(n1507), .A2(n1595), .A3(n1732), .B(n1738),
    .Z(n1508) );
OAI211D0 U2439 ( .A1(n1510), .A2(n1926), .B(n1509), .C(n1508)
   , .ZN(n1855) );
INVD0 U2440 ( .I(n2271), .ZN(n2354) );
NR2D0 U2441 ( .A1(n2544), .A2(n2354), .ZN(n2286) );
INVD0 U2442 ( .I(n1535), .ZN(n2622) );
OAI22D0 U2443 ( .A1(n2286), .A2(n1785), .B1(n4351), .B2(n2622
   ), .ZN(n1512)
      );
AO31D0 U2444 ( .A1(n1872), .A2(counter[6]), .A3(n1535), .B(
   n1766), .Z(n1525)
      );
INVD0 U2445 ( .I(n1657), .ZN(n1674) );
IND3D0 U2446 ( .A1(n1525), .B1(n1558), .B2(n1674), .ZN(n1511)
    );
AOI211D0 U2447 ( .A1(n1513), .A2(n1512), .B(n1691), .C(n1511)
   , .ZN(n1795) );
NR2D0 U2448 ( .A1(n1543), .A2(n1798), .ZN(n1517) );
CKND2D0 U2449 ( .A1(counter[4]), .A2(n4134), .ZN(n1708) );
NR2D0 U2450 ( .A1(n2062), .A2(n3897), .ZN(n1514) );
OAI21D0 U2451 ( .A1(n1515), .A2(n1514), .B(n1522), .ZN(n1540)
    );
OAI211D0 U2452 ( .A1(n1557), .A2(n2622), .B(n1708), .C(n1540)
   , .ZN(n1516) );
NR4D0 U2453 ( .A1(n1517), .A2(n1687), .A3(n1723), .A4(n1516),
    .ZN(n1782) );
NR2D0 U2454 ( .A1(n1800), .A2(n2618), .ZN(n1520) );
NR2D0 U2455 ( .A1(n1799), .A2(n2789), .ZN(n1547) );
NR3D0 U2456 ( .A1(n1797), .A2(n1802), .A3(n1759), .ZN(n1653)
   );
CKND2D0 U2457 ( .A1(n1518), .A2(n2799), .ZN(n1705) );
OAI211D0 U2458 ( .A1(n2120), .A2(n3955), .B(n1653), .C(n1705)
   , .ZN(n1519) );
NR4D0 U2459 ( .A1(n1520), .A2(n1803), .A3(n1547), .A4(n1519),
    .ZN(n1521) );
```

```
AOI31D0 U2460 ( .A1(n1795), .A2(n1782), .A3(n1521), .B(n1772)
   , .ZN(n1554) );
CKND2D0 U2461 ( .A1(n1536), .A2(n1820), .ZN(n1572) );
INVD0 U2462 ( .I(n1538), .ZN(n1638) );
CKND2D0 U2463 ( .A1(n1763), .A2(n1638), .ZN(n1683) );
CKND2D0 U2464 ( .A1(n2164), .A2(n1527), .ZN(n1726) );
INVD0 U2465 ( .I(n1726), .ZN(n1617) );
NR2D0 U2466 ( .A1(n1683), .A2(n1617), .ZN(n1822) );
OAI211D0 U2467 ( .A1(n1798), .A2(n1543), .B(n1572), .C(n1822)
   , .ZN(n1699) );
INVD0 U2468 ( .I(n1699), .ZN(n1534) );
CKND2D0 U2469 ( .A1(n1523), .A2(n1522), .ZN(n3075) );
AOI221D0 U2470 ( .A1(n1524), .A2(n3075), .B1(n3462), .B2(
   n3075), .C(
      counter[4]), .ZN(n1701) );
AOI211D0 U2471 ( .A1(n4257), .A2(n2799), .B(n1701), .C(n1525)
   , .ZN(n1804) );
CKND2D0 U2472 ( .A1(n1526), .A2(n4256), .ZN(n1786) );
INVD0 U2473 ( .I(n1786), .ZN(n1796) );
CKND2D0 U2474 ( .A1(n2173), .A2(n1527), .ZN(n1649) );
INVD0 U2475 ( .I(n1649), .ZN(n1747) );
INVD0 U2476 ( .I(n1832), .ZN(n1613) );
CKND2D0 U2477 ( .A1(n1513), .A2(n1528), .ZN(n1618) );
INVD0 U2478 ( .I(n1618), .ZN(n1724) );
AOI221D0 U2479 ( .A1(n1530), .A2(n2799), .B1(n1529), .B2(
   n2799), .C(n1724),
      .ZN(n1531) );
OAI211D0 U2480 ( .A1(n1789), .A2(n2622), .B(n1531), .C(n1741)
   , .ZN(n1532) );
NR4D0 U2481 ( .A1(n1796), .A2(n1747), .A3(n1613), .A4(n1532),
    .ZN(n1533) );
AOI31D0 U2482 ( .A1(n1534), .A2(n1804), .A3(n1533), .B(n1738)
   , .ZN(n1553) );
AOI22D0 U2483 ( .A1(n2164), .A2(n3732), .B1(n1536), .B2(n1535
   ), .ZN(n1542)
       );
NR2D0 U2484 ( .A1(n1691), .A2(n1724), .ZN(n1768) );
NR2D0 U2485 ( .A1(n1537), .A2(n2639), .ZN(n1650) );
INVD0 U2486 ( .I(n1650), .ZN(n1773) );
ND3D0 U2487 ( .A1(n1665), .A2(n1773), .A3(n1611), .ZN(n1740)
   );
CKND2D0 U2488 ( .A1(n1865), .A2(n1617), .ZN(n1818) );
INVD0 U2489 ( .I(n1818), .ZN(n1809) );
NR3D0 U2490 ( .A1(n1802), .A2(n1538), .A3(n1809), .ZN(n1596)
   );
ND3D0 U2491 ( .A1(n1596), .A2(n1741), .A3(n1704), .ZN(n1729)
   );
NR2D0 U2492 ( .A1(n3757), .A2(n4504), .ZN(n1707) );
```

```
INVD0 U2493 ( .I(n1707), .ZN(n1696) );
OAI211D0 U2494 ( .A1(n2538), .A2(n2126), .B(n1696), .C(n1539)
   , .ZN(n1791) );
INR4D0 U2495 ( .A1(n1768), .B1(n1740), .B2(n1729), .B3(n1791)
   , .ZN(n1541) );
AOI31D0 U2496 ( .A1(n1542), .A2(n1541), .A3(n1540), .B(n1792)
   , .ZN(n1552) );
INVD0 U2497 ( .I(n1572), .ZN(n1836) );
OAI22D0 U2498 ( .A1(n2062), .A2(n3897), .B1(n2126), .B2(n2538
   ), .ZN(n1546)
       );
INVD0 U2499 ( .I(n3671), .ZN(n2337) );
NR2D0 U2500 ( .A1(n1650), .A2(n2337), .ZN(n1671) );
NR2D0 U2501 ( .A1(n1543), .A2(n4018), .ZN(n4162) );
NR2D0 U2502 ( .A1(n3390), .A2(n2622), .ZN(n2783) );
NR4D0 U2503 ( .A1(n1600), .A2(n2615), .A3(n4162), .A4(n2783),
    .ZN(n1544) );
OAI211D0 U2504 ( .A1(n1789), .A2(n1798), .B(n1671), .C(n1544)
   , .ZN(n1545) );
NR4D0 U2505 ( .A1(n1836), .A2(n1586), .A3(n1546), .A4(n1545),
    .ZN(n1550) );
INVD0 U2506 ( .I(n1547), .ZN(n1549) );
INVD0 U2507 ( .I(n1682), .ZN(n1685) );
CKND2D0 U2508 ( .A1(n1685), .A2(n3732), .ZN(n1548) );
AOI31D0 U2509 ( .A1(n1550), .A2(n1549), .A3(n1548), .B(n1926)
   , .ZN(n1551) );
NR4D0 U2510 ( .A1(n1554), .A2(n1553), .A3(n1552), .A4(n1551),
    .ZN(n1568) );
INVD0 U2511 ( .I(n1673), .ZN(n1637) );
ND3D0 U2512 ( .A1(n1572), .A2(n1704), .A3(n1637), .ZN(n1569)
   );
INVD0 U2513 ( .I(n1653), .ZN(n1771) );
NR2D0 U2514 ( .A1(n1809), .A2(n1771), .ZN(n1584) );
NR2D0 U2515 ( .A1(n1556), .A2(n1555), .ZN(n1765) );
INVD0 U2516 ( .I(n1765), .ZN(n1598) );
ND4D0 U2517 ( .A1(n1584), .A2(n3324), .A3(n3671), .A4(n1598),
    .ZN(n1566) );
NR2D0 U2518 ( .A1(n1557), .A2(n2639), .ZN(n1684) );
INVD0 U2519 ( .I(n1684), .ZN(n1767) );
ND3D0 U2520 ( .A1(n1558), .A2(n1665), .A3(n1767), .ZN(n1806)
   );
NR2D0 U2521 ( .A1(n1797), .A2(n1765), .ZN(n1758) );
CKND2D0 U2522 ( .A1(n1758), .A2(n1618), .ZN(n1717) );
NR4D0 U2523 ( .A1(n1687), .A2(n1559), .A3(n1806), .A4(n1717),
    .ZN(n1564) );
IND2D0 U2524 ( .A1(n1600), .B1(n1767), .ZN(n1761) );
NR2D0 U2525 ( .A1(n1560), .A2(n1761), .ZN(n1825) );
```

```
ND4D0 U2526 ( .A1(n1671), .A2(n1825), .A3(n1741), .A4(n1637),
    .ZN(n1561) );
OAI31D0 U2527 ( .A1(n1803), .A2(n1797), .A3(n1561), .B(n1936)
   , .ZN(n1563) );
NR2D0 U2528 ( .A1(n1673), .A2(n1684), .ZN(n1750) );
CKND2D0 U2529 ( .A1(n1708), .A2(n3661), .ZN(n1730) );
INVD0 U2530 ( .I(n1730), .ZN(n1757) );
AO31D0 U2531 ( .A1(n1750), .A2(n1584), .A3(n1757), .B(n1738),
    .Z(n1562) );
OAI211D0 U2532 ( .A1(n1564), .A2(n1926), .B(n1563), .C(n1562)
   , .ZN(n1565) );
AOI221D0 U2533 ( .A1(n1569), .A2(n1931), .B1(n1566), .B2(
   n1931), .C(n1565),
      .ZN(n1567) );
OAI22D0 U2534 ( .A1(n1568), .A2(n4277), .B1(n1567), .B2(n3265
   ), .ZN(n1854)
      );
INVD0 U2535 ( .I(n1569), .ZN(n1681) );
CKND2D0 U2536 ( .A1(n1639), .A2(n1681), .ZN(n1788) );
NR2D0 U2537 ( .A1(n1803), .A2(n1684), .ZN(n1570) );
NR2D0 U2538 ( .A1(n1636), .A2(n1597), .ZN(n1819) );
ND4D0 U2539 ( .A1(n1570), .A2(n1819), .A3(n1574), .A4(n3661),
    .ZN(n1583) );
NR3D0 U2540 ( .A1(n1688), .A2(n1691), .A3(n1656), .ZN(n1702)
   );
NR4D0 U2541 ( .A1(n1600), .A2(n1724), .A3(n2796), .A4(n1642),
    .ZN(n1571) );
ND4D0 U2542 ( .A1(n1750), .A2(n1702), .A3(n1571), .A4(n1674),
    .ZN(n1576) );
INVD0 U2543 ( .I(n1738), .ZN(n1842) );
CKND2D0 U2544 ( .A1(n1572), .A2(n1665), .ZN(n1599) );
NR2D0 U2545 ( .A1(n1612), .A2(n1599), .ZN(n1610) );
CKND2D0 U2546 ( .A1(n2173), .A2(n1719), .ZN(n1573) );
ND4D0 U2547 ( .A1(n1832), .A2(n1610), .A3(n1574), .A4(n1573),
    .ZN(n1575) );
AOI22D0 U2548 ( .A1(n1577), .A2(n1576), .B1(n1842), .B2(n1575
   ), .ZN(n1580)
      );
INVD0 U2549 ( .I(n1578), .ZN(n1592) );
NR4D0 U2550 ( .A1(n1687), .A2(n1617), .A3(n1740), .A4(n1592),
    .ZN(n1579) );
AOI32D0 U2551 ( .A1(n1581), .A2(n1580), .A3(n1579), .B1(n1792
   ), .B2(n1580),
      .ZN(n1582) );
AOI221D0 U2552 ( .A1(n1788), .A2(n1931), .B1(n1583), .B2(
   n1931), .C(n1582),
      .ZN(n1852) );
```

```
ND4D0 U2553 ( .A1(n1825), .A2(n1584), .A3(n4504), .A4(n1618),
    .ZN(n1590) );
CKND2D0 U2554 ( .A1(n1610), .A2(n1762), .ZN(n1632) );
INR4D0 U2555 ( .A1(n1619), .B1(n1617), .B2(n1591), .B3(n1632)
   , .ZN(n1585) );
NR2D0 U2556 ( .A1(n1800), .A2(n3009), .ZN(n1641) );
INVD0 U2557 ( .I(n1641), .ZN(n1830) );
AOI31D0 U2558 ( .A1(n1585), .A2(n1767), .A3(n1830), .B(n1926)
   , .ZN(n1589) );
NR2D0 U2559 ( .A1(n1803), .A2(n1586), .ZN(n1669) );
AOI31D0 U2560 ( .A1(counter[2]), .A2(n2173), .A3(n4098), .B(
   n1632), .ZN(
      n1587) );
AOI31D0 U2561 ( .A1(n1669), .A2(n1822), .A3(n1587), .B(n1738)
   , .ZN(n1588) );
AOI211D0 U2562 ( .A1(n1931), .A2(n1590), .B(n1589), .C(n1588)
   , .ZN(n1594) );
INR4D0 U2563 ( .A1(n1669), .B1(n1592), .B2(n1591), .B3(n1806)
   , .ZN(n1593) );
AOI32D0 U2564 ( .A1(n1595), .A2(n1594), .A3(n1593), .B1(n1792
   ), .B2(n1594),
      .ZN(n1629) );
INVD0 U2565 ( .I(n1596), .ZN(n1746) );
NR2D0 U2566 ( .A1(n1707), .A2(n1597), .ZN(n1833) );
CKND2D0 U2567 ( .A1(n1833), .A2(n1598), .ZN(n1826) );
NR2D0 U2568 ( .A1(n1746), .A2(n1826), .ZN(n1666) );
IND4D0 U2569 ( .A1(n1599), .B1(n1666), .B2(n1611), .B3(n4233)
   , .ZN(n1609) );
INVD0 U2570 ( .I(n1833), .ZN(n1630) );
CKND2D0 U2571 ( .A1(n1611), .A2(n1631), .ZN(n1827) );
NR2D0 U2572 ( .A1(n1630), .A2(n1827), .ZN(n1731) );
OR2D0 U2573 ( .A1(n1600), .A2(n1650), .Z(n1728) );
INR4D0 U2574 ( .A1(n1731), .B1(n1683), .B2(n1730), .B3(n1728)
   , .ZN(n1601) );
AOI31D0 U2575 ( .A1(n1681), .A2(n1602), .A3(n1601), .B(n1926)
   , .ZN(n1608) );
CKND2D0 U2576 ( .A1(n1669), .A2(n1764), .ZN(n1695) );
CKND2D0 U2577 ( .A1(n2654), .A2(n1667), .ZN(n3647) );
ND4D0 U2578 ( .A1(n3617), .A2(n1708), .A3(n1773), .A4(n3647),
    .ZN(n1603) );
OAI31D0 U2579 ( .A1(n1695), .A2(n1616), .A3(n1603), .B(n1936)
   , .ZN(n1606) );
INR4D0 U2580 ( .A1(n1604), .B1(n1802), .B2(n3651), .B3(n1761)
   , .ZN(n1605) );
AOI32D0 U2581 ( .A1(n1681), .A2(n1606), .A3(n1605), .B1(n1772
   ), .B2(n1606),
      .ZN(n1607) );
```

```
AOI211D0 U2582 ( .A1(n1842), .A2(n1609), .B(n1608), .C(n1607)
   , .ZN(n1627) );
NR2D0 U2583 ( .A1(n1687), .A2(n4162), .ZN(n1633) );
ND4D0 U2584 ( .A1(n1733), .A2(n1633), .A3(n1610), .A4(n4504),
    .ZN(n1625) );
CKND2D0 U2585 ( .A1(n1611), .A2(n1671), .ZN(n1672) );
CKND2D0 U2586 ( .A1(n1614), .A2(n1830), .ZN(n1703) );
OR2D0 U2587 ( .A1(n1703), .A2(n1612), .Z(n1745) );
OA31D0 U2588 ( .A1(n1672), .A2(n1613), .A3(n1745), .B(n1842),
    .Z(n1624) );
NR2D0 U2589 ( .A1(n1797), .A2(n1707), .ZN(n1670) );
CKND2D0 U2590 ( .A1(n1818), .A2(n1614), .ZN(n1760) );
INR4D0 U2591 ( .A1(n1670), .B1(n1836), .B2(n1615), .B3(n1760)
   , .ZN(n1622) );
NR2D0 U2592 ( .A1(n1616), .A2(n1718), .ZN(n1635) );
INVD0 U2593 ( .I(n1635), .ZN(n1748) );
NR2D0 U2594 ( .A1(n1673), .A2(n1656), .ZN(n1831) );
CKND2D0 U2595 ( .A1(counter[1]), .A2(n1617), .ZN(n1643) );
ND4D0 U2596 ( .A1(n1619), .A2(n1831), .A3(n1618), .A4(n1643),
    .ZN(n1620) );
NR4D0 U2597 ( .A1(n1688), .A2(n1759), .A3(n1748), .A4(n1620),
    .ZN(n1621) );
OAI22D0 U2598 ( .A1(n1622), .A2(n1772), .B1(n1621), .B2(n1926
   ), .ZN(n1623)
      );
AOI211D0 U2599 ( .A1(n1936), .A2(n1625), .B(n1624), .C(n1623)
   , .ZN(n1626) );
OAI22D0 U2600 ( .A1(n1627), .A2(n3379), .B1(n1626), .B2(n4298
   ), .ZN(n1628)
      );
AOI21D0 U2601 ( .A1(n3850), .A2(n1629), .B(n1628), .ZN(n1851)
    );
NR3D0 U2602 ( .A1(n1771), .A2(n1630), .A3(n1728), .ZN(n1739)
   );
IND4D0 U2603 ( .A1(n1632), .B1(n1768), .B2(n1739), .B3(n1631)
   , .ZN(n1648) );
AN4D0 U2604 ( .A1(n1633), .A2(n1696), .A3(n1767), .A4(n1830),
    .Z(n1634) );
AOI31D0 U2605 ( .A1(n1653), .A2(n1635), .A3(n1634), .B(n1738)
   , .ZN(n1647) );
NR2D0 U2606 ( .A1(n1636), .A2(n1707), .ZN(n1680) );
ND4D0 U2607 ( .A1(n1639), .A2(n1638), .A3(n1637), .A4(n1643),
    .ZN(n1640) );
OAI31D0 U2608 ( .A1(n1695), .A2(n1827), .A3(n1640), .B(n1931)
   , .ZN(n1645) );
NR2D0 U2609 ( .A1(n1642), .A2(n1641), .ZN(n1824) );
AN4D0 U2610 ( .A1(n1756), .A2(n1643), .A3(n3140), .A4(n1824),
    .Z(n1644) );
```

```
AOI32D0 U2611 ( .A1(n1680), .A2(n1645), .A3(n1644), .B1(n1926
   ), .B2(n1645),
      .ZN(n1646) );
AOI211D0 U2612 ( .A1(n1936), .A2(n1648), .B(n1647), .C(n1646)
   , .ZN(n1664) );
ND4D0 U2613 ( .A1(n1702), .A2(n1680), .A3(n1649), .A4(n1830),
    .ZN(n1661) );
OR4D0 U2614 ( .A1(n1717), .A2(n1835), .A3(n1650), .A4(n1749),
    .Z(n1655) );
NR2D0 U2615 ( .A1(n1651), .A2(n2424), .ZN(n4037) );
NR4D0 U2616 ( .A1(n1688), .A2(n1724), .A3(n1718), .A4(n4037),
    .ZN(n1652) );
AOI31D0 U2617 ( .A1(n1653), .A2(n1680), .A3(n1652), .B(n1738)
   , .ZN(n1654) );
AOI221D0 U2618 ( .A1(n1746), .A2(n1931), .B1(n1655), .B2(
   n1931), .C(n1654),
      .ZN(n1659) );
NR4D0 U2619 ( .A1(n1657), .A2(n1802), .A3(n1656), .A4(n1717),
    .ZN(n1658) );
AOI32D0 U2620 ( .A1(n1833), .A2(n1659), .A3(n1658), .B1(n1926
   ), .B2(n1659),
      .ZN(n1660) );
AOI221D0 U2621 ( .A1(n1662), .A2(n1936), .B1(n1661), .B2(
   n1936), .C(n1660),
      .ZN(n1663) );
OAI22D0 U2622 ( .A1(n1664), .A2(n4527), .B1(n1663), .B2(n4325
   ), .ZN(n1849)
       );
ND4D0 U2623 ( .A1(n1702), .A2(n1666), .A3(n1764), .A4(n1665),
    .ZN(n1679) );
AOI211D0 U2624 ( .A1(n1667), .A2(n4351), .B(n1835), .C(n1761)
   , .ZN(n1668) );
AOI31D0 U2625 ( .A1(n1670), .A2(n1669), .A3(n1668), .B(n1738)
   , .ZN(n1678) );
INR4D0 U2626 ( .A1(n1671), .B1(n2706), .B2(n1759), .B3(n2664)
   , .ZN(n1676) );
OAI31D0 U2627 ( .A1(n1673), .A2(n1672), .A3(n1760), .B(n1931)
   , .ZN(n1675) );
AOI32D0 U2628 ( .A1(n1676), .A2(n1675), .A3(n1674), .B1(n1926
   ), .B2(n1675),
      .ZN(n1677) );
AOI211D0 U2629 ( .A1(n1936), .A2(n1679), .B(n1678), .C(n1677)
   , .ZN(n1716) );
OAI211D0 U2630 ( .A1(n1682), .A2(n3735), .B(n1681), .C(n1680)
   , .ZN(n1690) );
AOI211D0 U2631 ( .A1(n1686), .A2(n1685), .B(n1684), .C(n1683)
   , .ZN(n1709) );
```

```
AOI211D0 U2632 ( .A1(n3907), .A2(n2173), .B(n1688), .C(n1687)
   , .ZN(n1689) );
IND4D0 U2633 ( .A1(n1690), .B1(n1757), .B2(n1709), .B3(n1689)
   , .ZN(n1714) );
NR4D0 U2634 ( .A1(n1691), .A2(n1761), .A3(n1690), .A4(n1827),
    .ZN(n1693) );
AOI31D0 U2635 ( .A1(n1693), .A2(n1692), .A3(n1705), .B(n1792)
   , .ZN(n1713) );
AOI221D0 U2636 ( .A1(counter[11]), .A2(n1694), .B1(n3956), .
   B2(n2776), .C(
      n1799), .ZN(n1700) );
INVD0 U2637 ( .I(n1695), .ZN(n1839) );
OAI211D0 U2638 ( .A1(n3449), .A2(n1697), .B(n1839), .C(n1696)
   , .ZN(n1698) );
NR4D0 U2639 ( .A1(n1701), .A2(n1700), .A3(n1699), .A4(n1698),
    .ZN(n1711) );
INVD0 U2640 ( .I(n1702), .ZN(n1843) );
NR2D0 U2641 ( .A1(n1843), .A2(n1703), .ZN(n1742) );
ND4D0 U2642 ( .A1(n1742), .A2(n1705), .A3(n3075), .A4(n1704),
    .ZN(n1706) );
IINR4D0 U2643 ( .A1(n1709), .A2(n1708), .B1(n1707), .B2(n1706
   ), .ZN(n1710)
      );
OAI22D0 U2644 ( .A1(n1711), .A2(n1738), .B1(n1710), .B2(n1926
   ), .ZN(n1712)
      );
AOI211D0 U2645 ( .A1(n1931), .A2(n1714), .B(n1713), .C(n1712)
   , .ZN(n1715) );
OAI22D0 U2646 ( .A1(n1716), .A2(n4192), .B1(n1715), .B2(n3777
   ), .ZN(n1848)
      );
AOI211D0 U2647 ( .A1(n1719), .A2(n2173), .B(n1718), .C(n1717)
   , .ZN(n1721) );
AOI31D0 U2648 ( .A1(n1721), .A2(n1720), .A3(n1830), .B(n1738)
   , .ZN(n1737) );
NR2D0 U2649 ( .A1(n2120), .A2(n1722), .ZN(n1725) );
NR4D0 U2650 ( .A1(n1725), .A2(n1724), .A3(n1765), .A4(n1723),
    .ZN(n1727) );
AOI31D0 U2651 ( .A1(n1727), .A2(n1726), .A3(n4233), .B(n1926)
   , .ZN(n1736) );
OA31D0 U2652 ( .A1(n1730), .A2(n1729), .A3(n1728), .B(n1936),
    .Z(n1735) );
AOI31D0 U2653 ( .A1(n1733), .A2(n1732), .A3(n1731), .B(n1772)
   , .ZN(n1734) );
NR4D0 U2654 ( .A1(n1737), .A2(n1736), .A3(n1735), .A4(n1734),
    .ZN(n1781) );
AOI31D0 U2655 ( .A1(n1831), .A2(n1739), .A3(n4094), .B(n1738)
   , .ZN(n1755) );
```

```
INVD0 U2656 ( .I(n1740), .ZN(n1744) );
AN4D0 U2657 ( .A1(n1742), .A2(n1741), .A3(n3647), .A4(n1784),
    .Z(n1743) );
AOI21D0 U2658 ( .A1(n1744), .A2(n1743), .B(n1772), .ZN(n1754)
    );
NR4D0 U2659 ( .A1(n1747), .A2(n4494), .A3(n1746), .A4(n1745),
    .ZN(n1752) );
INR4D0 U2660 ( .A1(n1750), .B1(n1749), .B2(n1826), .B3(n1748)
   , .ZN(n1751) );
OAI22D0 U2661 ( .A1(n1752), .A2(n1792), .B1(n1751), .B2(n1926
   ), .ZN(n1753)
      );
OAI31D0 U2662 ( .A1(n1755), .A2(n1754), .A3(n1753), .B(n3437)
   , .ZN(n1780) );
AOI31D0 U2663 ( .A1(n1758), .A2(n1757), .A3(n1756), .B(n1926)
   , .ZN(n1778) );
OA31D0 U2664 ( .A1(n1761), .A2(n1760), .A3(n1759), .B(n1936),
    .Z(n1777) );
AN4D0 U2665 ( .A1(n1819), .A2(n1764), .A3(n1763), .A4(n1762),
    .Z(n1775) );
NR2D0 U2666 ( .A1(n1766), .A2(n1765), .ZN(n1821) );
INVD0 U2667 ( .I(n1821), .ZN(n1770) );
IND4D0 U2668 ( .A1(n1803), .B1(n1768), .B2(n3324), .B3(n1767)
   , .ZN(n1769) );
OAI31D0 U2669 ( .A1(n1771), .A2(n1770), .A3(n1769), .B(n1842)
   , .ZN(n1774) );
AOI32D0 U2670 ( .A1(n1775), .A2(n1774), .A3(n1773), .B1(n1772
   ), .B2(n1774),
     .ZN(n1776) );
OAI31D0 U2671 ( .A1(n1778), .A2(n1777), .A3(n1776), .B(n4492)
   , .ZN(n1779) );
OAI211D0 U2672 ( .A1(n1781), .A2(n4408), .B(n1780), .C(n1779)
   , .ZN(n1847) );
AN4D0 U2673 ( .A1(n1784), .A2(n1783), .A3(n1782), .A4(n1822),
    .Z(n1787) );
AOI32D0 U2674 ( .A1(n4244), .A2(n1787), .A3(n1786), .B1(n1785
   ), .B2(n1787),
     .ZN(n1817) );
INVD0 U2675 ( .I(n1788), .ZN(n1794) );
OAI21D0 U2676 ( .A1(n1789), .A2(n2789), .B(n1830), .ZN(n1790)
    );
AOI211D0 U2677 ( .A1(n1872), .A2(n4117), .B(n1791), .C(n1790)
   , .ZN(n1793) );
AOI31D0 U2678 ( .A1(n1795), .A2(n1794), .A3(n1793), .B(n1792)
   , .ZN(n1816) );
NR4D0 U2679 ( .A1(n1797), .A2(n1796), .A3(n4309), .A4(n2615),
    .ZN(n1814) );
```

```
AOI21D0 U2680 ( .A1(n1799), .A2(n4444), .B(n1798), .ZN(n1808)
    );
AOI21D0 U2681 ( .A1(n3009), .A2(n2618), .B(n1800), .ZN(n1810)
    );
OAI22D0 U2682 ( .A1(n1988), .A2(n3449), .B1(n1800), .B2(n3735
   ), .ZN(n1801)
       );
INR4D0 U2683 ( .A1(n1804), .B1(n1803), .B2(n1802), .B3(n1801)
   , .ZN(n1805) );
IIND4D0 U2684 ( .A1(n1806), .A2(n1810), .B1(n1831), .B2(n1805
   ), .ZN(n1807)
       );
OAI31D0 U2685 ( .A1(n1809), .A2(n1808), .A3(n1807), .B(n1842)
   , .ZN(n1813) );
AOI211D0 U2686 ( .A1(n1811), .A2(n3538), .B(n1826), .C(n1810)
   , .ZN(n1812) );
AOI32D0 U2687 ( .A1(n1814), .A2(n1813), .A3(n1812), .B1(n1926
   ), .B2(n1813),
      .ZN(n1815) );
AOI211D0 U2688 ( .A1(n1931), .A2(n1817), .B(n1816), .C(n1815)
   , .ZN(n1845) );
ND4D0 U2689 ( .A1(n1823), .A2(n1819), .A3(n1821), .A4(n1818),
    .ZN(n1841) );
CKND2D0 U2690 ( .A1(n2654), .A2(n1820), .ZN(n2197) );
ND4D0 U2691 ( .A1(n1823), .A2(n1822), .A3(n1821), .A4(n2197),
    .ZN(n1829) );
IIND4D0 U2692 ( .A1(n1827), .A2(n1826), .B1(n1825), .B2(n1824
   ), .ZN(n1828)
       );
AOI22D0 U2693 ( .A1(n1931), .A2(n1829), .B1(n1936), .B2(n1828
   ), .ZN(n1838)
       );
ND4D0 U2694 ( .A1(n1833), .A2(n1832), .A3(n1831), .A4(n1830),
    .ZN(n1834) );
NR4D0 U2695 ( .A1(n1836), .A2(n4472), .A3(n1835), .A4(n1834),
    .ZN(n1837) );
AOI32D0 U2696 ( .A1(n1839), .A2(n1838), .A3(n1837), .B1(n1926
   ), .B2(n1838),
      .ZN(n1840) );
AOI221D0 U2697 ( .A1(n1843), .A2(n1842), .B1(n1841), .B2(
   n1842), .C(n1840),
      .ZN(n1844) );
OAI22D0 U2698 ( .A1(n1845), .A2(n4129), .B1(n1844), .B2(n4190
   ), .ZN(n1846)
       );
NR4D0 U2699 ( .A1(n1849), .A2(n1848), .A3(n1847), .A4(n1846),
    .ZN(n1850) );
```

```
OAI211D0 U2700 ( .A1(n1852), .A2(n4148), .B(n1851), .C(n1850)
   , .ZN(n1853) );
AOI211D0 U2701 ( .A1(n4063), .A2(n1855), .B(n1854), .C(n1853)
   , .ZN(n2191) );
CKND2D0 U2702 ( .A1(counter[0]), .A2(n1872), .ZN(n2326) );
INVD0 U2703 ( .I(n2326), .ZN(n2355) );
NR2D0 U2704 ( .A1(n2565), .A2(n2355), .ZN(n2340) );
NR2D0 U2705 ( .A1(n2270), .A2(counter[3]), .ZN(n1949) );
INVD0 U2706 ( .I(n1949), .ZN(n1938) );
NR2D0 U2707 ( .A1(counter[0]), .A2(n1856), .ZN(n1962) );
CKND2D0 U2708 ( .A1(n1904), .A2(n2355), .ZN(n2073) );
INVD0 U2709 ( .I(n2073), .ZN(n1900) );
NR2D0 U2710 ( .A1(n1962), .A2(n1900), .ZN(n2052) );
CKND2D0 U2711 ( .A1(n2032), .A2(n2565), .ZN(n2465) );
INVD0 U2712 ( .I(n2465), .ZN(n2241) );
INVD0 U2713 ( .I(n2287), .ZN(n2568) );
NR2D0 U2714 ( .A1(n1859), .A2(n2260), .ZN(n1954) );
INVD0 U2715 ( .I(n1954), .ZN(n2090) );
AOI211D0 U2716 ( .A1(n2568), .A2(n3737), .B(n2315), .C(n2090)
   , .ZN(n1857) );
AOI211D0 U2717 ( .A1(n2321), .A2(n1931), .B(n2241), .C(n1857)
   , .ZN(n1858) );
OAI211D0 U2718 ( .A1(n2340), .A2(n1938), .B(n2052), .C(n1858)
   , .ZN(n1876) );
CKND2D0 U2719 ( .A1(n1859), .A2(n2263), .ZN(n1946) );
CKND2D0 U2720 ( .A1(counter[3]), .A2(n1860), .ZN(n1870) );
OAI22D0 U2721 ( .A1(n1941), .A2(n1946), .B1(n2326), .B2(n1870
   ), .ZN(n1940)
     );
NR2D0 U2722 ( .A1(n2092), .A2(n2540), .ZN(n2078) );
CKND2D0 U2723 ( .A1(n1925), .A2(n2316), .ZN(n2044) );
NR2D0 U2724 ( .A1(counter[1]), .A2(n2044), .ZN(n2042) );
NR2D0 U2725 ( .A1(n2078), .A2(n2042), .ZN(n2159) );
CKND2D0 U2726 ( .A1(n2568), .A2(n2540), .ZN(n2298) );
NR2D0 U2727 ( .A1(n2092), .A2(n2526), .ZN(n2142) );
AOI211D0 U2728 ( .A1(n1949), .A2(n2298), .B(n4110), .C(n2142)
   , .ZN(n1861) );
CKND2D0 U2729 ( .A1(n1925), .A2(n2354), .ZN(n1897) );
IND4D0 U2730 ( .A1(n1940), .B1(n2159), .B2(n1861), .B3(n1897)
   , .ZN(n1869) );
NR2D0 U2731 ( .A1(n2260), .A2(n1888), .ZN(n1953) );
NR2D0 U2732 ( .A1(n1905), .A2(n2568), .ZN(n2141) );
INVD0 U2733 ( .I(n2141), .ZN(n2070) );
CKND2D0 U2734 ( .A1(n1932), .A2(n2565), .ZN(n2110) );
INVD0 U2735 ( .I(n1870), .ZN(n1944) );
CKND2D0 U2736 ( .A1(n2565), .A2(n1944), .ZN(n1863) );
NR2D0 U2737 ( .A1(n3821), .A2(n1862), .ZN(n2653) );
INVD0 U2738 ( .I(n2653), .ZN(n3914) );
```

```
ND4D0 U2739 ( .A1(n2070), .A2(n2110), .A3(n1863), .A4(n3914),
    .ZN(n1864) );
NR2D0 U2740 ( .A1(n1953), .A2(n1864), .ZN(n1877) );
NR2D0 U2741 ( .A1(n1865), .A2(n2044), .ZN(n2150) );
NR2D0 U2742 ( .A1(counter[5]), .A2(n4167), .ZN(n2176) );
NR2D0 U2743 ( .A1(n2274), .A2(n4387), .ZN(n1995) );
NR3D0 U2744 ( .A1(n2150), .A2(n2176), .A3(n1995), .ZN(n1924)
   );
NR2D0 U2745 ( .A1(n2260), .A2(n1939), .ZN(n2560) );
OAI22D0 U2746 ( .A1(n2270), .A2(n4272), .B1(n1926), .B2(n2351
   ), .ZN(n1933)
       );
NR2D0 U2747 ( .A1(n2274), .A2(n4167), .ZN(n1986) );
NR2D0 U2748 ( .A1(n2092), .A2(n2326), .ZN(n1960) );
CKND2D0 U2749 ( .A1(n1925), .A2(n2321), .ZN(n2006) );
IND2D0 U2750 ( .A1(n1960), .B1(n2006), .ZN(n2060) );
NR2D0 U2751 ( .A1(n1986), .A2(n2060), .ZN(n1996) );
OAI211D0 U2752 ( .A1(n2271), .A2(n2270), .B(n1996), .C(n2073)
   , .ZN(n1866) );
AOI221D0 U2753 ( .A1(n2560), .A2(n3472), .B1(n1933), .B2(
   n3472), .C(n1866),
      .ZN(n1867) );
AOI31D0 U2754 ( .A1(n1877), .A2(n1924), .A3(n1867), .B(n2126)
   , .ZN(n1868) );
AOI221D0 U2755 ( .A1(n1876), .A2(n2173), .B1(n1869), .B2(
   n2173), .C(n1868),
      .ZN(n1886) );
INVD0 U2756 ( .I(n1924), .ZN(n1875) );
OR2D0 U2757 ( .A1(n2092), .A2(n2271), .Z(n2069) );
NR2D0 U2758 ( .A1(n1887), .A2(n2568), .ZN(n1992) );
INR2D0 U2759 ( .A1(n2069), .B1(n1992), .ZN(n2154) );
INVD0 U2760 ( .I(n1946), .ZN(n2080) );
NR2D0 U2761 ( .A1(n2565), .A2(n2320), .ZN(n2264) );
AOI21D0 U2762 ( .A1(n2264), .A2(n2423), .B(n1870), .ZN(n1871)
    );
CKND2D0 U2763 ( .A1(n1932), .A2(n2354), .ZN(n2082) );
CKND2D0 U2764 ( .A1(n2082), .A2(n2006), .ZN(n1969) );
AOI211D0 U2765 ( .A1(n1872), .A2(n2080), .B(n1871), .C(n1969)
   , .ZN(n1873) );
CKND2D0 U2766 ( .A1(n1932), .A2(n2320), .ZN(n1914) );
NR2D0 U2767 ( .A1(n2092), .A2(n2568), .ZN(n2014) );
INVD0 U2768 ( .I(n2014), .ZN(n2072) );
ND4D0 U2769 ( .A1(n2154), .A2(n1873), .A3(n1914), .A4(n2072),
    .ZN(n1874) );
OAI31D0 U2770 ( .A1(n1876), .A2(n1875), .A3(n1874), .B(n2164)
   , .ZN(n1885) );
INVD0 U2771 ( .I(n1877), .ZN(n1883) );
CKND2D0 U2772 ( .A1(n2032), .A2(n1878), .ZN(n1991) );
```

```
  INVD0 U2773 ( .I(n1991), .ZN(n1964) );
  INR2D0 U2774 ( .A1(n2082), .B1(n1964), .ZN(n2152) );
  INVD0 U2775 ( .I(n1995), .ZN(n2055) );
  CKND2D0 U2776 ( .A1(n2152), .A2(n2055), .ZN(n1972) );
  CKND2D0 U2777 ( .A1(n2568), .A2(n2326), .ZN(n2454) );
  NR2D0 U2778 ( .A1(n2422), .A2(n4100), .ZN(n2061) );
  INVD0 U2779 ( .I(n2540), .ZN(n2262) );
  CKND2D0 U2780 ( .A1(n1904), .A2(n2262), .ZN(n2043) );
  INVD0 U2781 ( .I(n2043), .ZN(n2076) );
  AOI211D0 U2782 ( .A1(n1944), .A2(n2454), .B(n2061), .C(n2076)
     , .ZN(n1880) );
  NR2D0 U2783 ( .A1(n2341), .A2(n2090), .ZN(n2087) );
  INVD0 U2784 ( .I(n2087), .ZN(n1879) );
  OAI211D0 U2785 ( .A1(n1881), .A2(n2274), .B(n1880), .C(n1879)
     , .ZN(n1882) );
  INVD0 U2786 ( .I(n2120), .ZN(n2162) );
  OAI31D0 U2787 ( .A1(n1883), .A2(n1972), .A3(n1882), .B(n2162)
     , .ZN(n1884) );
  AOI31D0 U2788 ( .A1(n1886), .A2(n1885), .A3(n1884), .B(n4277)
     , .ZN(n2189) );
  CKND2D0 U2789 ( .A1(n1925), .A2(n2320), .ZN(n2111) );
  INR2D0 U2790 ( .A1(n2111), .B1(n2078), .ZN(n2025) );
  INVD0 U2791 ( .I(n1897), .ZN(n1907) );
  NR2D0 U2792 ( .A1(n1907), .A2(n2076), .ZN(n2138) );
  CKAN2D0 U2793 ( .A1(n2025), .A2(n2138), .Z(n2065) );
  CKND2D0 U2794 ( .A1(n1932), .A2(n2321), .ZN(n2169) );
  ND4D0 U2795 ( .A1(n2065), .A2(n4167), .A3(n2169), .A4(n2082),
      .ZN(n1896) );
  NR2D0 U2796 ( .A1(n1887), .A2(n2540), .ZN(n2123) );
  NR2D0 U2797 ( .A1(n2123), .A2(n2176), .ZN(n2011) );
  NR2D0 U2798 ( .A1(n1905), .A2(n2526), .ZN(n2107) );
  INVD0 U2799 ( .I(n2107), .ZN(n1973) );
  CKND2D0 U2800 ( .A1(n1973), .A2(n2169), .ZN(n2040) );
  NR2D0 U2801 ( .A1(n1905), .A2(n2423), .ZN(n2149) );
  NR2D0 U2802 ( .A1(n1986), .A2(n2149), .ZN(n1901) );
  CKND2D0 U2803 ( .A1(n2055), .A2(n1901), .ZN(n1997) );
  NR2D0 U2804 ( .A1(n3749), .A2(n3750), .ZN(n4322) );
  NR2D0 U2805 ( .A1(n2193), .A2(n2281), .ZN(n2224) );
  INVD0 U2806 ( .I(n2224), .ZN(n2396) );
  CKND2D0 U2807 ( .A1(n2124), .A2(n2565), .ZN(n2505) );
  ND3D0 U2808 ( .A1(n2396), .A2(n2505), .A3(n2070), .ZN(n2116)
     );
  CKND2D0 U2809 ( .A1(n4322), .A2(n2116), .ZN(n2038) );
  IND4D0 U2810 ( .A1(n1997), .B1(n2038), .B2(n2073), .B3(n2082)
     , .ZN(n1892) );
  INR3D0 U2811 ( .A1(n2011), .B1(n1972), .B2(n1992), .ZN(n1990)
      );
  NR2D0 U2812 ( .A1(n2193), .A2(n1941), .ZN(n2426) );
```

