

UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería



Osciloscopio USB Multicanal

Módulo Software

Trabajo de graduación en modalidad de tesis presentado por
Aaron Fernando Mazariegos Quezada
para optar al grado académico de Licenciado en Ingeniería Electrónica

Guatemala
2014

UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería




Osciloscopio USB Multicanal

Módulo
Software

Trabajo de graduación en modalidad de tesis presentado por
Aaron Fernando Mazariegos Quezada
para optar al grado académico de Licenciado en Ingeniería Electrónica

Guatemala
2014

Vo. Bo. :

(f)  _____

MSc. Carlos Alberto Esquit Hernández

Tribunal Examinador:

(f)  _____

MSc. Carlos Esquit

(f)  _____

MSc. Roberto Saravia

(f)  _____

Ing. Martín Guzmán

Fecha de aprobación:

Guatemala 24 de junio de 2014

PREFACIO

El desarrollo de esta tesis ha sido una experiencia enriquecedora tanto en el ámbito académico como en el profesional. A lo largo del proceso, se abordaron desafíos técnicos y conceptuales que permitieron fortalecer conocimientos en el diseño de hardware, procesamiento de señales y desarrollo de software para sistemas de medición electrónica.

Este trabajo representa el resultado de meses de investigación, pruebas y optimización, con el objetivo de aportar una herramienta útil para el análisis de señales digitales y mixtas. La integración de un osciloscopio USB multicanal con un módulo de software de visualización abre oportunidades para futuras mejoras y aplicaciones en diversos ámbitos de la ingeniería electrónica.

Quiero expresar mi gratitud a la Universidad del Valle de Guatemala, a mis profesores y asesores, cuyo apoyo y orientación fueron fundamentales para el desarrollo de este proyecto. Asimismo, agradezco a mi madre y a mi padre por su motivación y apoyo incondicional en cada etapa del proceso.

Espero que este trabajo sirva como base para futuras investigaciones y proyectos en el área de instrumentación electrónica, y que pueda contribuir al crecimiento del conocimiento en la comunidad académica y profesional.

ÍNDICE

	Página
Prefacio	ii
Lista de cuadros	v
Lista de figuras.....	vi
Resumen	viii
I. Introducción.....	1
II. Objetivos	2
A. Objetivo general módulo.....	2
B. Objetivos específicos módulo	2
III. Justificación	3
IV. Marco teórico	4
A. Señales eléctricas.....	4
1. Clasificación de señales	5
2. Integridad de señal	11
B. Ondas.....	12
1. Frecuencia y período.....	13
2. Voltaje	13
3. Amplitud	13
4. Fase	14
C. Osciloscopio	14
1. Tipos de osciloscopio.....	14
2. Especificaciones de desempeño	15
D. Processing.....	22
V. Antecedentes	23
A. Del trabajo de graduación.....	23
B. Del módulo	26
VI. Diseño experimental	29
A. Investigación teórica.....	29

B.	Selección de hardware.....	30
C.	Selección de software.....	31
D.	Descripción de módulos.....	32
E.	Diseño hardware.....	34
F.	Diseño software.....	40
VII.	Discusión y resultados.....	45
VIII.	Conclusiones.....	56
IX.	Recomendaciones.....	57
X.	Bibliografía.....	58
XI.	Glosario.....	61

LISTA DE CUADROS

Página

1. Osciloscopios digitales ofrecidos por Tektronix.....	24
2. Tareas específicas por módulo	33
3. Tiempo visible por frecuencia de muestreo	37
4. Modos de operación del osciloscopio	41
5. Modos de operación del software	41

LISTA DE FIGURAS

	Página
6. Fuentes comunes de señales eléctricas. [18].....	4
7. Continuidad de señales (Izq. – Señal continua Der. – Señal discreta).....	5
8. Señales Analógica y Discreta (Izq. – Señal analógica Der. – Señal digital).....	6
9. Periodicidad de señales (Izq. – Señal periódica Der. – Señal aperiódica).....	7
10. Causalidad de señales (Izq. – Señal causal Der. – Señal anti-causal).....	8
11. Paridad de señales (Izq. – Señal par Der. – Señal impar).....	8
12. Señales determinísticas (Izq. – Señal aleatoria Der. – Señal determinística).....	9
13. Hemisferio de una Señal (Izq. – Hemisferio derecho Der. – Hemisferio izquierdo).....	10
14. Tamaño de señal (Izq. – Señal finita Der. – Señal infinita).....	10
15. Integridad de señal. [16].....	11
16. Onda en una cuerda, onda sinusoidal y onda de compresión.....	12
17. Tipos de osciloscopios (Analógico y Digital) [18].....	15
18. Ancho de banda [18].....	16
19. Frecuencia de muestreo [19].....	18
20. Tiempo de subida [19].....	19
21. Señales Mixtas [19].....	21
22. Osciloscopios desarrollados con Arduino [10] [5].....	25
23. DAQ32 desarrollado por X-io Technologies [7].....	26
24. Picoscope Software [4].....	26
25. OSC-illoscope [7].....	27
26. Análisis lógico por xoscillo [20].....	28
27. Diseño completo del Osciloscopio USB multicanal [11].....	34
28. Diseño Nanoboard [11].....	35
29. Diseño interno FPGA [11].....	35
30. Diagrama de flujo desarrollo de software.....	40
31. Diagrama de flujo ciclo principal.....	42
32. Diagrama de flujo módulo interpretador de control y datos.....	43
33. Diagrama de flujo módulo de muestras digitales.....	43
34. Diagrama de flujo módulo de muestras mixtas.....	44
35. Diagrama de flujo módulo de muestras analógicas.....	44
36. Modo inicial del software.....	45
37. Modo digital del software.....	46
38. Modo digital con grosor 1px.....	47
39. Modo digital con grosor 50px.....	47
40. Selector de grosor de dato.....	48
41. Modo digital con resolución 1024x640px.....	49

42. Modo digital con resolución 1440x900px	49
43. Posición inicio de despliegue.....	50
44. Posición inicio de despliegue.....	50
45. Modo analógico	51
46. Modo analógico 4 canales	52
47. Modo mixto	53
48. Preferencias modo mixto	54
49. Comparación. modo mixto	54

RESUMEN

Este trabajo de graduación consiste en diseñar y construir un *Osciloscopio USB Multicanal*, el cual se conecta a un computador por medio de USB para presentar los datos en pantalla. El osciloscopio es una herramienta con la cual se representan señales eléctricas de manera gráfica. Este instrumento de visualización es útil para el diseño, fabricación, mantenimiento y/o reparación de equipo electrónico.

La adquisición de datos se realiza por medio de un FPGA (Field Programmable Gate Array), el cual sirve como interfaz entre las señales de interés y el computador. Este dispositivo es un circuito integrado que puede ser configurado por el diseñador para describir comportamiento de hardware. El FPGA puede trabajar con altas tasas de muestreo y permite comunicación con el computador por medio USB.

El módulo de software se enfoca en la representación gráfica de las muestras de las señales. Se diseña un programa para realizar el despliegue de datos utilizando Processing, un lenguaje de programación con licencia libre de uso.

Con esto se obtiene un programa libre de plataforma en el cual se pueden visualizar las señales digitales desde 1 hasta 32 canales simultáneamente de una manera ordenada y amigable.

I. INTRODUCCIÓN

Este trabajo de graduación establece los métodos de diseño y desarrollo utilizados en la elaboración de un *Osciloscopio USB multicanal*, con el cual se pueden visualizar señales mixtas y señales digitales de alta frecuencia utilizando un computador. Actualmente, existen varias opciones para visualizar este tipo de señales, como los osciloscopios analógicos, digitales y digitizadores. Sin embargo, suelen presentar limitaciones en la cantidad de canales visibles, la capacidad de muestreo en señales de alta frecuencia (>10 MHz) y el costo.

El osciloscopio es un dispositivo vital para estudiantes universitarios que desean estudiar el comportamiento de señales de manera visual; lo cual es una situación común en laboratorios de electrónica. Este proyecto busca aportar un dispositivo portátil y de bajo costo capaz de capturar hasta 32 canales de datos digitales de alta frecuencia simultáneamente, desplegándolos en una computadora por medio del software de aplicación.

Para el desarrollo de este proyecto se realizó una investigación general acerca del funcionamiento y características del osciloscopio, junto con recopilación de información de proyectos similares previamente diseñados y producidos. Se utilizó el laboratorio y equipo del departamento de Electrónica de la Universidad del Valle de Guatemala para el desarrollo de hardware y para realizar pruebas de integración de módulos. El software fue desarrollado en una computadora personal utilizando el lenguaje Java en la plataforma de Processing, creando una aplicación libre de licencia, portátil y universal entre sistemas operativos.

II. OBJETIVOS

El objetivo principal de este proyecto es desarrollar el hardware y software necesarios para aportar un osciloscopio que permita realizar mediciones simultáneas de hasta 32 canales de datos digitales de alta frecuencia, con comunicación USB hacia una computadora para el despliegue de datos. El proyecto consta de dos módulos, uno de hardware el cual se encarga de comunicar la información obtenida de las señales eléctricas con la computadora, y un módulo de software el cual despliega estos datos al usuario.