```
INR4D0 U2813 ( .A1(n1990), .B1(n1962), .B2(n2426), .B3(n2040)
   , .ZN(n1890) );
NR2D0 U2814 ( .A1(n2193), .A2(n1888), .ZN(n2156) );
INVD0 U2815 ( .I(n2156), .ZN(n1889) );
NR2D0 U2816 ( .A1(n1905), .A2(n2271), .ZN(n2155) );
INVD0 U2817 ( .I(n2155), .ZN(n1993) );
AOI31D0 U2818 ( .A1(n1890), .A2(n1889), .A3(n1993), .B(n2062)
   , .ZN(n1891) );
AOI221D0 U2819 ( .A1(n2040), .A2(n2173), .B1(n1892), .B2(
   n2173), .C(n1891),
      .ZN(n1894) );
NR2D0 U2820 ( .A1(n1907), .A2(n2141), .ZN(n2053) );
ND3D0 U2821 ( .A1(n2053), .A2(n2073), .A3(n2006), .ZN(n2131)
   );
NR4D0 U2822 ( .A1(n2156), .A2(n2241), .A3(n2131), .A4(n1997),
    .ZN(n1893) );
AOI32D0 U2823 ( .A1(n2011), .A2(n1894), .A3(n1893), .B1(n2120
   ), .B2(n1894),
      .ZN(n1895) );
AOI221D0 U2824 ( .A1(n2042), .A2(n2165), .B1(n1896), .B2(
   n2165), .C(n1895),
      .ZN(n1923) );
CKND2D0 U2825 ( .A1(n2111), .A2(n2073), .ZN(n1975) );
INVD0 U2826 ( .I(n1972), .ZN(n2172) );
OAI211D0 U2827 ( .A1(n4263), .A2(n2193), .B(n2172), .C(n1897)
   , .ZN(n1898) );
NR3D0 U2828 ( .A1(n2042), .A2(n1975), .A3(n1898), .ZN(n2007)
   );
CKND2D0 U2829 ( .A1(n2154), .A2(n2043), .ZN(n2010) );
NR2D0 U2830 ( .A1(n2156), .A2(n2010), .ZN(n2021) );
ND4D0 U2831 ( .A1(n2025), .A2(n2021), .A3(n2053), .A4(n2039),
    .ZN(n1899) );
NR4D0 U2832 ( .A1(n1900), .A2(n2014), .A3(n2040), .A4(n1899),
    .ZN(n1903) );
CKND2D0 U2833 ( .A1(n1925), .A2(n2287), .ZN(n2170) );
INVD0 U2834 ( .I(n1901), .ZN(n2168) );
NR2D0 U2835 ( .A1(n2168), .A2(n2107), .ZN(n2109) );
AN4D0 U2836 ( .A1(n2070), .A2(n2170), .A3(n1924), .A4(n2109),
    .Z(n1902) );
OAI222D0 U2837 ( .A1(n2120), .A2(n2007), .B1(n2126), .B2(
   n1903), .C1(n2062),
      .C2(n1902), .ZN(n1912) );
INVD0 U2838 ( .I(n1962), .ZN(n2089) );
CKND2D0 U2839 ( .A1(n1904), .A2(n2321), .ZN(n2147) );
CKND2D0 U2840 ( .A1(n2089), .A2(n2147), .ZN(n2167) );
NR2D0 U2841 ( .A1(n2167), .A2(n2078), .ZN(n2128) );
NR2D0 U2842 ( .A1(n1986), .A2(n2142), .ZN(n2008) );
NR2D0 U2843 ( .A1(n1960), .A2(n2123), .ZN(n1979) );
```

```
OAI211D0 U2844 ( .A1(n2264), .A2(n1905), .B(n2008), .C(n1979)
   , .ZN(n1906) );
INR4D0 U2845 ( .A1(n2128), .B1(n1907), .B2(n2150), .B3(n1906)
   , .ZN(n1910) );
INVD0 U2846 ( .I(n2147), .ZN(n2015) );
INR3D0 U2847 ( .A1(n2038), .B1(n2015), .B2(n1992), .ZN(n2033)
    );
AN3D0 U2848 ( .A1(n2008), .A2(n2033), .A3(n2011), .Z(n1909) )
   ;
NR2D0 U2849 ( .A1(n1962), .A2(n1995), .ZN(n2139) );
CKND2D0 U2850 ( .A1(n4330), .A2(n1914), .ZN(n1917) );
INR4D0 U2851 ( .A1(n2139), .B1(n2061), .B2(n2426), .B3(n1917)
   , .ZN(n1908) );
OAI222D0 U2852 ( .A1(n2120), .A2(n1910), .B1(n2126), .B2(
   n1909), .C1(n2062),
      .C2(n1908), .ZN(n1911) );
AOI22D0 U2853 ( .A1(n4521), .A2(n1912), .B1(n3386), .B2(n1911
   ), .ZN(n1922)
       );
INR4D0 U2854 ( .A1(n2052), .B1(n1964), .B2(n2010), .B3(n2060)
   , .ZN(n1913) );
AOI31D0 U2855 ( .A1(n1913), .A2(n2505), .A3(n1993), .B(n4298)
   , .ZN(n1920) );
CKND2D0 U2856 ( .A1(n1914), .A2(n2170), .ZN(n2112) );
NR2D0 U2857 ( .A1(n2156), .A2(n2112), .ZN(n2091) );
NR4D0 U2858 ( .A1(n1915), .A2(n2150), .A3(n2149), .A4(n1972),
    .ZN(n1916) );
AOI31D0 U2859 ( .A1(n2091), .A2(n1916), .A3(n2038), .B(n3379)
   , .ZN(n1919) );
NR2D0 U2860 ( .A1(n2193), .A2(n2568), .ZN(n2213) );
NR2D0 U2861 ( .A1(n2155), .A2(n2213), .ZN(n1974) );
INVD0 U2862 ( .I(n1974), .ZN(n1961) );
OR3D0 U2863 ( .A1(n2123), .A2(n1992), .A3(n2149), .Z(n2077) )
   ;
OA31D0 U2864 ( .A1(n1961), .A2(n1917), .A3(n2077), .B(n4521),
    .Z(n1918) );
OAI31D0 U2865 ( .A1(n1920), .A2(n1919), .A3(n1918), .B(n2173)
   , .ZN(n1921) );
OAI211D0 U2866 ( .A1(n1923), .A2(n4192), .B(n1922), .C(n1921)
   , .ZN(n2188) );
CKND2D0 U2867 ( .A1(n4387), .A2(n1924), .ZN(n2068) );
CKND2D0 U2868 ( .A1(n1925), .A2(n2565), .ZN(n2146) );
ND4D0 U2869 ( .A1(n2147), .A2(n2069), .A3(n2082), .A4(n2146),
    .ZN(n1980) );
NR2D0 U2870 ( .A1(n1926), .A2(n2351), .ZN(n1927) );
AOI22D0 U2871 ( .A1(n2654), .A2(n1927), .B1(n2354), .B2(n1949
   ), .ZN(n1930)
       );
```

```
  NR2D0 U2872 ( .A1(n3441), .A2(n2351), .ZN(n1928) );
  NR2D0 U2873 ( .A1(n2270), .A2(n1941), .ZN(n2562) );
  NR4D0 U2874 ( .A1(n1960), .A2(n1928), .A3(n2653), .A4(n2562),
      .ZN(n1929) );
  IIND4D0 U2875 ( .A1(n2068), .A2(n1980), .B1(n1930), .B2(n1929
     ), .ZN(n1959)
        );
  AOI22D0 U2876 ( .A1(n1932), .A2(n2315), .B1(n1931), .B2(n3743
     ), .ZN(n1935)
        );
  NR2D0 U2877 ( .A1(n2078), .A2(n2141), .ZN(n2027) );
  NR3D0 U2878 ( .A1(n4110), .A2(n1953), .A3(n1933), .ZN(n1934)
     );
  AOI31D0 U2879 ( .A1(n1935), .A2(n2027), .A3(n1934), .B(n2062)
     , .ZN(n1958) );
  CKND2D0 U2880 ( .A1(n2139), .A2(n2110), .ZN(n2133) );
  CKND2D0 U2881 ( .A1(n1937), .A2(n1936), .ZN(n1965) );
  OAI211D0 U2882 ( .A1(n1939), .A2(n1938), .B(n2147), .C(n1965)
     , .ZN(n1948) );
  AOI211D0 U2883 ( .A1(n2080), .A2(n2298), .B(n2042), .C(n1940)
     , .ZN(n1942) );
  NR2D0 U2884 ( .A1(n2092), .A2(n1941), .ZN(n1963) );
  INVD0 U2885 ( .I(n1963), .ZN(n2054) );
  OAI211D0 U2886 ( .A1(n2538), .A2(n3529), .B(n1942), .C(n2054)
     , .ZN(n1943) );
  AOI211D0 U2887 ( .A1(n2565), .A2(n1944), .B(n2107), .C(n1943)
     , .ZN(n1951) );
  AOI22D0 U2888 ( .A1(n3335), .A2(n1945), .B1(n2263), .B2(n2354
     ), .ZN(n2079)
        );
  AOI32D0 U2889 ( .A1(n1946), .A2(n1951), .A3(n2079), .B1(n2286
     ), .B2(n1951),
        .ZN(n1947) );
  NR4D0 U2890 ( .A1(n2176), .A2(n2133), .A3(n1948), .A4(n1947),
      .ZN(n1956) );
  INVD0 U2891 ( .I(n1969), .ZN(n2071) );
  NR2D0 U2892 ( .A1(n2422), .A2(n2281), .ZN(n2195) );
  NR2D0 U2893 ( .A1(n2155), .A2(n2195), .ZN(n2136) );
  AOI211D0 U2894 ( .A1(n2287), .A2(n1949), .B(n2156), .C(n1986)
     , .ZN(n1950) );
  ND4D0 U2895 ( .A1(n2071), .A2(n2136), .A3(n1951), .A4(n1950),
      .ZN(n1952) );
  AOI211D0 U2896 ( .A1(n1954), .A2(n3544), .B(n1953), .C(n1952)
     , .ZN(n1955) );
  OAI22D0 U2897 ( .A1(n1956), .A2(n2126), .B1(n1955), .B2(n2120
     ), .ZN(n1957)
        );
```

```
AOI211D0 U2898 ( .A1(n2173), .A2(n1959), .B(n1958), .C(n1957)
   , .ZN(n2005) );
NR3D0 U2899 ( .A1(n2142), .A2(n1960), .A3(n2168), .ZN(n2022)
   );
ND4D0 U2900 ( .A1(n2091), .A2(n2022), .A3(n2043), .A4(n2082),
    .ZN(n1968) );
NR2D0 U2901 ( .A1(n1962), .A2(n1961), .ZN(n2024) );
AOI31D0 U2902 ( .A1(n2024), .A2(n1973), .A3(n2146), .B(n1988)
   , .ZN(n1967) );
NR4D0 U2903 ( .A1(n2176), .A2(n1964), .A3(n1963), .A4(n2112),
    .ZN(n2122) );
AOI31D0 U2904 ( .A1(n2122), .A2(n2069), .A3(n1965), .B(n2120)
   , .ZN(n1966) );
AOI211D0 U2905 ( .A1(n2165), .A2(n1968), .B(n1967), .C(n1966)
   , .ZN(n1971) );
NR4D0 U2906 ( .A1(n2141), .A2(n1995), .A3(n1975), .A4(n1969),
    .ZN(n1970) );
AOI32D0 U2907 ( .A1(n2091), .A2(n1971), .A3(n1970), .B1(n2062
   ), .B2(n1971),
      .ZN(n1985) );
INR4D0 U2908 ( .A1(n2154), .B1(n1972), .B2(n2131), .B3(n2133)
   , .ZN(n1983) );
NR2D0 U2909 ( .A1(n2149), .A2(n2060), .ZN(n2137) );
IND4D0 U2910 ( .A1(n1975), .B1(n1974), .B2(n2137), .B3(n1973)
   , .ZN(n1978) );
AOI211D0 U2911 ( .A1(n2032), .A2(n2454), .B(n2224), .C(n2150)
   , .ZN(n1976) );
AOI21D0 U2912 ( .A1(n1976), .A2(n2147), .B(n2120), .ZN(n1977)
    );
AOI221D0 U2913 ( .A1(n1980), .A2(n2165), .B1(n1978), .B2(
   n2165), .C(n1977),
      .ZN(n1982) );
CKND2D0 U2914 ( .A1(n1979), .A2(n2138), .ZN(n2034) );
OAI21D0 U2915 ( .A1(n2034), .A2(n1980), .B(n2173), .ZN(n1981)
    );
OAI211D0 U2916 ( .A1(n1983), .A2(n2062), .B(n1982), .C(n1981)
   , .ZN(n1984) );
AOI22D0 U2917 ( .A1(n4492), .A2(n1985), .B1(n3707), .B2(n1984
   ), .ZN(n2004)
       );
INVD0 U2918 ( .I(n1986), .ZN(n2157) );
OAI21D0 U2919 ( .A1(n1987), .A2(n2354), .B(n2124), .ZN(n1989)
    );
AOI31D0 U2920 ( .A1(n1990), .A2(n2157), .A3(n1989), .B(n1988)
   , .ZN(n2002) );
CKND2D0 U2921 ( .A1(n1991), .A2(n2110), .ZN(n2085) );
CKND2D0 U2922 ( .A1(n2111), .A2(n2054), .ZN(n2175) );
NR2D0 U2923 ( .A1(n2175), .A2(n1992), .ZN(n2035) );
```

```
INVD0 U2924 ( .I(n2035), .ZN(n2166) );
NR3D0 U2925 ( .A1(n2141), .A2(n2085), .A3(n2166), .ZN(n1994)
   );
AOI31D0 U2926 ( .A1(n1994), .A2(n4167), .A3(n1993), .B(n2062)
   , .ZN(n2001) );
CKND2D0 U2927 ( .A1(n2147), .A2(n2073), .ZN(n2009) );
NR2D0 U2928 ( .A1(n2009), .A2(n1995), .ZN(n2158) );
AN4D0 U2929 ( .A1(n2038), .A2(n2011), .A3(n2158), .A4(n1996),
    .Z(n1999) );
IINR4D0 U2930 ( .A1(n2069), .A2(n2024), .B1(n2078), .B2(n1997
   ), .ZN(n1998)
      );
OAI22D0 U2931 ( .A1(n1999), .A2(n2120), .B1(n1998), .B2(n2126
   ), .ZN(n2000)
      );
OAI31D0 U2932 ( .A1(n2002), .A2(n2001), .A3(n2000), .B(n3709)
   , .ZN(n2003) );
OAI211D0 U2933 ( .A1(n2005), .A2(n4129), .B(n2004), .C(n2003)
   , .ZN(n2187) );
INVD0 U2934 ( .I(n2040), .ZN(n2153) );
ND4D0 U2935 ( .A1(n2153), .A2(n2007), .A3(n2069), .A4(n2006),
    .ZN(n2020) );
CKND2D0 U2936 ( .A1(n2008), .A2(n2153), .ZN(n2047) );
NR4D0 U2937 ( .A1(n2010), .A2(n2085), .A3(n2047), .A4(n2009),
    .ZN(n2018) );
INVD0 U2938 ( .I(n2175), .ZN(n2084) );
CKND2D0 U2939 ( .A1(n2153), .A2(n2084), .ZN(n2094) );
OAI31D0 U2940 ( .A1(n2068), .A2(n2010), .A3(n2094), .B(n2164)
   , .ZN(n2017) );
ND3D0 U2941 ( .A1(n2011), .A2(n2065), .A3(n2136), .ZN(n2012)
   );
OAI31D0 U2942 ( .A1(n2015), .A2(n2014), .A3(n2012), .B(n2173)
   , .ZN(n2016) );
OAI211D0 U2943 ( .A1(n2018), .A2(n2120), .B(n2017), .C(n2016)
   , .ZN(n2019) );
AOI221D0 U2944 ( .A1(n2176), .A2(n2165), .B1(n2020), .B2(
   n2165), .C(n2019),
      .ZN(n2185) );
INVD0 U2945 ( .I(n2021), .ZN(n2134) );
ND4D0 U2946 ( .A1(n2052), .A2(n2022), .A3(n2056), .A4(n2169),
    .ZN(n2023) );
NR2D0 U2947 ( .A1(n2134), .A2(n2023), .ZN(n2031) );
NR2D0 U2948 ( .A1(n2042), .A2(n2060), .ZN(n2132) );
CKND2D0 U2949 ( .A1(n2032), .A2(n2355), .ZN(n2278) );
ND4D0 U2950 ( .A1(n2132), .A2(n2069), .A3(n2278), .A4(n2146),
    .ZN(n2106) );
ND4D0 U2951 ( .A1(n2025), .A2(n2152), .A3(n2024), .A4(n2169),
    .ZN(n2026) );
```

```
NR2D0 U2952 ( .A1(n2106), .A2(n2026), .ZN(n2030) );
ND4D0 U2953 ( .A1(n2132), .A2(n2027), .A3(n2157), .A4(n2110),
    .ZN(n2028) );
INR2D0 U2954 ( .A1(n2091), .B1(n2028), .ZN(n2029) );
OAI222D0 U2955 ( .A1(n2120), .A2(n2031), .B1(n2126), .B2(
   n2030), .C1(n2062),
      .C2(n2029), .ZN(n2105) );
CKND2D0 U2956 ( .A1(n2032), .A2(n2354), .ZN(n2220) );
IND4D0 U2957 ( .A1(n2034), .B1(n2153), .B2(n2033), .B3(n2220)
   , .ZN(n2037) );
ND4D0 U2958 ( .A1(n2132), .A2(n2035), .A3(n2055), .A4(n2465),
    .ZN(n2036) );
AOI22D0 U2959 ( .A1(n2162), .A2(n2037), .B1(n2165), .B2(n2036
   ), .ZN(n2050)
      );
INVD0 U2960 ( .I(n2085), .ZN(n2051) );
IND4D0 U2961 ( .A1(n2123), .B1(n2051), .B2(n2039), .B3(n2038)
   , .ZN(n2041) );
OAI31D0 U2962 ( .A1(n2042), .A2(n2041), .A3(n2040), .B(n2173)
   , .ZN(n2049) );
CKND2D0 U2963 ( .A1(n2146), .A2(n2054), .ZN(n2046) );
CKND2D0 U2964 ( .A1(n2043), .A2(n2128), .ZN(n2117) );
INVD0 U2965 ( .I(n2117), .ZN(n2171) );
OAI211D0 U2966 ( .A1(n2271), .A2(n2422), .B(n2171), .C(n2044)
   , .ZN(n2045) );
OAI31D0 U2967 ( .A1(n2047), .A2(n2046), .A3(n2045), .B(n2164)
   , .ZN(n2048) );
AOI31D0 U2968 ( .A1(n2050), .A2(n2049), .A3(n2048), .B(n4535)
   , .ZN(n2104) );
ND4D0 U2969 ( .A1(n2052), .A2(n2051), .A3(n2111), .A4(n3497),
    .ZN(n2067) );
AN4D0 U2970 ( .A1(n2396), .A2(n2146), .A3(n2054), .A4(n2053),
    .Z(n2108) );
NR2D0 U2971 ( .A1(n3821), .A2(n3822), .ZN(n4155) );
INVD0 U2972 ( .I(n4155), .ZN(n3793) );
ND4D0 U2973 ( .A1(n2159), .A2(n2108), .A3(n2073), .A4(n3793),
    .ZN(n2059) );
ND4D0 U2974 ( .A1(n2056), .A2(n2055), .A3(n2170), .A4(n2128),
    .ZN(n2074) );
INVD0 U2975 ( .I(n2074), .ZN(n2057) );
AOI31D0 U2976 ( .A1(n2057), .A2(n2146), .A3(n2873), .B(n2120)
   , .ZN(n2058) );
AOI221D0 U2977 ( .A1(n2133), .A2(n2173), .B1(n2059), .B2(
   n2173), .C(n2058),
      .ZN(n2064) );
INR4D0 U2978 ( .A1(n2170), .B1(n2061), .B2(n2060), .B3(n2085)
   , .ZN(n2063) );
```

```
AOI32D0 U2979 ( .A1(n2065), .A2(n2064), .A3(n2063), .B1(n2062
   ), .B2(n2064),
      .ZN(n2066) );
AOI221D0 U2980 ( .A1(n2068), .A2(n2165), .B1(n2067), .B2(
   n2165), .C(n2066),
      .ZN(n2102) );
ND4D0 U2981 ( .A1(n2138), .A2(n2071), .A3(n2070), .A4(n2069),
    .ZN(n2100) );
CKND2D0 U2982 ( .A1(n2073), .A2(n2072), .ZN(n2075) );
CKND2D0 U2983 ( .A1(n2153), .A2(n2137), .ZN(n2113) );
NR4D0 U2984 ( .A1(n2076), .A2(n2075), .A3(n2074), .A4(n2113),
    .ZN(n2098) );
INR4D0 U2985 ( .A1(n2079), .B1(n2078), .B2(n2156), .B3(n2077)
   , .ZN(n2083) );
CKND2D0 U2986 ( .A1(n2321), .A2(n2080), .ZN(n2081) );
ND4D0 U2987 ( .A1(n2084), .A2(n2083), .A3(n2082), .A4(n2081),
    .ZN(n2086) );
OAI31D0 U2988 ( .A1(n2087), .A2(n2086), .A3(n2085), .B(n2173)
   , .ZN(n2097) );
INVD0 U2989 ( .I(n2088), .ZN(n4145) );
OAI31D0 U2990 ( .A1(n4145), .A2(n2264), .A3(n2260), .B(n2089)
   , .ZN(n2095) );
AOI32D0 U2991 ( .A1(n2092), .A2(n2091), .A3(n2090), .B1(n2568
   ), .B2(n2091),
      .ZN(n2093) );
OAI31D0 U2992 ( .A1(n2095), .A2(n2094), .A3(n2093), .B(n2164)
   , .ZN(n2096) );
OAI211D0 U2993 ( .A1(n2098), .A2(n2120), .B(n2097), .C(n2096)
   , .ZN(n2099) );
AOI221D0 U2994 ( .A1(n2150), .A2(n2165), .B1(n2100), .B2(
   n2165), .C(n2099),
      .ZN(n2101) );
OAI22D0 U2995 ( .A1(n2102), .A2(n4148), .B1(n2101), .B2(n3777
   ), .ZN(n2103)
      );
AOI211D0 U2996 ( .A1(n4490), .A2(n2105), .B(n2104), .C(n2103)
   , .ZN(n2184) );
NR4D0 U2997 ( .A1(n2224), .A2(n2107), .A3(n2106), .A4(n2117),
    .ZN(n2121) );
CKND2D0 U2998 ( .A1(n2109), .A2(n2108), .ZN(n2115) );
IIND4D0 U2999 ( .A1(n2113), .A2(n2112), .B1(n2111), .B2(n2110
   ), .ZN(n2114)
      );
AOI22D0 U3000 ( .A1(n2164), .A2(n2115), .B1(n2165), .B2(n2114
   ), .ZN(n2119)
      );
OAI31D0 U3001 ( .A1(n2176), .A2(n2117), .A3(n2116), .B(n2173)
   , .ZN(n2118) );
```

```
OAI211D0 U3002 ( .A1(n2121), .A2(n2120), .B(n2119), .C(n2118)
   , .ZN(n2182) );
OAI211D0 U3003 ( .A1(n2540), .A2(n2193), .B(n2122), .C(n2169)
   , .ZN(n2130) );
AOI211D0 U3004 ( .A1(n2125), .A2(n2124), .B(n4236), .C(n2123)
   , .ZN(n2127) );
AOI31D0 U3005 ( .A1(n2132), .A2(n2128), .A3(n2127), .B(n2126)
   , .ZN(n2129) );
AOI221D0 U3006 ( .A1(n2131), .A2(n2162), .B1(n2130), .B2(
   n2162), .C(n2129),
      .ZN(n2145) );
INVD0 U3007 ( .I(n2132), .ZN(n2135) );
OAI31D0 U3008 ( .A1(n2135), .A2(n2134), .A3(n2133), .B(n2164)
   , .ZN(n2144) );
ND4D0 U3009 ( .A1(n2139), .A2(n2138), .A3(n2137), .A4(n2136),
    .ZN(n2140) );
OAI31D0 U3010 ( .A1(n2142), .A2(n2141), .A3(n2140), .B(n2173)
   , .ZN(n2143) );
AOI31D0 U3011 ( .A1(n2145), .A2(n2144), .A3(n2143), .B(n4410)
   , .ZN(n2181) );
CKND2D0 U3012 ( .A1(n2147), .A2(n2146), .ZN(n2148) );
NR4D0 U3013 ( .A1(n2150), .A2(n2426), .A3(n2149), .A4(n2148),
    .ZN(n2151) );
ND4D0 U3014 ( .A1(n2154), .A2(n2153), .A3(n2152), .A4(n2151),
    .ZN(n2163) );
NR2D0 U3015 ( .A1(n2156), .A2(n2155), .ZN(n2160) );
ND4D0 U3016 ( .A1(n2160), .A2(n2159), .A3(n2158), .A4(n2157),
    .ZN(n2161) );
AOI22D0 U3017 ( .A1(n2164), .A2(n2163), .B1(n2162), .B2(n2161
   ), .ZN(n2179)
      );
OAI31D0 U3018 ( .A1(n2168), .A2(n2167), .A3(n2166), .B(n2165)
   , .ZN(n2178) );
ND4D0 U3019 ( .A1(n2172), .A2(n2171), .A3(n2170), .A4(n2169),
    .ZN(n2174) );
OAI31D0 U3020 ( .A1(n2176), .A2(n2175), .A3(n2174), .B(n2173)
   , .ZN(n2177) );
AOI31D0 U3021 ( .A1(n2179), .A2(n2178), .A3(n2177), .B(n4408)
   , .ZN(n2180) );
AOI211D0 U3022 ( .A1(n4063), .A2(n2182), .B(n2181), .C(n2180)
   , .ZN(n2183) );
OAI211D0 U3023 ( .A1(n2185), .A2(n4190), .B(n2184), .C(n2183)
   , .ZN(n2186) );
NR4D0 U3024 ( .A1(n2189), .A2(n2188), .A3(n2187), .A4(n2186),
    .ZN(n2190) );
OAI222D0 U3025 ( .A1(\q_out[3]_BAR ), .A2(n4545), .B1(n4544),
    .B2(n2191),
      .C1(n4542), .C2(n2190), .ZN(n4815) );
```

```
  NR2D0 U3026 ( .A1(n3757), .A2(n2422), .ZN(n2277) );
  CKND2D0 U3027 ( .A1(n2354), .A2(n2277), .ZN(n2475) );
  INVD0 U3028 ( .I(n2475), .ZN(n2209) );
  CKND2D0 U3029 ( .A1(n2321), .A2(n2277), .ZN(n2494) );
  NR2D0 U3030 ( .A1(n2193), .A2(counter[4]), .ZN(n2455) );
  CKND2D0 U3031 ( .A1(n2565), .A2(n2455), .ZN(n2373) );
  CKND2D0 U3032 ( .A1(n2494), .A2(n2373), .ZN(n2479) );
  INVD0 U3033 ( .I(n2479), .ZN(n2246) );
  CKND2D0 U3034 ( .A1(n2544), .A2(n2277), .ZN(n2578) );
  CKND2D0 U3035 ( .A1(n2354), .A2(n2455), .ZN(n2429) );
  CKAN2D0 U3036 ( .A1(n2578), .A2(n2429), .Z(n2259) );
  CKND2D0 U3037 ( .A1(n2246), .A2(n2259), .ZN(n2585) );
  NR2D0 U3038 ( .A1(n2193), .A2(n2192), .ZN(n2405) );
  INVD0 U3039 ( .I(n2277), .ZN(n2336) );
  NR2D0 U3040 ( .A1(n2526), .A2(n2336), .ZN(n2404) );
  NR2D0 U3041 ( .A1(n2422), .A2(counter[4]), .ZN(n2322) );
  INVD0 U3042 ( .I(n2322), .ZN(n2208) );
  NR2D0 U3043 ( .A1(n2540), .A2(n2208), .ZN(n2268) );
  NR2D0 U3044 ( .A1(n2404), .A2(n2268), .ZN(n2591) );
  CKND2D0 U3045 ( .A1(n2287), .A2(n2277), .ZN(n2511) );
  OAI211D0 U3046 ( .A1(n2424), .A2(n4428), .B(n2591), .C(n2511)
     , .ZN(n2584) );
  NR2D0 U3047 ( .A1(n2405), .A2(n2584), .ZN(n2483) );
  CKND2D0 U3048 ( .A1(n2321), .A2(n2455), .ZN(n2219) );
  CKND2D0 U3049 ( .A1(n2320), .A2(n2322), .ZN(n2391) );
  CKND2D0 U3050 ( .A1(n2219), .A2(n2391), .ZN(n2301) );
  INVD0 U3051 ( .I(n2301), .ZN(n2512) );
  NR2D0 U3052 ( .A1(n2423), .A2(n2208), .ZN(n2503) );
  INVD0 U3053 ( .I(n2503), .ZN(n2367) );
  NR2D0 U3054 ( .A1(n2526), .A2(n2208), .ZN(n2273) );
  INVD0 U3055 ( .I(n2273), .ZN(n2546) );
  ND4D0 U3056 ( .A1(n2483), .A2(n2512), .A3(n2367), .A4(n2546),
      .ZN(n2194) );
  NR4D0 U3057 ( .A1(n2224), .A2(n2209), .A3(n2585), .A4(n2194),
      .ZN(n2204) );
  NR3D0 U3058 ( .A1(counter[11]), .A2(n2568), .A3(n4476), .ZN(
     n2411) );
  NR2D0 U3059 ( .A1(n2411), .A2(n2273), .ZN(n2223) );
  INVD0 U3060 ( .I(n2223), .ZN(n2478) );
  NR3D0 U3061 ( .A1(n4428), .A2(n2271), .A3(counter[11]), .ZN(
     n2412) );
  NR2D0 U3062 ( .A1(n2478), .A2(n2412), .ZN(n2557) );
  CKND2D0 U3063 ( .A1(n2511), .A2(n2557), .ZN(n2313) );
  CKND2D0 U3064 ( .A1(n2355), .A2(n2455), .ZN(n2395) );
  INVD0 U3065 ( .I(n2395), .ZN(n2516) );
  NR2D0 U3066 ( .A1(n2404), .A2(n2516), .ZN(n2495) );
  INVD0 U3067 ( .I(n2429), .ZN(n2385) );
  NR2D0 U3068 ( .A1(n2385), .A2(n2268), .ZN(n2243) );
```

```
  CKND2D0 U3069 ( .A1(n2355), .A2(n2277), .ZN(n2456) );
  INVD0 U3070 ( .I(n2456), .ZN(n2504) );
  INVD0 U3071 ( .I(n2219), .ZN(n2407) );
  NR2D0 U3072 ( .A1(n2195), .A2(n2503), .ZN(n2434) );
  NR2D0 U3073 ( .A1(n4428), .A2(n3009), .ZN(n2595) );
  INVD0 U3074 ( .I(n2595), .ZN(n2368) );
  CKND2D0 U3075 ( .A1(n2434), .A2(n2368), .ZN(n2486) );
  CKND2D0 U3076 ( .A1(n2287), .A2(n2455), .ZN(n2376) );
  CKND2D0 U3077 ( .A1(counter[4]), .A2(n2426), .ZN(n2410) );
  CKND2D0 U3078 ( .A1(n2376), .A2(n2410), .ZN(n2331) );
  NR4D0 U3079 ( .A1(n2504), .A2(n2407), .A3(n2486), .A4(n2331),
      .ZN(n2196) );
  IND4D0 U3080 ( .A1(n2313), .B1(n2495), .B2(n2243), .B3(n2196)
     , .ZN(n2200) );
  NR2D0 U3081 ( .A1(n4428), .A2(n3837), .ZN(n2559) );
  INVD0 U3082 ( .I(n2559), .ZN(n2258) );
  CKND2D0 U3083 ( .A1(n2456), .A2(n2258), .ZN(n2384) );
  CKND2D0 U3084 ( .A1(counter[5]), .A2(n4037), .ZN(n2413) );
  INVD0 U3085 ( .I(n2413), .ZN(n2210) );
  AOI21D0 U3086 ( .A1(n2322), .A2(n2454), .B(n2210), .ZN(n2272)
      );
  NR2D0 U3087 ( .A1(n3757), .A2(n2505), .ZN(n2480) );
  NR2D0 U3088 ( .A1(n2480), .A2(n2209), .ZN(n2451) );
  CKND2D0 U3089 ( .A1(n2320), .A2(n2455), .ZN(n2244) );
  NR2D0 U3090 ( .A1(n2281), .A2(n2336), .ZN(n2531) );
  CKND2D0 U3091 ( .A1(n2544), .A2(n2455), .ZN(n2226) );
  INVD0 U3092 ( .I(n2226), .ZN(n2436) );
  NR2D0 U3093 ( .A1(n2531), .A2(n2436), .ZN(n2554) );
  CKND2D0 U3094 ( .A1(n2554), .A2(n2223), .ZN(n2582) );
  CKND2D0 U3095 ( .A1(n2197), .A2(n2511), .ZN(n2302) );
  INR4D0 U3096 ( .A1(n2244), .B1(n2503), .B2(n2582), .B3(n2302)
     , .ZN(n2198) );
  IND4D0 U3097 ( .A1(n2384), .B1(n2272), .B2(n2451), .B3(n2198)
     , .ZN(n2199) );
  AOI22D0 U3098 ( .A1(n3443), .A2(n2200), .B1(n2508), .B2(n2199
     ), .ZN(n2203)
       );
  CKND2D0 U3099 ( .A1(n2494), .A2(n2367), .ZN(n2403) );
  INVD0 U3100 ( .I(n2403), .ZN(n2466) );
  NR2D0 U3101 ( .A1(n2271), .A2(n2208), .ZN(n2276) );
  NR2D0 U3102 ( .A1(n2276), .A2(n2480), .ZN(n2499) );
  INVD0 U3103 ( .I(n2499), .ZN(n2387) );
  NR2D0 U3104 ( .A1(n2404), .A2(n2387), .ZN(n2477) );
  ND3D0 U3105 ( .A1(n2466), .A2(n2477), .A3(n2410), .ZN(n2201)
     );
  OAI31D0 U3106 ( .A1(n2213), .A2(n2436), .A3(n2201), .B(n3114)
     , .ZN(n2202) );
```

```
OAI211D0 U3107 ( .A1(n2204), .A2(n2987), .B(n2203), .C(n2202)
   , .ZN(n2613) );
INVD0 U3108 ( .I(n2412), .ZN(n2229) );
CKND2D0 U3109 ( .A1(n2229), .A2(n2376), .ZN(n2502) );
INVD0 U3110 ( .I(n2502), .ZN(n2467) );
CKND2D0 U3111 ( .A1(n2355), .A2(n2322), .ZN(n2561) );
CKAN2D0 U3112 ( .A1(n2511), .A2(n2561), .Z(n2449) );
CKND2D0 U3113 ( .A1(n2467), .A2(n2449), .ZN(n2239) );
CKND2D0 U3114 ( .A1(counter[4]), .A2(n2224), .ZN(n2545) );
ND3D0 U3115 ( .A1(n2391), .A2(n2545), .A3(n2246), .ZN(n2221)
   );
NR2D0 U3116 ( .A1(n2796), .A2(n2436), .ZN(n2205) );
IIND4D0 U3117 ( .A1(n2239), .A2(n2221), .B1(n2205), .B2(n2495
   ), .ZN(n2218)
      );
CKND2D0 U3118 ( .A1(n2262), .A2(n2277), .ZN(n2464) );
ND3D0 U3119 ( .A1(n2396), .A2(n2464), .A3(n2449), .ZN(n2589)
   );
INVD0 U3120 ( .I(n2589), .ZN(n2207) );
CKND2D0 U3121 ( .A1(n2246), .A2(n2434), .ZN(n2418) );
NR3D0 U3122 ( .A1(n2405), .A2(n2210), .A3(n2418), .ZN(n2206)
   );
AOI31D0 U3123 ( .A1(n2207), .A2(n2451), .A3(n2206), .B(n3349)
   , .ZN(n2217) );
INVD0 U3124 ( .I(n2464), .ZN(n2444) );
INVD0 U3125 ( .I(n2477), .ZN(n2227) );
NR3D0 U3126 ( .A1(n2504), .A2(n2444), .A3(n2227), .ZN(n2393)
   );
NR2D0 U3127 ( .A1(n2568), .A2(n2208), .ZN(n2517) );
INVD0 U3128 ( .I(n2517), .ZN(n2230) );
ND3D0 U3129 ( .A1(n2456), .A2(n2230), .A3(n2391), .ZN(n2402)
   );
NR2D0 U3130 ( .A1(n2402), .A2(n2331), .ZN(n2580) );
CKND2D0 U3131 ( .A1(n2554), .A2(n2580), .ZN(n2374) );
ND4D0 U3132 ( .A1(n2243), .A2(n2413), .A3(n2494), .A4(n2545),
    .ZN(n2310) );
OAI31D0 U3133 ( .A1(n2209), .A2(n2374), .A3(n2310), .B(n3114)
   , .ZN(n2215) );
INVD0 U3134 ( .I(n2391), .ZN(n2212) );
INVD0 U3135 ( .I(n2545), .ZN(n2305) );
NR2D0 U3136 ( .A1(n2210), .A2(n2305), .ZN(n2211) );
CKND2D0 U3137 ( .A1(n2211), .A2(n2546), .ZN(n2421) );
CKND2D0 U3138 ( .A1(n2578), .A2(n2244), .ZN(n2285) );
NR4D0 U3139 ( .A1(n2213), .A2(n2212), .A3(n2421), .A4(n2285),
    .ZN(n2214) );
AOI32D0 U3140 ( .A1(n2393), .A2(n2215), .A3(n2214), .B1(n2596
   ), .B2(n2215),
      .ZN(n2216) );
```

```
AOI211D0 U3141 ( .A1(n2577), .A2(n2218), .B(n2217), .C(n2216)
   , .ZN(n2257) );
NR2D0 U3142 ( .A1(n2405), .A2(n2331), .ZN(n2474) );
CKND2D0 U3143 ( .A1(n2220), .A2(n2219), .ZN(n2553) );
NR4D0 U3144 ( .A1(n2444), .A2(n2553), .A3(n2221), .A4(n2313),
    .ZN(n2222) );
AOI21D0 U3145 ( .A1(n2474), .A2(n2222), .B(n3349), .ZN(n2238)
    );
NR2D0 U3146 ( .A1(n2517), .A2(n2412), .ZN(n2445) );
CKND2D0 U3147 ( .A1(n2259), .A2(n2223), .ZN(n2507) );
CKND2D0 U3148 ( .A1(n2321), .A2(n2322), .ZN(n2428) );
CKND2D0 U3149 ( .A1(n2391), .A2(n2428), .ZN(n2588) );
NR2D0 U3150 ( .A1(n2224), .A2(n2516), .ZN(n2225) );
IND2D0 U3151 ( .A1(n2588), .B1(n2225), .ZN(n2232) );
CKND2D0 U3152 ( .A1(n2226), .A2(n2561), .ZN(n2248) );
NR4D0 U3153 ( .A1(n2227), .A2(n2507), .A3(n2232), .A4(n2248),
    .ZN(n2228) );
AOI31D0 U3154 ( .A1(n2445), .A2(n2228), .A3(n2373), .B(n2987)
   , .ZN(n2237) );
CKND2D0 U3155 ( .A1(n2449), .A2(n2545), .ZN(n2500) );
CKND2D0 U3156 ( .A1(n2229), .A2(n2546), .ZN(n2592) );
NR2D0 U3157 ( .A1(n2268), .A2(n2585), .ZN(n2409) );
CKND2D0 U3158 ( .A1(counter[1]), .A2(n2405), .ZN(n2440) );
CKND2D0 U3159 ( .A1(n2410), .A2(n2440), .ZN(n2233) );
NR2D0 U3160 ( .A1(n2504), .A2(n2233), .ZN(n2534) );
CKND2D0 U3161 ( .A1(n2409), .A2(n2534), .ZN(n2304) );
CKND2D0 U3162 ( .A1(n2230), .A2(n2395), .ZN(n2231) );
NR4D0 U3163 ( .A1(n2500), .A2(n2592), .A3(n2304), .A4(n2231),
    .ZN(n2235) );
NR2D0 U3164 ( .A1(n2368), .A2(counter[0]), .ZN(n2571) );
INVD0 U3165 ( .I(n2571), .ZN(n2390) );
CKND2D0 U3166 ( .A1(n2390), .A2(n2413), .ZN(n2314) );
NR2D0 U3167 ( .A1(n2268), .A2(n2314), .ZN(n2498) );
INVD0 U3168 ( .I(n2276), .ZN(n2358) );
CKND2D0 U3169 ( .A1(n2498), .A2(n2358), .ZN(n2514) );
NR4D0 U3170 ( .A1(n2412), .A2(n2233), .A3(n2232), .A4(n2514),
    .ZN(n2234) );
OAI22D0 U3171 ( .A1(n2235), .A2(n3439), .B1(n2234), .B2(n2596
   ), .ZN(n2236)
       );
OAI31D0 U3172 ( .A1(n2238), .A2(n2237), .A3(n2236), .B(n4521)
   , .ZN(n2256) );
IND2D0 U3173 ( .A1(n2531), .B1(n2440), .ZN(n2433) );
NR4D0 U3174 ( .A1(n2796), .A2(n2411), .A3(n2433), .A4(n2239),
    .ZN(n2240) );
AOI31D0 U3175 ( .A1(n2259), .A2(n2240), .A3(n2413), .B(n3349)
   , .ZN(n2254) );
NR2D0 U3176 ( .A1(n2405), .A2(n2502), .ZN(n2392) );
```

```
  NR2D0 U3177 ( .A1(n2531), .A2(n2571), .ZN(n2476) );
  INVD0 U3178 ( .I(n2428), .ZN(n2303) );
  INR4D0 U3179 ( .A1(n2476), .B1(n2241), .B2(n2444), .B3(n2303)
     , .ZN(n2242) );
  AOI31D0 U3180 ( .A1(n2243), .A2(n2392), .A3(n2242), .B(n2987)
     , .ZN(n2253) );
  INVD0 U3181 ( .I(n2421), .ZN(n2251) );
  CKND2D0 U3182 ( .A1(n2475), .A2(n2440), .ZN(n2593) );
  CKND2D0 U3183 ( .A1(n2244), .A2(n2428), .ZN(n2386) );
  NR3D0 U3184 ( .A1(n2386), .A2(n2385), .A3(n2444), .ZN(n2497)
     );
  INVD0 U3185 ( .I(n2497), .ZN(n2420) );
  CKND2D0 U3186 ( .A1(n2410), .A2(n2476), .ZN(n2398) );
  INVD0 U3187 ( .I(n2398), .ZN(n2484) );
  AOI211D0 U3188 ( .A1(n2322), .A2(n2454), .B(n2503), .C(n2407)
     , .ZN(n2245) );
  ND4D0 U3189 ( .A1(n2484), .A2(n2246), .A3(n2245), .A4(n2258),
      .ZN(n2247) );
  OAI31D0 U3190 ( .A1(n2593), .A2(n2420), .A3(n2247), .B(n2508)
     , .ZN(n2250) );
  NR4D0 U3191 ( .A1(n2384), .A2(n2479), .A3(n2285), .A4(n2248),
      .ZN(n2249) );
  AOI32D0 U3192 ( .A1(n2251), .A2(n2250), .A3(n2249), .B1(n3439
     ), .B2(n2250),
        .ZN(n2252) );
  OAI31D0 U3193 ( .A1(n2254), .A2(n2253), .A3(n2252), .B(n3437)
     , .ZN(n2255) );
  OAI211D0 U3194 ( .A1(n2257), .A2(n4298), .B(n2256), .C(n2255)
     , .ZN(n2612) );
  NR2D0 U3195 ( .A1(n2517), .A2(n2553), .ZN(n2587) );
  ND4D0 U3196 ( .A1(n2587), .A2(n2392), .A3(n2259), .A4(n2258),
      .ZN(n2297) );
  NR2D0 U3197 ( .A1(n3757), .A2(n2260), .ZN(n2543) );
  NR2D0 U3198 ( .A1(n2444), .A2(n2516), .ZN(n2506) );
  INVD0 U3199 ( .I(n2506), .ZN(n2583) );
  NR2D0 U3200 ( .A1(n2270), .A2(counter[4]), .ZN(n2564) );
  INVD0 U3201 ( .I(n2564), .ZN(n2539) );
  AOI21D0 U3202 ( .A1(n2568), .A2(n2271), .B(n2539), .ZN(n2261)
      );
  AOI211D0 U3203 ( .A1(n2262), .A2(n2543), .B(n2583), .C(n2261)
     , .ZN(n2549) );
  CKND2D0 U3204 ( .A1(n2263), .A2(n3757), .ZN(n2338) );
  OAI22D0 U3205 ( .A1(n2264), .A2(n2338), .B1(n4351), .B2(n2622
     ), .ZN(n2266)
        );
  OAI22D0 U3206 ( .A1(n2265), .A2(n2622), .B1(n2568), .B2(n4517
     ), .ZN(n2348)
        );
```