A. OBJETIVO GENERAL MÓDULO

- Diseñar e implementar un software de aplicación para el despliegue de señales digitales en una computadora obteniendo los datos del módulo de hardware del proyecto.

B. OBJETIVOS ESPECÍFICOS MÓDULO

- Crear una aplicación que grafique hasta 32 canales de señales digitales simultáneamente.
- Crear una aplicación que grafique hasta 4 canales analógicos simultáneamente.
- Diseñar una interfaz gráfica de usuario ordenada y amigable.
- Crear archivos de registro con los datos recibidos.

III. JUSTIFICACIÓN

El osciloscopio digital es un instrumento de medición electrónica, con el cual se puede representar una señal eléctrica de manera gráfica. Esto permite medir con exactitud y precisión el momento en el cual ocurren ciertos eventos; lo cual es útil para el diseño, fabricación, mantenimiento y/o reparación de equipos electrónicos.

Este trabajo de graduación busca aportar un dispositivo de medición confiable que sirva como herramienta a ingenieros que desean analizar proyectos electrónicos en los cuales se manejen altas frecuencias y sea de interés el poder tener varios canales simultáneamente. El uso de varios canales simultáneos da la capacidad de analizar los tiempos y razones de disparo de los eventos, lo cual es sumamente importante para el control y análisis de sistemas y/o aplicaciones que manejen altas frecuencias de operación.

Se utiliza el Nanoboard 3000 como prototipo para un dispositivo con necesidad de la detección de señales que cambian rápidamente. El FPGA posee esta ventaja sobre microcontroladores convencionales.

Processing se utiliza como lenguaje de programación para el módulo de software ya que permite exportar la aplicación a computadoras con diferentes sistemas operativos haciéndolo un instrumento portátil, práctico y de bajo costo.

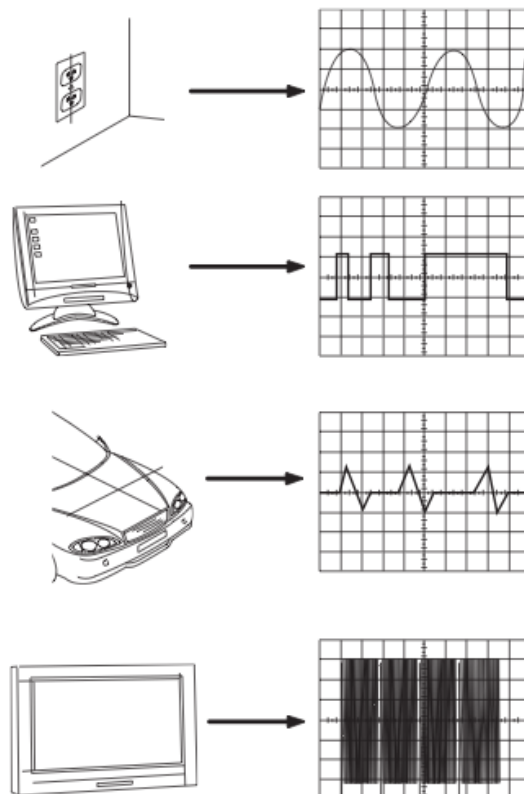
IV. MARCO TEÓRICO

A. SEÑALES ELÉCTRICAS

Una señal eléctrica es una función que contiene información acerca del comportamiento de un sistema o atributos de eventos físicos. (Priemer, 1991:1)

Las señales eléctricas pueden ocurrir naturalmente o pueden ser sintetizadas. Dado que no se pueden obtener señales eléctricas naturales de todos los fenómenos físicos, se utilizan transductores para transformar señales de un tipo de energía particular a energía eléctrica. Todas las señales eléctricas consideradas a continuación se tomarán en función de la variable independiente tiempo.

Figura 1. Fuentes comunes de señales eléctricas. [18].



Nota. Tektronix, 2008:4.

En la Figura 1 se muestran fuentes comunes de señales eléctricas, entre las cuales se encuentran las señales a/c, datos binarios de un computador, señales de sistema vehicular y una señal compleja de video, respectivamente.

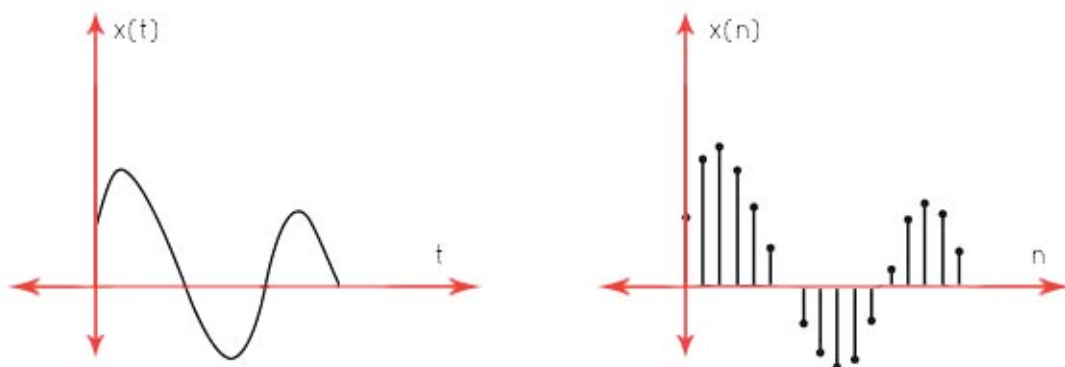
1. Clasificación de señales

Para comprender la forma en la que se comportan las señales se deben conocer las diferentes clasificaciones que estas pueden tener.

a. Continua o discreta

Una señal puede ser continua o discreta en el tiempo. Una señal continua es una función biyectiva con dominio en el tiempo y codominio en los números reales, representándose como $x(t)$. Mientras que una señal discreta en el tiempo es una función con dominio en tiempos discretos y codominio en números reales. Una señal discreta también se puede describir como una secuencia de números, representándose como $x(n)$. Las representaciones gráficas se pueden observar en la Figura 2.

Figura 2. Continuidad de señales (Izq. – Señal continua | Der. – Señal discreta)

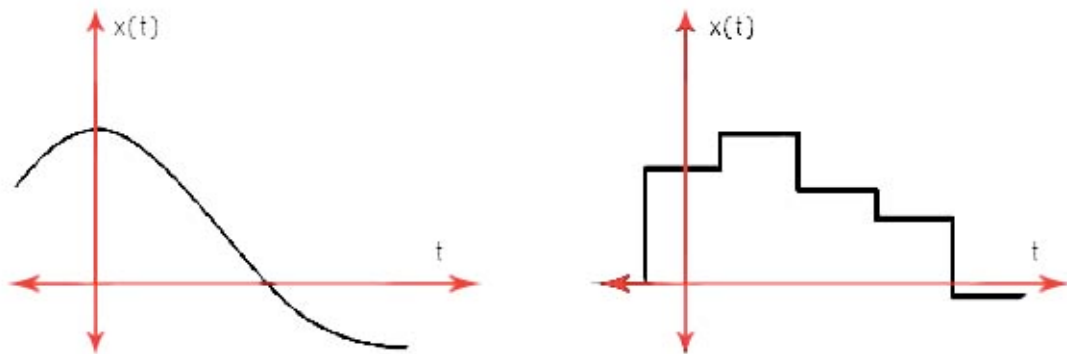


Se puede obtener una señal discreta muestreando una señal continua, lo cual es necesario si se desean utilizar un proceso digital en la señal continua. Mientras que se puede conseguir una señal continua interpolando o reconstruyendo los valores de una señal discreta.

b. Analógica o digital

La diferencia entre las señales analógicas y las digitales es que una señal analógica es una función con valores continuos en el codominio, mientras que en la digital la función puede obtener únicamente valores discretos. Comparando la clasificación de analógico y digital con la clasificación continua y discreta notamos que la diferencia es la discretización de ejes. Una señal digital es la combinación de tener tiempos discretos de muestreo en el eje t y valores discretos en el eje $x(t)$, lo cual se puede visualizar en la figura 3.

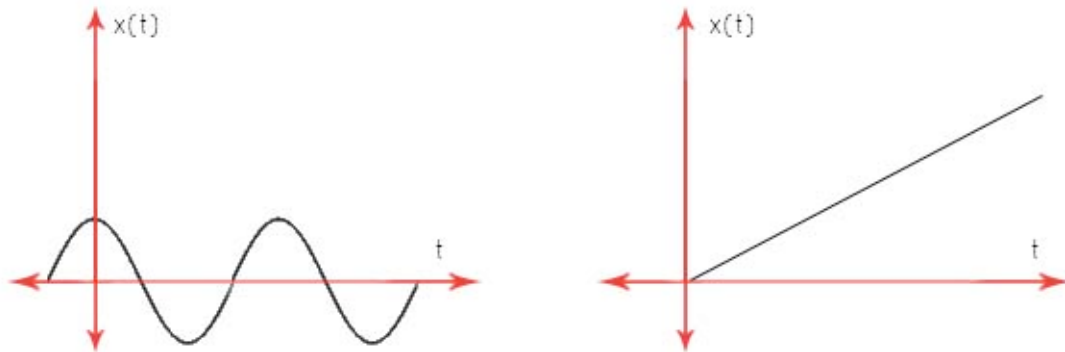
Figura 3. Señales analógica y discreta (Izq. – Señal analógica | Der. – Señal digital)



c. Periódica y aperiódica

Se define a una señal como periódica si repite sus valores en intervalos definidos por su período. El periodo de una señal es el tiempo que le toma volver a repetir sus valores, y está definido como el inverso de la frecuencia. Una señal aperiódica es aquella que no tiene un patrón de repetición definido. Este comportamiento se puede ver en la Figura 4, en la cual se muestra una señal sinusoidal del lado izquierdo, la cual es periódica, mientras que la rampa representada en el lado derecho es aperiódica.