```
NR4D0 U3207 ( .A1(n2426), .A2(n2531), .A3(n2266), .A4(n2348),
    .ZN(n2267) );
CKND2D0 U3208 ( .A1(n2654), .A2(n2543), .ZN(n2282) );
IND4D0 U3209 ( .A1(n2268), .B1(n2549), .B2(n2267), .B3(n2282)
   , .ZN(n2570) );
CKND2D0 U3210 ( .A1(n4480), .A2(n4256), .ZN(n2288) );
OAI22D0 U3211 ( .A1(n2269), .A2(n2288), .B1(n2423), .B2(n2539
   ), .ZN(n2361)
       );
NR2D0 U3212 ( .A1(n3757), .A2(n2270), .ZN(n2528) );
INVD0 U3213 ( .I(n2528), .ZN(n2537) );
OAI21D0 U3214 ( .A1(n2271), .A2(n2537), .B(n2578), .ZN(n2328)
    );
OR2D0 U3215 ( .A1(n2338), .A2(n2423), .Z(n2324) );
INVD0 U3216 ( .I(n2272), .ZN(n2487) );
NR2D0 U3217 ( .A1(n2487), .A2(n2273), .ZN(n2473) );
OAI211D0 U3218 ( .A1(n2526), .A2(n2537), .B(n2324), .C(n2473)
   , .ZN(n2291) );
CKND2D0 U3219 ( .A1(n2615), .A2(n2274), .ZN(n2334) );
IND4D0 U3220 ( .A1(n2291), .B1(n2499), .B2(n2334), .B3(n2440)
   , .ZN(n2275) );
NR4D0 U3221 ( .A1(n2570), .A2(n2361), .A3(n2328), .A4(n2275),
    .ZN(n2295) );
NR2D0 U3222 ( .A1(n3390), .A2(n2288), .ZN(n2563) );
NR2D0 U3223 ( .A1(n2568), .A2(n2338), .ZN(n2280) );
AOI21D0 U3224 ( .A1(n2277), .A2(n2316), .B(n2276), .ZN(n2446)
    );
NR2D0 U3225 ( .A1(n2796), .A2(n2571), .ZN(n2496) );
ND4D0 U3226 ( .A1(n2446), .A2(n2496), .A3(n2483), .A4(n2278),
    .ZN(n2279) );
OAI31D0 U3227 ( .A1(n2563), .A2(n2280), .A3(n2279), .B(n3443)
   , .ZN(n2294) );
OAI22D0 U3228 ( .A1(n2568), .A2(n2338), .B1(n2281), .B2(n2539
   ), .ZN(n2552)
       );
INVD0 U3229 ( .I(n2552), .ZN(n2283) );
OAI211D0 U3230 ( .A1(n2539), .A2(n3955), .B(n2283), .C(n2282)
   , .ZN(n2284) );
NR2D0 U3231 ( .A1(n2285), .A2(n2284), .ZN(n2533) );
OAI211D0 U3232 ( .A1(n2286), .A2(n2537), .B(n2512), .C(n2533)
   , .ZN(n2292) );
INR2D0 U3233 ( .A1(n2543), .B1(n2568), .ZN(n2333) );
INVD0 U3234 ( .I(n2494), .ZN(n2443) );
AOI211D0 U3235 ( .A1(n2287), .A2(n2564), .B(n2563), .C(n2443)
   , .ZN(n2317) );
AOI22D0 U3236 ( .A1(n4351), .A2(n2568), .B1(n2288), .B2(n2537
   ), .ZN(n2289)
       );
```

```
NR2D0 U3237 ( .A1(n2559), .A2(n2516), .ZN(n2490) );
INVD0 U3238 ( .I(n2490), .ZN(n2469) );
NR3D0 U3239 ( .A1(n2404), .A2(n2289), .A3(n2469), .ZN(n2359)
   );
IIND4D0 U3240 ( .A1(n2333), .A2(n2405), .B1(n2317), .B2(n2359
   ), .ZN(n2290)
       );
OAI31D0 U3241 ( .A1(n2292), .A2(n2291), .A3(n2290), .B(n2508)
   , .ZN(n2293) );
OAI211D0 U3242 ( .A1(n2295), .A2(n2987), .B(n2294), .C(n2293)
   , .ZN(n2296) );
AOI221D0 U3243 ( .A1(n2589), .A2(n3114), .B1(n2297), .B2(
   n3114), .C(n2296),
      .ZN(n2610) );
CKND2D0 U3244 ( .A1(n2393), .A2(n2474), .ZN(n2370) );
AOI211D0 U3245 ( .A1(n2322), .A2(n2298), .B(n2478), .C(n2370)
   , .ZN(n2312) );
ND4D0 U3246 ( .A1(n2476), .A2(n2534), .A3(n2299), .A4(n2561),
    .ZN(n2309) );
OAI211D0 U3247 ( .A1(n4321), .A2(n4476), .B(n2474), .C(n2373)
   , .ZN(n2300) );
OAI31D0 U3248 ( .A1(n2302), .A2(n2301), .A3(n2300), .B(n2577)
   , .ZN(n2307) );
NR2D0 U3249 ( .A1(n2531), .A2(n2303), .ZN(n2369) );
INR4D0 U3250 ( .A1(n2369), .B1(n2305), .B2(n2502), .B3(n2304)
   , .ZN(n2306) );
AOI32D0 U3251 ( .A1(n2473), .A2(n2307), .A3(n2306), .B1(n3439
   ), .B2(n2307),
      .ZN(n2308) );
AOI221D0 U3252 ( .A1(n2310), .A2(n2508), .B1(n2309), .B2(
   n2508), .C(n2308),
      .ZN(n2311) );
AOI32D0 U3253 ( .A1(n2312), .A2(n2311), .A3(n2494), .B1(n3349
   ), .B2(n2311),
      .ZN(n2383) );
AOI211D0 U3254 ( .A1(n2315), .A2(n2455), .B(n2314), .C(n2313)
   , .ZN(n2579) );
AOI22D0 U3255 ( .A1(n2316), .A2(n2543), .B1(n2544), .B2(n4452
   ), .ZN(n2330)
       );
NR2D0 U3256 ( .A1(n2326), .A2(n2539), .ZN(n2319) );
INVD0 U3257 ( .I(n2317), .ZN(n2318) );
AOI211D0 U3258 ( .A1(n2320), .A2(n2543), .B(n2319), .C(n2318)
   , .ZN(n2325) );
OAI21D0 U3259 ( .A1(n2322), .A2(n2528), .B(n2321), .ZN(n2323)
    );
ND4D0 U3260 ( .A1(n2325), .A2(n2334), .A3(n2324), .A4(n2323),
    .ZN(n2535) );
```

```
AOI21D0 U3261 ( .A1(n2540), .A2(n2326), .B(n2537), .ZN(n2327)
    );
NR4D0 U3262 ( .A1(n2469), .A2(n2535), .A3(n2328), .A4(n2327),
    .ZN(n2329) );
ND4D0 U3263 ( .A1(n2451), .A2(n2579), .A3(n2330), .A4(n2329),
    .ZN(n2347) );
NR2D0 U3264 ( .A1(n2540), .A2(n2537), .ZN(n2332) );
INR4D0 U3265 ( .A1(n2334), .B1(n2333), .B2(n2332), .B3(n2331)
   , .ZN(n2536) );
OR2D0 U3266 ( .A1(n2411), .A2(n2503), .Z(n2447) );
AO211D0 U3267 ( .A1(n2565), .A2(n2543), .B(n2447), .C(n2571),
    .Z(n2335) );
AOI211D0 U3268 ( .A1(n2355), .A2(n2564), .B(n2562), .C(n2335)
   , .ZN(n2550) );
NR2D0 U3269 ( .A1(n2340), .A2(n2336), .ZN(n2594) );
NR4D0 U3270 ( .A1(n2337), .A2(n2405), .A3(n2594), .A4(n2420),
    .ZN(n2345) );
AOI21D0 U3271 ( .A1(n2354), .A2(n2543), .B(n2500), .ZN(n2360)
    );
INVD0 U3272 ( .I(n2360), .ZN(n2343) );
CKND2D0 U3273 ( .A1(n2544), .A2(n2564), .ZN(n2339) );
AOI22D0 U3274 ( .A1(n2341), .A2(n2340), .B1(n2339), .B2(n2338
   ), .ZN(n2342)
      );
AOI211D0 U3275 ( .A1(n2355), .A2(n2543), .B(n2343), .C(n2342)
   , .ZN(n2344) );
ND4D0 U3276 ( .A1(n2536), .A2(n2550), .A3(n2345), .A4(n2344),
    .ZN(n2346) );
AOI22D0 U3277 ( .A1(n3443), .A2(n2347), .B1(n3114), .B2(n2346
   ), .ZN(n2366)
      );
INVD0 U3278 ( .I(n2474), .ZN(n2353) );
AOI211D0 U3279 ( .A1(n4480), .A2(n3743), .B(n2384), .C(n2348)
   , .ZN(n2350) );
AN4D0 U3280 ( .A1(n2533), .A2(n3715), .A3(n2391), .A4(n2367),
    .Z(n2349) );
OAI211D0 U3281 ( .A1(n4474), .A2(n2351), .B(n2350), .C(n2349)
   , .ZN(n2352) );
OAI31D0 U3282 ( .A1(n2353), .A2(n2421), .A3(n2352), .B(n2577)
   , .ZN(n2365) );
CKND2D0 U3283 ( .A1(n2354), .A2(n2564), .ZN(n2357) );
CKND2D0 U3284 ( .A1(n2355), .A2(n2528), .ZN(n2356) );
ND4D0 U3285 ( .A1(n2358), .A2(n2413), .A3(n2357), .A4(n2356),
    .ZN(n2363) );
NR2D0 U3286 ( .A1(n2593), .A2(n2411), .ZN(n2513) );
INVD0 U3287 ( .I(n2513), .ZN(n2419) );
IIND4D0 U3288 ( .A1(n2361), .A2(n2592), .B1(n2360), .B2(n2359
   ), .ZN(n2362)
```

```
        );
OAI31D0 U3289 ( .A1(n2363), .A2(n2419), .A3(n2362), .B(n2508)
   , .ZN(n2364) );
AOI31D0 U3290 ( .A1(n2366), .A2(n2365), .A3(n2364), .B(n4277)
   , .ZN(n2382) );
IND4D0 U3291 ( .A1(n2370), .B1(n2369), .B2(n2368), .B3(n2367)
   , .ZN(n2372) );
AOI21D0 U3292 ( .A1(n2978), .A2(n2455), .B(n2386), .ZN(n2441)
    );
ND4D0 U3293 ( .A1(n2441), .A2(n2534), .A3(n2490), .A4(n3647),
    .ZN(n2371) );
AOI22D0 U3294 ( .A1(n3443), .A2(n2372), .B1(n2508), .B2(n2371
   ), .ZN(n2380)
        );
INVD0 U3295 ( .I(n2373), .ZN(n2527) );
OR4D0 U3296 ( .A1(n2374), .A2(n2584), .A3(n2444), .A4(n2595),
    .Z(n2375) );
OAI31D0 U3297 ( .A1(n2796), .A2(n2527), .A3(n2375), .B(n3114)
   , .ZN(n2379) );
CKND2D0 U3298 ( .A1(n2506), .A2(n2376), .ZN(n2406) );
ND4D0 U3299 ( .A1(n2554), .A2(n2499), .A3(n2591), .A4(n2579),
    .ZN(n2377) );
OAI21D0 U3300 ( .A1(n2406), .A2(n2377), .B(n2577), .ZN(n2378)
    );
AOI31D0 U3301 ( .A1(n2380), .A2(n2379), .A3(n2378), .B(n4535)
   , .ZN(n2381) );
AOI211D0 U3302 ( .A1(n4063), .A2(n2383), .B(n2382), .C(n2381)
   , .ZN(n2609) );
NR2D0 U3303 ( .A1(n2385), .A2(n2384), .ZN(n2567) );
ND4D0 U3304 ( .A1(n2554), .A2(n2567), .A3(n2587), .A4(n2465),
    .ZN(n2389) );
NR3D0 U3305 ( .A1(n2516), .A2(n2402), .A3(n2386), .ZN(n2450)
   );
NR2D0 U3306 ( .A1(n2387), .A2(n2433), .ZN(n2463) );
ND4D0 U3307 ( .A1(n2557), .A2(n2450), .A3(n2463), .A4(n2578),
    .ZN(n2388) );
AOI22D0 U3308 ( .A1(n3443), .A2(n2389), .B1(n2577), .B2(n2388
   ), .ZN(n2401)
        );
ND4D0 U3309 ( .A1(n2393), .A2(n2392), .A3(n2391), .A4(n2390),
    .ZN(n2394) );
OAI31D0 U3310 ( .A1(n2403), .A2(n2487), .A3(n2394), .B(n2508)
   , .ZN(n2400) );
INVD0 U3311 ( .I(n2409), .ZN(n2448) );
ND4D0 U3312 ( .A1(n2451), .A2(n2396), .A3(n2465), .A4(n2395),
    .ZN(n2397) );
OAI31D0 U3313 ( .A1(n2398), .A2(n2448), .A3(n2397), .B(n3114)
   , .ZN(n2399) );
```

```
AOI31D0 U3314 ( .A1(n2401), .A2(n2400), .A3(n2399), .B(n4325)
   , .ZN(n2607) );
NR2D0 U3315 ( .A1(n2402), .A2(n2500), .ZN(n2427) );
NR3D0 U3316 ( .A1(n2405), .A2(n2404), .A3(n2403), .ZN(n2581)
   );
IND4D0 U3317 ( .A1(n2406), .B1(n2427), .B2(n2581), .B3(n2475)
   , .ZN(n2417) );
INR4D0 U3318 ( .A1(n2445), .B1(n2407), .B2(n2411), .B3(n2583)
   , .ZN(n2408) );
AOI31D0 U3319 ( .A1(n2409), .A2(n2499), .A3(n2408), .B(n2987)
   , .ZN(n2416) );
ND3D0 U3320 ( .A1(n2434), .A2(n2410), .A3(n2561), .ZN(n2510)
   );
INR4D0 U3321 ( .A1(n2496), .B1(n2412), .B2(n2411), .B3(n2510)
   , .ZN(n2414) );
AOI31D0 U3322 ( .A1(n2414), .A2(n2413), .A3(n2578), .B(n3439)
   , .ZN(n2415) );
AOI211D0 U3323 ( .A1(n2508), .A2(n2417), .B(n2416), .C(n2415)
   , .ZN(n2472) );
NR4D0 U3324 ( .A1(n2421), .A2(n2420), .A3(n2419), .A4(n2418),
    .ZN(n2439) );
OAI211D0 U3325 ( .A1(n2423), .A2(n2422), .B(n2498), .C(n3477)
   , .ZN(n2432) );
NR2D0 U3326 ( .A1(n4428), .A2(n2424), .ZN(n2425) );
INR4D0 U3327 ( .A1(n2427), .B1(n2426), .B2(n2480), .B3(n2425)
   , .ZN(n2430) );
AOI31D0 U3328 ( .A1(n2430), .A2(n2429), .A3(n2428), .B(n2987)
   , .ZN(n2431) );
AOI221D0 U3329 ( .A1(n2433), .A2(n3443), .B1(n2432), .B2(
   n3443), .C(n2431),
      .ZN(n2438) );
IND4D0 U3330 ( .A1(n2480), .B1(n2449), .B2(n2434), .B3(n2546)
   , .ZN(n2435) );
OAI31D0 U3331 ( .A1(n2796), .A2(n2436), .A3(n2435), .B(n2508)
   , .ZN(n2437) );
OAI211D0 U3332 ( .A1(n2439), .A2(n3439), .B(n2438), .C(n2437)
   , .ZN(n2462) );
INVD0 U3333 ( .I(n2557), .ZN(n2529) );
ND4D0 U3334 ( .A1(n2567), .A2(n2441), .A3(n2545), .A4(n2440),
    .ZN(n2442) );
NR4D0 U3335 ( .A1(n2444), .A2(n2443), .A3(n2529), .A4(n2442),
    .ZN(n2460) );
IND4D0 U3336 ( .A1(n2486), .B1(n2534), .B2(n2446), .B3(n2445)
   , .ZN(n2453) );
NR2D0 U3337 ( .A1(n2448), .A2(n2447), .ZN(n2485) );
ND4D0 U3338 ( .A1(n2451), .A2(n2450), .A3(n2449), .A4(n2485),
    .ZN(n2452) );
```

```
AOI22D0 U3339 ( .A1(n2577), .A2(n2453), .B1(n3114), .B2(n2452
   ), .ZN(n2459)
      );
CKND2D0 U3340 ( .A1(n2455), .A2(n2454), .ZN(n2481) );
ND4D0 U3341 ( .A1(n2497), .A2(n3671), .A3(n2456), .A4(n2481),
    .ZN(n2457) );
OAI31D0 U3342 ( .A1(n2510), .A2(n2584), .A3(n2457), .B(n2508)
   , .ZN(n2458) );
OAI211D0 U3343 ( .A1(n2460), .A2(n3349), .B(n2459), .C(n2458)
   , .ZN(n2461) );
AOI22D0 U3344 ( .A1(n4492), .A2(n2462), .B1(n3852), .B2(n2461
   ), .ZN(n2471)
      );
INVD0 U3345 ( .I(n2463), .ZN(n2590) );
ND4D0 U3346 ( .A1(n2467), .A2(n2466), .A3(n2465), .A4(n2464),
    .ZN(n2468) );
OAI31D0 U3347 ( .A1(n2469), .A2(n2590), .A3(n2468), .B(n3443)
   , .ZN(n2470) );
AOI32D0 U3348 ( .A1(n2472), .A2(n2471), .A3(n2470), .B1(n3379
   ), .B2(n2471),
      .ZN(n2606) );
CKND2D0 U3349 ( .A1(n2474), .A2(n2473), .ZN(n2523) );
IND4D0 U3350 ( .A1(n2523), .B1(n2477), .B2(n2476), .B3(n2475)
   , .ZN(n2493) );
NR4D0 U3351 ( .A1(n2480), .A2(n2479), .A3(n2478), .A4(n2588),
    .ZN(n2482) );
AOI31D0 U3352 ( .A1(n2483), .A2(n2482), .A3(n2481), .B(n2596)
   , .ZN(n2492) );
AO31D0 U3353 ( .A1(n2485), .A2(n2484), .A3(n2483), .B(n3349),
    .Z(n2489) );
NR4D0 U3354 ( .A1(n2796), .A2(n2487), .A3(n2585), .A4(n2486),
    .ZN(n2488) );
AOI32D0 U3355 ( .A1(n2490), .A2(n2489), .A3(n2488), .B1(n3439
   ), .B2(n2489),
      .ZN(n2491) );
AOI211D0 U3356 ( .A1(n2577), .A2(n2493), .B(n2492), .C(n2491)
   , .ZN(n2525) );
ND4D0 U3357 ( .A1(n2557), .A2(n2496), .A3(n2495), .A4(n2494),
    .ZN(n2522) );
IND4D0 U3358 ( .A1(n2500), .B1(n2499), .B2(n2498), .B3(n2497)
   , .ZN(n2501) );
NR4D0 U3359 ( .A1(n2504), .A2(n2503), .A3(n2502), .A4(n2501),
    .ZN(n2520) );
IND4D0 U3360 ( .A1(n2507), .B1(n2506), .B2(n3671), .B3(n2505)
   , .ZN(n2509) );
OAI31D0 U3361 ( .A1(n2514), .A2(n2510), .A3(n2509), .B(n2508)
   , .ZN(n2519) );
```

```
IND4D0 U3362 ( .A1(n2514), .B1(n2513), .B2(n2512), .B3(n2511)
   , .ZN(n2515) );
OAI31D0 U3363 ( .A1(n2517), .A2(n2516), .A3(n2515), .B(n3443)
   , .ZN(n2518) );
OAI211D0 U3364 ( .A1(n2520), .A2(n3439), .B(n2519), .C(n2518)
   , .ZN(n2521) );
AOI221D0 U3365 ( .A1(n2523), .A2(n2577), .B1(n2522), .B2(
   n2577), .C(n2521),
      .ZN(n2524) );
OAI22D0 U3366 ( .A1(n2525), .A2(n4192), .B1(n2524), .B2(n4148
   ), .ZN(n2605)
      );
NR2D0 U3367 ( .A1(n2526), .A2(n2537), .ZN(n2530) );
AO211D0 U3368 ( .A1(n2544), .A2(n2528), .B(n2584), .C(n2527),
    .Z(n2548) );
NR4D0 U3369 ( .A1(n2531), .A2(n2530), .A3(n2529), .A4(n2548),
    .ZN(n2532) );
IND4D0 U3370 ( .A1(n2535), .B1(n2534), .B2(n2533), .B3(n2532)
   , .ZN(n2576) );
INVD0 U3371 ( .I(n2536), .ZN(n2542) );
OAI22D0 U3372 ( .A1(n2540), .A2(n2539), .B1(n2538), .B2(n2537
   ), .ZN(n2541)
      );
AOI211D0 U3373 ( .A1(n2544), .A2(n2543), .B(n2542), .C(n2541)
   , .ZN(n2556) );
ND4D0 U3374 ( .A1(n2587), .A2(n2546), .A3(n2578), .A4(n2545),
    .ZN(n2547) );
IINR4D0 U3375 ( .A1(n2550), .A2(n2549), .B1(n2548), .B2(n2547
   ), .ZN(n2551)
      );
AOI21D0 U3376 ( .A1(n2556), .A2(n2551), .B(n2596), .ZN(n2575)
    );
INR4D0 U3377 ( .A1(n2554), .B1(n2553), .B2(n2588), .B3(n2552)
   , .ZN(n2555) );
IND4D0 U3378 ( .A1(n2585), .B1(n2557), .B2(n2556), .B3(n2555)
   , .ZN(n2558) );
INR4D0 U3379 ( .A1(n2561), .B1(n2560), .B2(n2559), .B3(n2558)
   , .ZN(n2573) );
AOI211D0 U3380 ( .A1(n2565), .A2(n2564), .B(n2563), .C(n2562)
   , .ZN(n2566) );
OAI211D0 U3381 ( .A1(n2568), .A2(n4428), .B(n2567), .C(n2566)
   , .ZN(n2569) );
NR4D0 U3382 ( .A1(n2571), .A2(n2588), .A3(n2570), .A4(n2569),
    .ZN(n2572) );
OAI22D0 U3383 ( .A1(n2573), .A2(n3439), .B1(n2572), .B2(n3349
   ), .ZN(n2574)
      );
```

```
AOI211D0 U3384 ( .A1(n2577), .A2(n2576), .B(n2575), .C(n2574)
   , .ZN(n2603) );
ND4D0 U3385 ( .A1(n2581), .A2(n2580), .A3(n2579), .A4(n2578),
    .ZN(n2601) );
NR4D0 U3386 ( .A1(n2585), .A2(n2584), .A3(n2583), .A4(n2582),
    .ZN(n2586) );
AOI21D0 U3387 ( .A1(n2587), .A2(n2586), .B(n3349), .ZN(n2600)
    );
INR4D0 U3388 ( .A1(n2591), .B1(n2590), .B2(n2589), .B3(n2588)
   , .ZN(n2598) );
NR4D0 U3389 ( .A1(n2595), .A2(n2594), .A3(n2593), .A4(n2592),
    .ZN(n2597) );
OAI22D0 U3390 ( .A1(n2598), .A2(n2987), .B1(n2597), .B2(n2596
   ), .ZN(n2599)
      );
AOI211D0 U3391 ( .A1(n3114), .A2(n2601), .B(n2600), .C(n2599)
   , .ZN(n2602) );
OAI22D0 U3392 ( .A1(n2603), .A2(n4129), .B1(n2602), .B2(n4527
   ), .ZN(n2604)
      );
NR4D0 U3393 ( .A1(n2607), .A2(n2606), .A3(n2605), .A4(n2604),
    .ZN(n2608) );
OAI211D0 U3394 ( .A1(n2610), .A2(n3777), .B(n2609), .C(n2608)
   , .ZN(n2611) );
AOI211D0 U3395 ( .A1(n3664), .A2(n2613), .B(n2612), .C(n2611)
   , .ZN(n2964) );
CKND2D0 U3396 ( .A1(n3657), .A2(n3913), .ZN(n2685) );
INVD0 U3397 ( .I(n2685), .ZN(n2838) );
OAI211D0 U3398 ( .A1(n2789), .A2(n2614), .B(n2838), .C(n3917)
   , .ZN(n2634) );
NR4D0 U3399 ( .A1(n2615), .A2(n2780), .A3(n2725), .A4(n2733),
    .ZN(n2621) );
CKND2D0 U3400 ( .A1(n3816), .A2(n3621), .ZN(n3771) );
NR2D0 U3401 ( .A1(n2919), .A2(n3723), .ZN(n3923) );
INVD0 U3402 ( .I(n3923), .ZN(n3827) );
CKND2D0 U3403 ( .A1(n3907), .A2(n2714), .ZN(n3915) );
OAI211D0 U3404 ( .A1(n2618), .A2(n2617), .B(n2616), .C(n3915)
   , .ZN(n2810) );
NR4D0 U3405 ( .A1(n3771), .A2(n3827), .A3(n2810), .A4(n2619),
    .ZN(n2620) );
OAI211D0 U3406 ( .A1(n2623), .A2(n2622), .B(n2621), .C(n2620)
   , .ZN(n2633) );
NR2D0 U3407 ( .A1(counter[2]), .A2(n3882), .ZN(n3899) );
NR2D0 U3408 ( .A1(n3892), .A2(n3899), .ZN(n2624) );
CKND2D0 U3409 ( .A1(n4110), .A2(n3757), .ZN(n3974) );
IND4D0 U3410 ( .A1(n2625), .B1(n3964), .B2(n2624), .B3(n3974)
   , .ZN(n2787) );
```

```
OAI211D0 U3411 ( .A1(n3955), .A2(n3946), .B(n3637), .C(n2626)
   , .ZN(n2853) );
OAI211D0 U3412 ( .A1(n4272), .A2(n3757), .B(n2628), .C(n2627)
   , .ZN(n2629) );
NR4D0 U3413 ( .A1(n2787), .A2(n2853), .A3(n2630), .A4(n2629),
    .ZN(n2631) );
AOI31D0 U3414 ( .A1(n2631), .A2(n3745), .A3(n3962), .B(n4059)
   , .ZN(n2632) );
AOI221D0 U3415 ( .A1(n2634), .A2(n3933), .B1(n2633), .B2(
   n3933), .C(n2632),
      .ZN(n2662) );
INVD0 U3416 ( .I(n2895), .ZN(n3597) );
OAI211D0 U3417 ( .A1(n3946), .A2(n3955), .B(n3597), .C(n2635)
   , .ZN(n2648) );
AOI22D0 U3418 ( .A1(n2638), .A2(n3732), .B1(n2637), .B2(n2636
   ), .ZN(n3894)
      );
OAI211D0 U3419 ( .A1(n2639), .A2(n3821), .B(n3894), .C(n3905)
   , .ZN(n2640) );
AOI211D0 U3420 ( .A1(counter[11]), .A2(n2642), .B(n2641), .C(
   n2640), .ZN(
      n2801) );
NR4D0 U3421 ( .A1(n2644), .A2(n2897), .A3(n3781), .A4(n2643),
    .ZN(n2645) );
IND4D0 U3422 ( .A1(n3690), .B1(n2801), .B2(n2646), .B3(n2645)
   , .ZN(n2647) );
OAI31D0 U3423 ( .A1(n4037), .A2(n2648), .A3(n2647), .B(n3844)
   , .ZN(n2661) );
NR2D0 U3424 ( .A1(n2650), .A2(n2649), .ZN(n2785) );
ND4D0 U3425 ( .A1(n2785), .A2(n2651), .A3(n3672), .A4(n3962),
    .ZN(n2659) );
NR2D0 U3426 ( .A1(n3832), .A2(n2762), .ZN(n2876) );
AOI211D0 U3427 ( .A1(n2799), .A2(n2654), .B(n2653), .C(n2652)
   , .ZN(n2656) );
ND4D0 U3428 ( .A1(n2876), .A2(n2656), .A3(n3987), .A4(n2655),
    .ZN(n2856) );
OR2D0 U3429 ( .A1(n2918), .A2(n2791), .Z(n2921) );
IND4D0 U3430 ( .A1(n2921), .B1(n2712), .B2(n3739), .B3(n2657)
   , .ZN(n2658) );
OAI31D0 U3431 ( .A1(n2659), .A2(n2856), .A3(n2658), .B(n3769)
   , .ZN(n2660) );
AOI31D0 U3432 ( .A1(n2662), .A2(n2661), .A3(n2660), .B(n4277)
   , .ZN(n2962) );
ND4D0 U3433 ( .A1(n3616), .A2(n4062), .A3(n2663), .A4(n2935),
    .ZN(n2670) );
NR3D0 U3434 ( .A1(n2664), .A2(n3890), .A3(n2880), .ZN(n2847)
   );
INVD0 U3435 ( .I(n4043), .ZN(n2665) );
```

```
CKND2D0 U3436 ( .A1(n3793), .A2(n2665), .ZN(n2674) );
NR2D0 U3437 ( .A1(n3904), .A2(n3660), .ZN(n4047) );
INVD0 U3438 ( .I(n2830), .ZN(n2818) );
ND4D0 U3439 ( .A1(n2786), .A2(n4047), .A3(n2709), .A4(n2818),
    .ZN(n2666) );
OAI31D0 U3440 ( .A1(n3608), .A2(n2674), .A3(n2666), .B(n3933)
   , .ZN(n2668) );
NR2D0 U3441 ( .A1(n2914), .A2(n3860), .ZN(n3630) );
AN4D0 U3442 ( .A1(n3630), .A2(n3609), .A3(n2930), .A4(n2765),
    .Z(n2667) );
AOI32D0 U3443 ( .A1(n2847), .A2(n2668), .A3(n2667), .B1(n4045
   ), .B2(n2668),
      .ZN(n2669) );
AOI221D0 U3444 ( .A1(n2685), .A2(n4000), .B1(n2670), .B2(
   n4000), .C(n2669),
      .ZN(n2701) );
CKND2D0 U3445 ( .A1(n3813), .A2(n4022), .ZN(n3869) );
NR2D0 U3446 ( .A1(n3869), .A2(n2671), .ZN(n2756) );
INVD0 U3447 ( .I(n2756), .ZN(n2863) );
NR4D0 U3448 ( .A1(n3756), .A2(n2673), .A3(n2863), .A4(n2672),
    .ZN(n2683) );
IND4D0 U3449 ( .A1(n2896), .B1(n2687), .B2(n4062), .B3(n2720)
   , .ZN(n2677) );
NR2D0 U3450 ( .A1(n3695), .A2(n4044), .ZN(n2916) );
IINR4D0 U3451 ( .A1(n2939), .A2(n2916), .B1(n3756), .B2(n2674
   ), .ZN(n2675)
      );
NR2D0 U3452 ( .A1(n3595), .A2(n2732), .ZN(n3615) );
INVD0 U3453 ( .I(n3615), .ZN(n4014) );
AOI31D0 U3454 ( .A1(n2675), .A2(n3631), .A3(n4014), .B(n4045)
   , .ZN(n2676) );
AOI221D0 U3455 ( .A1(n2872), .A2(n3769), .B1(n2677), .B2(
   n3769), .C(n2676),
      .ZN(n2682) );
INVD0 U3456 ( .I(n2678), .ZN(n2680) );
ND4D0 U3457 ( .A1(n2752), .A2(n3927), .A3(n3677), .A4(n2938),
    .ZN(n2679) );
OAI31D0 U3458 ( .A1(n2680), .A2(n3775), .A3(n2679), .B(n4000)
   , .ZN(n2681) );
OAI211D0 U3459 ( .A1(n2683), .A2(n4038), .B(n2682), .C(n2681)
   , .ZN(n2696) );
NR3D0 U3460 ( .A1(n2918), .A2(n2740), .A3(n2684), .ZN(n2851)
   );
CKND2D0 U3461 ( .A1(n3793), .A2(n2697), .ZN(n2933) );
INR4D0 U3462 ( .A1(n2851), .B1(n3716), .B2(n2908), .B3(n2933)
   , .ZN(n2694) );
NR2D0 U3463 ( .A1(n2872), .A2(n3775), .ZN(n2702) );
```

```
ND4D0 U3464 ( .A1(n2757), .A2(n2702), .A3(n4330), .A4(n3987),
    .ZN(n2689) );
CKND2D0 U3465 ( .A1(n3692), .A2(n3637), .ZN(n3798) );
NR4D0 U3466 ( .A1(n3798), .A2(n2685), .A3(n3649), .A4(n2819),
    .ZN(n2686) );
AOI31D0 U3467 ( .A1(n2687), .A2(n2686), .A3(n4050), .B(n4010)
   , .ZN(n2688) );
AOI221D0 U3468 ( .A1(n2690), .A2(n3933), .B1(n2689), .B2(
   n3933), .C(n2688),
      .ZN(n2693) );
INVD0 U3469 ( .I(n3892), .ZN(n3790) );
CKND2D0 U3470 ( .A1(n3633), .A2(n2941), .ZN(n2739) );
INR3D0 U3471 ( .A1(n2912), .B1(n2739), .B2(n2872), .ZN(n2691)
    );
AO31D0 U3472 ( .A1(n3790), .A2(n3653), .A3(n2691), .B(n4045),
    .Z(n2692) );
OAI211D0 U3473 ( .A1(n2694), .A2(n4059), .B(n2693), .C(n2692)
   , .ZN(n2695) );
AOI22D0 U3474 ( .A1(n4492), .A2(n2696), .B1(n3386), .B2(n2695
   ), .ZN(n2700)
       );
ND4D0 U3475 ( .A1(n2702), .A2(n2939), .A3(n3789), .A4(n2697),
    .ZN(n2698) );
OAI31D0 U3476 ( .A1(n2750), .A2(n2733), .A3(n2698), .B(n3769)
   , .ZN(n2699) );
AOI32D0 U3477 ( .A1(n2701), .A2(n2700), .A3(n2699), .B1(n4190
   ), .B2(n2700),
      .ZN(n2961) );
INVD0 U3478 ( .I(n2702), .ZN(n2843) );
INR4D0 U3479 ( .A1(n2929), .B1(n2908), .B2(n2843), .B3(n2703)
   , .ZN(n2704) );
AOI31D0 U3480 ( .A1(n2704), .A2(n3676), .A3(n3835), .B(n4045)
   , .ZN(n2719) );
NR4D0 U3481 ( .A1(n2706), .A2(n2705), .A3(n2791), .A4(n3632),
    .ZN(n2707) );
AOI31D0 U3482 ( .A1(n2707), .A2(n3671), .A3(n3676), .B(n4059)
   , .ZN(n2718) );
ND4D0 U3483 ( .A1(n2935), .A2(n3630), .A3(n2709), .A4(n2708),
    .ZN(n2710) );
NR4D0 U3484 ( .A1(n3918), .A2(n3870), .A3(n2872), .A4(n2710),
    .ZN(n2716) );
NR2D0 U3485 ( .A1(n3892), .A2(n2762), .ZN(n3785) );
ND4D0 U3486 ( .A1(n2712), .A2(n2939), .A3(n3785), .A4(n2711),
    .ZN(n2713) );
AOI211D0 U3487 ( .A1(n2714), .A2(n4098), .B(n4043), .C(n2713)
   , .ZN(n2715) );
OAI22D0 U3488 ( .A1(n2716), .A2(n4038), .B1(n2715), .B2(n4010
   ), .ZN(n2717)
```

```
        );
OAI31D0 U3489 ( .A1(n2719), .A2(n2718), .A3(n2717), .B(n3709)
   , .ZN(n2772) );
INR2D0 U3490 ( .A1(n2720), .B1(n3967), .ZN(n3773) );
AOI31D0 U3491 ( .A1(n3773), .A2(n2722), .A3(n2721), .B(n4038)
   , .ZN(n2731) );
INVD0 U3492 ( .I(n2900), .ZN(n2724) );
INVD0 U3493 ( .I(n2844), .ZN(n2828) );
NR2D0 U3494 ( .A1(n3751), .A2(n2820), .ZN(n3718) );
ND3D0 U3495 ( .A1(n2828), .A2(n3718), .A3(n4023), .ZN(n2738)
   );
NR3D0 U3496 ( .A1(n3674), .A2(n2733), .A3(n2738), .ZN(n2723)
   );
AOI31D0 U3497 ( .A1(n2724), .A2(n2752), .A3(n2723), .B(n4045)
   , .ZN(n2729) );
ND3D0 U3498 ( .A1(n2818), .A2(n2724), .A3(n3637), .ZN(n2850)
   );
CKND2D0 U3499 ( .A1(n3715), .A2(n3788), .ZN(n4036) );
INR4D0 U3500 ( .A1(n2821), .B1(n2791), .B2(n2850), .B3(n4036)
   , .ZN(n2727) );
NR4D0 U3501 ( .A1(n2725), .A2(n3843), .A3(n2739), .A4(n2824),
    .ZN(n2726) );
OAI22D0 U3502 ( .A1(n2727), .A2(n4010), .B1(n2726), .B2(n4059
   ), .ZN(n2728)
       );
OAI31D0 U3503 ( .A1(n2731), .A2(n2729), .A3(n2728), .B(n3311)
   , .ZN(n2771) );
NR2D0 U3504 ( .A1(n3822), .A2(n2732), .ZN(n2734) );
NR4D0 U3505 ( .A1(n3786), .A2(n3782), .A3(n2734), .A4(n2733),
    .ZN(n2735) );
AOI31D0 U3506 ( .A1(n2736), .A2(n3807), .A3(n2735), .B(n4038)
   , .ZN(n2747) );
INVD0 U3507 ( .I(n2840), .ZN(n4009) );
NR4D0 U3508 ( .A1(n3674), .A2(n2897), .A3(n2950), .A4(n4009),
    .ZN(n2737) );
AOI31D0 U3509 ( .A1(n3713), .A2(n3597), .A3(n2737), .B(n4059)
   , .ZN(n2746) );
CKND2D0 U3510 ( .A1(n3653), .A2(n2938), .ZN(n3858) );
NR4D0 U3511 ( .A1(n2740), .A2(n2739), .A3(n2738), .A4(n3858),
    .ZN(n2744) );
NR2D0 U3512 ( .A1(n3891), .A2(n2762), .ZN(n3888) );
ND4D0 U3513 ( .A1(n3888), .A2(n3715), .A3(n3813), .A4(n3835),
    .ZN(n2741) );
NR4D0 U3514 ( .A1(n2791), .A2(n2871), .A3(n2742), .A4(n2741),
    .ZN(n2743) );
OAI22D0 U3515 ( .A1(n2744), .A2(n4045), .B1(n2743), .B2(n4010
   ), .ZN(n2745)
       );
```

```
OAI31D0 U3516 ( .A1(n2747), .A2(n2746), .A3(n2745), .B(n3850)
   , .ZN(n2770) );
CKND2D0 U3517 ( .A1(n3621), .A2(n3986), .ZN(n2749) );
CKND2D0 U3518 ( .A1(n3913), .A2(n2748), .ZN(n2898) );
INR4D0 U3519 ( .A1(n2835), .B1(n2750), .B2(n2749), .B3(n2898)
   , .ZN(n2751) );
AOI31D0 U3520 ( .A1(n2916), .A2(n2752), .A3(n2751), .B(n4059)
   , .ZN(n2768) );
CKND2D0 U3521 ( .A1(n2753), .A2(n3883), .ZN(n2920) );
CKND2D0 U3522 ( .A1(n2859), .A2(n2754), .ZN(n2944) );
NR4D0 U3523 ( .A1(n3786), .A2(n3674), .A3(n2920), .A4(n2944),
    .ZN(n2755) );
AOI31D0 U3524 ( .A1(n2899), .A2(n3923), .A3(n2755), .B(n4038)
   , .ZN(n2767) );
ND4D0 U3525 ( .A1(n2757), .A2(n2756), .A3(n3789), .A4(n2877),
    .ZN(n2758) );
OAI31D0 U3526 ( .A1(n4037), .A2(n2759), .A3(n2758), .B(n3769)
   , .ZN(n2764) );
CKND2D0 U3527 ( .A1(n2760), .A2(n4445), .ZN(n4049) );
ND4D0 U3528 ( .A1(n3683), .A2(n3621), .A3(n3986), .A4(n4049),
    .ZN(n2761) );
NR4D0 U3529 ( .A1(n2796), .A2(n2762), .A3(n3810), .A4(n2761),
    .ZN(n2763) );
AOI32D0 U3530 ( .A1(n2765), .A2(n2764), .A3(n2763), .B1(n4045
   ), .B2(n2764),
      .ZN(n2766) );
OAI31D0 U3531 ( .A1(n2768), .A2(n2767), .A3(n2766), .B(n4063)
   , .ZN(n2769) );
ND4D0 U3532 ( .A1(n2772), .A2(n2771), .A3(n2770), .A4(n2769),
    .ZN(n2960) );
NR2D0 U3533 ( .A1(n3956), .A2(n2773), .ZN(n3945) );
AOI211D0 U3534 ( .A1(n4110), .A2(n3757), .B(n3738), .C(n3945)
   , .ZN(n3932) );
IND4D0 U3535 ( .A1(n2920), .B1(n3932), .B2(n3661), .B3(n3916)
   , .ZN(n2774) );
NR2D0 U3536 ( .A1(n2775), .A2(n2774), .ZN(n2800) );
OAI22D0 U3537 ( .A1(n2777), .A2(n3332), .B1(n2776), .B2(n3735
   ), .ZN(n2778)
      );
NR3D0 U3538 ( .A1(n2780), .A2(n2779), .A3(n2778), .ZN(n2792)
   );
ND4D0 U3539 ( .A1(n2792), .A2(n3657), .A3(n3912), .A4(n2854),
    .ZN(n2781) );
NR4D0 U3540 ( .A1(n2783), .A2(n3674), .A3(n2782), .A4(n2781),
    .ZN(n2784) );
ND4D0 U3541 ( .A1(n2786), .A2(n2785), .A3(n2800), .A4(n2784),
    .ZN(n2816) );
INVD0 U3542 ( .I(n2787), .ZN(n2794) );
```

```
OAI22D0 U3543 ( .A1(counter[1]), .A2(n2789), .B1(n2788), .B2(
   n3754), .ZN(
      n2790) );
INR4D0 U3544 ( .A1(n2792), .B1(n2872), .B2(n2791), .B3(n2790)
   , .ZN(n2793) );
AOI31D0 U3545 ( .A1(n2795), .A2(n2794), .A3(n2793), .B(n4045)
   , .ZN(n2815) );
AOI211D0 U3546 ( .A1(n2799), .A2(n2798), .B(n2797), .C(n2796)
   , .ZN(n2813) );
INVD0 U3547 ( .I(n2800), .ZN(n2809) );
INVD0 U3548 ( .I(n2801), .ZN(n2807) );
NR4D0 U3549 ( .A1(n3742), .A2(n2802), .A3(n3903), .A4(n2950),
    .ZN(n2805) );
NR4D0 U3550 ( .A1(n3674), .A2(n3870), .A3(n3639), .A4(n2908),
    .ZN(n2804) );
CKND2D0 U3551 ( .A1(counter[4]), .A2(n2803), .ZN(n3911) );
ND4D0 U3552 ( .A1(n2805), .A2(n2804), .A3(n3914), .A4(n3911),
    .ZN(n2806) );
OAI31D0 U3553 ( .A1(n2809), .A2(n2807), .A3(n2806), .B(n4000)
   , .ZN(n2812) );
NR4D0 U3554 ( .A1(n2810), .A2(n2944), .A3(n2809), .A4(n2808),
    .ZN(n2811) );
AOI32D0 U3555 ( .A1(n2813), .A2(n2812), .A3(n2811), .B1(n4038
   ), .B2(n2812),
      .ZN(n2814) );
AOI211D0 U3556 ( .A1(n3769), .A2(n2816), .B(n2815), .C(n2814)
   , .ZN(n2958) );
CKND2D0 U3557 ( .A1(n2828), .A2(n3672), .ZN(n2907) );
CKND2D0 U3558 ( .A1(n2817), .A2(n3633), .ZN(n3673) );
INR3D0 U3559 ( .A1(n3888), .B1(n2907), .B2(n3673), .ZN(n2922)
    );
AOI31D0 U3560 ( .A1(n2818), .A2(n3630), .A3(n2922), .B(n4038)
   , .ZN(n2893) );
OR2D0 U3561 ( .A1(n3771), .A2(n4021), .Z(n3776) );
NR4D0 U3562 ( .A1(n2820), .A2(n2819), .A3(n3673), .A4(n3776),
    .ZN(n2833) );
IND4D0 U3563 ( .A1(n2894), .B1(n2822), .B2(n2821), .B3(n3672)
   , .ZN(n2823) );
OAI31D0 U3564 ( .A1(n2825), .A2(n2824), .A3(n2823), .B(n4000)
   , .ZN(n2832) );
ND4D0 U3565 ( .A1(n2941), .A2(n2828), .A3(n2827), .A4(n2826),
    .ZN(n2829) );
OAI31D0 U3566 ( .A1(n3797), .A2(n2830), .A3(n2829), .B(n3844)
   , .ZN(n2831) );
OAI211D0 U3567 ( .A1(n2833), .A2(n4010), .B(n2832), .C(n2831)
   , .ZN(n2892) );
CKND2D0 U3568 ( .A1(n3987), .A2(n2938), .ZN(n3734) );
```

```
NR4D0 U3569 ( .A1(n3639), .A2(n2872), .A3(n3690), .A4(n3734),
    .ZN(n2834) );
ND4D0 U3570 ( .A1(n2939), .A2(n2834), .A3(n3672), .A4(n3913),
    .ZN(n2849) );
ND4D0 U3571 ( .A1(n2838), .A2(n2837), .A3(n2836), .A4(n2835),
    .ZN(n2842) );
AOI31D0 U3572 ( .A1(n2899), .A2(n2840), .A3(n2839), .B(n4045)
   , .ZN(n2841) );
AOI221D0 U3573 ( .A1(n2843), .A2(n3933), .B1(n2842), .B2(
   n3933), .C(n2841),
      .ZN(n2846) );
NR4D0 U3574 ( .A1(n4143), .A2(n4043), .A3(n2918), .A4(n2844),
    .ZN(n2845) );
AOI32D0 U3575 ( .A1(n2847), .A2(n2846), .A3(n2845), .B1(n4059
   ), .B2(n2846),
      .ZN(n2848) );
AOI221D0 U3576 ( .A1(n2850), .A2(n3769), .B1(n2849), .B2(
   n3769), .C(n2848),
      .ZN(n2890) );
CKND2D0 U3577 ( .A1(n3677), .A2(n3676), .ZN(n3957) );
NR2D0 U3578 ( .A1(n3854), .A2(n3957), .ZN(n4048) );
AOI31D0 U3579 ( .A1(n2851), .A2(n4048), .A3(n2873), .B(n4010)
   , .ZN(n2868) );
NR4D0 U3580 ( .A1(n2908), .A2(n3775), .A3(n2853), .A4(n2852),
    .ZN(n2858) );
INVD0 U3581 ( .I(n2854), .ZN(n3988) );
NR4D0 U3582 ( .A1(n3945), .A2(n3988), .A3(n2856), .A4(n2855),
    .ZN(n2857) );
AOI21D0 U3583 ( .A1(n2858), .A2(n2857), .B(n4038), .ZN(n2867)
    );
CKND2D0 U3584 ( .A1(n2859), .A2(n4023), .ZN(n3969) );
ND4D0 U3585 ( .A1(n2860), .A2(n3658), .A3(n3923), .A4(n3616),
    .ZN(n2861) );
NR4D0 U3586 ( .A1(n3690), .A2(n2862), .A3(n3969), .A4(n2861),
    .ZN(n2865) );
NR2D0 U3587 ( .A1(n3742), .A2(n2919), .ZN(n3794) );
CKND2D0 U3588 ( .A1(n3856), .A2(n3794), .ZN(n3620) );
NR4D0 U3589 ( .A1(n3782), .A2(n2863), .A3(n2921), .A4(n3620),
    .ZN(n2864) );
OAI22D0 U3590 ( .A1(n2865), .A2(n4045), .B1(n2864), .B2(n4059
   ), .ZN(n2866)
      );
OAI31D0 U3591 ( .A1(n2868), .A2(n2867), .A3(n2866), .B(n4305)
   , .ZN(n2889) );
AOI31D0 U3592 ( .A1(n2870), .A2(n3683), .A3(n2869), .B(n4059)
   , .ZN(n2887) );
NR4D0 U3593 ( .A1(n2872), .A2(n3869), .A3(n2880), .A4(n2871),
    .ZN(n2874) );
```

```
AOI31D0 U3594 ( .A1(n2875), .A2(n2874), .A3(n2873), .B(n4010)
   , .ZN(n2886) );
CKND2D0 U3595 ( .A1(n2876), .A2(n3835), .ZN(n3868) );
NR2D0 U3596 ( .A1(n3742), .A2(n3786), .ZN(n2878) );
IND4D0 U3597 ( .A1(n2907), .B1(n2878), .B2(n3633), .B3(n2877)
   , .ZN(n2879) );
NR4D0 U3598 ( .A1(n3608), .A2(n3783), .A3(n3868), .A4(n2879),
    .ZN(n2884) );
INVD0 U3599 ( .I(n2930), .ZN(n2937) );
OR4D0 U3600 ( .A1(n2880), .A2(n3756), .A3(n3674), .A4(n2919),
    .Z(n2881) );
NR4D0 U3601 ( .A1(n2882), .A2(n3783), .A3(n2937), .A4(n2881),
    .ZN(n2883) );
OAI22D0 U3602 ( .A1(n2884), .A2(n4038), .B1(n2883), .B2(n4045
   ), .ZN(n2885)
       );
OAI31D0 U3603 ( .A1(n2887), .A2(n2886), .A3(n2885), .B(n3437)
   , .ZN(n2888) );
OAI211D0 U3604 ( .A1(n2890), .A2(n4148), .B(n2889), .C(n2888)
   , .ZN(n2891) );
AOI221D0 U3605 ( .A1(n2893), .A2(n3707), .B1(n2892), .B2(
   n3707), .C(n2891),
      .ZN(n2957) );
NR3D0 U3606 ( .A1(n2894), .A2(n4143), .A3(n3650), .ZN(n2911)
   );
NR2D0 U3607 ( .A1(n3639), .A2(n2895), .ZN(n2934) );
IINR4D0 U3608 ( .A1(n2911), .A2(n2934), .B1(n2897), .B2(n2896
   ), .ZN(n2904)
       );
NR2D0 U3609 ( .A1(n3918), .A2(n2898), .ZN(n3928) );
ND3D0 U3610 ( .A1(n2899), .A2(n3928), .A3(n3883), .ZN(n2932)
   );
CKND2D0 U3611 ( .A1(n3672), .A2(n2938), .ZN(n3762) );
NR4D0 U3612 ( .A1(n2918), .A2(n3690), .A3(n2932), .A4(n3762),
    .ZN(n2903) );
NR3D0 U3613 ( .A1(n2900), .A2(n3716), .A3(n3782), .ZN(n2928)
   );
INR4D0 U3614 ( .A1(n2928), .B1(n4155), .B2(n2901), .B3(n3740)
   , .ZN(n2902) );
OAI222D0 U3615 ( .A1(n4059), .A2(n2904), .B1(n4045), .B2(
   n2903), .C1(n4010),
      .C2(n2902), .ZN(n2955) );
NR4D0 U3616 ( .A1(n2908), .A2(n2907), .A3(n2906), .A4(n2905),
    .ZN(n2909) );
AOI211D0 U3617 ( .A1(n3601), .A2(n2909), .B(n4038), .C(n3379)
   , .ZN(n2954) );
ND4D0 U3618 ( .A1(n2912), .A2(n2911), .A3(n2910), .A4(n3927),
    .ZN(n2927) );
```

```
NR4D0 U3619 ( .A1(n3918), .A2(n3782), .A3(n2914), .A4(n2913),
    .ZN(n2915) );
AOI31D0 U3620 ( .A1(n2916), .A2(n2941), .A3(n2915), .B(n4045)
   , .ZN(n2926) );
INR4D0 U3621 ( .A1(n3808), .B1(n2919), .B2(n2918), .B3(n2917)
   , .ZN(n2924) );
INR4D0 U3622 ( .A1(n2922), .B1(n3890), .B2(n2921), .B3(n2920)
   , .ZN(n2923) );
OAI22D0 U3623 ( .A1(n2924), .A2(n4059), .B1(n2923), .B2(n4038
   ), .ZN(n2925)
      );
AOI211D0 U3624 ( .A1(n3769), .A2(n2927), .B(n2926), .C(n2925)
   , .ZN(n2952) );
ND4D0 U3625 ( .A1(n2930), .A2(n2929), .A3(n2928), .A4(n3816),
    .ZN(n2949) );
NR4D0 U3626 ( .A1(n3650), .A2(n2933), .A3(n2932), .A4(n2931),
    .ZN(n2947) );
ND4D0 U3627 ( .A1(n2935), .A2(n2934), .A3(n3813), .A4(n4011),
    .ZN(n2936) );
OAI31D0 U3628 ( .A1(n3969), .A2(n2937), .A3(n2936), .B(n3769)
   , .ZN(n2946) );
ND4D0 U3629 ( .A1(n2941), .A2(n2940), .A3(n2939), .A4(n2938),
    .ZN(n2942) );
OAI31D0 U3630 ( .A1(n2944), .A2(n2943), .A3(n2942), .B(n3844)
   , .ZN(n2945) );
OAI211D0 U3631 ( .A1(n2947), .A2(n4038), .B(n2946), .C(n2945)
   , .ZN(n2948) );
AOI221D0 U3632 ( .A1(n2950), .A2(n4000), .B1(n2949), .B2(
   n4000), .C(n2948),
      .ZN(n2951) );
OAI22D0 U3633 ( .A1(n2952), .A2(n4408), .B1(n2951), .B2(n3265
   ), .ZN(n2953)
      );
AOI211D0 U3634 ( .A1(n4490), .A2(n2955), .B(n2954), .C(n2953)
   , .ZN(n2956) );
OAI211D0 U3635 ( .A1(n2958), .A2(n4129), .B(n2957), .C(n2956)
   , .ZN(n2959) );
NR4D0 U3636 ( .A1(n2962), .A2(n2961), .A3(n2960), .A4(n2959),
    .ZN(n2963) );
OAI222D0 U3637 ( .A1(\q_out[2]_BAR ), .A2(n4545), .B1(n4544),
    .B2(n2964),
      .C1(n4542), .C2(n2963), .ZN(n4814) );
NR4D0 U3638 ( .A1(n3319), .A2(n2965), .A3(n3237), .A4(n3195),
    .ZN(n2976) );
CKND2D0 U3639 ( .A1(n3392), .A2(n3142), .ZN(n3072) );
NR2D0 U3640 ( .A1(n4426), .A2(n4183), .ZN(n4498) );
IND4D0 U3641 ( .A1(n3072), .B1(n4498), .B2(n4126), .B3(n4333)
   , .ZN(n2971) );
```

```
NR4D0 U3642 ( .A1(n2967), .A2(n3447), .A3(n4183), .A4(n2966),
    .ZN(n2969) );
ND4D0 U3643 ( .A1(n2969), .A2(n4201), .A3(n2968), .A4(n4320),
    .ZN(n2970) );
AOI22D0 U3644 ( .A1(n4291), .A2(n2971), .B1(n4452), .B2(n2970
   ), .ZN(n2975)
      );
ND4D0 U3645 ( .A1(n4324), .A2(n3507), .A3(n3343), .A4(n4333),
    .ZN(n2972) );
OAI31D0 U3646 ( .A1(n2973), .A2(n3071), .A3(n2972), .B(n4342)
   , .ZN(n2974) );
OAI211D0 U3647 ( .A1(n2976), .A2(n3243), .B(n2975), .C(n2974)
   , .ZN(n3269) );
CKND2D0 U3648 ( .A1(n4194), .A2(n3199), .ZN(n3173) );
INVD0 U3649 ( .I(n3173), .ZN(n3093) );
OAI31D0 U3650 ( .A1(n2978), .A2(n3956), .A3(n2977), .B(n3093)
   , .ZN(n2979) );
NR4D0 U3651 ( .A1(n2980), .A2(n3125), .A3(n3041), .A4(n2979),
    .ZN(n2981) );
IND4D0 U3652 ( .A1(n3240), .B1(n2981), .B2(n3053), .B3(n3145)
   , .ZN(n3007) );
NR2D0 U3653 ( .A1(n3349), .A2(n3955), .ZN(n4248) );
NR4D0 U3654 ( .A1(n4124), .A2(n4119), .A3(n4248), .A4(n3459),
    .ZN(n2983) );
OAI211D0 U3655 ( .A1(n3009), .A2(n2987), .B(n2983), .C(n2982)
   , .ZN(n2984) );
NR4D0 U3656 ( .A1(n2986), .A2(n3008), .A3(n2985), .A4(n2984),
    .ZN(n3005) );
INVD0 U3657 ( .I(n3192), .ZN(n3238) );
INR4D0 U3658 ( .A1(n3053), .B1(n3065), .B2(n3524), .B3(n3238)
   , .ZN(n2992) );
NR2D0 U3659 ( .A1(n2987), .A2(n3009), .ZN(n2990) );
CKND2D0 U3660 ( .A1(n4418), .A2(n4335), .ZN(n4218) );
INVD0 U3661 ( .I(n2988), .ZN(n3213) );
NR4D0 U3662 ( .A1(n2990), .A2(n2989), .A3(n4218), .A4(n3213),
    .ZN(n2991) );
ND4D0 U3663 ( .A1(n2992), .A2(n2991), .A3(n4333), .A4(n4091),
    .ZN(n2993) );
OAI31D0 U3664 ( .A1(n2994), .A2(n3042), .A3(n2993), .B(n4452)
   , .ZN(n3004) );
AOI22D0 U3665 ( .A1(n3056), .A2(n2996), .B1(n4345), .B2(n2995
   ), .ZN(n3002)
      );
CKND2D0 U3666 ( .A1(n3320), .A2(n3044), .ZN(n2997) );
CKND2D0 U3667 ( .A1(n3198), .A2(n3074), .ZN(n3204) );
NR4D0 U3668 ( .A1(n4373), .A2(n4119), .A3(n2997), .A4(n3204),
    .ZN(n2999) );
```

```
ND4D0 U3669 ( .A1(n4141), .A2(n2999), .A3(n2998), .A4(n3081),
    .ZN(n3001) );
NR2D0 U3670 ( .A1(n4124), .A2(n4093), .ZN(n3000) );
OAI211D0 U3671 ( .A1(n3897), .A2(n3349), .B(n3000), .C(n3053)
   , .ZN(n3059) );
OAI31D0 U3672 ( .A1(n3002), .A2(n3001), .A3(n3059), .B(n4342)
   , .ZN(n3003) );
OAI211D0 U3673 ( .A1(n3005), .A2(n3243), .B(n3004), .C(n3003)
   , .ZN(n3006) );
AOI221D0 U3674 ( .A1(n3008), .A2(n4291), .B1(n3007), .B2(
   n4291), .C(n3006),
      .ZN(n3023) );
NR2D0 U3675 ( .A1(n3214), .A2(n3225), .ZN(n3127) );
INVD0 U3676 ( .I(n3009), .ZN(n3113) );
AOI211D0 U3677 ( .A1(n3114), .A2(n3113), .B(n4181), .C(n3233)
   , .ZN(n3181) );
IND4D0 U3678 ( .A1(n3025), .B1(n4177), .B2(n3127), .B3(n3181)
   , .ZN(n3015) );
CKND2D0 U3679 ( .A1(n4388), .A2(n4465), .ZN(n3029) );
INVD0 U3680 ( .I(n3029), .ZN(n3027) );
CKND2D0 U3681 ( .A1(n3373), .A2(n3094), .ZN(n3010) );
NR4D0 U3682 ( .A1(n4076), .A2(n3011), .A3(n3072), .A4(n3010),
    .ZN(n3013) );
AOI31D0 U3683 ( .A1(n3027), .A2(n4234), .A3(n3013), .B(n3012)
   , .ZN(n3014) );
AOI31D0 U3684 ( .A1(n4452), .A2(n3437), .A3(n3015), .B(n3014)
   , .ZN(n3022) );
IND4D0 U3685 ( .A1(n3308), .B1(n3016), .B2(n3095), .B3(n3149)
   , .ZN(n3020) );
AOI31D0 U3686 ( .A1(n3018), .A2(n3017), .A3(n4157), .B(n4527)
   , .ZN(n3019) );
AOI32D0 U3687 ( .A1(n3664), .A2(n4480), .A3(n3020), .B1(n3019
   ), .B2(n4480),
      .ZN(n3021) );
OAI211D0 U3688 ( .A1(n3023), .A2(n4277), .B(n3022), .C(n3021)
   , .ZN(n3268) );
NR2D0 U3689 ( .A1(n4311), .A2(n4403), .ZN(n3283) );
ND4D0 U3690 ( .A1(n3024), .A2(n3142), .A3(n3160), .A4(n3283),
    .ZN(n3035) );
INR4D0 U3691 ( .A1(n3026), .B1(n4309), .B2(n3214), .B3(n3025)
   , .ZN(n3033) );
ND4D0 U3692 ( .A1(n4177), .A2(n3218), .A3(n3074), .A4(n3027),
    .ZN(n3028) );
OAI31D0 U3693 ( .A1(n4373), .A2(n4494), .A3(n3028), .B(n4452)
   , .ZN(n3032) );
OAI31D0 U3694 ( .A1(n3030), .A2(n3234), .A3(n3029), .B(n4342)
   , .ZN(n3031) );
```

```
OAI211D0 U3695 ( .A1(n3033), .A2(n3243), .B(n3032), .C(n3031)
   , .ZN(n3034) );
AOI221D0 U3696 ( .A1(n3123), .A2(n4291), .B1(n3035), .B2(
   n4291), .C(n3034),
      .ZN(n3266) );
CKND2D0 U3697 ( .A1(n3235), .A2(n3094), .ZN(n3122) );
CKND2D0 U3698 ( .A1(n4367), .A2(n3036), .ZN(n3211) );
NR4D0 U3699 ( .A1(n3162), .A2(n3037), .A3(n4088), .A4(n4172),
    .ZN(n3039) );
IND4D0 U3700 ( .A1(n3211), .B1(n3039), .B2(n4389), .B3(n3038)
   , .ZN(n3040) );
NR4D0 U3701 ( .A1(n3042), .A2(n3041), .A3(n3122), .A4(n3040),
    .ZN(n3051) );
NR2D0 U3702 ( .A1(n4422), .A2(n3043), .ZN(n3171) );
IND4D0 U3703 ( .A1(n3064), .B1(n3171), .B2(n4078), .B3(n3044)
   , .ZN(n3070) );
INR4D0 U3704 ( .A1(n4241), .B1(n3162), .B2(n3045), .B3(n3070)
   , .ZN(n3046) );
IND4D0 U3705 ( .A1(n3059), .B1(n3046), .B2(n4120), .B3(n4465)
   , .ZN(n3047) );
AOI211D0 U3706 ( .A1(n3049), .A2(n3048), .B(n3524), .C(n3047)
   , .ZN(n3050) );
OAI22D0 U3707 ( .A1(n3051), .A2(n4476), .B1(n3050), .B2(n4474
   ), .ZN(n3108)
      );
ND3D0 U3708 ( .A1(n3392), .A2(n4209), .A3(n4240), .ZN(n3069)
   );
NR4D0 U3709 ( .A1(n3237), .A2(n3196), .A3(n3052), .A4(n3069),
    .ZN(n3063) );
AN4D0 U3710 ( .A1(n3182), .A2(n4334), .A3(n3054), .A4(n3053),
    .Z(n3055) );
OAI211D0 U3711 ( .A1(n3056), .A2(n3332), .B(n3055), .C(n4121)
   , .ZN(n3057) );
OAI31D0 U3712 ( .A1(n3058), .A2(n3085), .A3(n3057), .B(n4342)
   , .ZN(n3062) );
NR4D0 U3713 ( .A1(n3060), .A2(n4119), .A3(n4232), .A4(n3059),
    .ZN(n3061) );
AOI32D0 U3714 ( .A1(n3063), .A2(n3062), .A3(n3061), .B1(n3243
   ), .B2(n3062),
      .ZN(n3107) );
INR4D0 U3715 ( .A1(n4094), .B1(n3065), .B2(n3064), .B3(n4111)
   , .ZN(n3067) );
ND4D0 U3716 ( .A1(n3068), .A2(n3097), .A3(n3067), .A4(n3066),
    .ZN(n3091) );
NR4D0 U3717 ( .A1(n4248), .A2(n3071), .A3(n3070), .A4(n3069),
    .ZN(n3089) );
INR2D0 U3718 ( .A1(n4081), .B1(n3529), .ZN(n3073) );
```

```
INR4D0 U3719 ( .A1(n3074), .B1(n4111), .B2(n3073), .B3(n3072)
   , .ZN(n3076) );
ND4D0 U3720 ( .A1(n3076), .A2(n4157), .A3(n4078), .A4(n3075),
    .ZN(n3077) );
OAI31D0 U3721 ( .A1(n3196), .A2(n3147), .A3(n3077), .B(n4289)
   , .ZN(n3088) );
NR4D0 U3722 ( .A1(n3238), .A2(n3080), .A3(n3079), .A4(n3078),
    .ZN(n3083) );
ND4D0 U3723 ( .A1(n3083), .A2(n4388), .A3(n3082), .A4(n3081),
    .ZN(n3084) );
OAI31D0 U3724 ( .A1(n3086), .A2(n3085), .A3(n3084), .B(n4291)
   , .ZN(n3087) );
OAI211D0 U3725 ( .A1(n3089), .A2(n3243), .B(n3088), .C(n3087)
   , .ZN(n3090) );
AOI221D0 U3726 ( .A1(n3092), .A2(n4452), .B1(n3091), .B2(
   n4452), .C(n3090),
      .ZN(n3105) );
ND4D0 U3727 ( .A1(n3184), .A2(n3127), .A3(n4157), .A4(n4194),
    .ZN(n3103) );
CKND2D0 U3728 ( .A1(n4368), .A2(n3505), .ZN(n4436) );
CKND2D0 U3729 ( .A1(n3094), .A2(n3093), .ZN(n3194) );
NR4D0 U3730 ( .A1(n4195), .A2(n4436), .A3(n3132), .A4(n3194),
    .ZN(n3101) );
ND4D0 U3731 ( .A1(n4234), .A2(n3095), .A3(n4389), .A4(n3111),
    .ZN(n3096) );
OAI31D0 U3732 ( .A1(n4156), .A2(n4232), .A3(n3096), .B(n4480)
   , .ZN(n3100) );
ND3D0 U3733 ( .A1(n3097), .A2(n4324), .A3(n3142), .ZN(n3144)
   );
OAI31D0 U3734 ( .A1(n3098), .A2(n3158), .A3(n3144), .B(n4342)
   , .ZN(n3099) );
OAI211D0 U3735 ( .A1(n3101), .A2(n54), .B(n3100), .C(n3099),
   .ZN(n3102) );
AOI221D0 U3736 ( .A1(n3203), .A2(n4291), .B1(n3103), .B2(
   n4291), .C(n3102),
      .ZN(n3104) );
OAI22D0 U3737 ( .A1(n3105), .A2(n3777), .B1(n3104), .B2(n4298
   ), .ZN(n3106)
       );
AOI221D0 U3738 ( .A1(n3108), .A2(n3567), .B1(n3107), .B2(
   n3567), .C(n3106),
      .ZN(n3264) );
ND3D0 U3739 ( .A1(n4157), .A2(n4506), .A3(n3140), .ZN(n3130)
   );
NR3D0 U3740 ( .A1(n3109), .A2(n3123), .A3(n4436), .ZN(n3165)
   );
IND4D0 U3741 ( .A1(n3130), .B1(n3165), .B2(n4345), .B3(n4317)
   , .ZN(n3121) );
```

```
  NR2D0 U3742 ( .A1(n4143), .A2(n4494), .ZN(n3217) );
  INR4D0 U3743 ( .A1(n3217), .B1(n4495), .B2(n4181), .B3(n3110)
     , .ZN(n3119) );
  ND4D0 U3744 ( .A1(n3241), .A2(n4335), .A3(n4094), .A4(n3111),
      .ZN(n3112) );
  OAI31D0 U3745 ( .A1(n3211), .A2(n3173), .A3(n3112), .B(n4480)
     , .ZN(n3118) );
  CKND2D0 U3746 ( .A1(n3114), .A2(n3113), .ZN(n3115) );
  ND4D0 U3747 ( .A1(n4126), .A2(n4369), .A3(n3192), .A4(n3115),
      .ZN(n3116) );
  OAI31D0 U3748 ( .A1(n3213), .A2(n3131), .A3(n3116), .B(n4452)
     , .ZN(n3117) );
  OAI211D0 U3749 ( .A1(n3119), .A2(n4428), .B(n3118), .C(n3117)
     , .ZN(n3120) );
  AOI221D0 U3750 ( .A1(n3122), .A2(n4342), .B1(n3121), .B2(
     n4342), .C(n3120),
        .ZN(n3139) );
  NR2D0 U3751 ( .A1(n4438), .A2(n3197), .ZN(n4346) );
  CKND2D0 U3752 ( .A1(n4334), .A2(n4368), .ZN(n3302) );
  INVD0 U3753 ( .I(n3302), .ZN(n3476) );
  AOI211D0 U3754 ( .A1(n4507), .A2(n3123), .B(n3237), .C(n3158)
     , .ZN(n3124) );
  ND4D0 U3755 ( .A1(n3235), .A2(n4346), .A3(n3476), .A4(n3124),
      .ZN(n3137) );
  NR3D0 U3756 ( .A1(n3574), .A2(n4090), .A3(n4473), .ZN(n3216)
     );
  NR2D0 U3757 ( .A1(n3126), .A2(n3125), .ZN(n3250) );
  AOI31D0 U3758 ( .A1(n3216), .A2(n3127), .A3(n3250), .B(n4517)
     , .ZN(n3136) );
  INR4D0 U3759 ( .A1(n3160), .B1(n3129), .B2(n3128), .B3(n3367)
     , .ZN(n3134) );
  INR4D0 U3760 ( .A1(n3283), .B1(n3132), .B2(n3131), .B3(n3130)
     , .ZN(n3133) );
  OAI22D0 U3761 ( .A1(n3134), .A2(n3243), .B1(n3133), .B2(n54),
      .ZN(n3135) );
  AOI211D0 U3762 ( .A1(n4291), .A2(n3137), .B(n3136), .C(n3135)
     , .ZN(n3138) );
  OAI22D0 U3763 ( .A1(n3139), .A2(n4148), .B1(n3138), .B2(n4355
     ), .ZN(n3262)
        );
  CKND2D0 U3764 ( .A1(n3323), .A2(n3140), .ZN(n4484) );
  IND4D0 U3765 ( .A1(n4484), .B1(n3142), .B2(n3141), .B3(n4388)
     , .ZN(n3155) );
  CKND2D0 U3766 ( .A1(n4320), .A2(n3192), .ZN(n3143) );
  NR4D0 U3767 ( .A1(n3161), .A2(n3205), .A3(n3144), .A4(n3143),
      .ZN(n3153) );
  NR2D0 U3768 ( .A1(n3574), .A2(n4494), .ZN(n3337) );
```

```
ND4D0 U3769 ( .A1(n3218), .A2(n3337), .A3(n4137), .A4(n3145),
    .ZN(n3146) );
OAI31D0 U3770 ( .A1(n3148), .A2(n3147), .A3(n3146), .B(n4291)
   , .ZN(n3152) );
NR2D0 U3771 ( .A1(n4403), .A2(n4232), .ZN(n3219) );
ND4D0 U3772 ( .A1(n4153), .A2(n3149), .A3(n3171), .A4(n3219),
    .ZN(n3150) );
OAI31D0 U3773 ( .A1(n4494), .A2(n3195), .A3(n3150), .B(n4452)
   , .ZN(n3151) );
OAI211D0 U3774 ( .A1(n3153), .A2(n3243), .B(n3152), .C(n3151)
   , .ZN(n3154) );
AOI221D0 U3775 ( .A1(n3215), .A2(n4289), .B1(n3155), .B2(
   n4342), .C(n3154),
      .ZN(n3170) );
IND4D0 U3776 ( .A1(n3158), .B1(n3157), .B2(n3156), .B3(n3219)
   , .ZN(n3168) );
NR4D0 U3777 ( .A1(n4170), .A2(n4309), .A3(n3225), .A4(n3173),
    .ZN(n3159) );
AOI21D0 U3778 ( .A1(n3160), .A2(n3159), .B(n4474), .ZN(n3167)
    );
ND4D0 U3779 ( .A1(n3193), .A2(n3235), .A3(n3241), .A4(n4390),
    .ZN(n3185) );
OAI31D0 U3780 ( .A1(n3162), .A2(n3161), .A3(n3185), .B(n4291)
   , .ZN(n3164) );
NR4D0 U3781 ( .A1(n4373), .A2(n4156), .A3(n3238), .A4(n3303),
    .ZN(n3163) );
AOI32D0 U3782 ( .A1(n3165), .A2(n3164), .A3(n3163), .B1(n3243
   ), .B2(n3164),
      .ZN(n3166) );
AOI211D0 U3783 ( .A1(n4289), .A2(n3168), .B(n3167), .C(n3166)
   , .ZN(n3169) );
OAI22D0 U3784 ( .A1(n3170), .A2(n4535), .B1(n3169), .B2(n3379
   ), .ZN(n3261)
      );
ND4D0 U3785 ( .A1(n3199), .A2(n3172), .A3(n3178), .A4(n3171),
    .ZN(n3180) );
AOI211D0 U3786 ( .A1(n3176), .A2(n3175), .B(n3174), .C(n3173)
   , .ZN(n3177) );
AOI31D0 U3787 ( .A1(n3178), .A2(n3177), .A3(n4389), .B(n4517)
   , .ZN(n3179) );
AOI221D0 U3788 ( .A1(n3302), .A2(n4452), .B1(n3180), .B2(
   n4452), .C(n3179),
      .ZN(n3191) );
ND4D0 U3789 ( .A1(n3246), .A2(n3182), .A3(n3181), .A4(n4317),
    .ZN(n3189) );
NR4D0 U3790 ( .A1(n4472), .A2(n4306), .A3(n3248), .A4(n3367),
    .ZN(n3183) );
CKND2D0 U3791 ( .A1(n3184), .A2(n3183), .ZN(n3188) );
```

```
NR4D0 U3792 ( .A1(n3541), .A2(n3319), .A3(n4494), .A4(n3185),
    .ZN(n3186) );
CKND2D0 U3793 ( .A1(n3198), .A2(n3186), .ZN(n3187) );
AOI222D0 U3794 ( .A1(n3189), .A2(n4492), .B1(n3188), .B2(
   n3709), .C1(n3187),
      .C2(n3707), .ZN(n3190) );
OAI22D0 U3795 ( .A1(n3191), .A2(n4325), .B1(n3190), .B2(n4476
   ), .ZN(n3260)
       );
NR2D0 U3796 ( .A1(n3319), .A2(n4124), .ZN(n3286) );
ND4D0 U3797 ( .A1(n3193), .A2(n3337), .A3(n3286), .A4(n3192),
    .ZN(n3210) );
NR4D0 U3798 ( .A1(n3197), .A2(n3196), .A3(n3195), .A4(n3194),
    .ZN(n3208) );
ND4D0 U3799 ( .A1(n3201), .A2(n3200), .A3(n3199), .A4(n3198),
    .ZN(n3202) );
OAI31D0 U3800 ( .A1(n3203), .A2(n3509), .A3(n3202), .B(n4289)
   , .ZN(n3207) );
OAI31D0 U3801 ( .A1(n3205), .A2(n4376), .A3(n3204), .B(n4480)
   , .ZN(n3206) );
OAI211D0 U3802 ( .A1(n3208), .A2(n4476), .B(n3207), .C(n3206)
   , .ZN(n3209) );
AOI221D0 U3803 ( .A1(n3211), .A2(n4452), .B1(n3210), .B2(
   n4452), .C(n3209),
      .ZN(n3258) );
NR2D0 U3804 ( .A1(n4373), .A2(n4403), .ZN(n4466) );
ND4D0 U3805 ( .A1(n3246), .A2(n4466), .A3(n4201), .A4(n4320),
    .ZN(n3212) );
NR4D0 U3806 ( .A1(n3215), .A2(n3214), .A3(n3213), .A4(n3212),
    .ZN(n3224) );
AN4D0 U3807 ( .A1(n3246), .A2(n3218), .A3(n3217), .A4(n3216),
    .Z(n3223) );
ND4D0 U3808 ( .A1(n3220), .A2(n3219), .A3(n4201), .A4(n4137),
    .ZN(n3221) );
NR2D0 U3809 ( .A1(n4246), .A2(n3221), .ZN(n3222) );
OAI222D0 U3810 ( .A1(n4476), .A2(n3224), .B1(n4517), .B2(
   n3223), .C1(n3243),
      .C2(n3222), .ZN(n3232) );
CKND2D0 U3811 ( .A1(n4145), .A2(n4261), .ZN(n4329) );
CKND2D0 U3812 ( .A1(n4316), .A2(n4329), .ZN(n4365) );
NR4D0 U3813 ( .A1(n4232), .A2(n4307), .A3(n4365), .A4(n3225),
    .ZN(n3230) );
CKND2D0 U3814 ( .A1(n3235), .A2(n3241), .ZN(n3226) );
OAI31D0 U3815 ( .A1(n3240), .A2(n4359), .A3(n3226), .B(n4342)
   , .ZN(n3229) );
ND4D0 U3816 ( .A1(n3392), .A2(n3241), .A3(n3320), .A4(n3249),
    .ZN(n3227) );
```

```
OAI31D0 U3817 ( .A1(n4310), .A2(n4513), .A3(n3227), .B(n4452)
   , .ZN(n3228) );
OAI211D0 U3818 ( .A1(n3230), .A2(n4428), .B(n3229), .C(n3228)
   , .ZN(n3231) );
AOI22D0 U3819 ( .A1(n3437), .A2(n3232), .B1(n3664), .B2(n3231
   ), .ZN(n3257)
      );
IINR4D0 U3820 ( .A1(n3251), .A2(n4176), .B1(n3234), .B2(n3233
   ), .ZN(n3245)
      );
ND4D0 U3821 ( .A1(n3235), .A2(n3241), .A3(n4317), .A4(n4506),
    .ZN(n3236) );
NR4D0 U3822 ( .A1(n4422), .A2(n3238), .A3(n3237), .A4(n3236),
    .ZN(n3244) );
IINR4D0 U3823 ( .A1(n3241), .A2(n4488), .B1(n3240), .B2(n3239
   ), .ZN(n3242)
      );
OAI222D0 U3824 ( .A1(n54), .A2(n3245), .B1(n4517), .B2(n3244)
   , .C1(n3243),
     .C2(n3242), .ZN(n3255) );
CKND2D0 U3825 ( .A1(n3246), .A2(n4317), .ZN(n3247) );
NR4D0 U3826 ( .A1(n4172), .A2(n4232), .A3(n3248), .A4(n3247),
    .ZN(n3253) );
AN4D0 U3827 ( .A1(n4234), .A2(n3251), .A3(n3250), .A4(n3249),
    .Z(n3252) );
OAI22D0 U3828 ( .A1(n3253), .A2(n4517), .B1(n3252), .B2(n4474
   ), .ZN(n3254)
      );
AOI22D0 U3829 ( .A1(n4492), .A2(n3255), .B1(n3707), .B2(n3254
   ), .ZN(n3256)
      );
OAI211D0 U3830 ( .A1(n3258), .A2(n4192), .B(n3257), .C(n3256)
   , .ZN(n3259) );
NR4D0 U3831 ( .A1(n3262), .A2(n3261), .A3(n3260), .A4(n3259),
    .ZN(n3263) );
OAI211D0 U3832 ( .A1(n3266), .A2(n3265), .B(n3264), .C(n3263)
   , .ZN(n3267) );
AOI211D0 U3833 ( .A1(n3852), .A2(n3269), .B(n3268), .C(n3267)
   , .ZN(n3594) );
CKND2D0 U3834 ( .A1(n4460), .A2(n4137), .ZN(n4442) );
CKND2D0 U3835 ( .A1(n3476), .A2(n3286), .ZN(n3487) );
NR2D0 U3836 ( .A1(n4374), .A2(n4473), .ZN(n3466) );
INVD0 U3837 ( .I(n3466), .ZN(n3285) );
NR4D0 U3838 ( .A1(n4183), .A2(n4442), .A3(n3487), .A4(n3285),
    .ZN(n3280) );
ND4D0 U3839 ( .A1(n3321), .A2(n4320), .A3(n4212), .A4(n3505),
    .ZN(n3400) );
CKND2D0 U3840 ( .A1(n4363), .A2(n4368), .ZN(n3579) );
```

```
  CKND2D0 U3841 ( .A1(n4418), .A2(n3572), .ZN(n3296) );
  INVD0 U3842 ( .I(n3296), .ZN(n3498) );
  IND4D0 U3843 ( .A1(n3579), .B1(n3270), .B2(n4126), .B3(n3498)
     , .ZN(n3275) );
  ND3D0 U3844 ( .A1(n4136), .A2(n4133), .A3(n4363), .ZN(n3413)
     );
  INVD0 U3845 ( .I(n3413), .ZN(n3318) );
  CKND2D0 U3846 ( .A1(n3443), .A2(n3271), .ZN(n3568) );
  ND3D0 U3847 ( .A1(n3397), .A2(n4146), .A3(n3568), .ZN(n3372)
     );
  INVD0 U3848 ( .I(n3372), .ZN(n3272) );
  AOI31D0 U3849 ( .A1(n3318), .A2(n3273), .A3(n3272), .B(n4517)
     , .ZN(n3274) );
  AOI221D0 U3850 ( .A1(n3400), .A2(n4480), .B1(n3275), .B2(
     n4480), .C(n3274),
        .ZN(n3279) );
  CKND2D0 U3851 ( .A1(n4201), .A2(n4194), .ZN(n3405) );
  CKND2D0 U3852 ( .A1(n4389), .A2(n3497), .ZN(n3555) );
  INVD0 U3853 ( .I(n3555), .ZN(n4197) );
  NR2D0 U3854 ( .A1(n3276), .A2(n4232), .ZN(n3575) );
  CKAN2D0 U3855 ( .A1(n3575), .A2(n4335), .Z(n3398) );
  INR3D0 U3856 ( .A1(n3286), .B1(n4076), .B2(n4306), .ZN(n3421)
      );
  ND4D0 U3857 ( .A1(n3499), .A2(n4197), .A3(n3398), .A4(n3421),
      .ZN(n3277) );
  OAI31D0 U3858 ( .A1(n4181), .A2(n3405), .A3(n3277), .B(n4452)
     , .ZN(n3278) );
  OAI211D0 U3859 ( .A1(n3280), .A2(n4428), .B(n3279), .C(n3278)
     , .ZN(n3592) );
  ND4D0 U3860 ( .A1(n3498), .A2(n4334), .A3(n3343), .A4(n3521),
      .ZN(n3292) );
  CKND2D0 U3861 ( .A1(n4362), .A2(n4465), .ZN(n4210) );
  CKND2D0 U3862 ( .A1(n4388), .A2(n4418), .ZN(n3281) );
  NR4D0 U3863 ( .A1(n4384), .A2(n4210), .A3(n3281), .A4(n3302),
      .ZN(n3282) );
  AOI31D0 U3864 ( .A1(n3282), .A2(n3343), .A3(n3530), .B(n4474)
     , .ZN(n3291) );
  NR2D0 U3865 ( .A1(n3451), .A2(n4170), .ZN(n3506) );
  INVD0 U3866 ( .I(n3506), .ZN(n3339) );
  NR2D0 U3867 ( .A1(n4161), .A2(n3604), .ZN(n3408) );
  ND4D0 U3868 ( .A1(n3356), .A2(n4234), .A3(n3283), .A4(n3408),
      .ZN(n3284) );
  OAI31D0 U3869 ( .A1(n3285), .A2(n3339), .A3(n3284), .B(n4342)
     , .ZN(n3289) );
  CKND2D0 U3870 ( .A1(n4316), .A2(n3530), .ZN(n3287) );
  CKND2D0 U3871 ( .A1(n3286), .A2(n3498), .ZN(n3354) );
  NR4D0 U3872 ( .A1(n3542), .A2(n4250), .A3(n3287), .A4(n3354),
      .ZN(n3288) );
```

```
AOI32D0 U3873 ( .A1(n3398), .A2(n3289), .A3(n3288), .B1(n3243
   ), .B2(n3289),
      .ZN(n3290) );
AOI211D0 U3874 ( .A1(n4291), .A2(n3292), .B(n3291), .C(n3290)
   , .ZN(n3317) );
CKND2D0 U3875 ( .A1(n4317), .A2(n3295), .ZN(n3511) );
NR3D0 U3876 ( .A1(n4310), .A2(n4181), .A3(n3540), .ZN(n3369)
   );
ND4D0 U3877 ( .A1(n4286), .A2(n3369), .A3(n4367), .A4(n3323),
    .ZN(n3345) );
CKND2D0 U3878 ( .A1(n4362), .A2(n4157), .ZN(n3352) );
IND2D0 U3879 ( .A1(n3352), .B1(n4465), .ZN(n3550) );
NR4D0 U3880 ( .A1(n3405), .A2(n3511), .A3(n3345), .A4(n3550),
    .ZN(n3294) );
CKND2D0 U3881 ( .A1(n4294), .A2(n3569), .ZN(n3493) );
INR2D0 U3882 ( .A1(n3356), .B1(n3493), .ZN(n3423) );
ND3D0 U3883 ( .A1(n4136), .A2(n3505), .A3(n4194), .ZN(n3580)
   );
INR4D0 U3884 ( .A1(n3423), .B1(n4090), .B2(n3307), .B3(n3580)
   , .ZN(n3293) );
OAI22D0 U3885 ( .A1(n3294), .A2(n4428), .B1(n3293), .B2(n4474
   ), .ZN(n3300)
      );
CKND2D0 U3886 ( .A1(n4146), .A2(n3295), .ZN(n3513) );
NR4D0 U3887 ( .A1(n4210), .A2(n4436), .A3(n3296), .A4(n3513),
    .ZN(n3298) );
ND3D0 U3888 ( .A1(n3575), .A2(n3408), .A3(n4367), .ZN(n3322)
   );
CKND2D0 U3889 ( .A1(n3305), .A2(n4197), .ZN(n3358) );
NR4D0 U3890 ( .A1(n4310), .A2(n3322), .A3(n3358), .A4(n3352),
    .ZN(n3297) );
OAI22D0 U3891 ( .A1(n3298), .A2(n4517), .B1(n3297), .B2(n3243
   ), .ZN(n3299)
      );
OAI21D0 U3892 ( .A1(n3300), .A2(n3299), .B(n3850), .ZN(n3316)
    );
NR4D0 U3893 ( .A1(n4161), .A2(n4181), .A3(n3456), .A4(n3638),
    .ZN(n3301) );
NR2D0 U3894 ( .A1(n3555), .A2(n3339), .ZN(n3576) );
AOI31D0 U3895 ( .A1(n3301), .A2(n3392), .A3(n3576), .B(n4517)
   , .ZN(n3314) );
NR4D0 U3896 ( .A1(n4373), .A2(n3303), .A3(n3302), .A4(n3405),
    .ZN(n3304) );
AOI21D0 U3897 ( .A1(n3318), .A2(n3304), .B(n4476), .ZN(n3313)
    );
INVD0 U3898 ( .I(n3305), .ZN(n3409) );
CKND2D0 U3899 ( .A1(n4176), .A2(n4316), .ZN(n3366) );
CKND2D0 U3900 ( .A1(n3392), .A2(n3575), .ZN(n3412) );
```