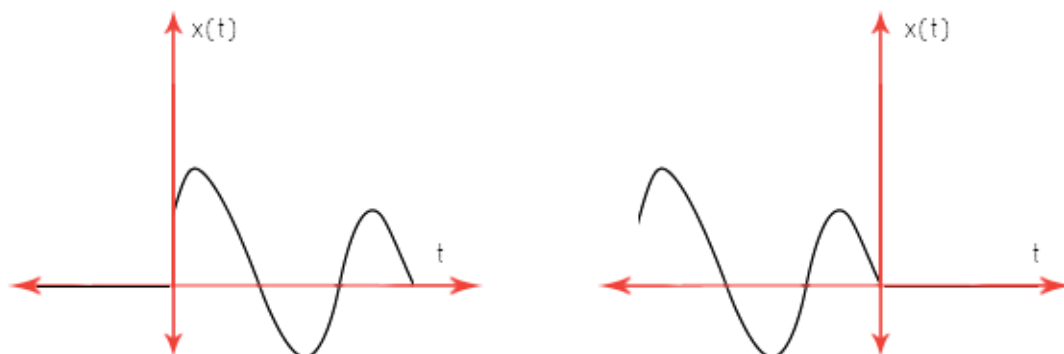
Figura 4. Periodicidad de señales (Izq. – Señal periódica | Der. – Señal aperiódica)



d. Causalidad

Una señal puede ser causal, anti-causal o no causal. Las señales causales son aquellas que tienen valor de cero para tiempos negativos, es decir $x(t) = 0$ para el intervalo $t < 0$. Las señales anti-causales tienen valor de cero para tiempos positivos, descrito por $x(t) = 0$ para el intervalo $t > 0$. Las señales no causales pueden tener cualquier valor en cualquier rango. Este tipo de clasificación sirve ya que una señal causal indica que se inició la señal luego de un evento en específico. En la Figura 5 se puede observar que en la gráfica de la señal causal, no se tiene ningún tipo de respuesta hasta que ocurra un evento en el tiempo cero, mientras que en el lado derecho tenemos una señal anti-causal, la cual se vuelve cero desde el tiempo cero.

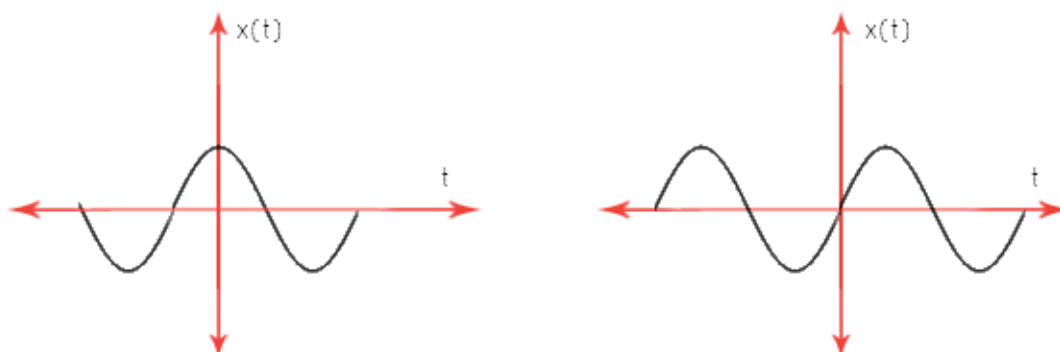
Figura 5. Causalidad de señales (Izq. – Señal causal | Der. – Señal anti-causal)



e. Paridad

Para que una señal $x(t)$ sea par, tiene que cumplir con que $x(t) = x(-t)$. Similar, para que una señal $x(t)$ sea impar, debe de satisfacer $x(t) = -x(-t)$. Que una señal sea par o impar facilita el análisis matemático dado a sus propiedades. Esta cualidad es fácil de reconocer Dado que una señal par es simétrica con respecto al eje vertical mientras que una señal impar es invertida y negada. (Selik, 2007)

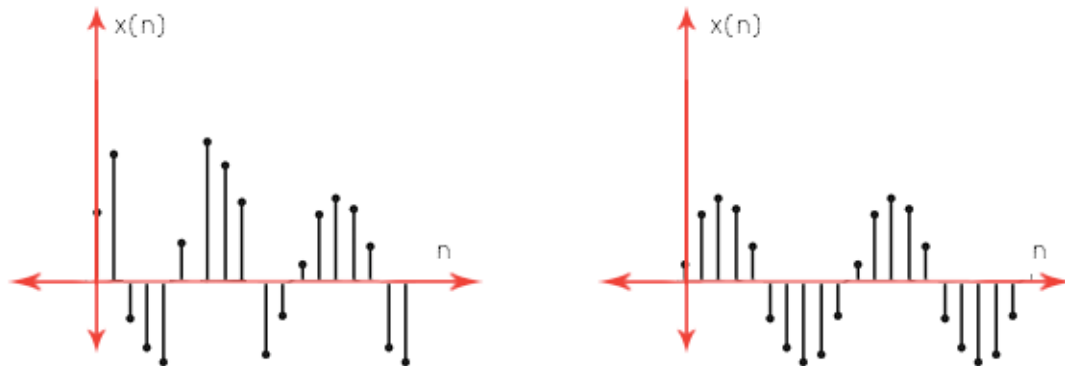
Figura 6. Paridad de señales (Izq. – Señal par | Der. – Señal impar)



f. Determinísticas y aleatorias

Las señales determinísticas son aquellas cuyo valor puede ser calculado usando una expresión matemática, una regla de asignación o tablas. Mientras que las aleatorias son impredecibles, es decir que su comportamiento no puede ser modelado de manera confiable. (Selik, 2007)

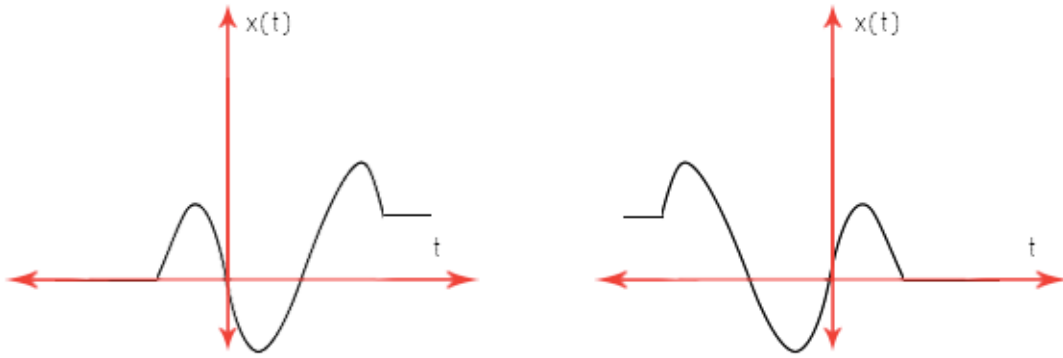
Figura 7. Señales determinísticas (Izq. – Señal aleatoria | Der. – Señal determinística)



g. Hemisferio

Una señal de hemisferio derecho tiene valores de $x(t) = 0$ para el intervalo $t < t_1 < \infty$, y una señal de hemisferio-izquierdo es aquella que cumple con que los valores de $x(t) = 0$ para $t > t_1 > -\infty$. (Selik, 2007)

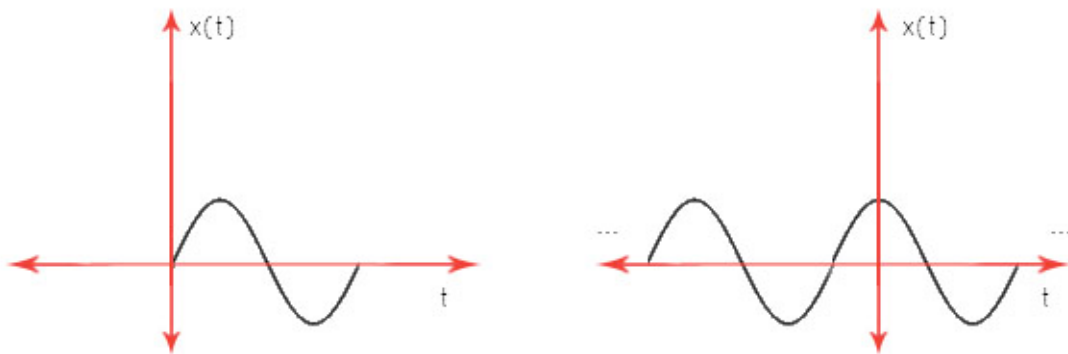
Figura 8. Hemisferio de una señal (Izq. – Hemisferio derecho | Der. – Hemisferio izquierdo)



h. Tamaño de señal

El tamaño de una señal puede ser finito o infinito. Las señales finitas tienen un intervalo finito en la cual la señal no es cero $t_1 < x(t) < t_2$, a diferencia de las infinitas, las cuales se extienden a lo largo de todo el eje de tiempo $-\infty \leq x(t) \leq \infty$. (Selik, 2007)

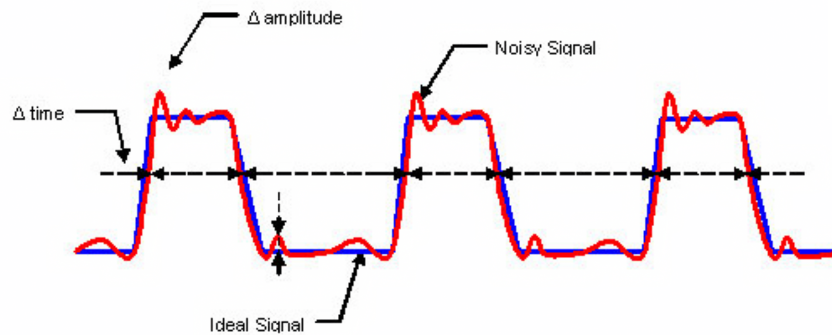
Figura 9. Tamaño de señal (Izq. – Señal finita | Der. – Señal infinita)



2. Integridad de señal

La integridad de una señal encapsula dos aspectos cruciales de diseño electrónico; la sincronización de eventos y calidad de la señal. La integridad describe los aspectos analógicos de las señales digitales y cómo las interconexiones afectan el desempeño del sistema. Identificar la integridad de las señales eléctricas de interés es útil cuando tenemos una frecuencia de reloj alta, ya que esto afecta la manera en la se comportan los componentes y las interconexiones en el circuito. (Fluke, 2014)

Figura 10. Integridad de señal. [16]



Nota. Fluke, 2014:1.

La verificación de integridad de una señal es primordial para el diseño arquitectónico y al momento de detectar errores. Las señales digitales reales están limitadas por tiempos de transición finitos, generan oscilaciones en la respuesta debido a las líneas de transmisión y se genera reactancia parásita por componentes en el circuito. También se toma en cuenta la sobreelevación, el crosstalk generado por campos magnéticos y el ruido introducido al sistema.

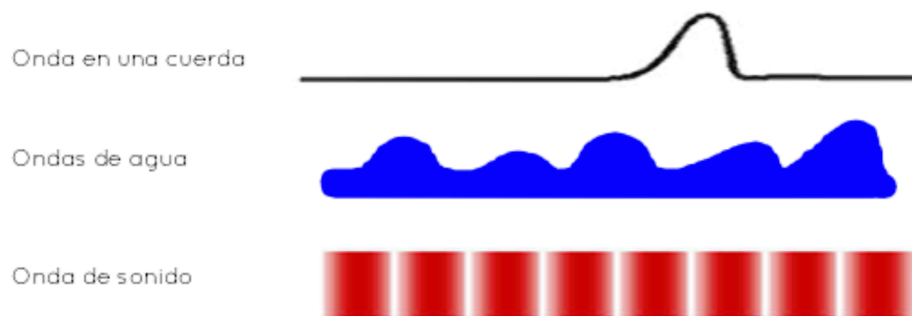
El estudio de la integridad de una señal se puede reducir al análisis de la variación en amplitud y tiempo del valor esperado. Los errores más comunes se dan por el ruido, los cuales son:

- Ruido por reflexión – generado por discordancia en impedancias.
- Ruido por Crosstalk – generado por acoplamiento electromagnético entre las trayectorias de las vías.
- Ruido de Tierra – generado por el switch simultáneo de salidas. (Chou, 1994:78)

B. ONDAS

Una onda es una perturbación que viaja a través de un medio de una ubicación a otra. La mayoría de movimientos generados naturalmente se comportan como una onda. Como ejemplo está el comportamiento de mareas, los movimiento de tierras en terremotos, explosiones, la propagación de sonido, o la frecuencia de vibración de un cuerpo en movimiento. En la Figura 11 se muestran ejemplos de ondas simples, primero encontramos un pulso en una cuerda, la segunda imagen muestra olas en forma de onda moviéndose a través de un cuerpo de agua, y por último se muestran ondas de presión de aire por las cuales se transmite sonido. (Tektronix, 2008:4)

Figura 11. Onda en una cuerda, onda sinusoidal y onda de compresión



Para el análisis de estos eventos es necesario identificar las características principales de la onda que describen el evento. Estas características son la frecuencia de la onda, la intensidad de la señal, su amplitud y su fase.

1. Frecuencia y período

Cuando una onda se repite cada cierto tiempo, se dice que es una onda periódica. En donde el periodo (T) es el tiempo en que se tarda en completar un ciclo. La tasa en la que esta se repite se conoce como frecuencia. La frecuencia (f) se mide en Hertz (Hz) y es igual al número de veces que la señal se repite en un segundo. El período y la frecuencia son recíprocos entre sí, de manera que $1/T$ es igual a la frecuencia y el período es igual a $1/f$. (Tektronix, 2008:11)

2. Voltaje

El voltaje es la cantidad de potencial eléctrico, interpretado como la intensidad de la señal, entre dos puntos en un circuito. Por lo general, uno de estos puntos es de tierra, pero puede ser cualquier otra señal de referencia. Es posible que desee medir el voltaje desde el pico máximo al pico mínimo de onda, esto se conoce como el voltaje de pico a pico de la onda. (Tektronix, 2008:11)

3. Amplitud

La amplitud de una onda eléctrica es la cantidad de voltaje entre dos puntos de un circuito. Esta se refiere comúnmente a la tensión máxima entre una señal y tierra. (Tektronix, 2008:12)

4. Fase

La fase denota la ubicación actual en el ciclo de una onda, medido en grados. La fase de una onda está definida como la distancia entre el primero de cruce por cero y el punto en el espacio definido como el origen. Si dos o más ondas tienen la misma frecuencia y fase, se dice que están en fase, lo que significa que estarán alineadas en todas partes. Cuando dos o más ondas tienen la misma frecuencia pero fases diferentes se dice que están desfasadas. (Tektronix, 2008:12)

C. OSCILOSCOPIO

El osciloscopio es un instrumento electrónico que permite visualizar señales eléctricas. Este dispositivo genera una gráfica de la señal, mostrando una onda, en la cual observa cómo varía el voltaje con respecto al tiempo. El osciloscopio ayuda a visualizar valores de voltaje en un tiempo específico, indicar la frecuencia de oscilación si la señal es periódica, identificar distorsión en la señal y verificar si se ha introducido ruido en el sistema, entre otros. Esta información es crucial para que el ingeniero, técnico, científico y/o educador, pueda analizar y visualizar el comportamiento de los eventos de interés. (Tektronix, 2008:13)

1. Tipos de osciloscopio

Los osciloscopios se pueden clasificar en dos categorías generales: analógicos y digitales. El osciloscopio analógico funciona con voltajes que varían continuamente, mientras que el digital trabaja muestreando una señal analógica, convirtiéndola en números binarios discretos que representan el voltaje.

Un osciloscopio digital también se puede utilizar como digitizador si el sistema ya se encontrara en modo digital, evitando el uso de un convertidor, lo cual resulta útil al visualizar secuencias de datos. En la Figura 12 se muestra un osciloscopio analógico, el cual por medio de un barrido de la pantalla, genera la imagen de la onda. Del lado derecho se encuentra un osciloscopio digital, el cual tiene capacidad de mostrar múltiples canales simultáneamente.

Figura 12. Tipos de osciloscopios (Izq. – Analógico | Der. – Digital) [18]



Nota. Tektronix, 2014.

2. Especificaciones de desempeño

Las consideraciones de desempeño de un osciloscopio afectan de manera significativa la capacidad de obtener una integridad de señal deseada. Esta sección describe algunos términos de desempeño del osciloscopio que se utilizan para elegir el instrumento adecuado para la aplicación de interés. (Tektronix, 2008:35)