```
NR4D0 U3901 ( .A1(n4473), .A2(n3409), .A3(n3366), .A4(n3412),
    .ZN(n3310) );
CKND2D0 U3902 ( .A1(n3343), .A2(n4333), .ZN(n3306) );
NR4D0 U3903 ( .A1(n3308), .A2(n3307), .A3(n3306), .A4(n3511),
    .ZN(n3309) );
OAI22D0 U3904 ( .A1(n3310), .A2(n3243), .B1(n3309), .B2(n54),
    .ZN(n3312) );
OAI31D0 U3905 ( .A1(n3314), .A2(n3313), .A3(n3312), .B(n3311)
   , .ZN(n3315) );
OAI211D0 U3906 ( .A1(n3317), .A2(n4355), .B(n3316), .C(n3315)
   , .ZN(n3591) );
CKND2D0 U3907 ( .A1(n4126), .A2(n3318), .ZN(n3508) );
NR2D0 U3908 ( .A1(n3574), .A2(n4426), .ZN(n4175) );
CKND2D0 U3909 ( .A1(n4175), .A2(n3323), .ZN(n3577) );
NR4D0 U3910 ( .A1(n3319), .A2(n3395), .A3(n3508), .A4(n3577),
    .ZN(n3331) );
IND4D0 U3911 ( .A1(n3322), .B1(n3321), .B2(n3320), .B3(n4330)
   , .ZN(n3327) );
CKND2D0 U3912 ( .A1(n4367), .A2(n3323), .ZN(n3501) );
CKND2D0 U3913 ( .A1(n3343), .A2(n4133), .ZN(n3387) );
NR2D0 U3914 ( .A1(n3501), .A2(n3387), .ZN(n3494) );
CKND2D0 U3915 ( .A1(n3396), .A2(n4418), .ZN(n3578) );
INVD0 U3916 ( .I(n3578), .ZN(n3325) );
AOI31D0 U3917 ( .A1(n3494), .A2(n3325), .A3(n3324), .B(n4517)
   , .ZN(n3326) );
AOI221D0 U3918 ( .A1(n3540), .A2(n4291), .B1(n3327), .B2(
   n4291), .C(n3326),
      .ZN(n3330) );
ND3D0 U3919 ( .A1(n4364), .A2(n4136), .A3(n4320), .ZN(n3533)
   );
ND3D0 U3920 ( .A1(n4212), .A2(n4157), .A3(n4390), .ZN(n3328)
   );
OAI31D0 U3921 ( .A1(n3533), .A2(n3387), .A3(n3328), .B(n4480)
   , .ZN(n3329) );
OAI211D0 U3922 ( .A1(n3331), .A2(n54), .B(n3330), .C(n3329),
   .ZN(n3385) );
NR2D0 U3923 ( .A1(n3333), .A2(n3332), .ZN(n3334) );
NR4D0 U3924 ( .A1(n3537), .A2(n3334), .A3(n4436), .A4(n3366),
    .ZN(n3336) );
CKND2D0 U3925 ( .A1(n3335), .A2(n4081), .ZN(n4211) );
ND4D0 U3926 ( .A1(n3337), .A2(n3336), .A3(n4265), .A4(n4211),
    .ZN(n3342) );
NR2D0 U3927 ( .A1(n4426), .A2(n3338), .ZN(n3445) );
CKAN2D0 U3928 ( .A1(n4107), .A2(n4211), .Z(n4220) );
INVD0 U3929 ( .I(n4232), .ZN(n4391) );
ND3D0 U3930 ( .A1(n4176), .A2(n4109), .A3(n4391), .ZN(n3424)
   );
```

```
INR4D0 U3931 ( .A1(n4220), .B1(n3473), .B2(n3339), .B3(n3424)
   , .ZN(n3340) );
ND4D0 U3932 ( .A1(n3445), .A2(n3340), .A3(n4334), .A4(n4078),
    .ZN(n3341) );
AOI22D0 U3933 ( .A1(n4480), .A2(n3342), .B1(n4342), .B2(n3341
   ), .ZN(n3348)
      );
CKND2D0 U3934 ( .A1(n4336), .A2(n3476), .ZN(n3401) );
INVD0 U3935 ( .I(n3511), .ZN(n3489) );
NR2D0 U3936 ( .A1(n4263), .A2(n4018), .ZN(n4516) );
INVD0 U3937 ( .I(n4516), .ZN(n3812) );
ND4D0 U3938 ( .A1(n3486), .A2(n3489), .A3(n3812), .A4(n3343),
    .ZN(n3344) );
OAI31D0 U3939 ( .A1(n3401), .A2(n3509), .A3(n3344), .B(n4291)
   , .ZN(n3347) );
OAI31D0 U3940 ( .A1(n4413), .A2(n4173), .A3(n3345), .B(n4452)
   , .ZN(n3346) );
AOI31D0 U3941 ( .A1(n3348), .A2(n3347), .A3(n3346), .B(n3777)
   , .ZN(n3384) );
NR2D0 U3942 ( .A1(n3349), .A2(n3822), .ZN(n4443) );
INR2D0 U3943 ( .A1(n3350), .B1(n4443), .ZN(n3488) );
ND4D0 U3944 ( .A1(n3351), .A2(n4286), .A3(n3575), .A4(n3488),
    .ZN(n3363) );
NR2D0 U3945 ( .A1(n3353), .A2(n3352), .ZN(n3426) );
NR4D0 U3946 ( .A1(n4373), .A2(n4090), .A3(n3354), .A4(n3387),
    .ZN(n3355) );
AOI31D0 U3947 ( .A1(n3426), .A2(n3356), .A3(n3355), .B(n54),
   .ZN(n3362) );
OAI21D0 U3948 ( .A1(n4114), .A2(n3357), .B(n4362), .ZN(n4424)
    );
OAI31D0 U3949 ( .A1(n4403), .A2(n4424), .A3(n3358), .B(n4291)
   , .ZN(n3360) );
CKND2D0 U3950 ( .A1(n4391), .A2(n4507), .ZN(n4394) );
NR4D0 U3951 ( .A1(n3451), .A2(n4394), .A3(n4210), .A4(n3533),
    .ZN(n3359) );
AOI32D0 U3952 ( .A1(n3369), .A2(n3360), .A3(n3359), .B1(n4517
   ), .B2(n3360),
      .ZN(n3361) );
AOI211D0 U3953 ( .A1(n4480), .A2(n3363), .B(n3362), .C(n3361)
   , .ZN(n3382) );
CKND2D0 U3954 ( .A1(n4098), .A2(n3406), .ZN(n3365) );
ND4D0 U3955 ( .A1(n3421), .A2(n3494), .A3(n4335), .A4(n3365),
    .ZN(n3378) );
INR4D0 U3956 ( .A1(n3369), .B1(n3368), .B2(n3367), .B3(n3366)
   , .ZN(n3370) );
AOI31D0 U3957 ( .A1(n3408), .A2(n3370), .A3(n4388), .B(n4474)
   , .ZN(n3377) );
```

```
ND3D0 U3958 ( .A1(n3426), .A2(n3396), .A3(n4368), .ZN(n3371)
   );
OAI31D0 U3959 ( .A1(n3372), .A2(n4394), .A3(n3371), .B(n4342)
   , .ZN(n3375) );
CKND2D0 U3960 ( .A1(n4447), .A2(n3373), .ZN(n3551) );
NR4D0 U3961 ( .A1(n3540), .A2(n4384), .A3(n3578), .A4(n3551),
    .ZN(n3374) );
AOI32D0 U3962 ( .A1(n3466), .A2(n3375), .A3(n3374), .B1(n3243
   ), .B2(n3375),
      .ZN(n3376) );
AOI211D0 U3963 ( .A1(n4291), .A2(n3378), .B(n3377), .C(n3376)
   , .ZN(n3380) );
OAI22D0 U3964 ( .A1(n3382), .A2(n3381), .B1(n3380), .B2(n3379
   ), .ZN(n3383)
       );
AOI211D0 U3965 ( .A1(n3386), .A2(n3385), .B(n3384), .C(n3383)
   , .ZN(n3589) );
NR4D0 U3966 ( .A1(n4124), .A2(n4250), .A3(n4219), .A4(n3387),
    .ZN(n3388) );
AOI31D0 U3967 ( .A1(n3498), .A2(n3506), .A3(n3388), .B(n4476)
   , .ZN(n3438) );
CKND2D0 U3968 ( .A1(n4212), .A2(n3505), .ZN(n3389) );
AOI211D0 U3969 ( .A1(n3391), .A2(n3390), .B(n3451), .C(n3389)
   , .ZN(n3404) );
NR2D0 U3970 ( .A1(n3443), .A2(n4145), .ZN(n3393) );
OAI211D0 U3971 ( .A1(n3393), .A2(n3822), .B(n3392), .C(n3489)
   , .ZN(n3394) );
OAI31D0 U3972 ( .A1(n3395), .A2(n3455), .A3(n3394), .B(n4289)
   , .ZN(n3403) );
ND4D0 U3973 ( .A1(n3398), .A2(n4367), .A3(n3397), .A4(n3396),
    .ZN(n3399) );
OAI31D0 U3974 ( .A1(n3401), .A2(n3400), .A3(n3399), .B(n4452)
   , .ZN(n3402) );
OAI211D0 U3975 ( .A1(n3404), .A2(n3243), .B(n3403), .C(n3402)
   , .ZN(n3436) );
AOI211D0 U3976 ( .A1(n4098), .A2(n3406), .B(n4210), .C(n3405)
   , .ZN(n3407) );
ND4D0 U3977 ( .A1(n3408), .A2(n3407), .A3(n4316), .A4(n3530),
    .ZN(n3419) );
NR2D0 U3978 ( .A1(n4373), .A2(n4306), .ZN(n4169) );
NR3D0 U3979 ( .A1(n3574), .A2(n3456), .A3(n3409), .ZN(n3410)
   );
AOI31D0 U3980 ( .A1(n4286), .A2(n4169), .A3(n3410), .B(n4476)
   , .ZN(n3418) );
ND4D0 U3981 ( .A1(n4447), .A2(n4467), .A3(n4369), .A4(n4390),
    .ZN(n3411) );
OAI31D0 U3982 ( .A1(n3493), .A2(n3412), .A3(n3411), .B(n4289)
   , .ZN(n3416) );
```

```
AOI211D0 U3983 ( .A1(n3414), .A2(n4349), .B(n3413), .C(n3577)
   , .ZN(n3415) );
AOI32D0 U3984 ( .A1(n3506), .A2(n3416), .A3(n3415), .B1(n4474
   ), .B2(n3416),
      .ZN(n3417) );
AOI211D0 U3985 ( .A1(n4480), .A2(n3419), .B(n3418), .C(n3417)
   , .ZN(n3434) );
ND4D0 U3986 ( .A1(n3422), .A2(n3421), .A3(n4194), .A4(n3420),
    .ZN(n3432) );
AOI31D0 U3987 ( .A1(n3423), .A2(n3476), .A3(n3568), .B(n3243)
   , .ZN(n3431) );
INR4D0 U3988 ( .A1(n3568), .B1(n3540), .B2(n3425), .B3(n3424)
   , .ZN(n3429) );
INVD0 U3989 ( .I(n3577), .ZN(n3571) );
AN4D0 U3990 ( .A1(n4133), .A2(n3427), .A3(n3571), .A4(n3426),
    .Z(n3428) );
OAI22D0 U3991 ( .A1(n3429), .A2(n54), .B1(n3428), .B2(n4476),
    .ZN(n3430) );
AOI211D0 U3992 ( .A1(n4342), .A2(n3432), .B(n3431), .C(n3430)
   , .ZN(n3433) );
OAI22D0 U3993 ( .A1(n3434), .A2(n4190), .B1(n3433), .B2(n4408
   ), .ZN(n3435)
      );
AOI221D0 U3994 ( .A1(n3438), .A2(n3437), .B1(n3436), .B2(
   n3437), .C(n3435),
      .ZN(n3588) );
NR2D0 U3995 ( .A1(n3439), .A2(counter[2]), .ZN(n3539) );
NR2D0 U3996 ( .A1(n3441), .A2(n3440), .ZN(n3471) );
OA22D0 U3997 ( .A1(n3442), .A2(n3539), .B1(n3544), .B2(n3471)
   , .Z(n3448) );
NR3D0 U3998 ( .A1(n3449), .A2(n3443), .A3(n4145), .ZN(n3464)
   );
NR2D0 U3999 ( .A1(n4124), .A2(n3444), .ZN(n4267) );
IND4D0 U4000 ( .A1(n3464), .B1(n4267), .B2(n3445), .B3(n4320)
   , .ZN(n3446) );
NR4D0 U4001 ( .A1(n3448), .A2(n3550), .A3(n3447), .A4(n3446),
    .ZN(n3484) );
OAI211D0 U4002 ( .A1(n3528), .A2(n3449), .B(n4137), .C(n4121)
   , .ZN(n3450) );
NR4D0 U4003 ( .A1(n3452), .A2(n3451), .A3(n3487), .A4(n3450),
    .ZN(n3454) );
ND4D0 U4004 ( .A1(n3553), .A2(n3454), .A3(n3453), .A4(n4329),
    .ZN(n3468) );
NR3D0 U4005 ( .A1(n3457), .A2(n3456), .A3(n3455), .ZN(n3510)
   );
INR3D0 U4006 ( .A1(n4078), .B1(n3459), .B2(n3458), .ZN(n3475)
    );
```

```
OAI211D0 U4007 ( .A1(n3462), .A2(n3461), .B(n3475), .C(n3460)
   , .ZN(n3463) );
NR4D0 U4008 ( .A1(n4203), .A2(n3464), .A3(n3511), .A4(n3463),
    .ZN(n3465) );
AOI31D0 U4009 ( .A1(n3510), .A2(n3466), .A3(n3465), .B(n4428)
   , .ZN(n3467) );
AOI221D0 U4010 ( .A1(n3469), .A2(n4480), .B1(n3468), .B2(
   n4480), .C(n3467),
      .ZN(n3483) );
NR2D0 U4011 ( .A1(n4113), .A2(n3470), .ZN(n4101) );
INVD0 U4012 ( .I(n4101), .ZN(n4260) );
NR2D0 U4013 ( .A1(n4093), .A2(n3471), .ZN(n3552) );
OAI31D0 U4014 ( .A1(n3956), .A2(n3472), .A3(n4260), .B(n3552)
   , .ZN(n3481) );
IND2D0 U4015 ( .A1(n3473), .B1(n4220), .ZN(n3480) );
CKND2D0 U4016 ( .A1(n4223), .A2(n3474), .ZN(n3543) );
INVD0 U4017 ( .I(n3543), .ZN(n3478) );
OAI211D0 U4018 ( .A1(n3478), .A2(n3477), .B(n3476), .C(n3475)
   , .ZN(n3479) );
OAI31D0 U4019 ( .A1(n3481), .A2(n3480), .A3(n3479), .B(n4289)
   , .ZN(n3482) );
OAI211D0 U4020 ( .A1(n3484), .A2(n54), .B(n3483), .C(n3482),
   .ZN(n3566) );
ND4D0 U4021 ( .A1(n3486), .A2(n4197), .A3(n3485), .A4(n4334),
    .ZN(n3492) );
INVD0 U4022 ( .I(n3487), .ZN(n3490) );
AOI31D0 U4023 ( .A1(n3490), .A2(n3489), .A3(n3488), .B(n4517)
   , .ZN(n3491) );
AOI221D0 U4024 ( .A1(n3493), .A2(n4452), .B1(n3492), .B2(
   n4452), .C(n3491),
      .ZN(n3504) );
INVD0 U4025 ( .I(n3494), .ZN(n3496) );
OAI31D0 U4026 ( .A1(n3496), .A2(n3495), .A3(n3551), .B(n4480)
   , .ZN(n3503) );
ND4D0 U4027 ( .A1(n3499), .A2(n3498), .A3(n4194), .A4(n3497),
    .ZN(n3500) );
OAI31D0 U4028 ( .A1(n4438), .A2(n3501), .A3(n3500), .B(n4291)
   , .ZN(n3502) );
AOI31D0 U4029 ( .A1(n3504), .A2(n3503), .A3(n3502), .B(n4527)
   , .ZN(n3565) );
CKND2D0 U4030 ( .A1(n4169), .A2(n3505), .ZN(n4440) );
ND4D0 U4031 ( .A1(n3507), .A2(n3506), .A3(n4362), .A4(n3568),
    .ZN(n3520) );
INR4D0 U4032 ( .A1(n3510), .B1(n3509), .B2(n4219), .B3(n3508)
   , .ZN(n3518) );
OAI31D0 U4033 ( .A1(n4210), .A2(n3512), .A3(n3511), .B(n4291)
   , .ZN(n3517) );
CKND2D0 U4034 ( .A1(n4234), .A2(n4137), .ZN(n3515) );
```

```
IND4D0 U4035 ( .A1(n3513), .B1(n3569), .B2(n4109), .B3(n4157)
   , .ZN(n3514) );
OAI31D0 U4036 ( .A1(n3577), .A2(n3515), .A3(n3514), .B(n4342)
   , .ZN(n3516) );
OAI211D0 U4037 ( .A1(n3518), .A2(n3243), .B(n3517), .C(n3516)
   , .ZN(n3519) );
AOI221D0 U4038 ( .A1(n4440), .A2(n4452), .B1(n3520), .B2(
   n4452), .C(n3519),
      .ZN(n3563) );
OAI211D0 U4039 ( .A1(n4018), .A2(n4223), .B(n3521), .C(n4329)
   , .ZN(n3522) );
NR4D0 U4040 ( .A1(n3524), .A2(n3523), .A3(n3580), .A4(n3522),
    .ZN(n3525) );
ND4D0 U4041 ( .A1(n3526), .A2(n3525), .A3(n4121), .A4(n4078),
    .ZN(n3561) );
AOI221D0 U4042 ( .A1(n3529), .A2(n3528), .B1(n1513), .B2(
   n3528), .C(n3527),
      .ZN(n3534) );
OAI31D0 U4043 ( .A1(counter[6]), .A2(n3531), .A3(n3737), .B(
   n3530), .ZN(
      n3532) );
AOI211D0 U4044 ( .A1(counter[11]), .A2(n3534), .B(n3533), .C(
   n3532), .ZN(
      n3536) );
CKND2D0 U4045 ( .A1(n4079), .A2(n4117), .ZN(n4077) );
AOI31D0 U4046 ( .A1(n3536), .A2(n3535), .A3(n4077), .B(n3243)
   , .ZN(n3560) );
AOI211D0 U4047 ( .A1(n3539), .A2(n3538), .B(n4310), .C(n3537)
   , .ZN(n3558) );
NR2D0 U4048 ( .A1(n3541), .A2(n3540), .ZN(n3547) );
AOI211D0 U4049 ( .A1(n4256), .A2(n3543), .B(n3542), .C(n4075)
   , .ZN(n3546) );
CKND2D0 U4050 ( .A1(n4145), .A2(n3544), .ZN(n3545) );
ND4D0 U4051 ( .A1(n3548), .A2(n3547), .A3(n3546), .A4(n3545),
    .ZN(n3549) );
OAI31D0 U4052 ( .A1(n3551), .A2(n3550), .A3(n3549), .B(n4342)
   , .ZN(n3557) );
IND4D0 U4053 ( .A1(n4210), .B1(n4447), .B2(n3553), .B3(n3552)
   , .ZN(n3554) );
NR4D0 U4054 ( .A1(n4413), .A2(n4089), .A3(n3555), .A4(n3554),
    .ZN(n3556) );
AOI32D0 U4055 ( .A1(n3558), .A2(n3557), .A3(n3556), .B1(n4474
   ), .B2(n3557),
      .ZN(n3559) );
AOI211D0 U4056 ( .A1(n4291), .A2(n3561), .B(n3560), .C(n3559)
   , .ZN(n3562) );
OAI22D0 U4057 ( .A1(n3563), .A2(n4148), .B1(n3562), .B2(n4277
   ), .ZN(n3564)
```

```
        );
AOI211D0 U4058 ( .A1(n3567), .A2(n3566), .B(n3565), .C(n3564)
   , .ZN(n3587) );
AOI31D0 U4059 ( .A1(n3570), .A2(n3569), .A3(n3568), .B(n54),
   .ZN(n3585) );
AOI31D0 U4060 ( .A1(n3573), .A2(n3572), .A3(n3571), .B(n4428)
   , .ZN(n3584) );
NR2D0 U4061 ( .A1(n3574), .A2(n4310), .ZN(n4360) );
AN4D0 U4062 ( .A1(n4109), .A2(n4360), .A3(n3576), .A4(n3575),
    .Z(n3582) );
NR4D0 U4063 ( .A1(n3580), .A2(n3579), .A3(n3578), .A4(n3577),
    .ZN(n3581) );
OAI22D0 U4064 ( .A1(n3582), .A2(n3243), .B1(n3581), .B2(n4517
   ), .ZN(n3583)
        );
OAI31D0 U4065 ( .A1(n3585), .A2(n3584), .A3(n3583), .B(n4521)
   , .ZN(n3586) );
ND4D0 U4066 ( .A1(n3589), .A2(n3588), .A3(n3587), .A4(n3586),
    .ZN(n3590) );
AOI211D0 U4067 ( .A1(n3709), .A2(n3592), .B(n3591), .C(n3590)
   , .ZN(n3593) );
OAI222D0 U4068 ( .A1(\q_out[1]_BAR ), .A2(n4545), .B1(n4544),
    .B2(n3594),
      .C1(n4542), .C2(n3593), .ZN(n4813) );
NR2D0 U4069 ( .A1(n4024), .A2(n3595), .ZN(n3831) );
NR2D0 U4070 ( .A1(n3832), .A2(n3655), .ZN(n3840) );
INVD0 U4071 ( .I(n3840), .ZN(n4057) );
NR4D0 U4072 ( .A1(n3831), .A2(n3734), .A3(n4021), .A4(n4057),
    .ZN(n3596) );
ND4D0 U4073 ( .A1(n3597), .A2(n3794), .A3(n3928), .A4(n3596),
    .ZN(n3603) );
CKND2D0 U4074 ( .A1(n3737), .A2(n3831), .ZN(n3805) );
INVD0 U4075 ( .I(n3805), .ZN(n3970) );
NR2D0 U4076 ( .A1(n4044), .A2(n3970), .ZN(n3979) );
INR4D0 U4077 ( .A1(n3700), .B1(n3656), .B2(n3615), .B3(n3598)
   , .ZN(n3599) );
ND4D0 U4078 ( .A1(n3601), .A2(n3979), .A3(n3600), .A4(n3599),
    .ZN(n3602) );
AOI22D0 U4079 ( .A1(n3933), .A2(n3603), .B1(n3844), .B2(n3602
   ), .ZN(n3613)
        );
ND3D0 U4080 ( .A1(n3718), .A2(n3979), .A3(n3986), .ZN(n4029)
   );
NR3D0 U4081 ( .A1(n3605), .A2(n3604), .A3(n3786), .ZN(n3717)
   );
CKND2D0 U4082 ( .A1(n3657), .A2(n3717), .ZN(n4056) );
INVD0 U4083 ( .I(n4056), .ZN(n3873) );
```

```
ND4D0 U4084 ( .A1(n3606), .A2(n3873), .A3(n3883), .A4(n4049),
    .ZN(n3607) );
OAI21D0 U4085 ( .A1(n4029), .A2(n3607), .B(n3769), .ZN(n3612)
    );
NR2D0 U4086 ( .A1(n3608), .A2(n3870), .ZN(n3725) );
INVD0 U4087 ( .I(n3725), .ZN(n3689) );
ND4D0 U4088 ( .A1(n3794), .A2(n4023), .A3(n3788), .A4(n3609),
    .ZN(n3981) );
INVD0 U4089 ( .I(n3931), .ZN(n3853) );
NR2D0 U4090 ( .A1(n3981), .A2(n3853), .ZN(n3701) );
NR3D0 U4091 ( .A1(n3787), .A2(n3786), .A3(n3898), .ZN(n3814)
   );
ND4D0 U4092 ( .A1(n3979), .A2(n3701), .A3(n3814), .A4(n3789),
    .ZN(n3610) );
OAI31D0 U4093 ( .A1(n3689), .A2(n3967), .A3(n3610), .B(n4000)
   , .ZN(n3611) );
AOI31D0 U4094 ( .A1(n3613), .A2(n3612), .A3(n3611), .B(n4192)
   , .ZN(n4073) );
NR3D0 U4095 ( .A1(n3614), .A2(n3970), .A3(n3740), .ZN(n3908)
   );
CKND2D0 U4096 ( .A1(n3745), .A2(n3637), .ZN(n4008) );
NR3D0 U4097 ( .A1(n3646), .A2(n4008), .A3(n3716), .ZN(n3766)
   );
INVD0 U4098 ( .I(n3766), .ZN(n3796) );
NR3D0 U4099 ( .A1(n3890), .A2(n3615), .A3(n3796), .ZN(n3678)
   );
ND4D0 U4100 ( .A1(n3616), .A2(n3928), .A3(n3908), .A4(n3678),
    .ZN(n3626) );
NR2D0 U4101 ( .A1(n4057), .A2(n4008), .ZN(n3628) );
AN3D0 U4102 ( .A1(n3794), .A2(n3814), .A3(n3628), .Z(n3618) )
   ;
AOI31D0 U4103 ( .A1(n3785), .A2(n3618), .A3(n3617), .B(n4038)
   , .ZN(n3625) );
ND3D0 U4104 ( .A1(n4047), .A2(n3619), .A3(n4023), .ZN(n4054)
   );
OAI31D0 U4105 ( .A1(n4056), .A2(n3620), .A3(n4054), .B(n4000)
   , .ZN(n3623) );
CKND2D0 U4106 ( .A1(n4047), .A2(n3621), .ZN(n3640) );
NR2D0 U4107 ( .A1(n3649), .A2(n3742), .ZN(n3714) );
INVD0 U4108 ( .I(n3714), .ZN(n3635) );
NR4D0 U4109 ( .A1(n3967), .A2(n3858), .A3(n3640), .A4(n3635),
    .ZN(n3622) );
AOI32D0 U4110 ( .A1(n3628), .A2(n3623), .A3(n3622), .B1(n4010
   ), .B2(n3623),
      .ZN(n3624) );
AOI211D0 U4111 ( .A1(n3844), .A2(n3626), .B(n3625), .C(n3624)
   , .ZN(n3670) );
NR2D0 U4112 ( .A1(n3890), .A2(n3632), .ZN(n3675) );
```

```
INR4D0 U4113 ( .A1(n3628), .B1(n3891), .B2(n3831), .B3(n3627)
   , .ZN(n3629) );
AOI31D0 U4114 ( .A1(n3675), .A2(n3923), .A3(n3629), .B(n4038)
   , .ZN(n3645) );
CKND2D0 U4115 ( .A1(n3630), .A2(n3813), .ZN(n3767) );
CKND2D0 U4116 ( .A1(n4016), .A2(n3631), .ZN(n4053) );
NR4D0 U4117 ( .A1(n3890), .A2(n3632), .A3(n3767), .A4(n4053),
    .ZN(n3634) );
AOI31D0 U4118 ( .A1(n3725), .A2(n3634), .A3(n3633), .B(n4045)
   , .ZN(n3644) );
NR2D0 U4119 ( .A1(n4029), .A2(n3635), .ZN(n3694) );
INR4D0 U4120 ( .A1(n3694), .B1(n3798), .B2(n3646), .B3(n3640)
   , .ZN(n3642) );
NR3D0 U4121 ( .A1(n3636), .A2(n3918), .A3(n3650), .ZN(n3682)
   );
CKND2D0 U4122 ( .A1(n3927), .A2(n3637), .ZN(n4035) );
NR3D0 U4123 ( .A1(n3639), .A2(n4035), .A3(n3638), .ZN(n3839)
   );
INVD0 U4124 ( .I(n3640), .ZN(n3922) );
NR2D0 U4125 ( .A1(n4042), .A2(n3651), .ZN(n3693) );
AN4D0 U4126 ( .A1(n3682), .A2(n3839), .A3(n3922), .A4(n3693),
    .Z(n3641) );
OAI22D0 U4127 ( .A1(n3642), .A2(n4010), .B1(n3641), .B2(n4059
   ), .ZN(n3643)
      );
OAI31D0 U4128 ( .A1(n3645), .A2(n3644), .A3(n3643), .B(n4490)
   , .ZN(n3669) );
INR4D0 U4129 ( .A1(n4016), .B1(n4008), .B2(n3646), .B3(n3832)
   , .ZN(n3887) );
INR4D0 U4130 ( .A1(n3647), .B1(n3751), .B2(n3649), .B3(n3734)
   , .ZN(n3648) );
AOI31D0 U4131 ( .A1(n3887), .A2(n3717), .A3(n3648), .B(n4010)
   , .ZN(n3667) );
INR2D0 U4132 ( .A1(n3908), .B1(n3958), .ZN(n4040) );
NR4D0 U4133 ( .A1(n3651), .A2(n3918), .A3(n3650), .A4(n3649),
    .ZN(n3652) );
AOI31D0 U4134 ( .A1(n4040), .A2(n3653), .A3(n3652), .B(n4059)
   , .ZN(n3666) );
CKND2D0 U4135 ( .A1(n3814), .A2(n3682), .ZN(n3691) );
NR4D0 U4136 ( .A1(n3656), .A2(n3655), .A3(n3654), .A4(n3691),
    .ZN(n3663) );
ND4D0 U4137 ( .A1(n3658), .A2(n3923), .A3(n3839), .A4(n3657),
    .ZN(n3659) );
OAI31D0 U4138 ( .A1(n4028), .A2(n3660), .A3(n3659), .B(n3844)
   , .ZN(n3662) );
AOI32D0 U4139 ( .A1(n3663), .A2(n3662), .A3(n3661), .B1(n4038
   ), .B2(n3662),
      .ZN(n3665) );
```

```
OAI31D0 U4140 ( .A1(n3667), .A2(n3666), .A3(n3665), .B(n3664)
   , .ZN(n3668) );
OAI211D0 U4141 ( .A1(n3670), .A2(n4148), .B(n3669), .C(n3668)
   , .ZN(n4072) );
ND3D0 U4142 ( .A1(n3714), .A2(n3671), .A3(n3692), .ZN(n4041)
   );
CKND2D0 U4143 ( .A1(n3700), .A2(n3672), .ZN(n4007) );
NR4D0 U4144 ( .A1(n3674), .A2(n3673), .A3(n4041), .A4(n4007),
    .ZN(n3688) );
INVD0 U4145 ( .I(n3981), .ZN(n3712) );
INR2D0 U4146 ( .A1(n3675), .B1(n3740), .ZN(n3799) );
ND4D0 U4147 ( .A1(n3856), .A2(n3712), .A3(n3799), .A4(n3676),
    .ZN(n3680) );
IND4D0 U4148 ( .A1(n4041), .B1(n3979), .B2(n3678), .B3(n3677)
   , .ZN(n3679) );
AOI22D0 U4149 ( .A1(n3933), .A2(n3680), .B1(n4000), .B2(n3679
   ), .ZN(n3687)
      );
ND4D0 U4150 ( .A1(n3683), .A2(n4047), .A3(n3682), .A4(n3681),
    .ZN(n3684) );
OAI31D0 U4151 ( .A1(n3685), .A2(n4058), .A3(n3684), .B(n3769)
   , .ZN(n3686) );
OAI211D0 U4152 ( .A1(n3688), .A2(n4045), .B(n3687), .C(n3686)
   , .ZN(n3708) );
NR2D0 U4153 ( .A1(n3690), .A2(n3689), .ZN(n3711) );
INVD0 U4154 ( .I(n3711), .ZN(n3815) );
INR4D0 U4155 ( .A1(n3979), .B1(n3691), .B2(n3796), .B3(n3815)
   , .ZN(n3705) );
ND4D0 U4156 ( .A1(n3694), .A2(n3693), .A3(n4419), .A4(n3692),
    .ZN(n3699) );
INVD0 U4157 ( .I(n3717), .ZN(n3830) );
INR4D0 U4158 ( .A1(n3928), .B1(n3695), .B2(n3969), .B3(n3830)
   , .ZN(n3696) );
ND4D0 U4159 ( .A1(n3794), .A2(n3979), .A3(n3697), .A4(n3696),
    .ZN(n3698) );
AOI22D0 U4160 ( .A1(n4000), .A2(n3699), .B1(n3844), .B2(n3698
   ), .ZN(n3704)
      );
ND4D0 U4161 ( .A1(n4048), .A2(n3785), .A3(n3701), .A4(n3700),
    .ZN(n3702) );
OAI21D0 U4162 ( .A1(n4028), .A2(n3702), .B(n3933), .ZN(n3703)
    );
OAI211D0 U4163 ( .A1(n3705), .A2(n4010), .B(n3704), .C(n3703)
   , .ZN(n3706) );
AOI22D0 U4164 ( .A1(n3709), .A2(n3708), .B1(n3707), .B2(n3706
   ), .ZN(n4070)
      );
NR2D0 U4165 ( .A1(n3710), .A2(n4044), .ZN(n3838) );
```

```
ND4D0 U4166 ( .A1(n3713), .A2(n3712), .A3(n3711), .A4(n3838),
    .ZN(n3720) );
CKND2D0 U4167 ( .A1(n3715), .A2(n3714), .ZN(n3820) );
INVD0 U4168 ( .I(n3820), .ZN(n3952) );
NR2D0 U4169 ( .A1(n3716), .A2(n3767), .ZN(n3721) );
ND4D0 U4170 ( .A1(n3718), .A2(n3717), .A3(n3952), .A4(n3721),
    .ZN(n3719) );
AOI22D0 U4171 ( .A1(n3933), .A2(n3720), .B1(n3769), .B2(n3719
   ), .ZN(n3729)
      );
CKND2D0 U4172 ( .A1(n3856), .A2(n3725), .ZN(n3859) );
INVD0 U4173 ( .I(n3721), .ZN(n3724) );
OR4D0 U4174 ( .A1(n3859), .A2(n3771), .A3(n4042), .A4(n3724),
    .Z(n3722) );
OAI31D0 U4175 ( .A1(n3786), .A2(n3723), .A3(n3722), .B(n4000)
   , .ZN(n3728) );
INVD0 U4176 ( .I(n4035), .ZN(n3806) );
NR2D0 U4177 ( .A1(n4029), .A2(n3724), .ZN(n3744) );
ND4D0 U4178 ( .A1(n3725), .A2(n3806), .A3(n3744), .A4(n3987),
    .ZN(n3726) );
OAI21D0 U4179 ( .A1(n3771), .A2(n3726), .B(n3844), .ZN(n3727)
    );
AOI31D0 U4180 ( .A1(n3729), .A2(n3728), .A3(n3727), .B(n4298)
   , .ZN(n4006) );
OAI21D0 U4181 ( .A1(n3950), .A2(n3881), .B(n3730), .ZN(n3930)
    );
CKND2D0 U4182 ( .A1(n3732), .A2(n3731), .ZN(n3772) );
ND4D0 U4183 ( .A1(n3916), .A2(n3915), .A3(n3985), .A4(n3912),
    .ZN(n3733) );
IINR4D0 U4184 ( .A1(n3930), .A2(n3772), .B1(n3734), .B2(n3733
   ), .ZN(n3886)
      );
NR2D0 U4185 ( .A1(n3736), .A2(n3735), .ZN(n3741) );
CKND2D0 U4186 ( .A1(n3738), .A2(n3737), .ZN(n3764) );
CKND2D0 U4187 ( .A1(n3739), .A2(n3764), .ZN(n3895) );
NR4D0 U4188 ( .A1(n3742), .A2(n3741), .A3(n3740), .A4(n3895),
    .ZN(n3747) );
AOI211D0 U4189 ( .A1(n3743), .A2(n3949), .B(n3903), .C(n3889)
   , .ZN(n3978) );
AN4D0 U4190 ( .A1(n3745), .A2(n3789), .A3(n3744), .A4(n3978),
    .Z(n3746) );
ND4D0 U4191 ( .A1(n3886), .A2(n3747), .A3(n3746), .A4(n3911),
    .ZN(n3770) );
NR2D0 U4192 ( .A1(n3749), .A2(n3748), .ZN(n3752) );
AOI22D0 U4193 ( .A1(counter[4]), .A2(n3750), .B1(n4322), .B2(
   n3757), .ZN(
      n3954) );
```

```
AOI211D0 U4194 ( .A1(n3752), .A2(n3954), .B(n3751), .C(n3970)
   , .ZN(n3884) );
OAI211D0 U4195 ( .A1(n3754), .A2(n3753), .B(n3978), .C(n3884)
   , .ZN(n3755) );
NR2D0 U4196 ( .A1(n3756), .A2(n3755), .ZN(n3921) );
NR3D0 U4197 ( .A1(n3956), .A2(n3757), .A3(n4272), .ZN(n3763)
   );
OA211D0 U4198 ( .A1(n3759), .A2(n3956), .B(n3758), .C(n3984),
    .Z(n3885) );
AOI32D0 U4199 ( .A1(n3914), .A2(n3885), .A3(n3760), .B1(n3956
   ), .B2(n3885),
      .ZN(n3761) );
INR4D0 U4200 ( .A1(n3764), .B1(n3763), .B2(n3762), .B3(n3761)
   , .ZN(n3765) );
IND4D0 U4201 ( .A1(n3767), .B1(n3766), .B2(n3921), .B3(n3765)
   , .ZN(n3768) );
AOI22D0 U4202 ( .A1(n3933), .A2(n3770), .B1(n3769), .B2(n3768
   ), .ZN(n3780)
       );
INVD0 U4203 ( .I(n3887), .ZN(n3829) );
OAI31D0 U4204 ( .A1(n3829), .A2(n3771), .A3(n4041), .B(n3844)
   , .ZN(n3779) );
ND4D0 U4205 ( .A1(n3773), .A2(n3840), .A3(n3984), .A4(n3772),
    .ZN(n3774) );
OAI31D0 U4206 ( .A1(n3776), .A2(n3775), .A3(n3774), .B(n4000)
   , .ZN(n3778) );
AOI31D0 U4207 ( .A1(n3780), .A2(n3779), .A3(n3778), .B(n3777)
   , .ZN(n4005) );
NR4D0 U4208 ( .A1(n3783), .A2(n3782), .A3(n3827), .A4(n3781),
    .ZN(n3784) );
ND4D0 U4209 ( .A1(n3785), .A2(n3839), .A3(n3784), .A4(n4050),
    .ZN(n3804) );
NR2D0 U4210 ( .A1(n3787), .A2(n3786), .ZN(n3792) );
CKND2D0 U4211 ( .A1(n3789), .A2(n3788), .ZN(n4020) );
NR4D0 U4212 ( .A1(n3827), .A2(n3958), .A3(n4053), .A4(n4020),
    .ZN(n3791) );
AOI31D0 U4213 ( .A1(n3792), .A2(n3791), .A3(n3790), .B(n4010)
   , .ZN(n3803) );
ND4D0 U4214 ( .A1(n3808), .A2(n4048), .A3(n3794), .A4(n3793),
    .ZN(n3795) );
NR4D0 U4215 ( .A1(n3797), .A2(n3810), .A3(n3796), .A4(n3795),
    .ZN(n3801) );
NR2D0 U4216 ( .A1(n4058), .A2(n4053), .ZN(n3863) );
IINR4D0 U4217 ( .A1(n3799), .A2(n3863), .B1(n3919), .B2(n3798
   ), .ZN(n3800)
       );
OAI22D0 U4218 ( .A1(n3801), .A2(n4059), .B1(n3800), .B2(n4045
   ), .ZN(n3802)
```

```
        );
AOI211D0 U4219 ( .A1(n3933), .A2(n3804), .B(n3803), .C(n3802)
   , .ZN(n3879) );
ND4D0 U4220 ( .A1(n3808), .A2(n3807), .A3(n3806), .A4(n3805),
    .ZN(n3809) );
NR4D0 U4221 ( .A1(n3811), .A2(n3810), .A3(n4054), .A4(n3809),
    .ZN(n3826) );
ND4D0 U4222 ( .A1(n3908), .A2(n3861), .A3(n4049), .A4(n3812),
    .ZN(n3819) );
CKND2D0 U4223 ( .A1(n3814), .A2(n3813), .ZN(n4027) );
NR4D0 U4224 ( .A1(n3919), .A2(n4008), .A3(n4027), .A4(n3815),
    .ZN(n3817) );
AOI31D0 U4225 ( .A1(n3817), .A2(n3816), .A3(n3986), .B(n4010)
   , .ZN(n3818) );
AOI221D0 U4226 ( .A1(n3820), .A2(n3844), .B1(n3819), .B2(
   n3844), .C(n3818),
      .ZN(n3825) );
CKND2D0 U4227 ( .A1(n3821), .A2(n4024), .ZN(n3900) );
OAI211D0 U4228 ( .A1(n3822), .A2(n3900), .B(n3931), .C(n3987)
   , .ZN(n3823) );
OAI31D0 U4229 ( .A1(n3971), .A2(n3858), .A3(n3823), .B(n3933)
   , .ZN(n3824) );
OAI211D0 U4230 ( .A1(n3826), .A2(n4059), .B(n3825), .C(n3824)
   , .ZN(n3851) );
OR2D0 U4231 ( .A1(n3958), .A2(n3827), .Z(n3828) );
NR4D0 U4232 ( .A1(n3970), .A2(n3858), .A3(n3829), .A4(n3828),
    .ZN(n3848) );
NR4D0 U4233 ( .A1(n3832), .A2(n3831), .A3(n3830), .A4(n4008),
    .ZN(n3836) );
CKND2D0 U4234 ( .A1(n3949), .A2(n3833), .ZN(n3834) );
ND4D0 U4235 ( .A1(n3928), .A2(n3836), .A3(n3835), .A4(n3834),
    .ZN(n3842) );
ND4D0 U4236 ( .A1(n3840), .A2(n3839), .A3(n3838), .A4(n3837),
    .ZN(n3841) );
AOI22D0 U4237 ( .A1(n3933), .A2(n3842), .B1(n4000), .B2(n3841
   ), .ZN(n3847)
      );
NR2D0 U4238 ( .A1(n3843), .A2(n3858), .ZN(n3983) );
ND4D0 U4239 ( .A1(n3983), .A2(n3863), .A3(n4022), .A4(n4011),
    .ZN(n3845) );
OAI21D0 U4240 ( .A1(n4008), .A2(n3845), .B(n3844), .ZN(n3846)
    );
OAI211D0 U4241 ( .A1(n3848), .A2(n4010), .B(n3847), .C(n3846)
   , .ZN(n3849) );
AOI22D0 U4242 ( .A1(n3852), .A2(n3851), .B1(n3850), .B2(n3849
   ), .ZN(n3878)
      );
INVD0 U4243 ( .I(n4027), .ZN(n3857) );
```