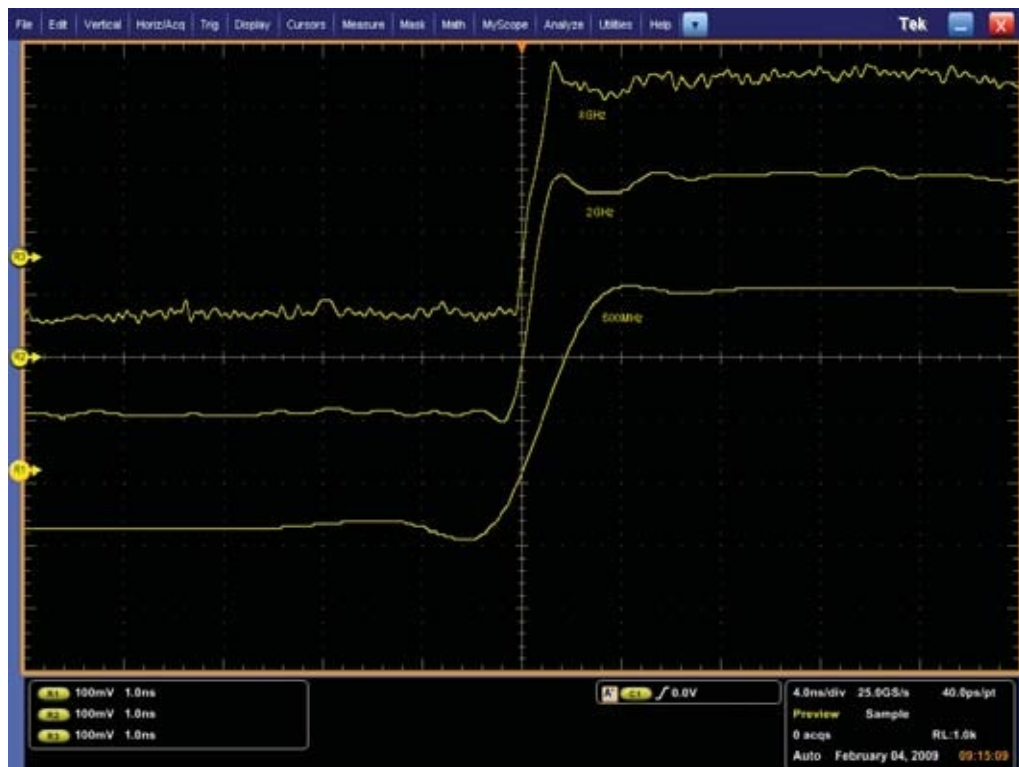
a. Ancho de banda

El ancho de banda determina la capacidad de un osciloscopio para medir una señal. A medida que aumenta la frecuencia de la señal, la capacidad del osciloscopio para visualizar con precisión las disminuciones de señal. Esta especificación indica el rango de

frecuencias que el osciloscopio puede medir con precisión. Sin un ancho de banda adecuado, el osciloscopio no será capaz de identificar los cambios de alta frecuencia, se distorsionará la amplitud y se perderán los detalles. (Tektronix, 2008:36)

El ancho de banda se especifica como la frecuencia en la que una señal de entrada sinusoidal es atenuada al 70.7% de su amplitud original, también conocido como el punto de -3 dB. (National Instruments, 2011)

Figura 13. Ancho de banda [18]



Nota. Tektronix, 2011.

En la Figura 13 se observa que mientras mayor sea el ancho de banda, la representación de la señal será más precisa. Esto se ilustra con una señal capturada a 250 MHz, 1 GHz y 4 GHz de ancho de banda. Esto muestra por qué un digitalizador de banda ancha es importante al momento de capturar ondas con componentes de alta frecuencia.

La diferencia que hace el ancho de banda en las gráficas es evidente, ya que al tener un ancho de banda mayor se logra representar la señal de una manera más real. (Tektronix, 2008:35)

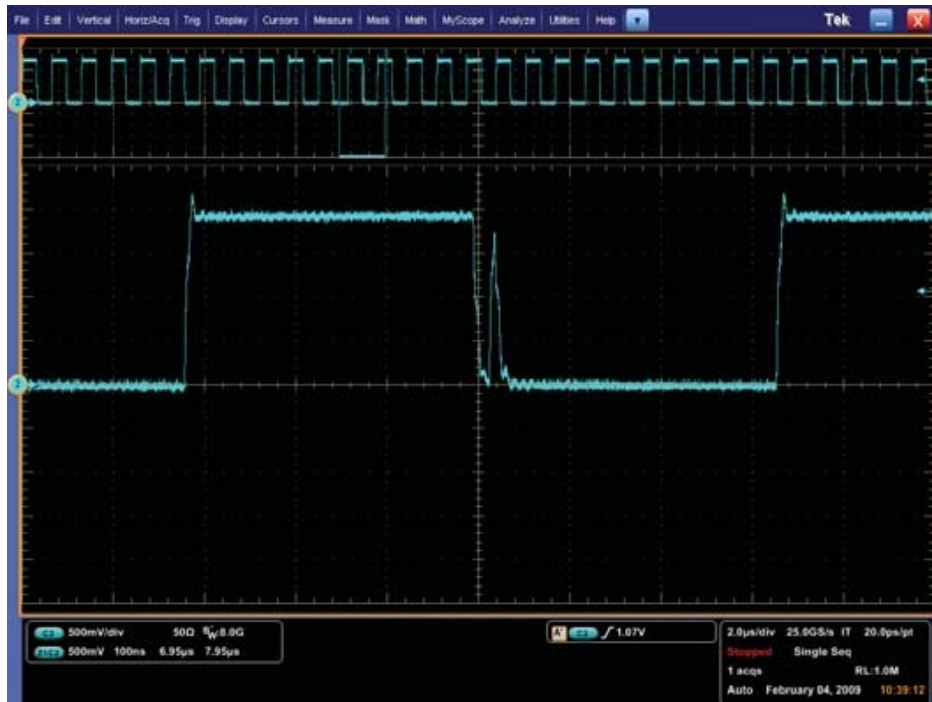
Se puede mejorar el ancho de banda de un osciloscopio utilizando procesamiento de señales digitales. Un filtro de ecualización arbitraria DSP puede ser utilizado para mejorar la respuesta del canal del osciloscopio. Este filtro extiende el ancho de banda, aplanando la respuesta del canal, mejora la linealidad de fase y proporciona una mejor correspondencia entre los canales. También disminuye el tiempo de subida y mejora la respuesta de paso de dominio de tiempo. (Tektronix, 2008:36)

b. Frecuencia de muestreo

La frecuencia de muestreo se refiere a la frecuencia con la cual el osciloscopio digital toma muestras de una señal. Mientras más rápido muestree un osciloscopio, mayor será la resolución y el detalle en la forma de onda, haciendo menos probable la pérdida de datos o eventos críticos. (National Instruments, 2011)

Para reconstruir con precisión una señal y evitar el aliasing, el teorema de Nyquist establece que la señal debe ser muestreada al menos al doble de la frecuencia de su componente más alto. Este teorema, sin embargo, asume una longitud de registro infinita y una señal continua. Dado que ningún osciloscopio ofrece una longitud de registro infinita, la toma de muestra utilizando este teorema es por lo general insuficiente. (National Instruments, 2011)

Figura 14. Frecuencia de muestreo [19]



Nota. National Instruments, 2011.

Mientras que la frecuencia de muestreo indica la tasa con la cual el osciloscopio muestrea la señal de entrada dentro de una onda, la tasa de captura se refiere a la rapidez con que un osciloscopio adquiere la onda. La tasa de captura es expresada como ondas por segundo (ondas/s). (Tektronix, 2008:38)

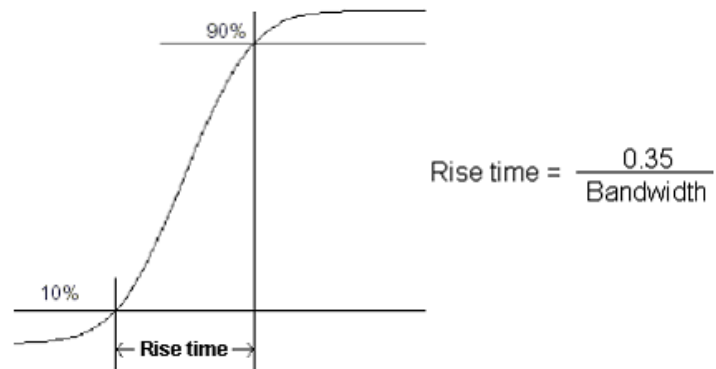
Osciloscopios que poseen altas tasas de captura aumentan drásticamente la probabilidad de capturar anomalías transitorias como el jitter, pulsos pequeños, fallos y errores de transición. (Tektronix, 2008:38)

c. Tiempo de subida

El tiempo de subida se define como el tiempo que tarda una señal en pasar desde el 10% hasta el 90% de su valor final. El osciloscopio debe tener suficiente tiempo de

subida para capturar con precisión los detalles de las transiciones rápidas. Se utiliza el tiempo de subida para describir el rango de frecuencia útil de un osciloscopio. (National Instruments, 2011)

Figura 15. Tiempo de subida [19]



Nota. National Instruments, 2011.

La Figura 15 muestra la relación entre el ancho de banda (Bandwidth) y el tiempo de subida (Rise Time).

d. Capacidad de registro

La capacidad de registro se expresa como el número de puntos o muestra que comprenden un registro de una onda completa. Esto determina la cantidad de datos que se pueden capturar con cada canal. Dado que un osciloscopio puede almacenar un número limitado de muestras, la duración de la onda (tiempo) será inversamente proporcional a la tasa de muestreo del osciloscopio. (Tektronix, 2008:39)

El intervalo de tiempo visible queda definido como la división del tamaño de registro sobre la frecuencia de muestreo. Se puede seleccionar un tamaño de registro específico

dependiendo de la aplicación, de manera que se optimice el nivel de detalle hasta donde sea necesario.

e. Función de disparo

Típicamente los osciloscopios y digitalizadores se utilizan para adquirir señales basadas en un determinado evento. La capacidad de disparo del instrumento permite aislar el evento de interés, capturando señal antes y después del evento. Esta función en un osciloscopio sincroniza el barrido horizontal en el punto correcto de la señal, lo cual es esencial para la caracterización clara de la señal.