```
NR4D0 U4244 ( .A1(n3918), .A2(n3854), .A3(n4057), .A4(n3853),
    .ZN(n3855) );
AOI31D0 U4245 ( .A1(n3857), .A2(n3856), .A3(n3855), .B(n4010)
   , .ZN(n3876) );
NR2D0 U4246 ( .A1(n3957), .A2(n3858), .ZN(n4051) );
INR4D0 U4247 ( .A1(n3861), .B1(n3860), .B2(n3918), .B3(n3859)
   , .ZN(n3862) );
AOI31D0 U4248 ( .A1(n3863), .A2(n4051), .A3(n3862), .B(n4045)
   , .ZN(n3875) );
ND4D0 U4249 ( .A1(n3865), .A2(n3928), .A3(n3864), .A4(n3983),
    .ZN(n3866) );
OAI31D0 U4250 ( .A1(n3868), .A2(n3867), .A3(n3866), .B(n4000)
   , .ZN(n3872) );
NR4D0 U4251 ( .A1(n3870), .A2(n3869), .A3(n3981), .A4(n4053),
    .ZN(n3871) );
AOI32D0 U4252 ( .A1(n3873), .A2(n3872), .A3(n3871), .B1(n4038
   ), .B2(n3872),
      .ZN(n3874) );
OAI31D0 U4253 ( .A1(n3876), .A2(n3875), .A3(n3874), .B(n4492)
   , .ZN(n3877) );
OAI211D0 U4254 ( .A1(n3879), .A2(n4410), .B(n3878), .C(n3877)
   , .ZN(n4004) );
OAI21D0 U4255 ( .A1(n3907), .A2(n3881), .B(n3880), .ZN(n3963)
    );
ND4D0 U4256 ( .A1(n3884), .A2(n3883), .A3(n3882), .A4(n3963),
    .ZN(n3942) );
ND4D0 U4257 ( .A1(n3888), .A2(n3887), .A3(n3886), .A4(n3885),
    .ZN(n3941) );
NR4D0 U4258 ( .A1(n3892), .A2(n3891), .A3(n3890), .A4(n3889),
    .ZN(n3893) );
IIND4D0 U4259 ( .A1(n3895), .A2(n3944), .B1(n3894), .B2(n3893
   ), .ZN(n3968)
       );
AOI32D0 U4260 ( .A1(n4109), .A2(n3915), .A3(n3897), .B1(n3896
   ), .B2(n3915),
      .ZN(n3910) );
NR2D0 U4261 ( .A1(counter[4]), .A2(n4322), .ZN(n3989) );
AOI211D0 U4262 ( .A1(n3901), .A2(n3900), .B(n3899), .C(n3898)
   , .ZN(n3975) );
CKND2D0 U4263 ( .A1(n4023), .A2(n3984), .ZN(n3902) );
NR4D0 U4264 ( .A1(n3904), .A2(n3903), .A3(n4008), .A4(n3902),
    .ZN(n3906) );
ND4D0 U4265 ( .A1(n3975), .A2(n3906), .A3(n4011), .A4(n3905),
    .ZN(n3953) );
AOI211D0 U4266 ( .A1(n3907), .A2(n3989), .B(n3919), .C(n3953)
   , .ZN(n3929) );
IND4D0 U4267 ( .A1(n4037), .B1(n3908), .B2(n3929), .B3(n3911)
   , .ZN(n3909) );
```

```
NR4D0 U4268 ( .A1(n4044), .A2(n3968), .A3(n3910), .A4(n3909),
    .ZN(n3939) );
ND4D0 U4269 ( .A1(n3914), .A2(n3913), .A3(n3912), .A4(n3911),
    .ZN(n3947) );
ND3D0 U4270 ( .A1(n3917), .A2(n3916), .A3(n3915), .ZN(n3966)
   );
NR4D0 U4271 ( .A1(n3919), .A2(n3918), .A3(n3944), .A4(n4008),
    .ZN(n3920) );
ND4D0 U4272 ( .A1(n3923), .A2(n3922), .A3(n3921), .A4(n3920),
    .ZN(n3924) );
OAI31D0 U4273 ( .A1(n3947), .A2(n3966), .A3(n3924), .B(n3769)
   , .ZN(n3938) );
IND4D0 U4274 ( .A1(n3957), .B1(n3928), .B2(n3927), .B3(n3925)
   , .ZN(n3936) );
INVD0 U4275 ( .I(n3929), .ZN(n3935) );
ND4D0 U4276 ( .A1(n3932), .A2(n3931), .A3(n3962), .A4(n3930),
    .ZN(n3934) );
OAI31D0 U4277 ( .A1(n3936), .A2(n3935), .A3(n3934), .B(n3933)
   , .ZN(n3937) );
OAI211D0 U4278 ( .A1(n3939), .A2(n4045), .B(n3938), .C(n3937)
   , .ZN(n3940) );
AOI221D0 U4279 ( .A1(n3942), .A2(n4000), .B1(n3941), .B2(
   n4000), .C(n3940),
      .ZN(n4002) );
NR4D0 U4280 ( .A1(n4110), .A2(n3945), .A3(n3944), .A4(n3943),
    .ZN(n3996) );
NR2D0 U4281 ( .A1(n3946), .A2(n3955), .ZN(n3948) );
AOI211D0 U4282 ( .A1(n3950), .A2(n3949), .B(n3948), .C(n3947)
   , .ZN(n3951) );
IND4D0 U4283 ( .A1(n3953), .B1(n3952), .B2(n3996), .B3(n3951)
   , .ZN(n3999) );
INVD0 U4284 ( .I(n3954), .ZN(n3960) );
OAI21D0 U4285 ( .A1(n3956), .A2(n4272), .B(n3955), .ZN(n3959)
    );
AOI211D0 U4286 ( .A1(n3960), .A2(n3959), .B(n3958), .C(n3957)
   , .ZN(n3973) );
ND4D0 U4287 ( .A1(n3964), .A2(n3963), .A3(n3962), .A4(n3961),
    .ZN(n3965) );
NR3D0 U4288 ( .A1(n3967), .A2(n3966), .A3(n3965), .ZN(n3993)
   );
NR4D0 U4289 ( .A1(n3971), .A2(n3970), .A3(n3969), .A4(n3968),
    .ZN(n3972) );
AOI31D0 U4290 ( .A1(n3973), .A2(n3993), .A3(n3972), .B(n4045)
   , .ZN(n3998) );
OAI211D0 U4291 ( .A1(n3977), .A2(n3976), .B(n3975), .C(n3974)
   , .ZN(n3982) );
ND4D0 U4292 ( .A1(n4047), .A2(n3979), .A3(n3978), .A4(n3993),
    .ZN(n3980) );
```

```
OAI31D0 U4293 ( .A1(n3982), .A2(n3981), .A3(n3980), .B(n3769)
   , .ZN(n3995) );
INVD0 U4294 ( .I(n3983), .ZN(n4055) );
ND4D0 U4295 ( .A1(n3987), .A2(n3986), .A3(n3985), .A4(n3984),
    .ZN(n3992) );
AO211D0 U4296 ( .A1(n3990), .A2(n3989), .B(n4021), .C(n3988),
    .Z(n3991) );
INR4D0 U4297 ( .A1(n3993), .B1(n4055), .B2(n3992), .B3(n3991)
   , .ZN(n3994) );
AOI32D0 U4298 ( .A1(n3996), .A2(n3995), .A3(n3994), .B1(n4038
   ), .B2(n3995),
      .ZN(n3997) );
AOI211D0 U4299 ( .A1(n4000), .A2(n3999), .B(n3998), .C(n3997)
   , .ZN(n4001) );
OAI22D0 U4300 ( .A1(n4002), .A2(n4129), .B1(n4001), .B2(n4277
   ), .ZN(n4003)
       );
NR4D0 U4301 ( .A1(n4006), .A2(n4005), .A3(n4004), .A4(n4003),
    .ZN(n4069) );
INVD0 U4302 ( .I(n4007), .ZN(n4013) );
NR4D0 U4303 ( .A1(n4009), .A2(n4008), .A3(n4028), .A4(n4027),
    .ZN(n4012) );
AOI31D0 U4304 ( .A1(n4013), .A2(n4012), .A3(n4011), .B(n4010)
   , .ZN(n4034) );
NR4D0 U4305 ( .A1(n4021), .A2(n4058), .A3(n4029), .A4(n4035),
    .ZN(n4015) );
AOI31D0 U4306 ( .A1(n4015), .A2(n4050), .A3(n4014), .B(n4059)
   , .ZN(n4033) );
OAI211D0 U4307 ( .A1(n4018), .A2(n4017), .B(n4016), .C(n4040)
   , .ZN(n4019) );
NR4D0 U4308 ( .A1(n4021), .A2(n4027), .A3(n4020), .A4(n4019),
    .ZN(n4031) );
OAI211D0 U4309 ( .A1(n4025), .A2(n4024), .B(n4023), .C(n4022)
   , .ZN(n4026) );
NR4D0 U4310 ( .A1(n4029), .A2(n4028), .A3(n4027), .A4(n4026),
    .ZN(n4030) );
OAI22D0 U4311 ( .A1(n4031), .A2(n4045), .B1(n4030), .B2(n4038
   ), .ZN(n4032)
       );
OAI31D0 U4312 ( .A1(n4034), .A2(n4033), .A3(n4032), .B(n4521)
   , .ZN(n4068) );
NR3D0 U4313 ( .A1(n4037), .A2(n4036), .A3(n4035), .ZN(n4039)
   );
AOI31D0 U4314 ( .A1(n4040), .A2(n4048), .A3(n4039), .B(n4038)
   , .ZN(n4066) );
NR4D0 U4315 ( .A1(n4044), .A2(n4043), .A3(n4042), .A4(n4041),
    .ZN(n4046) );
```

```
AOI31D0 U4316 ( .A1(n4048), .A2(n4047), .A3(n4046), .B(n4045)
   , .ZN(n4065) );
ND4D0 U4317 ( .A1(n4051), .A2(n4330), .A3(n4050), .A4(n4049),
    .ZN(n4052) );
OAI31D0 U4318 ( .A1(n4054), .A2(n4053), .A3(n4052), .B(n3769)
   , .ZN(n4061) );
NR4D0 U4319 ( .A1(n4058), .A2(n4057), .A3(n4056), .A4(n4055),
    .ZN(n4060) );
AOI32D0 U4320 ( .A1(n4062), .A2(n4061), .A3(n4060), .B1(n4059
   ), .B2(n4061),
      .ZN(n4064) );
OAI31D0 U4321 ( .A1(n4066), .A2(n4065), .A3(n4064), .B(n4063)
   , .ZN(n4067) );
ND4D0 U4322 ( .A1(n4070), .A2(n4069), .A3(n4068), .A4(n4067),
    .ZN(n4071) );
NR3D0 U4323 ( .A1(n4073), .A2(n4072), .A3(n4071), .ZN(n4543)
   );
AOI21D0 U4324 ( .A1(n4117), .A2(n4112), .B(n4111), .ZN(n4253)
    );
INR4D0 U4325 ( .A1(n4077), .B1(n4076), .B2(n4075), .B3(n4074)
   , .ZN(n4086) );
CKND2D0 U4326 ( .A1(n4167), .A2(n4418), .ZN(n4455) );
NR2D0 U4327 ( .A1(n4181), .A2(n4171), .ZN(n4485) );
ND4D0 U4328 ( .A1(n4108), .A2(n4078), .A3(n4121), .A4(n4465),
    .ZN(n4247) );
AOI211D0 U4329 ( .A1(n4079), .A2(n4081), .B(n4437), .C(n4247)
   , .ZN(n4080) );
ND4D0 U4330 ( .A1(n4485), .A2(n4080), .A3(n4363), .A4(n4120),
    .ZN(n4087) );
AOI211D0 U4331 ( .A1(n4081), .A2(n4083), .B(n4455), .C(n4087)
   , .ZN(n4085) );
OAI21D0 U4332 ( .A1(n4084), .A2(n4083), .B(n4082), .ZN(n4414)
    );
ND4D0 U4333 ( .A1(n4253), .A2(n4086), .A3(n4085), .A4(n4414),
    .ZN(n4097) );
CKND2D0 U4334 ( .A1(n4334), .A2(n4137), .ZN(n4281) );
INVD0 U4335 ( .I(n4281), .ZN(n4433) );
IND2D0 U4336 ( .A1(n4087), .B1(n4498), .ZN(n4242) );
INR4D0 U4337 ( .A1(n4221), .B1(n4090), .B2(n4089), .B3(n4088)
   , .ZN(n4273) );
ND3D0 U4338 ( .A1(n4464), .A2(n4273), .A3(n4091), .ZN(n4102)
   );
NR4D0 U4339 ( .A1(n4093), .A2(n4242), .A3(n4102), .A4(n4092),
    .ZN(n4095) );
ND4D0 U4340 ( .A1(n4433), .A2(n4095), .A3(n4369), .A4(n4094),
    .ZN(n4096) );
AOI22D0 U4341 ( .A1(n4452), .A2(n4097), .B1(n4289), .B2(n4096
   ), .ZN(n4132)
```

```
        );
CKND2D0 U4342 ( .A1(n4099), .A2(n4098), .ZN(n4196) );
INVD0 U4343 ( .I(n4196), .ZN(n4231) );
OAI211D0 U4344 ( .A1(n4101), .A2(n4100), .B(n4391), .C(n4362)
   , .ZN(n4106) );
NR2D0 U4345 ( .A1(n4310), .A2(n4484), .ZN(n4509) );
INVD0 U4346 ( .I(n4509), .ZN(n4375) );
CKND2D0 U4347 ( .A1(n4363), .A2(n4233), .ZN(n4514) );
NR2D0 U4348 ( .A1(n4375), .A2(n4514), .ZN(n4461) );
INR4D0 U4349 ( .A1(n4461), .B1(n4103), .B2(n4376), .B3(n4102)
   , .ZN(n4104) );
ND4D0 U4350 ( .A1(n4253), .A2(n4104), .A3(n4167), .A4(n4108),
    .ZN(n4105) );
OAI31D0 U4351 ( .A1(n4231), .A2(n4106), .A3(n4105), .B(n4291)
   , .ZN(n4131) );
IND4D0 U4352 ( .A1(n4110), .B1(n4109), .B2(n4108), .B3(n4107)
   , .ZN(n4128) );
AOI211D0 U4353 ( .A1(n4113), .A2(n4112), .B(n4111), .C(n4232)
   , .ZN(n4226) );
INVD0 U4354 ( .I(n4114), .ZN(n4116) );
AOI221D0 U4355 ( .A1(n4118), .A2(n4117), .B1(n4116), .B2(
   n4117), .C(n4115),
      .ZN(n4123) );
IINR4D0 U4356 ( .A1(n4121), .A2(n4120), .B1(n4438), .B2(n4119
   ), .ZN(n4122)
       );
ND4D0 U4357 ( .A1(n4226), .A2(n4123), .A3(n4122), .A4(n4506),
    .ZN(n4268) );
NR2D0 U4358 ( .A1(n4124), .A2(n4181), .ZN(n4202) );
ND4D0 U4359 ( .A1(n4126), .A2(n4202), .A3(n4503), .A4(n4125),
    .ZN(n4127) );
OAI31D0 U4360 ( .A1(n4128), .A2(n4268), .A3(n4127), .B(n4480)
   , .ZN(n4130) );
AOI31D0 U4361 ( .A1(n4132), .A2(n4131), .A3(n4130), .B(n4129)
   , .ZN(n4540) );
CKND2D0 U4362 ( .A1(n4447), .A2(n4137), .ZN(n4180) );
CKND2D0 U4363 ( .A1(n4336), .A2(n4133), .ZN(n4154) );
NR2D0 U4364 ( .A1(n4250), .A2(n4154), .ZN(n4423) );
AOI211D0 U4365 ( .A1(n4145), .A2(n4144), .B(n4134), .C(n4440)
   , .ZN(n4135) );
ND4D0 U4366 ( .A1(n4423), .A2(n4135), .A3(n4362), .A4(n4503),
    .ZN(n4140) );
CKND2D0 U4367 ( .A1(n4296), .A2(n4467), .ZN(n4385) );
CKND2D0 U4368 ( .A1(n4136), .A2(n4345), .ZN(n4386) );
NR2D0 U4369 ( .A1(n4386), .A2(n4359), .ZN(n4152) );
INVD0 U4370 ( .I(n4152), .ZN(n4482) );
INVD0 U4371 ( .I(n4498), .ZN(n4293) );
```

```
NR4D0 U4372 ( .A1(n4375), .A2(n4385), .A3(n4482), .A4(n4293),
    .ZN(n4138) );
AOI31D0 U4373 ( .A1(n4138), .A2(n4137), .A3(n4196), .B(n4476)
   , .ZN(n4139) );
AOI221D0 U4374 ( .A1(n4180), .A2(n4452), .B1(n4140), .B2(
   n4452), .C(n4139),
      .ZN(n4151) );
CKND2D0 U4375 ( .A1(n4335), .A2(n4459), .ZN(n4401) );
ND3D0 U4376 ( .A1(n4334), .A2(n4460), .A3(n4141), .ZN(n4483)
   );
NR2D0 U4377 ( .A1(n4401), .A2(n4483), .ZN(n4287) );
INVD0 U4378 ( .I(n4287), .ZN(n4142) );
OAI31D0 U4379 ( .A1(n4143), .A2(n4378), .A3(n4142), .B(n4480)
   , .ZN(n4150) );
AOI211D0 U4380 ( .A1(n4145), .A2(n4144), .B(n4156), .C(n4386)
   , .ZN(n4392) );
INVD0 U4381 ( .I(n4440), .ZN(n4451) );
ND4D0 U4382 ( .A1(n4392), .A2(n4451), .A3(n4389), .A4(n4146),
    .ZN(n4147) );
OAI31D0 U4383 ( .A1(n4455), .A2(n4376), .A3(n4147), .B(n4342)
   , .ZN(n4149) );
AOI31D0 U4384 ( .A1(n4151), .A2(n4150), .A3(n4149), .B(n4148)
   , .ZN(n4539) );
INR2D0 U4385 ( .A1(n4202), .B1(n4306), .ZN(n4284) );
CKND2D0 U4386 ( .A1(n4284), .A2(n4503), .ZN(n4512) );
INVD0 U4387 ( .I(n4401), .ZN(n4285) );
ND4D0 U4388 ( .A1(n4285), .A2(n4152), .A3(n4391), .A4(n4329),
    .ZN(n4166) );
CKAN2D0 U4389 ( .A1(n4153), .A2(n4317), .Z(n4508) );
CKAN2D0 U4390 ( .A1(n4508), .A2(n4202), .Z(n4228) );
INVD0 U4391 ( .I(n4512), .ZN(n4158) );
ND3D0 U4392 ( .A1(n4158), .A2(n4201), .A3(n4369), .ZN(n4425)
   );
NR2D0 U4393 ( .A1(n4425), .A2(n4180), .ZN(n4198) );
NR2D0 U4394 ( .A1(n4154), .A2(n4394), .ZN(n4204) );
ND4D0 U4395 ( .A1(n4198), .A2(n4204), .A3(n4506), .A4(n4316),
    .ZN(n4160) );
INVD0 U4396 ( .I(n4420), .ZN(n4454) );
NR2D0 U4397 ( .A1(n4422), .A2(n4155), .ZN(n4332) );
INVD0 U4398 ( .I(n4332), .ZN(n4282) );
NR4D0 U4399 ( .A1(n4516), .A2(n4156), .A3(n4454), .A4(n4282),
    .ZN(n4168) );
IND4D0 U4400 ( .A1(n4180), .B1(n4158), .B2(n4168), .B3(n4157)
   , .ZN(n4159) );
AOI22D0 U4401 ( .A1(n4291), .A2(n4160), .B1(n4342), .B2(n4159
   ), .ZN(n4164)
     );
NR2D0 U4402 ( .A1(n4161), .A2(n4170), .ZN(n4350) );
```

```
INR4D0 U4403 ( .A1(n4350), .B1(n4172), .B2(n4162), .B3(n4483)
   , .ZN(n4163) );
AOI32D0 U4404 ( .A1(n4228), .A2(n4164), .A3(n4163), .B1(n4474
   ), .B2(n4164),
      .ZN(n4165) );
AOI221D0 U4405 ( .A1(n4512), .A2(n4480), .B1(n4166), .B2(
   n4480), .C(n4165),
      .ZN(n4193) );
IND4D0 U4406 ( .A1(n4514), .B1(n4169), .B2(n4168), .B3(n4167)
   , .ZN(n4189) );
NR2D0 U4407 ( .A1(n4473), .A2(n4170), .ZN(n4500) );
NR2D0 U4408 ( .A1(n4172), .A2(n4171), .ZN(n4313) );
INVD0 U4409 ( .I(n4313), .ZN(n4493) );
NR4D0 U4410 ( .A1(n4173), .A2(n4493), .A3(n4385), .A4(n4218),
    .ZN(n4174) );
AOI31D0 U4411 ( .A1(n4175), .A2(n4500), .A3(n4174), .B(n4428)
   , .ZN(n4188) );
ND3D0 U4412 ( .A1(n4388), .A2(n4367), .A3(n4332), .ZN(n4400)
   );
INVD0 U4413 ( .I(n4400), .ZN(n4186) );
ND3D0 U4414 ( .A1(n4465), .A2(n4176), .A3(n4420), .ZN(n4354)
   );
IND4D0 U4415 ( .A1(n4354), .B1(n4177), .B2(n4418), .B3(n4367)
   , .ZN(n4178) );
OAI31D0 U4416 ( .A1(n4180), .A2(n4179), .A3(n4178), .B(n4342)
   , .ZN(n4185) );
NR2D0 U4417 ( .A1(n4181), .A2(n4511), .ZN(n4501) );
CKND2D0 U4418 ( .A1(n4460), .A2(n4501), .ZN(n4319) );
INR4D0 U4419 ( .A1(n4360), .B1(n4183), .B2(n4182), .B3(n4319)
   , .ZN(n4184) );
AOI32D0 U4420 ( .A1(n4186), .A2(n4185), .A3(n4184), .B1(n4474
   ), .B2(n4185),
      .ZN(n4187) );
AOI211D0 U4421 ( .A1(n4480), .A2(n4189), .B(n4188), .C(n4187)
   , .ZN(n4191) );
OAI22D0 U4422 ( .A1(n4193), .A2(n4192), .B1(n4191), .B2(n4190
   ), .ZN(n4538)
       );
IND4D0 U4423 ( .A1(n4195), .B1(n4217), .B2(n4501), .B3(n4194)
   , .ZN(n4208) );
ND3D0 U4424 ( .A1(n4334), .A2(n4362), .A3(n4196), .ZN(n4348)
   );
INVD0 U4425 ( .I(n4348), .ZN(n4216) );
ND4D0 U4426 ( .A1(n4467), .A2(n4216), .A3(n4459), .A4(n4500),
    .ZN(n4200) );
INVD0 U4427 ( .I(n4455), .ZN(n4361) );
AOI31D0 U4428 ( .A1(n4198), .A2(n4361), .A3(n4197), .B(n4517)
   , .ZN(n4199) );
```

```
AOI221D0 U4429 ( .A1(n4354), .A2(n4480), .B1(n4200), .B2(
   n4480), .C(n4199),
      .ZN(n4206) );
CKND2D0 U4430 ( .A1(n4202), .A2(n4201), .ZN(n4441) );
INR4D0 U4431 ( .A1(n4204), .B1(n4203), .B2(n4441), .B3(n4393)
   , .ZN(n4205) );
AOI32D0 U4432 ( .A1(n4332), .A2(n4206), .A3(n4205), .B1(n4428
   ), .B2(n4206),
      .ZN(n4207) );
AOI221D0 U4433 ( .A1(n4375), .A2(n4452), .B1(n4208), .B2(
   n4452), .C(n4207),
      .ZN(n4536) );
ND4D0 U4434 ( .A1(n4361), .A2(n4221), .A3(n4503), .A4(n4209),
    .ZN(n4214) );
NR2D0 U4435 ( .A1(n4231), .A2(n4210), .ZN(n4497) );
ND4D0 U4436 ( .A1(n4497), .A2(n4433), .A3(n4212), .A4(n4211),
    .ZN(n4213) );
NR4D0 U4437 ( .A1(n4473), .A2(n4215), .A3(n4214), .A4(n4213),
    .ZN(n4239) );
IND4D0 U4438 ( .A1(n4218), .B1(n4336), .B2(n4217), .B3(n4216)
   , .ZN(n4230) );
NR2D0 U4439 ( .A1(n4281), .A2(n4219), .ZN(n4396) );
NR2D0 U4440 ( .A1(n4484), .A2(n4250), .ZN(n4397) );
OAI211D0 U4441 ( .A1(n4223), .A2(n4222), .B(n4221), .C(n4220)
   , .ZN(n4224) );
IINR4D0 U4442 ( .A1(n4226), .A2(n4397), .B1(n4225), .B2(n4224
   ), .ZN(n4227)
      );
AOI31D0 U4443 ( .A1(n4228), .A2(n4396), .A3(n4227), .B(n3243)
   , .ZN(n4229) );
AOI221D0 U4444 ( .A1(n4319), .A2(n4291), .B1(n4230), .B2(
   n4291), .C(n4229),
      .ZN(n4238) );
NR2D0 U4445 ( .A1(n4232), .A2(n4231), .ZN(n4462) );
CKND2D0 U4446 ( .A1(n4462), .A2(n4333), .ZN(n4470) );
NR2D0 U4447 ( .A1(n4437), .A2(n4470), .ZN(n4499) );
ND4D0 U4448 ( .A1(n4499), .A2(n4234), .A3(n4330), .A4(n4233),
    .ZN(n4235) );
OAI31D0 U4449 ( .A1(n4323), .A2(n4236), .A3(n4235), .B(n4452)
   , .ZN(n4237) );
OAI211D0 U4450 ( .A1(n4239), .A2(n4517), .B(n4238), .C(n4237)
   , .ZN(n4304) );
CKND2D0 U4451 ( .A1(n4241), .A2(n4240), .ZN(n4249) );
NR4D0 U4452 ( .A1(n4243), .A2(n4442), .A3(n4242), .A4(n4249),
    .ZN(n4245) );
IND4D0 U4453 ( .A1(n4246), .B1(n4259), .B2(n4245), .B3(n4244)
   , .ZN(n4255) );
```

```
NR3D0 U4454 ( .A1(n4248), .A2(n4247), .A3(n4483), .ZN(n4271)
   );
INR4D0 U4455 ( .A1(n4501), .B1(n4250), .B2(n4378), .B3(n4249)
   , .ZN(n4251) );
ND4D0 U4456 ( .A1(n4253), .A2(n4271), .A3(n4252), .A4(n4251),
    .ZN(n4254) );
AOI22D0 U4457 ( .A1(n4291), .A2(n4255), .B1(n4452), .B2(n4254
   ), .ZN(n4280)
       );
CKND2D0 U4458 ( .A1(n4257), .A2(n4256), .ZN(n4258) );
ND4D0 U4459 ( .A1(n4259), .A2(n4324), .A3(n4360), .A4(n4258),
    .ZN(n4270) );
NR2D0 U4460 ( .A1(n4261), .A2(n4260), .ZN(n4264) );
INVD0 U4461 ( .I(n4385), .ZN(n4434) );
OAI211D0 U4462 ( .A1(n4264), .A2(n4263), .B(n4434), .C(n4262)
   , .ZN(n4275) );
IND4D0 U4463 ( .A1(n4268), .B1(n4267), .B2(n4266), .B3(n4265)
   , .ZN(n4269) );
OAI31D0 U4464 ( .A1(n4270), .A2(n4275), .A3(n4269), .B(n4342)
   , .ZN(n4279) );
INVD0 U4465 ( .I(n4271), .ZN(n4276) );
ND4D0 U4466 ( .A1(n4273), .A2(n4272), .A3(n4317), .A4(n4507),
    .ZN(n4274) );
OAI31D0 U4467 ( .A1(n4276), .A2(n4275), .A3(n4274), .B(n4480)
   , .ZN(n4278) );
AOI31D0 U4468 ( .A1(n4280), .A2(n4279), .A3(n4278), .B(n4277)
   , .ZN(n4303) );
NR4D0 U4469 ( .A1(n4438), .A2(n4282), .A3(n4365), .A4(n4281),
    .ZN(n4283) );
IND4D0 U4470 ( .A1(n4470), .B1(n4285), .B2(n4284), .B3(n4283)
   , .ZN(n4290) );
ND4D0 U4471 ( .A1(n4350), .A2(n4466), .A3(n4287), .A4(n4286),
    .ZN(n4288) );
AOI22D0 U4472 ( .A1(n4291), .A2(n4290), .B1(n4289), .B2(n4288
   ), .ZN(n4301)
       );
ND4D0 U4473 ( .A1(n4500), .A2(n4345), .A3(n4465), .A4(n4343),
    .ZN(n4292) );
OAI31D0 U4474 ( .A1(n4293), .A2(n4483), .A3(n4292), .B(n4452)
   , .ZN(n4300) );
ND4D0 U4475 ( .A1(n4296), .A2(n4392), .A3(n4295), .A4(n4294),
    .ZN(n4297) );
OAI21D0 U4476 ( .A1(n4436), .A2(n4297), .B(n4480), .ZN(n4299)
    );
AOI31D0 U4477 ( .A1(n4301), .A2(n4300), .A3(n4299), .B(n4298)
   , .ZN(n4302) );
AOI211D0 U4478 ( .A1(n4305), .A2(n4304), .B(n4303), .C(n4302)
   , .ZN(n4534) );
```

```
NR3D0 U4479 ( .A1(n4438), .A2(n4307), .A3(n4306), .ZN(n4416)
   );
NR2D0 U4480 ( .A1(n4493), .A2(n4514), .ZN(n4308) );
ND4D0 U4481 ( .A1(n4467), .A2(n4416), .A3(n4308), .A4(n4344),
    .ZN(n4315) );
NR4D0 U4482 ( .A1(n4311), .A2(n4310), .A3(n4309), .A4(n4400),
    .ZN(n4312) );
AOI31D0 U4483 ( .A1(n4423), .A2(n4313), .A3(n4312), .B(n3243)
   , .ZN(n4314) );
AOI221D0 U4484 ( .A1(n4394), .A2(n4452), .B1(n4315), .B2(
   n4452), .C(n4314),
      .ZN(n4328) );
ND4D0 U4485 ( .A1(n4509), .A2(n4463), .A3(n4317), .A4(n4316),
    .ZN(n4318) );
OAI31D0 U4486 ( .A1(n4354), .A2(n4319), .A3(n4318), .B(n4342)
   , .ZN(n4327) );
OAI21D0 U4487 ( .A1(n4322), .A2(n4321), .B(n4320), .ZN(n4371)
    );
INR3D0 U4488 ( .A1(n4416), .B1(n4371), .B2(n4323), .ZN(n4331)
    );
AO31D0 U4489 ( .A1(n4331), .A2(n4502), .A3(n4324), .B(n4428),
    .Z(n4326) );
AOI31D0 U4490 ( .A1(n4328), .A2(n4327), .A3(n4326), .B(n4325)
   , .ZN(n4532) );
ND4D0 U4491 ( .A1(n4332), .A2(n4331), .A3(n4330), .A4(n4329),
    .ZN(n4341) );
INVD0 U4492 ( .I(n4359), .ZN(n4339) );
CKND2D0 U4493 ( .A1(n4334), .A2(n4333), .ZN(n4337) );
ND4D0 U4494 ( .A1(n4336), .A2(n4485), .A3(n4434), .A4(n4335),
    .ZN(n4471) );
NR4D0 U4495 ( .A1(n4437), .A2(n4337), .A3(n4514), .A4(n4471),
    .ZN(n4338) );
AOI31D0 U4496 ( .A1(n4339), .A2(n4364), .A3(n4338), .B(n3243)
   , .ZN(n4340) );
AOI221D0 U4497 ( .A1(n4426), .A2(n4342), .B1(n4341), .B2(
   n4342), .C(n4340),
      .ZN(n4358) );
ND4D0 U4498 ( .A1(n4346), .A2(n4345), .A3(n4344), .A4(n4343),
    .ZN(n4347) );
OAI31D0 U4499 ( .A1(n4348), .A2(n4375), .A3(n4347), .B(n4452)
   , .ZN(n4357) );
INVD0 U4500 ( .I(n4349), .ZN(n4352) );
OAI211D0 U4501 ( .A1(n4352), .A2(n4351), .B(n4460), .C(n4350)
   , .ZN(n4353) );
OAI31D0 U4502 ( .A1(n4470), .A2(n4354), .A3(n4353), .B(n4291)
   , .ZN(n4356) );
AOI31D0 U4503 ( .A1(n4358), .A2(n4357), .A3(n4356), .B(n4355)
   , .ZN(n4531) );
```

```
NR3D0 U4504 ( .A1(n4426), .A2(n4371), .A3(n4359), .ZN(n4412)
   );
ND4D0 U4505 ( .A1(n4412), .A2(n4466), .A3(n4361), .A4(n4360),
    .ZN(n4383) );
CKND2D0 U4506 ( .A1(n4362), .A2(n4391), .ZN(n4366) );
ND3D0 U4507 ( .A1(n4364), .A2(n4392), .A3(n4363), .ZN(n4402)
   );
NR4D0 U4508 ( .A1(n4454), .A2(n4366), .A3(n4402), .A4(n4365),
    .ZN(n4381) );
INVD0 U4509 ( .I(n4497), .ZN(n4372) );
ND4D0 U4510 ( .A1(n4467), .A2(n4369), .A3(n4368), .A4(n4367),
    .ZN(n4370) );
OAI31D0 U4511 ( .A1(n4372), .A2(n4371), .A3(n4370), .B(n4291)
   , .ZN(n4380) );
OR4D0 U4512 ( .A1(n4376), .A2(n4375), .A3(n4374), .A4(n4373),
    .Z(n4377) );
OAI31D0 U4513 ( .A1(n4441), .A2(n4378), .A3(n4377), .B(n4452)
   , .ZN(n4379) );
OAI211D0 U4514 ( .A1(n4381), .A2(n4517), .B(n4380), .C(n4379)
   , .ZN(n4382) );
AOI221D0 U4515 ( .A1(n4384), .A2(n4480), .B1(n4383), .B2(
   n4480), .C(n4382),
      .ZN(n4411) );
NR2D0 U4516 ( .A1(n4386), .A2(n4385), .ZN(n4417) );
ND4D0 U4517 ( .A1(n4461), .A2(n4417), .A3(n4388), .A4(n4387),
    .ZN(n4407) );
CKND2D0 U4518 ( .A1(n4389), .A2(n4420), .ZN(n4469) );
IND4D0 U4519 ( .A1(n4469), .B1(n4392), .B2(n4391), .B3(n4390)
   , .ZN(n4399) );
NR4D0 U4520 ( .A1(n4413), .A2(n4422), .A3(n4394), .A4(n4393),
    .ZN(n4395) );
AOI31D0 U4521 ( .A1(n4397), .A2(n4396), .A3(n4395), .B(n4517)
   , .ZN(n4398) );
AOI221D0 U4522 ( .A1(n4400), .A2(n4480), .B1(n4399), .B2(
   n4480), .C(n4398),
      .ZN(n4405) );
NR4D0 U4523 ( .A1(n4438), .A2(n4403), .A3(n4402), .A4(n4401),
    .ZN(n4404) );
AOI32D0 U4524 ( .A1(n4501), .A2(n4405), .A3(n4404), .B1(n4476
   ), .B2(n4405),
      .ZN(n4406) );
AOI221D0 U4525 ( .A1(n4495), .A2(n4452), .B1(n4407), .B2(
   n4452), .C(n4406),
      .ZN(n4409) );
OAI22D0 U4526 ( .A1(n4411), .A2(n4410), .B1(n4409), .B2(n4408
   ), .ZN(n4530)
       );
```

```
ND4D0 U4527 ( .A1(n4412), .A2(n4459), .A3(n4418), .A4(n4414),
    .ZN(n4432) );
INR4D0 U4528 ( .A1(n4414), .B1(n4413), .B2(n4422), .B3(n4454)
   , .ZN(n4415) );
AOI31D0 U4529 ( .A1(n4417), .A2(n4416), .A3(n4415), .B(n4474)
   , .ZN(n4431) );
ND4D0 U4530 ( .A1(n4420), .A2(n4423), .A3(n4419), .A4(n4418),
    .ZN(n4421) );
NR4D0 U4531 ( .A1(n4438), .A2(n4422), .A3(n4470), .A4(n4421),
    .ZN(n4429) );
INVD0 U4532 ( .I(n4423), .ZN(n4515) );
NR4D0 U4533 ( .A1(n4426), .A2(n4515), .A3(n4425), .A4(n4424),
    .ZN(n4427) );
OAI22D0 U4534 ( .A1(n4429), .A2(n4428), .B1(n4427), .B2(n4517
   ), .ZN(n4430)
       );
AOI211D0 U4535 ( .A1(n4480), .A2(n4432), .B(n4431), .C(n4430)
   , .ZN(n4528) );
ND4D0 U4536 ( .A1(n4488), .A2(n4434), .A3(n4433), .A4(n4500),
    .ZN(n4435) );
NR4D0 U4537 ( .A1(n4437), .A2(n4484), .A3(n4436), .A4(n4435),
    .ZN(n4458) );
OR4D0 U4538 ( .A1(n4441), .A2(n4440), .A3(n4439), .A4(n4438),
    .Z(n4449) );
AOI221D0 U4539 ( .A1(n4445), .A2(n4444), .B1(n4443), .B2(
   n4444), .C(n4442),
      .ZN(n4446) );
AOI31D0 U4540 ( .A1(n4508), .A2(n4447), .A3(n4446), .B(n4476)
   , .ZN(n4448) );
AOI221D0 U4541 ( .A1(n4450), .A2(n4480), .B1(n4449), .B2(
   n4480), .C(n4448),
      .ZN(n4457) );
ND4D0 U4542 ( .A1(n4460), .A2(n4459), .A3(n4451), .A4(n4499),
    .ZN(n4453) );
OAI31D0 U4543 ( .A1(n4455), .A2(n4454), .A3(n4453), .B(n4452)
   , .ZN(n4456) );
OAI211D0 U4544 ( .A1(n4458), .A2(n4517), .B(n4457), .C(n4456)
   , .ZN(n4491) );
CKND2D0 U4545 ( .A1(n4460), .A2(n4459), .ZN(n4481) );
ND4D0 U4546 ( .A1(n4464), .A2(n4463), .A3(n4462), .A4(n4461),
    .ZN(n4479) );
ND4D0 U4547 ( .A1(n4467), .A2(n4466), .A3(n4503), .A4(n4465),
    .ZN(n4468) );
NR4D0 U4548 ( .A1(n4493), .A2(n4470), .A3(n4469), .A4(n4468),
    .ZN(n4477) );
NR4D0 U4549 ( .A1(n4473), .A2(n4472), .A3(n4471), .A4(n4470),
    .ZN(n4475) );
```

```
  OAI22D0 U4550 ( .A1(n4477), .A2(n4476), .B1(n4475), .B2(n4474
     ), .ZN(n4478)
          );
  AOI221D0 U4551 ( .A1(n4481), .A2(n4480), .B1(n4479), .B2(
     n4480), .C(n4478),
        .ZN(n4487) );
  INR4D0 U4552 ( .A1(n4485), .B1(n4484), .B2(n4483), .B3(n4482)
     , .ZN(n4486) );
  AOI32D0 U4553 ( .A1(n4488), .A2(n4487), .A3(n4486), .B1(n4517
     ), .B2(n4487),
        .ZN(n4489) );
  AOI22D0 U4554 ( .A1(n4492), .A2(n4491), .B1(n4490), .B2(n4489
     ), .ZN(n4526)
          );
  NR4D0 U4555 ( .A1(n4495), .A2(n4494), .A3(n4493), .A4(n4514),
      .ZN(n4496) );
  AOI31D0 U4556 ( .A1(n4498), .A2(n4497), .A3(n4496), .B(n54),
     .ZN(n4524) );
  AN4D0 U4557 ( .A1(n4502), .A2(n4501), .A3(n4500), .A4(n4499),
      .Z(n4505) );
  AOI31D0 U4558 ( .A1(n4505), .A2(n4504), .A3(n4503), .B(n3243)
     , .ZN(n4523) );
  ND4D0 U4559 ( .A1(n4509), .A2(n4508), .A3(n4507), .A4(n4506),
      .ZN(n4510) );
  OAI31D0 U4560 ( .A1(n4512), .A2(n4511), .A3(n4510), .B(n4291)
     , .ZN(n4519) );
  NR4D0 U4561 ( .A1(n4516), .A2(n4515), .A3(n4514), .A4(n4513),
      .ZN(n4518) );
  AOI32D0 U4562 ( .A1(n4520), .A2(n4519), .A3(n4518), .B1(n4517
     ), .B2(n4519),
        .ZN(n4522) );
  OAI31D0 U4563 ( .A1(n4524), .A2(n4523), .A3(n4522), .B(n4521)
     , .ZN(n4525) );
  OAI211D0 U4564 ( .A1(n4528), .A2(n4527), .B(n4526), .C(n4525)
     , .ZN(n4529) );
  NR4D0 U4565 ( .A1(n4532), .A2(n4531), .A3(n4530), .A4(n4529),
      .ZN(n4533) );
  OAI211D0 U4566 ( .A1(n4536), .A2(n4535), .B(n4534), .C(n4533)
     , .ZN(n4537) );
  NR4D0 U4567 ( .A1(n4540), .A2(n4539), .A3(n4538), .A4(n4537),
      .ZN(n4541) );
  OAI222D0 U4568 ( .A1(\q_out[0]_BAR ), .A2(n4545), .B1(n4544),
      .B2(n4543),
        .C1(n4542), .C2(n4541), .ZN(n4812) );
endmodule
```

```
module IO_nanochip ( q_out, clk, reset_nanochip, select,
   reset_ring_osc,
        out_ring_osc );
  output [7:0] q_out;
  input [1:0] select;
  input clk, reset_nanochip, reset_ring_osc;
  output out_ring_osc;
  wire   \*Logic1* , net23682, clk_w, reset_nanochip_w,
     reset_ring_osc_w,
         out_ring_osc_w, n1, n2, n3, n4, n5, n6, n7;
  wire   [7:0] q_out_w;
  wire   [1:0] select_w;

  nanochip U10 ( .clk(clk_w), .reset_nanochip(reset_nanochip_w)
     , .select(
        select_w), .reset_ring_osc(reset_ring_osc_w), .
           out_ring_osc(
        out_ring_osc_w), .\q_out[6]_BAR (q_out_w[6]), .\q_out
           [5]_BAR (
        q_out_w[5]), .\q_out[4]_BAR (q_out_w[4]), .\q_out[3]
           _BAR (q_out_w[3]),
        .\q_out[2]_BAR (q_out_w[2]), .\q_out[1]_BAR (q_out_w
           [1]),
        .\q_out[0]_BAR (q_out_w[0]) );
  PDDW0204SCDG IN0 ( .I(net23682), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(
        reset_ring_osc), .DS(net23682), .PE(net23682), .C(
           reset_ring_osc_w) );
  PDDW0204SCDG IN1 ( .I(net23682), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(clk), .DS(net23682), .PE(net23682), .C(
     clk_w) );
  PDDW0204SCDG IN2 ( .I(net23682), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(
        reset_nanochip), .DS(net23682), .PE(net23682), .C(
           reset_nanochip_w) );
  PDDW0204SCDG IN3 ( .I(net23682), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(
        select[0]), .DS(net23682), .PE(net23682), .C(select_w
           [0]) );
  PDDW0204SCDG IN4 ( .I(net23682), .OEN(\*Logic1* ), .IE(\*
     Logic1* ), .PAD(
        select[1]), .DS(net23682), .PE(net23682), .C(select_w
           [1]) );
  PDDW0204SCDG OUT0 ( .I(out_ring_osc_w), .OEN(net23682), .IE
     (\*Logic1* ),
        .PAD(out_ring_osc), .DS(net23682), .PE(net23682) );
  PDDW0204SCDG OUT1 ( .I(n1), .OEN(net23682), .IE(\*Logic1* ),
     .PAD(q_out[0]),
```

```
        .DS(net23682), .PE(net23682) );
  PDDW0204SCDG OUT2 ( .I(n2), .OEN(net23682), .IE(\*Logic1* ),
     .PAD(q_out[1]),
        .DS(net23682), .PE(net23682) );
  PDDW0204SCDG OUT3 ( .I(n3), .OEN(net23682), .IE(\*Logic1* ),
     .PAD(q_out[2]),
        .DS(net23682), .PE(net23682) );
  PDDW0204SCDG OUT4 ( .I(n4), .OEN(net23682), .IE(\*Logic1* ),
     .PAD(q_out[3]),
        .DS(net23682), .PE(net23682) );
  PDDW0204SCDG OUT5 ( .I(n5), .OEN(net23682), .IE(\*Logic1* ),
     .PAD(q_out[4]),
        .DS(net23682), .PE(net23682) );
  PDDW0204SCDG OUT6 ( .I(n6), .OEN(net23682), .IE(\*Logic1* ),
     .PAD(q_out[5]),
        .DS(net23682), .PE(net23682) );
  PDDW0204SCDG OUT7 ( .I(n7), .OEN(net23682), .IE(\*Logic1* ),
     .PAD(q_out[6]),
        .DS(net23682), .PE(net23682) );
  PDDW0204SCDG OUT8 ( .I(net23682), .OEN(net23682), .IE(\*
     Logic1* ), .PAD(
        q_out[7]), .DS(net23682), .PE(net23682) );
  TIEH U6 ( .Z(\*Logic1* ) );
  TIEL U7 ( .ZN(net23682) );
  CKND1 U8 ( .I(q_out_w[6]), .ZN(n7) );
  CKND1 U9 ( .I(q_out_w[0]), .ZN(n1) );
  CKND1 U11 ( .I(q_out_w[5]), .ZN(n6) );
  CKND1 U12 ( .I(q_out_w[2]), .ZN(n3) );
  CKND1 U13 ( .I(q_out_w[4]), .ZN(n5) );
  CKND1 U14 ( .I(q_out_w[1]), .ZN(n2) );
  CKND1 U15 ( .I(q_out_w[3]), .ZN(n4) );
endmodule
```

18.3. Códigos para la simulación e implementación del *gate-level netlist* del *core* de "El Gran Jaguar" en el FPGA Xilinx Virtex-5 Genesys

18.3.1. Código de Verilog donde se definen las *blackboxes* del *core* de "El Gran Jaguar"

Cuadro 101. Archivo Blackboxes.v

```
// ======================================
// Blackboxes.v - Biblioteca de celdas (9t) para la sintesis
   del core del nanochip en FPGA
```

```
// ========================================

//-----------------------------------------------
module OA211D0 (A1, A2, B, C, Z);
    input A1, A2, B, C;
    output Z;
        assign Z = (A1 | A2) & B & C;
endmodule
//-----------------------------------------------

module OA21D0 (A1, A2, B, Z);
    input A1, A2, B;
    output Z;
        assign Z = (A1 | A2) & B;
endmodule
//-----------------------------------------------

module OA221D0 (A1, A2, B1, B2, C, Z);
    input A1, A2, B1, B2, C;
    output Z;
        assign Z = (A1 | A2) & (B1 | B2) & C;
endmodule
//-----------------------------------------------

module OA22D0 (A1, A2, B1, B2, Z);
    input A1, A2, B1, B2;
    output Z;
        assign Z = (A1 | A2) & (B1 | B2);
endmodule
//-----------------------------------------------

module OA31D0 (A1, A2, A3, B, Z);
    input A1, A2, A3, B;
    output Z;
        assign Z = (A1 | A2 | A3) & B;
endmodule
//-----------------------------------------------

module OA32D0 (A1, A2, A3, B1, B2, Z);
    input A1, A2, A3, B1, B2;
    output Z;
        assign Z = (A1 | A2 | A3) & (B1 | B2);
endmodule
//-----------------------------------------------

module OAI211D0 (A1, A2, B, C, ZN);
    input A1, A2, B, C;
    output ZN;
```

```
        assign ZN = ~((A1 | A2) & B & C);
endmodule
//-------------------------------------------

module OAI21D0 (A1, A2, B, ZN);
    input A1, A2, B;
    output ZN;
        assign ZN = ~((A1 | A2) & B);
endmodule
//-------------------------------------------

module OAI221D0 (A1, A2, B1, B2, C, ZN);
    input A1, A2, B1, B2, C;
    output ZN;
        assign ZN = ~((A1 | A2) & (B1 | B2) & C);
endmodule
//-------------------------------------------
module OAI222D0 (A1, A2, B1, B2, C1, C2, ZN);
    input A1, A2, B1, B2, C1, C2;
    output ZN;
        assign ZN = ~((A1 | A2) & (B1 | B2) & (C1 | C2));
endmodule

//-------------------------------------------
module OAI22D0 (A1, A2, B1, B2, ZN);
    input A1, A2, B1, B2;
    output ZN;
        assign ZN = ~((A1 | A2) & (B1 | B2));
endmodule

//-------------------------------------------
module OAI31D0 (A1, A2, A3, B, ZN);
    input A1, A2, A3, B;
    output ZN;
        assign ZN = ~((A1 | A2 | A3) & B);
endmodule

//-------------------------------------------
module OAI32D0 (A1, A2, A3, B1, B2, ZN);
    input A1, A2, A3, B1, B2;
    output ZN;
        assign ZN = ~((A1 | A2 | A3) & (B1 | B2));
endmodule

//-------------------------------------------
module OR2D0 (A1, A2, Z);
    input A1, A2;
    output Z;
```

```
        assign Z = A1 | A2;
endmodule

//-------------------------------------------
module OR2D1 (A1, A2, Z);
    input A1, A2;
    output Z;
    assign #0.1 Z = A1 | A2;
endmodule

//-------------------------------------------
module OR3D0 (A1, A2, A3, Z);
    input A1, A2, A3;
    output Z;
        assign Z = A1 | A2 | A3;
endmodule

//-------------------------------------------
module OR4D0 (A1, A2, A3, A4, Z);
    input A1, A2, A3, A4;
    output Z;
        assign Z = A1 | A2 | A3 | A4;
endmodule

//-------------------------------------------
module TIEL (ZN);
  output  ZN;
  assign ZN  = 1'b0;
endmodule

//-------------------------------------------
module XNR2D0 (A1, A2, ZN);
  input         A1, A2;
  output        ZN;
  assign ZN = ~(A1 ^ A2);
endmodule

//-------------------------------------------
// Cells that are missing
//-------------------------------------------
module AO222D0 (A1, A2, B1, B2, C1, C2, Z);
    input A1, A2, B1, B2, C1, C2;
    output Z;
        assign Z = (A1 & A2) | (B1 & B2) | (C1 & C2);
endmodule

// ---------------- CKND2D1 ----------------
// ZN = ~(A1 & A2)
```

```
module CKND2D1 (
    input  A1, input A2,
    output ZN
);
    assign ZN = ~(A1 & A2);
endmodule

// Celda: IIND4D0 - Compuerta NAND de 4 entradas con inversores
    en A1 y A2
module IIND4D0 (A1, A2, B1, B2, ZN);
    input A1, A2, B1, B2;
    output ZN;
        assign ZN = ~(B1 & B2 & ~A1 & ~A2);
endmodule

// Celda: IINR4D0 - Compuerta NOR de 4 entradas con inversores
   en A1 y A2
module IINR4D0 (A1, A2, B1, B2, ZN);
    input A1, A2, B1, B2;
    output ZN;
        assign ZN = ~(B1 | B2 | ~A1 | ~A2);
endmodule

// Celda: IND2D0 - Compuerta NAND de 2 entradas con inversor en
    A1
module IND2D0 (A1, B1, ZN);
    input A1, B1;
    output ZN;
        assign ZN = ~(B1 & ~A1);
endmodule

// Celda: IND3D0 - Compuerta NAND de 3 entradas con inversor en
    A1
module IND3D0 (A1, B1, B2, ZN);
    input A1, B1, B2;
    output ZN;
        assign ZN = ~(B1 & B2 & ~A1);
endmodule

// Celda: IND4D0 - Compuerta NAND de 4 entradas con inversor en
    A1
module IND4D0 (A1, B1, B2, B3, ZN);
    input A1, B1, B2, B3;
    output ZN;
        assign ZN = ~(B1 & B2 & ~A1 & B3);
endmodule
```

```
// Celda: INR2D0 - Compuerta NOR de 2 entradas con inversor en
   A1
module INR2D0 (A1, B1, ZN);
    input A1, B1;
    output ZN;
        assign ZN = ~(B1 | ~A1);
endmodule

// Celda: INR3D0 - Compuerta NOR de 3 entradas con inversor en
   A1
module INR3D0 (A1, B1, B2, ZN);
    input A1, B1, B2;
    output ZN;
        assign ZN = ~(B1 | B2 | ~A1);
endmodule

// Celda: INR4D0 - Compuerta NOR de 4 entradas con inversor en
   A1
module INR4D0 (A1, B1, B2, B3, ZN);
    input A1, B1, B2, B3;
    output ZN;
        assign ZN = ~(B1 | B2 | ~A1 | B3);
endmodule

// Celda: INVD0 - Inversor simple
module INVD0 (I, ZN);
    input I;
    output ZN;
        assign ZN = ~I;
endmodule

// Celda: INVD1 - Inversor simple (version D1)
module INVD1 (I, ZN);
    input I;
    output ZN;
        assign #0.1 ZN = ~I;
endmodule

// Celda: IOA21D0 - Compuerta AND-OR-INVERT con inversores en
   entradas A
module IOA21D0 (A1, A2, B, ZN);
    input A1, A2, B;
    output ZN;
        assign ZN = ~((~A1 | ~A2) & B);
endmodule

// Celda: MOAI22D0 - Compuerta OR-NAND (Majority OAI)
module MOAI22D0 (A1, A2, B1, B2, ZN);
```

```
    input A1, A2, B1, B2;
    output ZN;
        assign ZN = ~((A1 | A2) & ~(B1 & B2));
endmodule

// Celda: ND3D0 - Compuerta NAND de 3 entradas
module ND3D0 (A1, A2, A3, ZN);
    input A1, A2, A3;
    output ZN;
        assign ZN = ~(A1 & A2 & A3);
endmodule

// Celda: ND4D0 - Compuerta NAND de 4 entradas
module ND4D0 (A1, A2, A3, A4, ZN);
    input A1, A2, A3, A4;
    output ZN;
        assign ZN = ~(A1 & A2 & A3 & A4);
endmodule

// Celda: NR2D0 - Compuerta NOR de 2 entradas
module NR2D0 (A1, A2, ZN);
    input A1, A2;
    output ZN;
    assign ZN = ~(A1 | A2);
endmodule

// Celda: NR2XD0 - Compuerta NOR de 2 entradas (version XD0)
module NR2XD0 (A1, A2, ZN);
    input A1, A2;
    output ZN;
    assign ZN = ~(A1 | A2);
endmodule

// Celda: NR3D0 - Compuerta NOR de 3 entradas
module NR3D0 (A1, A2, A3, ZN);
    input A1, A2, A3;
    output ZN;
    assign ZN = ~(A1 | A2 | A3);
endmodule

// Celda: NR4D0 - Compuerta NOR de 4 entradas
module NR4D0 (A1, A2, A3, A4, ZN);
    input A1, A2, A3, A4;
    output ZN;
    assign ZN = ~(A1 | A2 | A3 | A4);
endmodule

// ---------------- AN3D0 ----------------
```

```
module AN3D0 (
    input  A1, input A2, input A3,
    output Z
);
    assign Z = A1 & A2 & A3;
endmodule

// ---------------- AN4D0 ----------------
module AN4D0 (
    input  A1, input A2, input A3, input A4,
    output Z
);
    assign Z = A1 & A2 & A3 & A4;
endmodule

// ---------------- AO211D0 ----------------
// Z = (A1 & A2) | B | C
module AO211D0 (
    input  A1, input A2, input B, input C,
    output Z
);
    assign Z = (A1 & A2) | B | C;
endmodule

// ---------------- AO21D0 ----------------
// Z = (A1 & A2) | B
module AO21D0 (
    input  A1, input A2, input B,
    output Z
);
    assign Z = (A1 & A2) | B;
endmodule

// ---------------- AO22D0 ----------------
// Z = (A1 & A2) | (B1 & B2)
module AO22D0 (
    input  A1, input A2, input B1, input B2,
    output Z
);
    assign Z = (A1 & A2) | (B1 & B2);
endmodule

// ---------------- AO31D0 ----------------
// Z = (A1 & A2 & A3) | B
module AO31D0 (
    input  A1, input A2, input A3, input B,
    output Z
);
```

```
    assign Z = (A1 & A2 & A3) | B;
endmodule

// ---------------- AOI211D0 ----------------
// ZN = ~((A1 & A2) | B | C)
module AOI211D0 (
    input  A1, input A2, input B, input C,
    output ZN
);
    assign ZN = ~((A1 & A2) | B | C);
endmodule

// ---------------- AOI21D0 ----------------
// ZN = ~((A1 & A2) | B)
module AOI21D0 (
    input  A1, input A2, input B,
    output ZN
);
    assign ZN = ~((A1 & A2) | B);
endmodule

// ---------------- AOI221D1 ----------------
// ZN = ~((A1 & A2) | (B1 & B2) | C)
module AOI221D1 (
    input  A1, input A2, input B1, input B2, input C,
    output ZN
);
    assign ZN = ~((A1 & A2) | (B1 & B2) | C);
endmodule

// ---------------- AOI221D0 ----------------
// ZN = ~((A1 & A2) | (B1 & B2) | C)
module AOI221D0 (
    input  A1, input A2, input B1, input B2, input C,
    output ZN
);
    assign ZN = ~((A1 & A2) | (B1 & B2) | C);
endmodule

// ---------------- AOI222D0 ----------------
// ZN = ~((A1 & A2) | (B1 & B2) | (C1 & C2))
module AOI222D0 (
    input  A1, input A2, input B1, input B2, input C1, input C2
       ,
    output ZN
);
    assign ZN = ~((A1 & A2) | (B1 & B2) | (C1 & C2));
endmodule
```

```
// ---------------- AOI22D0 ----------------
// ZN = ~((A1 & A2) | (B1 & B2))
module AOI22D0 (
    input  A1, input A2, input B1, input B2,
    output ZN
);
    assign ZN = ~((A1 & A2) | (B1 & B2));
endmodule

// ---------------- AOI31D0 ----------------
// ZN = ~((A1 & A2 & A3) | B)
module AOI31D0 (
    input  A1, input A2, input A3, input B,
    output ZN
);
    assign ZN = ~((A1 & A2 & A3) | B);
endmodule

// ---------------- AOI32D0 ----------------
// ZN = ~((A1 & A2 & A3) | (B1 & B2))
module AOI32D0 (
    input  A1, input A2, input A3, input B1, input B2,
    output ZN
);
    assign ZN = ~((A1 & A2 & A3) | (B1 & B2));
endmodule

/* ---------------- AOI33D0 ----------------
// ZN = ~((A1 & A2 & A3) | (B1 & B2 & B3))
module AOI33D0 (
    input  A1, input A2, input A3, input B1, input B2, input B3
       ,
    output ZN
);
    assign ZN = ~((A1 & A2 & A3) | (B1 & B2 & B3));
endmodule*/

// ---------------- CKAN2D0 ----------------
// Z = A1 & A2
module CKAN2D0 (
    input  A1, input A2,
    output Z
);
    assign Z = A1 & A2;
endmodule

// ---------------- CKND1 ----------------
```

```
// ZN = ~I
module CKND1 (
    input  I,
    output ZN
);
    assign ZN = ~I;
endmodule

// ---------------- CKND2D0 ----------------
// ZN = ~(A1 & A2)
module CKND2D0 (
    input  A1, input A2,
    output ZN
);
    assign ZN = ~(A1 & A2);
endmodule

// ---------------- DFCNQD1 ----------------
// Flip-Flop D con reset asincrono activo en bajo (clear)
module DFCNQD1 (
    input  wire D,
    input  wire CP,    // reloj (flanco positivo)
    input  wire CDN,   // reset asincrono activo en 0
    output reg  Q
);
    always @(posedge CP or negedge CDN) begin
        if (!CDN)
            Q <= 1'b0;
        else
            Q <= D;
    end
endmodule

// ---------------- DFQD1 ----------------
// Flip-Flop D sin reset (flanco positivo)
module DFQD1 (
    input  wire D,
    input  wire CP,   // reloj (flanco positivo)
    output reg  Q
);
    always @(posedge CP) begin
        Q <= D;
    end
endmodule

// ---------------- IAO21D0 ----------------
// ZN = ~((~A1 & ~A2) | B) = (~B) & (A1 | A2)
module IAO21D0 (
```

```
    input  A1, input A2, input B,
    output ZN
);
    assign ZN = (~B) & (A1 | A2);
endmodule
```

18.3.2. Código para la asignación de pines del FPGA

Cuadro 102. Código UCF para la asignación de pines del FPGA

```
// Comando para definir clk como pin de entrada
NET "clk" CLOCK_DEDICATED_ROUTE = FALSE;
NET "clk" LOC = "AM13";

// pines simulacion -------------
NET "reset_nanochip" LOC = "J19" | PULLDOWN;
NET "select[0]" LOC = "L18" | PULLDOWN;
NET "select[1]" LOC = "K18" | PULLDOWN;

NET "reset_ring_osc" LOC = "H18" | PULLDOWN;
// ----------------------

// pines implementacion ----------
//NET "reset_nanochip" LOC = "AM12";
//NET "select[0]" LOC = "AD10";
//NET "select[1]" LOC = "AE8";

//NET "reset_ring_osc" LOC = "AF10";
// ----------------------

// NO CAMBIAR
NET "out_ring_osc" LOC = "AL11";

// pines simulacion -------------
NET "q_out[0]" LOC = "AG8";
NET "q_out[1]" LOC = "AH8";
NET "q_out[2]" LOC = "AH9";
NET "q_out[3]" LOC = "AG10";

NET "q_out[4]" LOC = "AH10";
NET "q_out[5]" LOC = "AG11";
NET "q_out[6]" LOC = "AF11";
NET "q_out[7]" LOC = "AE11";
// ----------------------

// pines implementacion ----------
//NET "q_out[7]" LOC = "AE9";
```

```
//NET "q_out[6]" LOC = "AC8";
//NET "q_out[5]" LOC = "AB10";
//NET "q_out[4]" LOC = "AC9";

//NET "q_out[3]" LOC = "AF8";
//NET "q_out[2]" LOC = "AB8";
//NET "q_out[1]" LOC = "AA10";
//NET "q_out[0]" LOC = "AA9";
// ---------------------
```

18.3.3. Código de Python ejecutado en la Raspberry Pi 4B para la recepción y reproducción del texto ASCII

Cuadro 103. Archivo nano_RPI_modified.py

```
from gtts import gTTS
import os
import re
import RPi.GPIO as GPIO
import time
from multiprocessing import Process, Value, Array, Queue
from threading import Thread
import subprocess  # Para manejar subprocesos
from langdetect import detect

# Limpiar GPIOs al correr de nuevo
GPIO.cleanup()

# Ignorar advertencias del GPIO
GPIO.setwarnings(False)

# Definir los pines GPIO que se utilizaran para recibir la
   salida q[7:0]
gpio_pins = [17, 18, 27, 22, 23, 24, 25, 4]

# Pines para las entradas para inicio, reloj, reset y seleccion
    de idioma
reset_nanochip = 20 #Pin reset nanochip
clock_pin = 21          # Pim clk
reset_ring_osc = 26  # Pin para el reset del ring osc
select_1 = 19  # Primer bit de seleccion de idioma
select_2 = 13  # Segundo bit de seleccion de idioma

# Configurar los pines GPIO
GPIO.setmode(GPIO.BCM)

# Configurar cada pin de entrada para leer datos
```

```
for pin in gpio_pins:
    GPIO.setup(pin, GPIO.IN)

# Configuracion de los pines de inicio, reloj, reset y
   seleccion de idioma
GPIO.setup(reset_nanochip, GPIO.OUT)
GPIO.setup(clock_pin, GPIO.OUT)
GPIO.setup(reset_ring_osc, GPIO.OUT)
GPIO.setup(select_1, GPIO.OUT)
GPIO.setup(select_2, GPIO.OUT)

#max_length = 2371
# Bandera compartida entre procesos para detener la lectura
stop_flag = Value('b', False)
#received_text = Array('b', max_length)
# Indice compartido para rastrear la posicion actual en el
   texto recibido
index_value = Value('i', 0)

def send_clock_pulse():
    """Envia un pulso en el pin de reloj."""
    GPIO.output(clock_pin, GPIO.HIGH)
    time.sleep(0.001)
    GPIO.output(clock_pin, GPIO.LOW)

def initialize_pins():
    """Inicializa las senales de inicio e idioma."""
    GPIO.output(reset_nanochip, GPIO.HIGH)  # Reset
    GPIO.output(select_1, GPIO.HIGH)  # Configura idioma
       inicial
    GPIO.output(select_2, GPIO.LOW)
    GPIO.output(reset_ring_osc, GPIO.HIGH) # Reset del
       oscilador
    time.sleep(0.1)
    GPIO.output(reset_ring_osc, GPIO.LOW)
    GPIO.output(reset_nanochip, GPIO.LOW)

def read_byte():
    """Lee un byte a traves de los pines GPIO."""
    byte = 0
    # Leer cada pin y construir el byte bit por bit
    for i, pin in enumerate(gpio_pins):
        if GPIO.input(pin):
            byte |= (1 << i)
    return byte

def get_language():
```

```
    """Obtiene el idioma en funcion de los pines de seleccion
       de idioma."""
    # Leer el estado de los pines de seleccion
    lang_bit1 = GPIO.input(select_1)
    lang_bit2 = GPIO.input(select_2)

    # Determinar idioma segun la combinacion de bits
    if lang_bit1 == GPIO.HIGH and lang_bit2 == GPIO.HIGH:
        return 'es'  # Espanol
    elif lang_bit1 == GPIO.LOW and lang_bit2 == GPIO.HIGH:
        return 'en'  # Ingles
    else:
        return 'en'  # Valor predeterminado si no se reconoce
           la combinacion

def looks_like_name(phrase):
    """Heuristica para saber si la mayoria de palabras son en
       espanol."""
    # Dividir la frase en palabras
    words = phrase.strip().split()
    if not words:
        return False
    # Calcular la proporcion de palabras que comienzan con
       mayuscula
    uppercase_ratio = sum(1 for w in words if w[0].isupper())/
       len(words)
    # Verificar si contiene comas o 'j' (comun en nombres
       espanoles)
    has_commas = ',' in phrase or 'j' in phrase
    return uppercase_ratio > 0.5 and has_commas

def speak_phrase(phrase_queue):
    """Funcion que dice las frases utilizando gTTS en el idioma
       correspondiente."""
    while True:
        # Esperar hasta que haya una frase en la cola
        if not phrase_queue.empty():
            phrase = phrase_queue.get()
            # Senal para detener el proceso de voz
            if phrase == "STOP":
                break

            try:
                # Detectar automaticamente el idioma de la
                   frase
                detected = detect(phrase)
                lang = 'es' if detected.startswith('es') else '
                   en'
```

```
                # Aplicar heuristica para nombres y forzar
                    espanol si aplica
                if looks_like_name(phrase):
                    lang = 'es'

                print(f"Diciendo frase en {lang}: '{phrase}'")
                # Generar el archivo de audio con gTTS
                tts = gTTS(text=phrase, lang=lang)
                tts.save("output.mp3")

                # Reproducir la frase de manera bloqueante para
                    reducir silencios entre frases
                subprocess.run(['mpg321', '-q', 'output.mp3'])
                    # -q silencia los mensajes de mpg321

            except Exception as e:
                print(f"Error al ejecutar gTTS: {e}")

def reset_monitor(index_value, stop_flag, phrase_queue,
    idle_timeout=5):
    """Supervisa el estado del indice y activa el reset si
        alcanza el maximo."""
    last_index = index_value.value
    last_change_time = time.time()

    while not stop_flag.value:
        current_index = index_value.value
        # Verificar si el indice ha cambiado
        if current_index != last_index:
            last_index = curent_index
            last_change_time = time.time()
        # Si ha pasado demasiado tiempo sin cambios, reiniciar
        elif time.time() - last_change_time > idle_timeout:
        #if index_value.value >= :  # Verifica si se alcanzo el
            limite
            print("Inactividad detectada. Reiniciando...")
            #print("Limite alcanzado. Activando reset y
                reinicio del proceso...")
            # activate_reset()
            reset_parameters(phrase_queue)
            break
        time.sleep(0.1)

#def activate_reset():
#    """Activa el pin de reset por un breve momento."""
#    GPIO.output(reset_ring_osc, GPIO.HIGH)
#    time.sleep(0.1)
```

```
#    GPIO.output(reset_ring_osc, GPIO.LOW)

def reset_parameters(phrase_queue):
    """Reinicia parametros importantes del sistema y prepara
       para continuar."""
    #global received_text, index_value
    # Reiniciar los datos recibidos
    # Reiniciar el indice a cero
    index_value.value = 0
    # Reiniciar la bandera de parada
    stop_flag.value = False
    #for i in range(max_length):
    #    received_text[i] = 0
    #index_value.value = 0
    #stop_flag.value = False  # Reiniciar la bandera de parada

    # Vaciar la cola de frases
    while not phrase_queue.empty():
        phrase_queue.get()

    print("Parametros reiniciados. Listo para nuevo proceso.")

def read_data_with_voice(stop_flag, phrase_queue, index_value):
    """Funcion para leer los datos a traves de los pines GPIO y
       anadir frases a la cola."""
    # Lista para almacenar el texto recibido
    received_text = []
    phrase_buffer = []  # Inicializa phrase_buffer como una
       lista vacia
    while not stop_flag.value:
        # Enviar pulso de reloj para leer el siguiente byte
        send_clock_pulse()
        # Leer el byte de los pines GPIO
        byte = read_byte()
        #index = index_value.value
        #if index >= max_length:  # Detener si se alcanza el
           limite
        #    break

        # Verificar que el byte sea valido
        if 0 <= byte <= 255:
            # Convertir el byte a caracter
            char = chr(byte)
            # Agregar el caracter al texto recibido
            received_text.append(char)
            # Incrementar el indice
            index_value.value += 1
            print(f"Byte leido: {byte} -> '{char}'")
```

```
            # Frases separadas por .
            # Acumular caracteres hasta encontrar un punto
            if char != '.':
                phrase_buffer.append(char)
            else:
                # Formar la frase completa
                phrase = ''.join(phrase_buffer).strip()
                phrase_buffer.clear()  # Reinicia el buffer
                   tras formar la frase
                # Agregar la frase a la cola si no esta vacia
                if phrase:
                    phrase_queue.put(phrase)

            # Detectar el fin del texto
            if char == '\n':  # Fin del texto
                # Activar bandera de parada
                stop_flag.value = True
                # Enviar senal de parada al proceso de voz
                phrase_queue.put("STOP")

        time.sleep(0.01)
    # Retornar el texto completo recibido
    return ''.join(received_text)

def main():
    try:
        initialize_pins()  # Inicializar los pines de inicio e
           idioma

        while True:  # Bucle principal para permitir reinicio
           continuo
            phrase_queue = Queue()  # Crear una nueva cola para
                cada ciclo
            stop_flag.value = False  # Reiniciar bandera de
               parada

            # Iniciar procesos e hilos
            # Proceso para leer datos desde GPIO
            read_process = Process(target=read_data_with_voice,
                args=(stop_flag, phrase_queue, index_value))
            # Proceso para reproducir las frases con voz
            speak_process = Process(target=speak_phrase, args=(
               phrase_queue,))
            # Hilo para monitorear inactividad y reiniciar si
               es necesario
            reset_thread = Thread(target=reset_monitor, args=(
               index_value, stop_flag, phrase_queue))
```

```
            # Iniciar los procesos e hilos
            read_process.start()
            speak_process.start()
            reset_thread.start()

            read_process.join()  # Esperar a que termine la
               lectura
            reset_thread.join()  # Esperar a que el hilo de
               monitoreo termine

            # Mostrar el texto recibido tras completar la
               lectura
            print("Texto␣recibido:")
            # Filtrar solo bytes validos
            valid_bytes = [b for b in received_text[:
               index_value.value] if 0 <= b <= 255]
            # Decodificar a texto UTF-8
            decoded_text = bytes(valid_bytes).decode('utf-8',
               errors='replace')
            print(decoded_text)

            # Indicar que el proceso de voz debe detenerse
            phrase_queue.put("STOP")
            speak_process.join()  # Esperar a que termine el
               proceso de voz

            print("Reiniciando␣para␣nuevo␣ciclo...")
            reset_parameters(phrase_queue)  # Reiniciar
               parametros y limpiar
            pass
    finally:
        # Limpiar configuracion de GPIO al finalizar
        GPIO.cleanup()

# Punto de entrada del programa
if __name__ == "__main__":
    main()
```

18.4. Códigos para la automatización de la instalación de las aplicaciones de Synopsys

18.4.1. Código del *script* APPS.sh editado

Cuadro 104. Archivo APPS.sh

```
#! /bin/bash
# Autor > Paola Alejandra Mendizabal Batres
# Objetivo > Descarga e Instalacion de Aplicaciones y Librerias
    de Synopsys para el Laboratorio de Nanoelectronica y
   proyecto Gran Jaguar.
# Fecha > Enero de 2023
#
   ------------------------------------------------------------------------

# Editor: Carlos Valdes
# Fecha: Octubre 2025
#------------------------------------------------------------------------

# ++++++++++++++++++++++++++++++++++++++++ VARIABLES DE
   USUARIO ++++++++++++++++++++++++++++++
SRC_DIR=$HOME/Downloads
TRGT_DIR=/usr/synopsys

EDK_DIR=$HOME/Desktop/Folder_de_Trabajo
PDK_DIR=/usr/synopsys/iPDK

VERSION_ALL=0
NUMB_VERSIONS=6

PRODUCT_DIR=/site31759/MyProducts

DIR_AUTH=auth
DIR_REV=rev
DIR_REVO=rev_o

LIB_EFT_DIR=/Libraries

# ++++++++++++++++++++++++++++++++++++++++++++ VARIABLES
   ++++++++++++++++++++++++++++++++++++
SFTP_SERVER=eft.synopsys.com

HERE=$(pwd)

#
   ------------------------------------------------------------------------

# Este codigo es necesario para realizar la conexion ssh con el
    servidor sin verificar la Host Key (NO BORRAR)
export LFTP_HOME="$HOME/.lftp"
mkdir -p "$LFTP_HOME"
echo "set␣sftp:connect-program␣\"ssh␣-a␣-x␣-o␣
   StrictHostKeyChecking=no␣-o␣UserKnownHostsFile=/dev/null\""
```

```
    > "$LFTP_HOME/rc"
#
    ------------------------------------------------------------------------


apps_formal=('VCS y DVE' 'Verdi' 'ICC' 'ICC2' 'IC Validator' '
    Cosmos Scope' 'Custom Compiler' 'Custom Waveview' 'Formality
    ' 'STAR RC' 'PrimeWave' 'IC Validator Workbench' 'Primetime'
     'Design Vision, Design Compiler y TestMAX DFT (Synthesis)'
    'TestMax ATPG' 'PrimeSim' 'PrimeSim XA' 'PrimeSim HSPICE' '
    Todas')

apps_real=('vcs_all dve' 'verdi' 'ic_compiler' 'icc2' '
    icvalidator' 'cosmosscope' 'cust_compiler' 'wv_adv' 'fm' '
    starrc' 'p_wave' 'icv_wb' 'pt' 'syn' 'tx' 'p_sim' 'customsim
    ' 'hspice')

re='^[0-9]+$'
shopt -s extglob
contador=0
flag=0
new_array=()
gui=0

# ++++++++++++++++++++++++++++++++++++++++ BANDERAS
    ++++++++++++++++++++++++++++++++++++++++

no_inst=0           # Se activa si el installer no esta
    instalado.
#
    ++++++++++++++++++++++++++++++++++++++
```

```
cd $PRODUCT_DIR/$DIR_AUTH
ls > myproductsauth.txt
cd ..
cd $PRODUCT_DIR/$DIR_REVO
ls > myproductsrevold.txt
bye
EOF
awk '{print $9, "rev"}' myproductsrev.txt > tmp_file && mv
   tmp_file myproductsrev.txt
awk '{print $9, "rev_o"}' myproductsrevold.txt > tmp_file && mv
    tmp_file myproductsrevold.txt
awk '{print $9, "auth"}' myproductsauth.txt > tmp_file && mv
   tmp_file myproductsauth.txt
awk '{print $9}' libsyn.txt > tmp_file && mv tmp_file libsyn.
   txt
date >> myproductsrev.txt
}

# OBJETIVO: Enumerar los elementos de un array.
display_list() {
contador=0
arr=("$@")
for j in "${arr[@]}"; do
        echo "$contador␣$j"
        let contador++
done
}

# OBJETIVO: Verificar que la variable no este vacia y sea un
   numero.
verif_input() {
if [[ -z $1 ]] ; then
    echo "Variable␣vacia.␣Adios!"
    exit 0
fi

if [[ ! $1 =~ $re && "$1" != "b" ]] ; then
    echo "La␣variable␣no␣es␣un␣numero.␣Adios!"
    exit 0
fi
}

# OBJETIVO: Verificar que la variable se encuentre dentro del
   rango especificado.
verif_input2() {
if (($1 < 0 || $1 >= $2)); then
echo "Fuera␣de␣rango.␣Adios!␣"
exit 0
```

```
fi
}

# OBJETIVO: Verificar que el directorio exista.
verif_folder() {
if [ ! -d $1 ]; then
        echo "ERROR > El directorio $1 no existe. "
        exit 0
fi
}

# OBJETIVO: Verificar que el archivo exista.
verif_file() {
if [ ! -f $1 ]; then
        echo "ERROR > El archivo $1 no existe. "
        exit 0
fi
}

# OBJETIVO: Verificar si los archivos de texto donde se guardan
    las versiones de cada aplicacion existen.
# Si no existen, actualizar el listado.
verif_file_ls() {
if [ ! -f $1 ]; then
        echo "El archivo $1 no existe. Actualizando el listado
           de versiones.. "
    ls_products
fi
}

# OBJETIVO: Verificar si la variable esta vacia.
var_empty() {
var=$3
if [[ -z ${var// } ]] ; then
        echo "ERROR > $1."
        if (($2==1)) ; then
                exit 0
        else
                flag=1
        fi
fi
}

# OBJETIVO: Obtener el listado de versiones de la app
   seleccionada.
obtener_versiones() {
echo -e "01 > Obteniendo el listado de versiones de ${
   apps_formal[$numapp]}... \n"
```

```
if (($numapp==0)); then
    read -a var_sep <<< "${apps_real[$numapp]}"
    ver_app=($(awk 'match($0, '/${var_sep[0]}/') {print $1}'
       myproductsrev.txt myproductsrevold.txt myproductsauth.
       txt | sort -r | head -$NUMB_VERSIONS))
    ver_app2=($(awk 'match($0, '/${var_sep[1]}/') {print $1}'
       myproductsrev.txt myproductsrevold.txt myproductsauth.
       txt | sort -r | head -$NUMB_VERSIONS))
echo $ver_app2
    var_empty "No se encuentra la aplicacion dve" 0 $ver_app2
else
    ver_app=($(awk 'match($0, '/${apps_real[$numapp]}/') {print
        $1}' myproductsrev.txt myproductsrevold.txt
       myproductsauth.txt | sort -r | head -$NUMB_VERSIONS))
fi
var_empty "No se encuentra la aplicacion ${apps_real[$numapp]}"
    1 $ver_app
}

# OBJETIVO: Desplegar el listado de versiones de la app
   seleccionada.
desplegar_versiones() {
echo "Desplegando las ultimas $NUMB_VERSIONS versiones de ${
   apps_formal[$numapp]}..."
echo -e "*Para cambiar el numero de versiones cambiar la
   variable NUMB_VERSIONS en el script. \n"
if (($numapp==0)); then
    contador=0
    for i in "${ver_app[@]}"; do
    echo "$contador $i ---- ${ver_app2[$contador]}"
    let contador++
    done
else
    display_list "${ver_app[@]}"
fi
}

# OBJETIVO: Eliminar las carpetas de descarga que existan de la
    app y version seleccionada, para evitar conflictos con el
   checksum.
dlt_old_dir() {
echo -e "02 > Eliminando carpetas de descarga viejas > $1 \n"
var_empty "La version seleccionada no existe." 1 $1
rm -rf $SRC_DIR/$1
mkdir $SRC_DIR/$1

if (($numapp==0 & $flag==0)); then
```

```
    var_empty "La version seleccionada no existe." 1 ${ver_app2
       [$num_ver]}
    rm -rf $SRC_DIR/${ver_app2[$num_ver]}
    mkdir $SRC_DIR/${ver_app2[$num_ver]}
fi
# Obtener nombre de directorio en el servidor de la version
   seleccionada.
dir_inst=($(awk 'match($0, '/$1/') {print $2}' myproductsrev.
   txt myproductsrevold.txt myproductsauth.txt))
var_empty "No se encontro el directorio en el que se encuentra
   la aplicacion seleccionada." 1 $dir_inst
}

# OBJETIVO: Descarga de archivos de la app y version
   seleccionada.
descarga_apps() {
if (($numapp==0 & $flag==0)); then
echo "Conectando al servidor de Synopsys ..."
lftp sftp://$SFTP_USER:"$SFTP_PSW"@$SFTP_SERVER << EOF
cd $PRODUCT_DIR/$dir_inst
cd ${ver_app[$num_ver]}
ls
mget * -O "$SRC_DIR/${ver_app[$num_ver]}"
cd ..
cd ${ver_app2[$num_ver]}
ls
mget * -O "$SRC_DIR/${ver_app2[$num_ver]}"
bye
EOF
else
echo "Conectando al servidor de Synopsys ..."
lftp sftp://$SFTP_USER:"$SFTP_PSW"@$SFTP_SERVER << EOF
cd $PRODUCT_DIR/$dir_inst
cd $1
ls
mget * -O "$SRC_DIR/${ver_app[$num_ver]}"
bye
EOF
fi
}

# OBJETIVO: Verificar el checksum de los archivos descargados
   con la info de checksum_info.txt.
chcksm() {
echo -e "04 > Verificando checksum de $1... \n"
cd "$SRC_DIR/$1"
> checksum_real.txt
if [ ! -f checksum_info.txt ]; then
```

```
        echo "El␣archivo␣checksum_info.txt␣no␣existe.␣Verifique
           ␣la␣descarga␣de␣archivos."
        exit 0
fi
sed -n '1,2p' checksum_info.txt >> checksum_real.txt
cksum !(installer.log|*checksum*) >> checksum_real.txt

if [[ $(diff -w checksum_info.txt checksum_real.txt) = "" ]];
   then
    echo "Checksum␣Correcto.␣Descarga␣de␣$1␣exitosa."
else
    echo "Los␣archivos␣de␣$1␣no␣se␣han␣descargado␣correctamente
       .␣Intentelo␣de␣nuevo."
    exit 0
fi

# Descomprimir archivos para la instalacion
if ls *.gz &>/dev/null; then
        gzip -d *.gz
fi

cd "$HERE"
}

# OBETIVO: Separar el nombre y version del producto de todos
   los archivos common que existan en la carpeta de instalacion
   .
# Esto nos servira para actualizar la variable de entorno.
sep_cmn() {
echo -e "05␣>␣Obteniendo␣el␣nombre␣y␣version␣del␣archivo␣common
   ...␣\n"
arr=()
name_app=()
version=()
delimiter="_v"
ver_arr=()
contador=0

verif_folder "$SRC_DIR/$1"
cd "$SRC_DIR/$1"

name_cmn=($(ls *common*))
var_empty "No␣se␣encuentro␣el␣archivo␣common␣de␣$1" 1 $name_cmn

for item in "${name_cmn[@]}"; do
    string=$item$delimiter
    while [[ $string ]]; do
        if [[ $contador -eq 0 ]]; then
```

```
            name_app+=( "${string%%"$delimiter"*}" )
            string=${string#*"$delimiter"}
            let contador++
        else
            ver_arr+=( "${string%%"$delimiter"*}" )
            string=${string#*"$delimiter"}
            let contador--
        fi
    done
done

for value in ${ver_arr[@]}; do
    IFS_OLD=$IFS
    IFS='_'
    read -a var_sep <<< "$value"
    version+=( ${var_sep[0]} )
    IFS=$IFS_OLD
done

var_empty "No se pudo obtener el nombre del archivo common" 1
   $name_app
var_empty "No se pudo obtener la version del archivo common" 1
   $version

echo "NOMBRES:"
for value in ${name_app[@]}; do
        echo $value
done

echo "VERSIONES:"
for value in ${version[@]}; do
        echo $value
done

cd "$HERE"
}

# OBJETIVO: Eliminar archivos de instalacion de versiones
   diferentes y llamar al instalador de Synopsys.
instalacion() {
contador=0
echo -e "06 > Eliminando archivos de instalacion diferentes a $
   {ver_app[$num_ver]} \n"
for z in "${name_app[@]}"; do
        if [ -d "$TRGT_DIR/$z" ]; then
                cd "$TRGT_DIR/$z"
                rm -rf !(${version[contador]})
        fi
```

```
        echo "DIRECTORIO > $TRGT_DIR/$z/${version[contador]}"
        let contador++
done
# Llamar al installer
echo "07 > Instalando ${ver_app[$num_ver]}.."
installer -source "$SRC_DIR/${ver_app[$num_ver]}" -target "
   $TRGT_DIR"
cd "$HERE"
}

# OBJETIVO: Actualizar bashrc.
set_env() {
contador=0
echo "07 > Actualizando las variables de entorno de ${ver_app[
   $num_ver]}..."
for y in "${name_app[@]}"; do
    sed -i '/\/'$y'\//d' ~/.bashrc
    echo "Nombre: $y"
    echo -e "Version: ${version[contador]} \n"

    # Variable de entorno para HSPICE
    if (($numapp==17)); then
    echo 'PATH='$TRGT_DIR'/'$y'/'${version[contador]}'/'$y'/bin
       :$PATH' >> ~/.bashrc
    echo 'PATH='$TRGT_DIR'/'$y'/'${version[contador]}'/'$y'/bin
       :$PATH'
    # Variable de entorno para CSCOPE
    elif (($numapp==5)); then
    echo 'PATH='$TRGT_DIR'/'$y'/'${version[contador]}'/ai_bin:
       $PATH' >> ~/.bashrc
    echo 'PATH='$TRGT_DIR'/'$y'/'${version[contador]}'/ai_bin:
       $PATH'
    else
    # Variable de Entorno para las demas Apps
    echo 'PATH='$TRGT_DIR'/'$y'/'${version[contador]}'/bin:
       $PATH' >> ~/.bashrc
    echo 'PATH='$TRGT_DIR'/'$y'/'${version[contador]}'/bin:
       $PATH'
    fi

    # Instalacion y variables de entorno extra para DVE
    if (($numapp==0 & $flag==0)); then
    echo "Instalando ${ver_app2[$num_ver]}"
    installer -source "$SRC_DIR/${ver_app2[$num_ver]}" -target
       "$TRGT_DIR"
    echo "Actualizando las variables de entorno de ${ver_app2[
       $num_ver]}..."
    sed -i '/VCS_HOME/d' ~/.bashrc
```

```
    echo 'export VCS_HOME='$TRGT_DIR'/'$y'/'${version[contador
       ]}'' >> ~/.bashrc
    echo 'export VCS_HOME='$TRGT_DIR'/'$y'/'${version[contador
       ]}''
    elif (($numapp==4)); then
    # Variables de entorno extra para IC Validator
    sed -i '/ICV_HOME_DIR/d' ~/.bashrc
    echo 'export ICV_HOME_DIR='$TRGT_DIR'/'$y'/'${version[
       contador]}'' >> ~/.bashrc
    echo 'export ICV_HOME_DIR='$TRGT_DIR'/'$y'/'${version[
       contador]}''
    elif (($numapp==1)); then
    # Variables de entorno extra para IC Validator
    sed -i '/VERDI_HOME/d' ~/.bashrc
    echo 'export VERDI_HOME='$TRGT_DIR'/'$y'/'${version[
       contador]}'' >> ~/.bashrc
    echo 'export VERDI_HOME='$TRGT_DIR'/'$y'/'${version[
       contador]}''
    fi

    let contador++
done
sed -i '/export PATH/d' ~/.bashrc
echo 'export PATH' >> ~/.bashrc
echo 'export PATH'

echo ">> Descarga e Instalacion de ${ver_app[$num_ver]}
   completada. Para mayor informacion sobre la instalacion
   referirse al archivo README o al trabajo de graduacion de
   Paola Mendizabal."
}