La función de disparo hace que las ondas repetitivas aparezcan estáticas en la gráfica del osciloscopio, mostrando repetidamente la misma porción de la señal de entrada. Esta función puede reaccionar a varios parámetros, como lo es un cambio de polaridad, umbral de amplitud o por duración de pulsos entre otros.

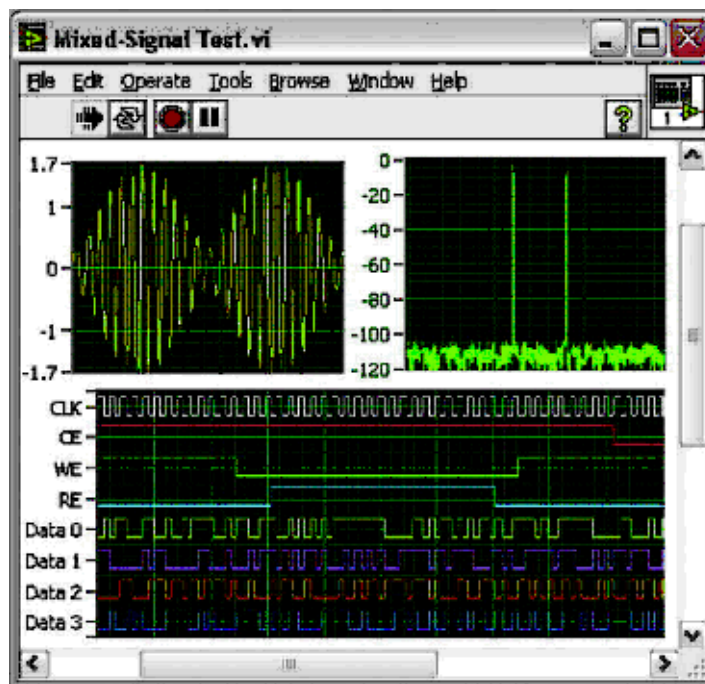
f. Densidad de canales

Un factor de desempeño de un osciloscopio o digitalizador es el número de canales del instrumento. La mayoría de los osciloscopios tienen de dos a cuatro canales, cada uno muestreado simultáneamente a una velocidad determinada. Es importante tener cuidado con cómo se ve afectada la velocidad de muestreo al utilizar todos los canales del digitalizador. Esto es debido a una técnica de uso común llamado muestreo en tiempo entrelazado, que intercala múltiples canales para lograr una tasa de muestreo más alta. Si el digitalizador o un osciloscopio utiliza este método y está utilizando todos los canales, es posible que no sea capaz de adquirir a la tasa máxima de adquisición, Dado que se divide la frecuencia de muestreo entre el número de canales activos. (National Instruments, 2011)

g. Capacidad de manejo de señales mixtas

Existen aplicaciones en las cuales es de interés visualizar tanto señales analógicas como digitales, esto se conoce como la capacidad de manejo de señales mixtas. Al tener señales digitales como usualmente son las señales de control y las analógicas desplegadas en conjunto se puede analizar el comportamiento del sistema con mayor profundidad. También se puede utilizar los canales de señales analógicas para desplegar las señales digitales, mostrando la verdadera imagen de los datos siendo recibidos. Esto facilita el análisis de integridad de la señal digital, ayudando a corregir errores.

Figura 16. Señales mixtas [19]



Nota. National Instruments, 2011.

La Figura 16 muestra señales mixtas, en donde las analógicas se encuentran en la parte superior, mientras que en la parte inferior se muestran las señales digitales de control y datos digitales de interés.

D. PROCESSING

Processing es un lenguaje de programación, entorno de desarrollo, y comunidad en línea. Desde el año 2001, ha promovido la producción de software dentro de las artes visuales y culturas visuales por medio de la tecnología. En su inicio fue creado para servir como un cuaderno de bocetos de software y para enseñar los fundamentos de programación dentro de un contexto visual, evolucionó hasta convertirse en una herramienta de desarrollo para profesionales. Hoy en día, hay miles de estudiantes, artistas, diseñadores, investigadores y aficionados que utilizan el procesamiento para el aprendizaje, la creación de prototipos y producción. (Processing, 2014)

Este software es una alternativa a las herramientas de software propietario con licencias restrictivas y costosas, por lo que es accesible para las escuelas y los estudiantes. Su condición de código abierto fomenta la participación comunitaria y la colaboración que es vital para el crecimiento de Procesamiento. Colaboradores comparten programas funcionales, contribuyen con trozos de código, construyen bibliotecas, herramientas y modos que amplían las posibilidades del software. (Processing, 2014)

La comunidad Processing ha escrito más de un centenar de bibliotecas para facilitar la visión por ordenador, visualización de datos, la composición musical, la creación de redes, la exportación de archivos en 3D, y la electrónica de programación. Basado en el lenguaje Java, pero utiliza una sintaxis simplificada y un modelo modificado para la programación de gráficos. (Processing, 2014)

V. ANTECEDENTES



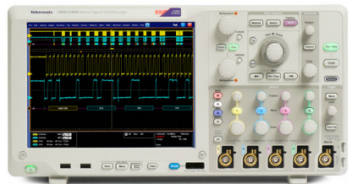
A. DEL TRABAJO DE GRADUACIÓN

Dada la gran diversidad de dispositivos electrónicos que son utilizados hoy en día, la necesidad de visualizar las señales utilizadas entre módulos es indispensable para la verificación de funcionamiento correcto y detección de errores. Por esta necesidad existen varios tipos de instrumentos electrónicos que cumplen esta finalidad, entre los cuales están los osciloscopios analógicos y digitales.

Los osciloscopios digitales han reemplazado a los analógicos por varias razones, como la capacidad de memoria, frecuencia de muestreo, tamaño y costo. Gracias a estas razones existen varios proyectos que tratan de crear su propia versión del osciloscopio que cumpla con necesidades de diseño específicas.

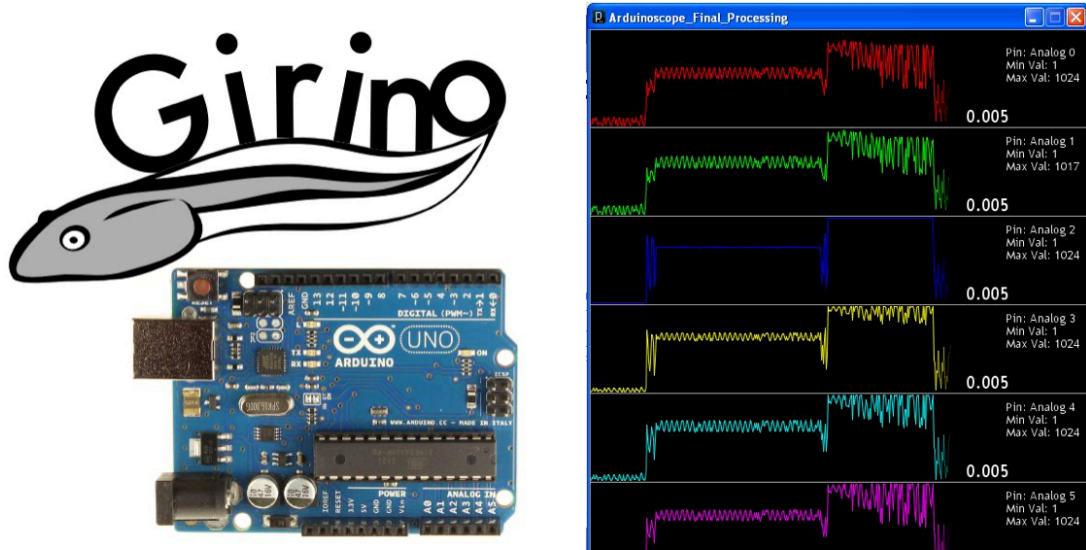
Entre los osciloscopios digitales comerciales más destacados se encuentran los creados por Tektronix. Esta compañía ofrece tres categorías de osciloscopios digitales, entre los cuales están los básicos, de señales mixtas y por desempeño. En el siguiente cuadro se muestra un osciloscopio de cada una de estas categorías junto a una pequeña descripción. El Cuadro 1 muestra osciloscopios con características similares al desarrollado.

Cuadro 1. Osciloscopios digitales ofrecidos por Tektronix

TBS1104	MDO4014B-3	MSO5034B
		
Ancho de Banda Analógico: 100 MHz Frecuencia de Muestreo: 1 Gs/s Canales: 4 Analógicos	Ancho de Banda Analógico: 100 MHz Frecuencia de Muestreo: 2.5 Gs/s Canales: 4 Analógicos – 16 Digitales	Ancho de Banda Analógico: 350 MHz Frecuencia de Muestreo: 5 Gs/s Canales: 4 Analógicos– 16 Digitales
Precio de lista: \$1490	Precio de lista: \$9500	Precio de lista: \$15,600

El alto precio de un osciloscopio comercial ha motivado a individuos y pequeñas empresas a crear su versión de este instrumento. La mayoría de osciloscopios desarrollados por estudiantes y aficionados utilizan la plataforma de Arduino y microprocesadores para muestrear las señales, Dado que incluyen convertidores analógicos-digitales, lo cual facilita la adquisición de datos y reduce el precio del dispositivo significativamente. Entre los proyectos de osciloscopio personales, se destaca el Girino y el Arduinoscope. El costo total de estas propuestas no supera los \$60. La principal limitación que estos dispositivos presentan es la cantidad de canales simultáneos que pueden manejar de manera confiable, graficando hasta 6 canales. También se compromete el ancho de banda, ya que en los modos de muestreo más altos, este dispositivo llega a muestrear hasta 153kSa/s, en el Girino.

Figura 17. Osciloscopios desarrollados con Arduino [10] [5]

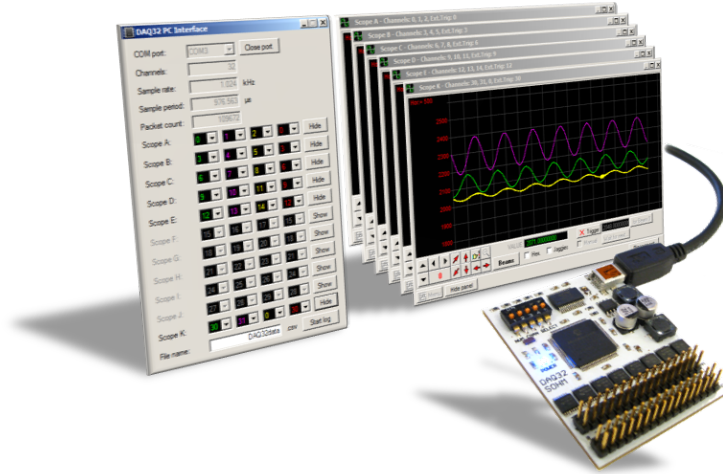


Nota. Instructables: Girino/Arduinoscope, 2014.

La Figura 17 muestra del lado derecho el despliegue de los canales analógicos muestreados usando los convertidores incluidos en el Arduinoscope, se debe notar que usando este acercamiento, mientras se agreguen canales el ancho de banda disminuye.

La empresa X-io Technologies desarrolló, como proyecto libre de uso, un dispositivo capaz de muestrear 32 canales simultáneamente. Posee una frecuencia de muestreo de 32 kHz para un canal, lo cual se convierte en 1 kHz por canal al habilitar todos los canales. Aunque este dispositivo tenga la capacidad de mostrar gran cantidad de canales, aún no soluciona el percance de la frecuencia de muestreo, Dado que 1 kHz por canal es muy bajo para la mayoría de comunicaciones digitales inter-modulares relacionadas en proyectos electrónicos. En la Figura 18 se muestra el dispositivo y el software que despliega las gráficas.

Figura 18. DAQ32 desarrollado por X-io Technologies [7]

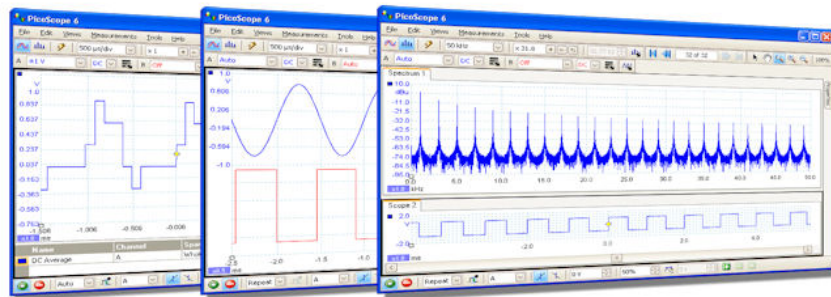


Nota. Github, 2014.

B. DEL MÓDULO

El diseño del hardware es solamente la mitad del trabajo, Dado que en el software se le interpretan y despliegan los datos de manera que sean entendibles por el usuario. Del lado comercial, Picoscope ofrece un software gratuito del mismo nombre el cual al momento de unirlo con el hardware que ofrece la compañía. Este software solamente funciona en computadoras con sistema operativo Windows.

Figura 19. Picoscope Software [4]



Nota. Picoscope, 2014.

Para el sector de software público, la empresa X-io Technologies provee un software creado en Processing capaz de desplegar gráficas de 16 canales. Esta aplicación fue diseñada para graficar las señales de sensores por medio del hardware OSC que ofrece la compañía. En la Figura 20 se muestra la distribución de gráficas en el OSC-illoscope.

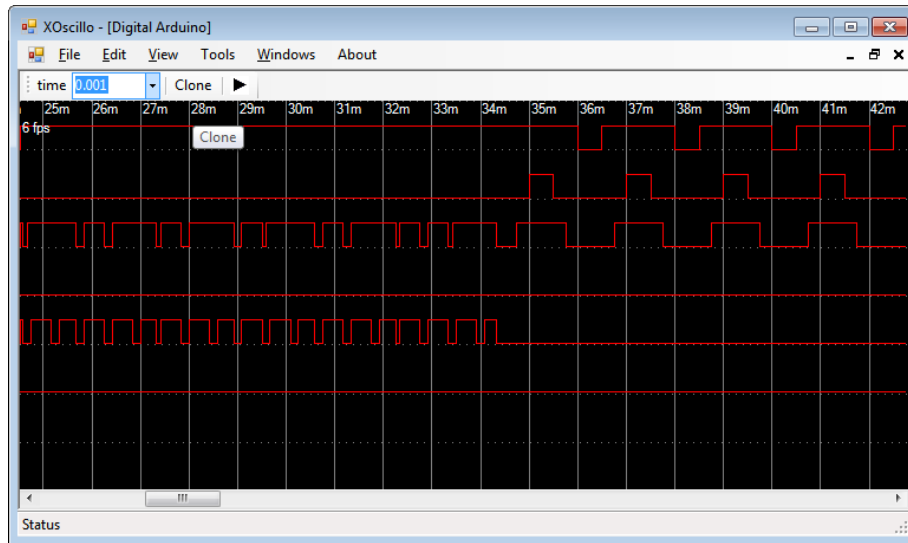
Figura 20. OSC-illoscope [7]



Nota. X-io Technologies, 2014.

De igual manera xoscillo es un software libre que puede ser utilizado con distintas plataformas, por ejemplo Arduino y Parallax. Este paquete funciona con Windows y Linux, y posee características de grado comercial, como lo es vista panorámica, capacidad de guardar y cargar ondas, manejo múltiple de ondas y análisis frecuencial. Es capaz de manejar hasta 4 canales simultáneos a frecuencia máxima de 7 kHz, lo cual es suficiente para proyecto de aficionados, pero no para aplicaciones serias de alta frecuencia. En la Figura 21 se muestra la distribución de canales al momento de analizar secuencias lógicas.

Figura 21. Análisis lógico por xoscillo [20]



Nota. Google Code, 2014.

VI. DISEÑO EXPERIMENTAL

Para el diseño y desarrollo del dispositivo y del software de este proyecto, se llevaron a cabo distintas etapas de investigación, selección de equipo, descripción y diseño de módulos. Por lo anterior, a continuación se presenta con detalle, la manera en que se desarrollaron las distintas fases que comprenden el presente trabajo de graduación.

A. INVESTIGACIÓN TEÓRICA

El trabajo de graduación se inició estableciendo las bases teóricas necesarias que permitirían el desarrollo de los temas de cada módulo. Para realizar esta investigación se utilizó como fuente principal de información: el internet y la Biblioteca de la Universidad del Valle de Guatemala.

En esta fase se inició recopilando información sobre el funcionamiento del osciloscopio y sus características. Luego se buscaron los dispositivos ofrecidos en el mercado, junto con la descripción de sus capacidades de muestreo. Con la información del hardware definida se procedió investigando sobre propuestas de software comerciales y libres, lo cual sirvió para delimitar el alcance del proyecto.

Al tener conocimiento básico acerca del funcionamiento del osciloscopio, el alcance del proyecto establecido y los objetivos específicos definidos se procedió a realizar una investigación sobre el hardware y software necesario para construir el dispositivo y su aplicación. Se recopiló información acerca de los osciloscopios digitales con interfaz USB a la venta y del software que estos dispositivos utilizan. Conociendo distintos instrumentos de representación de señales se establecieron los parámetros requeridos para la obtención de datos, su procesamiento y representación gráfica.

Teniendo el hardware y software necesario para el diseño y construcción del osciloscopio USB multicanal definido, se procedió a realizar el diseño de cada módulo. Cada módulo requirió su respectiva investigación de funcionamiento para poder ser diseñado de la manera adecuada. Al concluir el diseño solo restaba la integración de módulos. Se profundiza en el desarrollo de estas etapas en las siguientes secciones.

B. SELECCIÓN DE HARDWARE

Para las etapas de investigación, diseño y desarrollo se utilizó equipo propio junto con equipo provisto por la Universidad del Valle de Guatemala. El equipo propio principal utilizado fue una computadora portátil y un celular con conexión a internet. Los materiales provistos por la Universidad del Valle de Guatemala fueron utilizados en el laboratorio de electrónica, ya que cuenta con la instrumentación electrónica necesaria para realizar el diseño y hacer pruebas. Esta instrumentación es comprendida por equipos como: un osciloscopio analógico, un generador de ondas, fuentes de voltaje, computadora de escritorio y router con conexión a internet.

A continuación se listan los equipos utilizados para la elaboración de este trabajo de graduación de manera detallada:

Computadora personal portátil Macbook Pro, utilizada principalmente para la etapa de investigación, procesamiento de datos y programación de módulo de software. Esta computadora cuenta con sistema operativo OS X versión 10.9.2 , 8 GB RAM DDR3 y 750 GB de capacidad de almacenamiento de estado sólido. Esta computadora posee la especificación adecuada para poder realizar el desarrollo y pruebas de software necesarias.