# OBJETIVO: Descargar los archivos de la carpeta seleccionada (
   PDK o EDK) y descomprimir los archivos descargados.
descarga_libs() {

cd $LIBS_DIR
pwd

echo "Conectando al servidor de Synopsys ..."
lftp sftp://$SFTP_USER:"$SFTP_PSW"@$SFTP_SERVER << EOF
echo "Conexion exitosa. Descargando archivos de $1."
cd $LIB_EFT_DIR
cd $1
ls
mirror
bye
EOF
```

```
ls *.tar.gz | xargs -n1 tar -xvf
ls *.tar | xargs -n1 tar -xvf

cd "$HERE"
}

# OBJETIVO: Llamar al script INSTALL.sh para actualizar la
   variable de entorno.
set_env_libs() {
if (($option==0)); then

echo "Actualizando variables de entorno de ${tech_PDK[$num_lib
   ]}..."
PDK_DIR_ENV=$(find $PDK_DIR/${tech_PDK[$num_lib]} -iname
   INSTALL.sh)

if [[ -z ${PDK_DIR_ENV// } ]] ; then
        echo "WARNING > No se encontro el archivo INSTALL.sh.
           Se deben actualizar las variables de entorno
           manualmente."
else
        cd ${PDK_DIR_ENV:0:-10}
        bash INSTALL.sh
        read -r line < SOURCEME
        echo "Actualizando bashrc: $line"
        echo ${line:7:6}
        sed -i '/'${line:7:6}'/d' ~/.bashrc
        echo $line >> ~/.bashrc
fi
fi
cd $HERE
}

inst_installer() {
cd "$SRC_DIR/${ver_installer[$num_ver]}"

chmod 755 *.run
./*.run << EOF
$TRGT_DIR/installer
EOF

verif_folder "$TRGT_DIR/installer"

echo "Actualizando variables de entorno.."
if ! grep -q PATH=$TRGT_DIR/installer ~/.bashrc; then
echo 'PATH='$TRGT_DIR'/installer:$PATH' >> ~/.bashrc
fi
```

```
sed -i '/export PATH/d' ~/.bashrc
echo 'export PATH' >> ~/.bashrc

cd $HERE
}

# OBJETIVO: Agregar numero a cada item del array para listarlo
   con whiptail.
addNum() {
new_array=()
counter=0
old_array=("$@")
for x in "${old_array[@]}"; do
        new_array+=(" $counter")
        new_array+=(" ${old_array[$counter]}")
        let counter++
done
}

# OBJETIVO: Mostrar menu de ayuda y salir.
HelpandQuit() {
    echo ""
    echo "Objetivo: "
    echo "Descargar aplicaciones de Synopsys utiles para el
       proyecto Gran Jaguar y los cursos de Nanoelectronica -
       UVG."
    echo "Uso:"
    echo "SNPS_INST [-gui] [-help]"
    echo ""
    echo "    -gui             : Mostrar interfaz grafica."
    echo ""
    echo "Pasos para ejecutarlo por primera vez:"
    echo "    1. Correr el siguiente comando como root : yum
       install lftp"
    echo "    2. Verificar que los siguientes directorios
       existan o modificar las variables en el script :"
    echo "          SRC_DIR=$HOME/Downloads"
    echo "          TRGT_DIR=/usr/synopsys"
    echo "          EDK_DIR=$HOME/Desktop/Folder_de_Trabajo"
    echo "          PDK_DIR=/usr/synopsys/iPDK"
    echo "      IMPORTANTE >> Los nombres de los directorios no
        deben de contener espacios."
    echo "    3. Si no cuenta con el directorio /usr/synopsys
       debe de correr los siguientes comandos como root:"
    echo "              # mkdir /usr/synopsys"
    echo "              # chown <usuario> /usr/synopsys"
```

```
    echo "    4*. Si se desea, cambiar las siguientes variables
       :"
    echo "              NUMB_VERSIONS > Cantidad de versiones a
        desplegar (por default 6)"
    echo "              VERSION_ALL > Numero de version a
       instalar al escoger la opcion Todas las aplicaciones (
       por default 0, la cual representa la version mas
       reciente)."
    echo "    5. Asegurarse de tener acceso a internet."
    echo ""
    echo "*Para mas informacion leer el archivo README.txt o el
        documento de trabajo de graduacion de Paola Mendizabal
       ."
    exit 1
}

# OBJETIVO: Algoritmo de descarga
dwnld_algoritmo() {
dlt_old_dir ${ver_app[$num_ver]}
descarga_apps ${ver_app[$num_ver]}
# -------------- Checksum ----------------------
# Para VCS y DVE
if (($numapp==0 & $flag==0)); then
#    chcksm ${ver_app2[$num_ver]}
echo "verificando checksum de ${ver_app2[$num_ver]}"
fi
# Para las demas apps
#chcksm ${ver_app[$num_ver]}
echo "verificando checksum de ${ver_app[$num_ver]}"
}

# OBJETIVO: Inicializar para algoritmo de librerias.
init_libs() {
    verif_folder "$EDK_DIR"
    verif_folder "$PDK_DIR"

    tech_EDK=($(awk 'match($0, '/EDK/') {print $1}' libsyn.txt
       | sort -r ))
    tech_PDK=($(awk 'match($0, '/PDK/') {print $1}' libsyn.txt
       | sort -r ))

    var_empty "La variable tech_PDK esta vacia. Verifique el
       archivo de texto libsyn.txt" 1 $tech_PDK
    var_empty "La variable tech_EDK esta vacia. Verifique el
       archivo de texto libsyn.txt" 1 $tech_EDK

}
```

```
# OBJETIVO: Inicializar para algoritmo de aplicaciones.
init_apps_installer() {
    verif_folder "$SRC_DIR"
    verif_folder "$TRGT_DIR"
    # ----------- Buscar en la lista las versiones del
       Installer ----------------------------
    ver_installer=($(awk 'match($0, /installer/) {print $1}'
       myproductsrev.txt myproductsrevold.txt myproductsauth.
       txt | sort -r | head -$NUMB_VERSIONS))
    var_empty "No se encuentra installer en el servidor de
       Synopsys" 1 $ver_installer
}

pausa() {
    echo "Pausa. (y para continuar)"
    read resp
    case $resp in
        [Yy]* ) echo "...";;
        * ) exit 0;;
    esac
}

# ++++++++++++++++++++++++++++++++++++++ INICIALIZACION
   ++++++++++++++++++++++++++++++++++++++
if [ "$#" -gt 0 ]; then

copt="$1"
case "$copt" in
        -gui)
        gui=1
;;
        -help)
        HelpandQuit
;;
*)
        echo ""
        echo "ERROR > El argumento no existe."
        exit 1
;;
esac

fi

echo "Ingrese el usuario de SolvnetPlus de Synopsys"
read SFTP_USER
echo "Ingrese la constrasena de SolvnetPlus de Synopsys"
read -s SFTP_PSW
```

```
verif_input $VERSION_ALL
verif_input $NUMB_VERSIONS

verif_file_ls "myproductsauth.txt"
verif_file_ls "myproductsrev.txt"
verif_file_ls "myproductsrevold.txt"
verif_file_ls "libsyn.txt"


# ++++++++++++++++++++++++++++++++++ MENUS GUI
   ++++++++++++++++++++++++++++++++++++++++++

menuPrincipal() {
whiptail --title "Descarga e Instalacion de Software Synopsys -
    UVG" --menu " " 15 60 5 --cancel-button "Salir"\
        "1" "Actualizar versiones" \
        "2" "Synopsys Installer" \
        "3" "Aplicaciones" \
        "4" "PDK" \
        "5" "EDK" \
        3>&2 2>&1 1>&3
}

menu_apps() {
whiptail --title "Aplicaciones - Gran Jaguar" --menu "Choose"
   15 70 8 "${new_array[@]}" --cancel-button "Salir" \
        3>&2 2>&1 1>&3
}

# ++++++++++++++++++++++++++++++++++ INTERFAZ GRAFICA
   ++++++++++++++++++++++++++++++++++++++++++
if (($gui==1)); then

CHOICE=$(menuPrincipal)

while [ 1 ]; do
case $CHOICE in

"1") if (whiptail --title "Actualizar Versiones" --yesno "El
   listado de aplicaciones se actualizo por ultima vez en la
   siguiente fecha > $(tail -1 myproductsrev.txt). Desea volver
    a actualizarlo? " 8 78 --no-button "Regresar" --yes-button
   "Si"); then
        ls_products
        whiptail --msgbox "La lista de aplicaciones se
           actualizo exitosamente. OK, para regresar al Menu
           Principal." 8 78
        fi
```

```
;;
"2") init_apps_installer
addNum "${ver_installer[@]}"
num_ver=$(menu_apps)
if [[ ! -z ${num_ver// } ]] ; then
numapp=1
dlt_old_dir ${ver_installer[$num_ver]}
descarga_apps ${ver_installer[$num_ver]}
chcksm ${ver_installer[$num_ver]}
echo "Iniciando proceso de instalacion para Synopsys Installer
   ."
inst_installer
fi
;;

"3") init_apps_installer
if command -v installer &>/dev/null; then
installer > prueba.txt << EOF
EOF
whiptail --msgbox "Actualmente tiene la siguiente version del
   instalador > $(head -4 prueba.txt)" 10 65

addNum "${apps_formal[@]}"
numapp=$(menu_apps)

if [[ ! -z ${numapp// } ]] ; then

if (($numapp==${#apps_formal[*]})); then
echo "SE DESCARGA E INSTALA TODO"

else
obtener_versiones
addNum "${ver_app[@]}"
desplegar_versiones
num_ver=$(menu_apps)

if [[ ! -z ${num_ver// } ]] ; then
    dwnld_algoritmo
    verif_folder "$SRC_DIR/${ver_app[$num_ver]}"
    sep_cmn ${ver_app[$num_ver]}
    instalacion $name_app $version
    set_env
fi

fi
fi

else
```

```
whiptail --msgbox "Synopsys Installer no se encuentra instalado
    . Sin el no se pueden instalar las aplicaciones." 10 65
fi
;;

"4") init_libs
option=0
LIBS_DIR+=(${PDK_DIR[@]})
addNum "${tech_PDK[@]}"
num_lib=$(menu_apps)
if [[ ! -z ${num_lib// } ]] ; then
descarga_libs ${tech_PDK[$num_lib]}
set_env_libs
fi
;;

"5") init_libs
option=1
LIBS_DIR+=(${EDK_DIR[@]})
addNum "${tech_EDK[@]}"
num_lib=$(menu_apps)
if [[ ! -z ${num_lib// } ]] ; then
descarga_libs ${tech_EDK[$num_lib]}
set_env_libs
fi
;;

*) break ;;

esac

CHOICE=$(menuPrincipal)
done


else
# ++++++++++++++++++++++++++++++++++ INTERFAZ CLI
   ++++++++++++++++++++++++++++++++++++++++
while [ "$option" != "b" ]; do
echo "---------------------- Descarga e Instalacion de Software
    Synopsys -------------------------"
echo "0. PDKs"
echo "1. EDKs"
echo -e "2. Aplicaciones \n"

echo "Seleccione la opcion que desea descargar. Cualquier otra
   tecla para salir."
read option
```

```
verif_input $option
verif_input2 $option 3

#---------------------- DESCARGA E INSTALACION
   ----------------------------------
if (($option==2)); then
#init_apps_installer
# --------- Tiene el Installer? -------------------------
#if command -v installer &>/dev/null; then
#       echo "Actualmente tiene la siguiente version de
   Synopsys Installer >"
#installer > prueba.txt << EOF
#EOF
#head -4 prueba.txt
#       echo "Desea instalar una version diferente?"
#    echo "0. Si"
#    echo "1. No"
#    echo "* Cualquier otra tecla para salir."
#       read yn_inst
#else
#       echo -e "\nSynopsys Installer no se encuentra instalado
   . Sin el no podra instalar otras aplicaciones de Synopsys."
#       no_inst=1
#    echo -e "Desea instalarlo?\n"
#    echo "0. Si"
#    echo "1. No"
#    echo "* Cualquier otra tecla para salir."
#       read yn_inst
#fi

#verif_input $yn_inst
#verif_input2 $yn_inst 2

#if [ "$yn_inst" == "b" ]; then
#    echo "Saliendo de la aplicacion..."
#    echo " pr"
#    exit 0
#fi

#if [[ $no_inst == 1 && $yn_inst == 1 ]]; then
#    echo "Saliendo de la aplicacion..."
#    exit 0
#fi

#if [[ $yn_inst == 0 ]]; then

#display_list "${ver_installer[@]}"
```

```
#echo -e "* Cualquier otra tecla para salir. \n"
#echo "Seleccione la version que desea instalar."
#read num_ver

#verif_input $num_ver
#verif_input2 $num_ver $contador

#if [ "$num_ver" == "b" ]; then
#    exit 0
#fi

#echo "Ha escogido > ${ver_installer[$num_ver]}"
#numapp=1
#dlt_old_dir ${ver_installer[$num_ver]}
#descarga_apps ${ver_installer[$num_ver]}
#chcksm ${ver_installer[$num_ver]}
# ------------ Instalacion --------------
#echo "Iniciando proceso de instalacion para Synopsys Installer
   ."
#inst_installer
#fi

while [ "$numapp" != "b" ]; do
echo -e "\n ------------- APPS > Proyecto Gran Jaguar
   --------------------"
display_list "${apps_formal[@]}"

echo "b - Regresar al menu anterior."
echo -e "* Cualquier otra tecla para salir. \n"

echo "Que aplicacion desea descargar?"
read numapp

# ----- Entradas no deseadas ---------
up_lim=$((contador-1))

verif_input $numapp
verif_input2 $numapp $contador

if (($numapp==$up_lim)); then

for ((x = 0 ; x < ${#apps_formal[*]} ; x++)); do
    echo "Iniciando proceso para ${apps_formal[$x]}..."
    numapp=$x
    num_ver=$VERSION_ALL
    obtener_versiones
    desplegar_versiones
```

```
    if (( $num_ver > ${#ver_app[*]} )); then
        num_ver=${#ver_app[*]}
        echo "Para la app ${apps_formal[$x]} ahora la version
           escogida es $num_ver"
    fi
    dwnld_algoritmo
    verif_folder "$SRC_DIR/${ver_app[$num_ver]}"
    sep_cmn ${ver_app[$num_ver]}
    instalacion $name_app $version
    set_env
done

elif [ "$numapp" != "b" ]; then

while [ "$num_ver" != "b" ]; do

if [ "$num_ver" != "b" ]; then
echo -e "\n Iniciando proceso para ${apps_formal[$numapp]}..."
obtener_versiones
desplegar_versiones
echo -e "b - Regresar al menu anterior. \n"
echo "Seleccione la version que desea descargar. Cualquier otra
    tecla para salir."
read num_ver
verif_input $num_ver
verif_input2 $num_ver $contador

if [ "$num_ver" != "b" ]; then
dwnld_algoritmo
verif_folder "$SRC_DIR/${ver_app[$num_ver]}"
sep_cmn ${ver_app[$num_ver]}
instalacion $name_app $version
set_env
fi

fi

done
num_ver=0

fi

done
numapp=0

# ---------------------------------- LIBRERIAS
   ----------------------------
elif [ "$option" != "b" ]; then
```

```
init_libs

tech_LIBS=()

if (($option==0)); then
    tech_LIBS+=(${tech_PDK[@]})
    LIBS_DIR+=(${PDK_DIR[@]})
else
    tech_LIBS+=(${tech_EDK[@]})
    LIBS_DIR+=(${EDK_DIR[@]})
fi

while [ "$num_lib" != "b" ]; do
echo -e "\n ------------- LIBRERIAS > EDKS y PDKS
   --------------------"
display_list "${tech_LIBS[@]}"
echo $contador "Todas."
echo "b - Regresar al menu anterior."
echo -e "* Cualquier otra tecla para salir. \n"

up_lim=$((contador+1))

echo "Seleccione la libreria que desea descargar."
read num_lib

verif_input $num_lib
verif_input2 $num_lib $up_lim

if [ "$num_lib" != "b" ]; then

if (($num_lib==$contador)); then
for ((x=0; x<${#tech_LIBS[*]}; x++));do
        num_lib=$x
        echo "Descargando ${tech_LIBS[$num_lib]}..."
        descarga_libs ${tech_LIBS[$num_lib]}
        set_env_libs
done
else
echo "Descargando ${tech_LIBS[$num_lib]}..."
descarga_libs ${tech_LIBS[$num_lib]}
set_env_libs
fi

fi

done
num_lib=0
```

```
fi

done
fi
```

18.4.2. Código del *script* Synopsys_Apps_Inst.sh

Cuadro 105. Archivo Synopsys_Apps_Inst.sh

```
#! /bin/bash
# SCRIPT FUNCIONAL PARA LA INSTALACION DE LAS APLICACIONES DE
   SYNOPSYS
# Autor: Carlos Valdes
# Fecha: Noviembre 2025
#-------------------------------------------------------

# --------------------------------| CONFIGURACION | ----
#-------------------------------------------------------
shopt -s extglob
# Variables de usuario
SRC_DIR=$HOME/Downloads
TRGT_DIR=/usr/synopsys
INSTALLER_DIR=/usr/synopsys/installer
SFTP_SERVER=eft.synopsys.com
PRODUCT_DIR=/site31759/MyProducts
NUMB_VERSIONS=6

DIR_AUTH=auth
DIR_REV=rev
DIR_REVO=rev_o

APPS_SCRIPT="./APPS.sh"
ESTADO_FILE="$HOME/Downloads/Installer_Flags/dependencias_state
   "

# Variables de control
HERE=$(pwd)
flag=0
numapp=1  # Para installer, siempre es diferente de 0
re='^[0-9]+$'
# -------------------------------------------------------
# Este codigo es necesario para realizar la conexion ssh con el
    servidor sin verificar la Host Key (NO BORRAR)
export LFTP_HOME="$HOME/.lftp"
mkdir -p "$LFTP_HOME"
```

```
echo "set␣sftp:connect-program␣\"ssh␣-a␣-x␣-o␣
   StrictHostKeyChecking=no␣-o␣UserKnownHostsFile=/dev/null\""
   > "$LFTP_HOME/rc"
# ------------------------------------------------------
#______________________________________________________

# ---------------------------| FUNCIONES | ------------
#______________________________________________________

# ++++++++++++++++++++++++++++++++++++++++ FUNCIONES
   GENERALES++++++++++++++++++++++++++++++
# OBJETIVO: Verificar que la variable no este vacia y sea un
   numero.
verif_input() {
if [[ -z $1 ]] ; then
    echo "Variable␣vacia.␣Adios!"
    exit 0
fi

if [[ ! $1 =~ $re && "$1" != "b" ]] ; then
    echo "La␣variable␣no␣es␣un␣numero.␣Adios!"
    exit 0
fi
}

# OBJETIVO: Verificar que la variable se encuentre dentro del
   rango especificado.
verif_input2() {
if (($1 < 0 || $1 >= $2)); then
echo "Fuera␣de␣rango.␣Adios!␣"
exit 0
fi
}

# OBJETIVO: Enumerar los elementos de un array.
display_list() {
contador=0
arr=("$@")
for j in "${arr[@]}"; do
        echo "$contador␣$j"
        let contador++
done
}

# OBJETIVO: Eliminar las carpetas de descarga que existan de la
    app y version seleccionada
dlt_old_dir() {
```

```
    echo -e "02 > Eliminando carpetas de descarga viejas > $1 \
       n"

    # Validar que el parametro no este vacio
    if [[ -z "$1" ]]; then
        echo "ERROR: La version seleccionada no existe."
        exit 1
    fi

    # Limpiar y crear directorio
    rm -rf "$SRC_DIR/$1"
    mkdir -p "$SRC_DIR/$1"

    # Solo para VCS (numapp==0) necesita manejar DVE tambien
    if [[ $numapp -eq 0 && $flag -eq 0 ]]; then
        if [[ -n "${ver_app2[$num_ver]}" ]]; then
            rm -rf "$SRC_DIR/${ver_app2[$num_ver]}"
            mkdir -p "$SRC_DIR/${ver_app2[$num_ver]}"
        fi
    fi

    # Obtener nombre de directorio en el servidor
    dir_inst=($(awk -v ver="$1" '$1 == ver {print $2}'
       myproductsrev.txt myproductsrevold.txt myproductsauth.
       txt))

    if [[ -z "$dir_inst" ]]; then
        echo "ERROR: No se encontro el directorio en el que se
           encuentra la aplicacion seleccionada."
        exit 1
    fi

    echo "Directorio en servidor: $PRODUCT_DIR/$dir_inst/$1"
}

# OBJETIVO: Descarga de archivos de la app y version
   seleccionada.
descarga_apps() {
    local version="$1"
    local target_dir="$SRC_DIR/$version"

    # Crear directorio si no existe
    mkdir -p "$target_dir"

    if [[ $numapp -eq 0 && $flag -eq 0 ]]; then
        # Caso especial para VCS y DVE
        echo "Conectando al servidor de Synopsys ..."
        lftp sftp://$SFTP_USER:"$SFTP_PSW"@$SFTP_SERVER << EOF
```

```
cd $PRODUCT_DIR/$dir_inst
cd ${ver_app[$num_ver]}
lcd $SRC_DIR/${ver_app[$num_ver]}
mget *
cd ..
cd ${ver_app2[$num_ver]}
lcd $SRC_DIR/${ver_app2[$num_ver]}
mget *
bye
EOF
    else
        # Caso normal para el resto de aplicaciones
        echo "Conectando al servidor de Synopsys ..."
        lftp sftp://$SFTP_USER:"$SFTP_PSW"@$SFTP_SERVER << EOF
cd $PRODUCT_DIR/$dir_inst
cd $version
lcd $target_dir
mget *
bye
EOF
    fi

    echo ""
    echo "Descarga completada en: $target_dir"
}

# OBJETIVO: Verificar el checksum de los archivos descargados
chcksm() {
    echo -e "04 > Verificando checksum de $1 ... \n"

    local check_dir="$SRC_DIR/$1"

    # Verificar que el directorio exista
    if [ ! -d "$check_dir" ]; then
        echo "ERROR: El directorio $check_dir no existe."
        exit 1
    fi

    cd "$check_dir"

    # Verificar que exista checksum_info.txt
    if [ ! -f checksum_info.txt ]; then
        echo "ERROR: El archivo checksum_info.txt no existe en
           $check_dir"
        echo "Archivos disponibles:"
        ls -la
        exit 1
    fi
```

```
    # Crear archivo de checksum real
    > checksum_real.txt
    sed -n '1,2p' checksum_info.txt >> checksum_real.txt
    cksum !(installer.log|*checksum*) >> checksum_real.txt

    # Comparar checksums
    if [[ $(diff -w checksum_info.txt checksum_real.txt) = ""
       ]]; then
        echo "Checksum correcto. Descarga de $1 exitosa."
    else
        echo "ERROR: Los archivos de $1 no se han descargado
           correctamente."
        echo "Diferencias encontradas:"
        diff -w checksum_info.txt checksum_real.txt
        exit 1
    fi

    # Descomprimir archivos para la instalacion
    if ls *.gz &>/dev/null; then
        echo "Descomprimiendo archivos .gz..."
        gzip -d *.gz
    fi

    cd "$HERE"
}

# OBJETIVO: Obtener el listado de productos en el servidor de
   Synopsys
ls_products() {
echo "Conectando al servidor de Synopsys para actualizar
   listado..."
lftp sftp://$SFTP_USER:"$SFTP_PSW"@$SFTP_SERVER << EOF
cd $PRODUCT_DIR/$DIR_REV
ls > myproductsrev.txt
cd ..
cd $PRODUCT_DIR/$DIR_AUTH
ls > myproductsauth.txt
cd ..
cd $PRODUCT_DIR/$DIR_REVO
ls > myproductsrevold.txt
bye
EOF

awk '{print $9, "rev"}' myproductsrev.txt > tmp_file && mv
   tmp_file myproductsrev.txt
awk '{print $9, "rev_o"}' myproductsrevold.txt > tmp_file && mv
    tmp_file myproductsrevold.txt
```

```
awk '{print $9, "auth"}' myproductsauth.txt > tmp_file && mv
   tmp_file myproductsauth.txt
date >> myproductsrev.txt

echo "Listado␣actualizado␣exitosamente."
}

# OBJETIVO: Verificar si la variable esta vacia.
var_empty() {
    var=$3
    if [[ -z ${var// } ]] ; then
        echo "ERROR␣>␣$1."
        if (($2==1)) ; then
            exit 0
        else
            flag=1
        fi
    fi
}
# ++++++++++++++++++++++++++++++++++++++++
# ++++++++++++++++++++++++++++++++++++++++++++Funcion
   Dependencias Iniciales ++++++++++++

Dependencias_Iniciales() {
# Detectar el usuario actual
current_user=$USER

# Detectar el grupo principal del usuario actual
current_group=$(id -gn $USER)

ESTADO_FILE="/$HOME/Downloads/Installer_Flags/
   dependencias_state"

echo "Ejecutando␣con␣el␣usuario:␣$current_user␣y␣grupo:␣
   $current_group"

# ----------------------------------------AGREGAR SERVIDOR DE
   LICENCIAS A .BASHRC--------

echo -e "Agregrando␣servidor␣de␣licencias␣al␣BASHRC..."
# Ruta al archivo .bashrc
BASHRC_FILE="$HOME/.bashrc"

# Servidor de licencias UVG
IP="192.168.74.1"  # IP Del servidor
LINE_SERVER_LICENSE="LM_LICENSE_FILE=\"27020@$IP\";␣export␣
   LM_LICENSE_FILE"
```

```
# Verificar si la linea de servidor de licencias ya existe en
   el archivo .bashrc
if grep -q "LM_LICENSE_FILE" "$BASHRC_FILE"; then
    # Si existe, reemplazar la linea con la nueva IP
    sed -i "s|LM_LICENSE_FILE=\"[^\"]*\";␣export␣
       LM_LICENSE_FILE|$LINE_SERVER_LICENSE|" "$BASHRC_FILE"
    echo -e "La␣linea␣del␣servidor␣UVG␣ha␣sido␣actualizada␣en␣
       el␣archivo␣.bashrc\n"
else
    # Si no existe, agregarla antes de la linea "export␣PATH"
    sed -i "/export␣PATH/i␣$LINE_SERVER_LICENSE" "$BASHRC_FILE"
    echo -e "Linea␣del␣servidor␣UVG␣agregada␣al␣archivo␣.bashrc
       \n"
fi

# Verificar si la linea del PATH INSTALLER ya existe en el
   archivo .bashrc
LINE_INSTALLER='PATH=/usr/synopsys/installer:$PATH'
if ! grep -q "$LINE_INSTALLER" "$BASHRC_FILE"; then
    # Si no existe, agregarla antes de la linea "export␣PATH"
    sed -i "/^export␣PATH/i␣$LINE_INSTALLER" "$BASHRC_FILE"
    echo -e "Linea␣del␣PATH␣INSTALLER␣agregada␣al␣archivo␣.
       bashrc␣\n"
else
    echo -e "La␣linea␣del␣PATH␣INSTALLER␣ya␣existe␣en␣el␣
       archivo␣.bashrc␣\n"
fi

# ----------------------------------------CREACION DE CARPETAS
   DE ESPACIO DE TRABAJO-----------------

# Crear la carpeta 'Folder_de_Trabajo' en HOME/Desktop
#echo "Creando␣carpeta␣de␣Folder_de_Trabajo␣en␣la␣ruta␣/Desktop
   /Folder_de_Trabajo"
mkdir -p ~/Desktop/Folder_de_Trabajo

# Configurar permisos para ~/Desktop/Folder_de_Trabajo
chmod 775 ~/Desktop/Folder_de_Trabajo

mkdir -p ~/Downloads/Installer_Flags

# Configurar permisos para ~/Desktop/Folder_de_Trabajo
chmod 775 ~/Downloads/Installer_Flags


# ----------------------------------------CREACION DE
   CARPETAS CON PERMISO DE ROOT-------------
# Crear la carpeta 'synopsys' en /usr
```

```
echo -e "Creando␣directorio␣/usr/synopsys...\n"
sudo mkdir -p /usr/synopsys

# Crear la carpeta 'iPDK' dentro de 'synopsys'
echo "Creando␣carpeta␣de␣iPDK␣en␣la␣ruta␣/usr/synopsys/iPDK"
sudo mkdir -p /usr/synopsys/iPDK

sudo mkdir -p /usr/synopsys/installer

# Cambiar la propiedad de /usr/synopsys al usuario y grupo
   actual
sudo chown -R $current_user:$current_group /usr/synopsys/

# Configurar permisos para /usr/synopsys
sudo chmod -R 775 /usr/synopsys/

echo -e "Instalando␣aplicaciones␣y␣librerias..."
sudo dnf update -y
sudo dnf install -y lftp
sudo dnf install -y libXScrnSaver #installer
sudo dnf install -y libnsl #waveview
sudo dnf install -y tcsh #cdesigner
#sudo dnf install -y rsh

touch "$ESTADO_FILE"
}
# ------------------------------------------------
# ++++++++++++++++++++++++++++++++++++++++++++++++

# ++++ DESCARGA E INSTALACION DE "SYNOPSYS␣INSTALLER" ++++++++
Install_Synopsys_Installer() {

echo "␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣VERIFICACION␣DE␣SYNOPSYS␣INSTALLER"
echo "============================================================"

# Verificar si 'installer' esta disponible en el PATH
if command -v installer &>/dev/null; then
    echo ">>>␣Synopsys␣Installer␣ya␣esta␣instalado␣en␣el␣
       sistema."
    echo ""

    # Mostrar version actual si es posible
    echo "Version␣actual␣detectada:"
    if command -v installer &>/dev/null; then
        echo "Actualmente␣tiene␣la␣siguiente␣version␣de␣
           Synopsys␣Installer:"
installer > prueba.txt << EOF
EOF
```

```
head -4 prueba.txt
    else
        echo "No se pudo determinar la version actual."
    fi
    echo ""

    # Preguntar si desea instalar otra version
    echo "Desea instalar una version diferente?"
    echo "0) Si"
    echo "1) No"
    echo "b) Volver al menu principal"
    echo "* Cualquier otra tecla para salir"
    read -r yn_inst

else
    echo ""
    echo ">>> Synopsys Installer no esta instalado."
    echo "Sin el, no podra instalar otras aplicaciones de
       Synopsys."
    echo ""
    no_inst=1
    echo "Desea instalarlo?"
    echo "0) Si"
    echo "1) No"
    echo "b) Volver al menu principal"
    echo "* Cualquier otra tecla para salir"
    read -r yn_inst
fi

# Validacion de entrada
verif_input "$yn_inst"

# Salir si el usuario lo solicita
if [ "$yn_inst" == "b" ]; then
    echo "Volviendo al menu principal..."
    return 0
fi

# Si ya tiene el installer y elige 'No', salir
if [[ $no_inst != 1 && $yn_inst == 1 ]]; then
    echo "Volviendo al menu principal..."
    return 0
fi

# Si elige instalar o reinstalar
if [[ $yn_inst == 0 ]]; then
    echo ""
```

```
    echo "     LISTADO DE VERSIONES DE SYNOPSYS INSTALLER
       DISPONIBLES"
    echo "
       ============================================================
       "

    # Verificar que existan los archivos de productos
    if [ ! -f "myproductsrev.txt" ] || [ ! -f "myproductsrevold
       .txt" ] || [ ! -f "myproductsauth.txt" ]; then
        echo "ERROR: No se encontraron los archivos de
           productos."
        echo "Actualizando el listado desde el servidor..."

        # Solicitar credenciales si no estan definidas
        if [[ -z "$SFTP_USER" ]] || [[ -z "$SFTP_PSW" ]]; then
            echo "Ingrese el usuario de SolvnetPlus de Synopsys
               :"
            read SFTP_USER
            echo "Ingrese la contrasena de SolvnetPlus de
               Synopsys:"
            read -s SFTP_PSW
            echo ""
        fi

        # Actualizar listado de productos
        ls_products
    fi

    # Buscar versiones disponibles del installer
    ver_installer=($(awk 'match($0, /installer/) {print $1}'
       myproductsrev.txt myproductsrevold.txt myproductsauth.
       txt 2>/dev/null | sort -r | head -$NUMB_VERSIONS))

    if [ ${#ver_installer[@]} -eq 0 ]; then
        echo "ERROR: No se encontraron versiones del installer
           en el servidor."
        echo "Volviendo al menu principal..."
        return 1
    fi

    # Mostrar lista de versiones disponibles
    display_list "${ver_installer[@]}"
    echo "b) Volver al menu principal"
    echo "* Cualquier otra tecla para salir"
    echo ""
    echo "Seleccione la version que desea instalar:"
    read -r num_ver
```

```
    verif_input "$num_ver"

    if [ "$num_ver" == "b" ]; then
        echo "Volviendo al menu principal..."
        return 0
    fi

    verif_input2 "$num_ver" "${#ver_installer[@]}"

    echo ""
    echo "======================================================"
    echo "Ha escogido: ${ver_installer[$num_ver]}"
    echo "======================================================"
    numapp=1

    # Solicitar credenciales si aun no estan definidas
    if [[ -z "$SFTP_USER" ]] || [[ -z "$SFTP_PSW" ]]; then
        echo ""
        echo "Ingrese el usuario de SolvnetPlus de Synopsys:"
        read SFTP_USER
        echo "Ingrese la contrasena de SolvnetPlus de Synopsys:
           "
        read -s SFTP_PSW
        echo ""
    fi

    # Limpiar directorios previos
    echo ""
    echo "01 > Preparando directorio de descarga..."
    dlt_old_dir "${ver_installer[$num_ver]}"

    # Descargar archivos
    echo ""
    echo "03 > Descargando archivos desde el servidor..."
    descarga_apps "${ver_installer[$num_ver]}"

    # Verificar checksum
    echo ""
    chcksm "${ver_installer[$num_ver]}"

    # Buscar el archivo .run en el directorio correcto
    echo ""
    echo "05 > Buscando instalador..."
    INSTALLER_FILE=$(find "$SRC_DIR/${ver_installer[$num_ver]}"
        -type f -name "SynopsysInstaller_*.run" 2>/dev/null |
       sort -V | tail -n 1)

    if [ -z "$INSTALLER_FILE" ]; then
```

```
        echo "ERROR: No se encontro el archivo instalador .run"
        echo "Ubicacion buscada: $SRC_DIR/${ver_installer[
            $num_ver]}"
        echo "Volviendo al menu principal..."
        return 1
    fi

    echo "Instalador encontrado: $INSTALLER_FILE"

    # Dar permisos de ejecucion
    echo ""
    echo "06 > Configurando permisos..."
    chmod +x "$INSTALLER_FILE"

    # Crear directorio de instalacion
    echo ""
    echo "07 > Creando directorio de instalacion..."
    sudo mkdir -p "$INSTALLER_DIR"

    # Ejecutar instalador
    echo ""
    echo "08 > Ejecutando instalador de Synopsys..."
    echo "Esto puede tomar varios minutos..."
    sudo "$INSTALLER_FILE" -dir "$INSTALLER_DIR"

    # Verificar que la instalacion fue exitosa
    if [ ! -f "$INSTALLER_DIR/installer" ]; then
        echo ""
        echo "ERROR: La instalacion no se completo
            correctamente."
        echo "No se encontro el ejecutable en $INSTALLER_DIR/
            installer"
        return 1
    fi

    # Configurar permisos
    echo ""
    echo "09 > Configurando permisos finales..."
    sudo chown -R $USER:$USER "$INSTALLER_DIR"
    sudo chmod -R +x "$INSTALLER_DIR"

    # Actualizar PATH en .bashrc si es necesario
    echo ""
    echo "10 > Actualizando variables de entorno..."
    BASHRC_FILE="$HOME/.bashrc"
    LINE_INSTALLER="PATH=$INSTALLER_DIR:\$PATH"

    if ! grep -q "PATH=$INSTALLER_DIR" "$BASHRC_FILE"; then
```

```
        sed -i "/^export␣PATH/i␣$LINE_INSTALLER" "$BASHRC_FILE"
        echo "Variable␣PATH␣actualizada␣en␣.bashrc"
    else
        echo "Variable␣PATH␣ya␣estaba␣configurada"
    fi

    echo ""
    echo "=====================================================
␣␣␣␣echo␣">>> Synopsys Installer instalado con exito en
   $INSTALLER_DIR"
␣␣␣␣echo␣"=====================================================
    echo ""
    echo "NOTA:␣Debera␣reiniciar␣la␣terminal␣o␣ejecutar␣'source
       ␣~/.bashrc'"
    echo "␣␣␣␣␣␣para␣que␣los␣cambios␣surtan␣efecto."
    echo ""

    return 0
fi
}

# ++++++++++++++++++++++++++++++++++++++++++++++++

#  +++++++++ DESCARGA E INSTALACION DE APPS ++++++++++++
Install_Synopsys_Apps() {
    if [ ! -x "$INSTALLER_DIR/installer" ]; then
        echo "ERROR:␣Synopsys␣Installer␣not␣found.␣Please␣run␣
           option␣2␣first."
        exit 1
    fi

    # Verificar si ya tiene permisos de escritura
    if [ -w "$TRGT_DIR" ]; then
        echo "Permisos␣correctos.␣Continuando␣con␣la␣
           instalacion..."
    else
        echo "Se␣requieren␣permisos␣de␣administrador."
        echo "Ingrese␣su␣contrasena␣sudo"

        # Configurar permisos (pide contrasena una sola vez)
        if sudo chown -R $USER:$USER "$TRGT_DIR" && sudo chmod
           -R 775 "$TRGT_DIR"; then
            echo "Permisos␣configurados␣correctamente."
        else
            echo "ERROR:␣No␣se␣pudieron␣configurar␣los␣permisos
               ."
            return 1
        fi
```

```
    fi

    bash "$APPS_SCRIPT"
    echo ">>> Application installation completed."
}
# ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

#------------------------------------------------------------

# ------------------------------| MAIN PRINCIPAL |
   ---------------
#------------------------------------------------------------

while true; do
    clear
    echo "
       ============================================================"
    echo "         SYNOPSYS APPLICATION INSTALLER - UVG"
    echo "
       ============================================================
       "
    echo "1) Configurar Dependencias Iniciales"
    echo "2) Descarga e instalacion de SYNOPSYS INSTALLER"
    echo "3) Descarga e instalacion de aplicaciones de Synopsys
       "
    echo "4) Salir"
    echo "
       ============================================================"
    read -p "Seleccionar una opcion [1-4]: " option
    echo ""

    case $option in
# ------------------------------------------------------
        1)
        if [ ! -f "$ESTADO_FILE" ]; then
            echo "Ejecutando Funcion de Dependencias_Iniciales
               ..."
            echo " "
            Dependencias_Iniciales

            # Verificar si dependencias_iniciales.sh se ejecuto
                correctamente
            if [ $? -eq 0 ]; then

                echo "Dependencias_Iniciales se ejecuto
                   correctamente."
                echo ""
            else
```

```
                echo "Error al ejecutar Dependencias_Iniciales.
                   Saliendo..."
                exit 1
            fi

        else
            echo "Las dependencias ya se han instalado
               anteriormente."
            #secho " "
        fi
        ;;
# ------------------------------------------------------
        2) Install_Synopsys_Installer ;;
# ------------------------------------------------------
        3)
        # Verificar si ya se instalo el SYNOPSYS INSTALLER
        if [ -z "$(ls -A /usr/synopsys/installer)" ]; then
                echo "ADVERTENCIA! -> No se puede instalar las
                   aplicaciones de Synopsys."
                echo "Es necesario seleccionar la opcion 2 para
                   instalar el SYNOPSYS INSTALLER."
                echo "Ejecute la opcion 2 y vuelva a intentar."

        else
                echo "Instalador ya disponible, puede descargar
                   la app que desee."
                Install_Synopsys_Apps
        fi
        ;;
# ------------------------------------------------------
        4) echo "Saliendo..."; exit 0 ;;
# ------------------------------------------------------
        *) echo "Opcion invalida. Intentar de nuevo." ;;
    esac
    echo ""
    read -p "Presionar ENTER para volver al menu..."
done
```

CAPÍTULO 19

Glosario

***Back-End*:** etapa del flujo de diseño también conocida como síntesis física. Incluye el *floor-planning*, el *placement and routing* (PnR) y la generación de archivos para fabricación.

***Blackbox*:** caja negra que representa una celda, de la cual se conoce únicamente su funcionamiento lógico, más se desconoce su funcionamiento interno.

***Front-End*:** etapa del flujo de diseño también conocida como síntesis lógica. Incluye la descripción RTL, la síntesis para obtener el *gate-level netlist* y la verificación funcional (simulaciones y verificaciones formales).

Gate-level netlist: representación del circuito como una lista de conexiones entre celdas estándar de una tecnología específica.

***Ring Oscillator*:** circuito compuesto por una cadena impar de inversores que produce una señal periódica debido a su realimentación.

***Script*:** archivo de texto que contiene una serie de comandos para automatizar tareas en un entorno de *software*.

ASIC *(Application-Specific Integrated Circuit)*: circuito integrado diseñado para una aplicación o tarea específica.

Bash *(Bourne Again Shell)*: intérprete de comandos y lenguaje de *scripting* utilizado en sistemas Unix/Linux.

Celda estándar: bloque lógico predefinido en una biblioteca tecnológica, como compuertas lógicas, *flip-flops*, *buffers*, etc.

CMOS *(Complementary Metal-Oxide-Semiconductor)*: tecnología utilizada para construir circuitos integrados que utiliza transistores complementarios (NMOS y PMOS) para diseños con baja potencia de consumo.

Design Compiler: herramienta de Synopsys utilizada para la síntesis lógica de diseños digitales a partir de código RTL.

Digilent Genesys plataforma de desarrollo FPGA basada en la lógica programable de Xilinx. Es utilizada para prototipado y pruebas de hardware.

EDA *(Electronic Design Automation)*: herramientas de *software* usadas para diseñar, simular y sintetizar circuitos integrados.

Emulación en FPGA (*FPGA Prototyping*): validación funcional de un diseño digital mediante su implementación en un FPGA.

Flujo de diseño: conjunto ordenado de etapas utilizadas para diseñar al completo un circuito integrado, desde su descripción RTL hasta su la generación de archivos para fabricación.

FPGA *(Field Programmable Gate Array)*: dispositivo lógico programable utilizado para validar prototipos y realizar pruebas de hardware.

HDL *(Hardware Description Language)*: lenguaje utilizado para describir circuitos digitales, como Verilog o VHDL.

Librería tecnológica: conjunto de celdas estándar y modelos proporcionados por un fabricante para una tecnología CMOS específica.

PVT *(Process, Voltage, Temperature)*: condiciones bajo las cuales se caracterizan las celdas estándar, considerando variaciones en el proceso de fabricación, el voltaje y la temperatura de operación.

RTL *(Register Transfer Level)*: nivel de abstracción en el que se describe el flujo de datos entre registros y las operaciones lógicas aplicadas.

SSH *(Secure Shell)*: protocolo de red que permite la comunicación segura y encriptada entre dos dispositivos a través de una red, utilizado comúnmente para acceder a servidores de forma remota.

Synopsys: empresa líder en herramientas EDA utilizadas para el diseño de circuitos integrados.

TSMC *(Taiwan Semiconductor Manufacturing Company)*: compañía líder en fabricación de semiconductores y tecnología CMOS.

Verilog: lenguaje de descripción de hardware utilizado para modelar, simular y sintetizar circuitos digitales.

VLSI *(Very Large Scale Integration)*: se refiere a la integración de millones de transistores en un único circuito integrado.