Teléfono celular iPhone 5s, utilizado principalmente como acceso a internet portátil. Útil para todas las etapas ya que funciona como medio principal de comunicación y de organización. Se usó el calendario para organizar de manera cronológica las etapas del proyecto, administrando de mejor manera el tiempo. Habilitó la posibilidad de trabajar remotamente, removiendo la necesidad de estar en el laboratorio de la universidad para utilizar internet.

Osciloscopio TEKTRONIX - TDS 3032C es el equipo que permite la verificación de un funcionamiento correcto del dispositivo creado y para detección de errores entre las interconexiones modulares.

Computadora de escritorio Dell, con Windows XP. Computadora que cuenta con licencias para el uso del paquete de software Altium Designer. Este software se utilizó para la configuración del Nanoboard 3000, y para la programación de la interfaz gráfica residente en este dispositivo. Este software también fue útil para la creación de esquemáticos, por lo que se utilizó mayormente en el módulo de hardware. Gracias a que esta computadora tiene un sistema operativo diferente al de la computadora personal en la que se desarrollo el software, también sirve para confirmar la portabilidad de la aplicación.

C. SELECCIÓN DE SOFTWARE

Para las etapas de investigación, diseño y desarrollo se utilizó software propio junto software provisto por la Universidad del Valle de Guatemala. Entre el software propio usado esta un editor de texto, navegador de internet y plataforma de programación. El software de la Universidad utilizado fue principalmente Altium Designer, ya que es necesario tener una licencia vigente para utilizarlo.

A continuación se listan los paquetes de software y aplicaciones de programación que se utilizaron para la redacción del trabajo escrito y para el diseño/desarrollo del software de la aplicación e interfaz gráfica en el Nanoboard:

Dentro del software propio utilizado esta el editor de texto, Microsoft Word:mac2011, el cual fue utilizado para la redacción del presente trabajo escrito. Para el proceso de investigación se usó el navegador nativo de la computadora portátil personal, Safari. Esta fue la herramienta principal de recopilación de información.

Processing fue seleccionado para la programación del módulo de software principal. Se eligió esta plataforma dado a su enfoque en las artes visuales y a su portabilidad. El lenguaje de programación utilizado por Processing es Java, el cual deja exportar la aplicación a los sistemas operativos usados en la actualidad como lo es Windows, Mac OS y Linux.

Altium Designer es un paquete de software que permite programar el Nanoboard 3000. Siendo este el dispositivo principal utilizado para el módulo de hardware, este software es necesario. Además del software que despliega las señales en la computadora, se realizó un programa para la interfaz con el usuario desde este dispositivo. En este programa se presentan los controles del dispositivo, los cuales son la frecuencia de muestreo, el tiempo que captura y el modo de operación.

D. DESCRIPCIÓN DE MÓDULOS

En esta fase del proyecto se delimita el alcance de cada módulo, lo cual es necesario para realizar el diseño. Este trabajo de graduación se dividió en dos partes, un módulo de hardware y un módulo de software. El módulo de hardware se encarga del muestreo de señales analógicas y la captura de señales digitales.

El módulo de software incluye tanto el software realizado para la computadora como el software realizado para el Nanoboard 3000. Dado que estos módulos deben comunicarse eficientemente entre sí, se completa el proyecto utilizando comunicación serial por medio de USB 2.0.

En el Cuadro 2 se muestran las tareas delegadas a cada módulo. El presente documento trata el módulo de software, para referencia detallada del módulo de hardware se pide revisar documento adecuado.

Cuadro 2. Tareas específicas por módulo

Módulo hardware	
Diseño del hardware	Manejo de memoria Nanoboard 3000
Muestreo de señales analógicas (FlashyD Board)	Selección puntas canales analógicos
Multiplexación de canales de entrada	Selección conductores canales digitales
Captura de señales digitales (Nanoboard 3000)	Opción de trigger
Manejo de datos muestreados	Señales de control
Envío de información por medio USB 2.0	Integración módulos
Módulo software	
Diseño del software	Manejo de archivos
Despliegue de señales analógicas	Exportar aplicación multiplataforma
Manejo de despliegue de canales	Control de modo de visualización
Despliegue de señales digitales	GUI Nanoboard 3000
Interpretación de datos recibidos	Despliegue de información de referencia
Recepción de información por medio USB 2.0	Integración módulos

E. DISEÑO HARDWARE

La fase de diseño de hardware encapsula la configuración del FPGA del Nanoboard 3000 para aplicación de osciloscopio digital con alta frecuencia de muestreo (100MSa/s). También es el encargado del desarrollo de módulos externos de hardware, los cuales son una serie de multiplexores y convertidores analógicos a digital FlashyD.

Este módulo maneja la comunicación USB y es el delegado a la selección de puntas del osciloscopio tanto para las entradas analógicas como las digitales. El detalle acerca de este módulo proviene del trabajo de graduación Osciloscopio USB Multicanal módulo Hardware elaborado por Juan Pablo Ruiz, el cual se muestra en esta sección.

Figura 22. Diseño completo del osciloscopio USB multicanal [11]

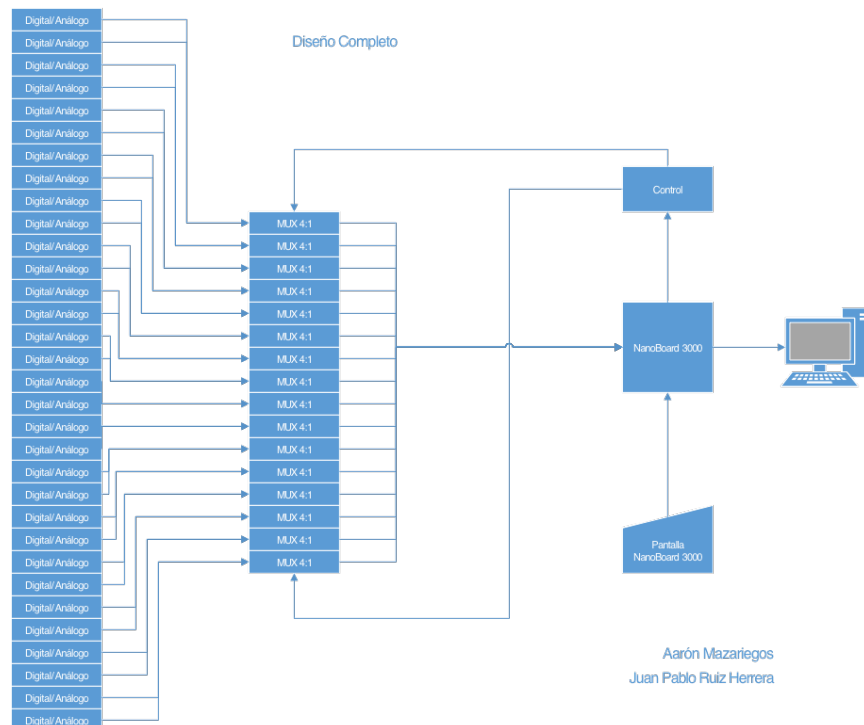


Figura 23. Diseño Nanoboard [11]

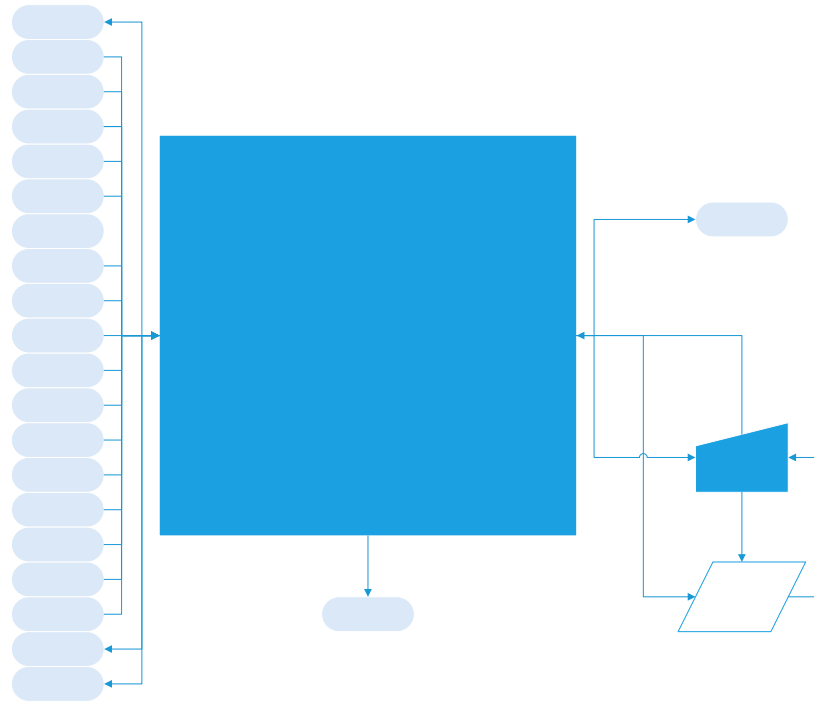
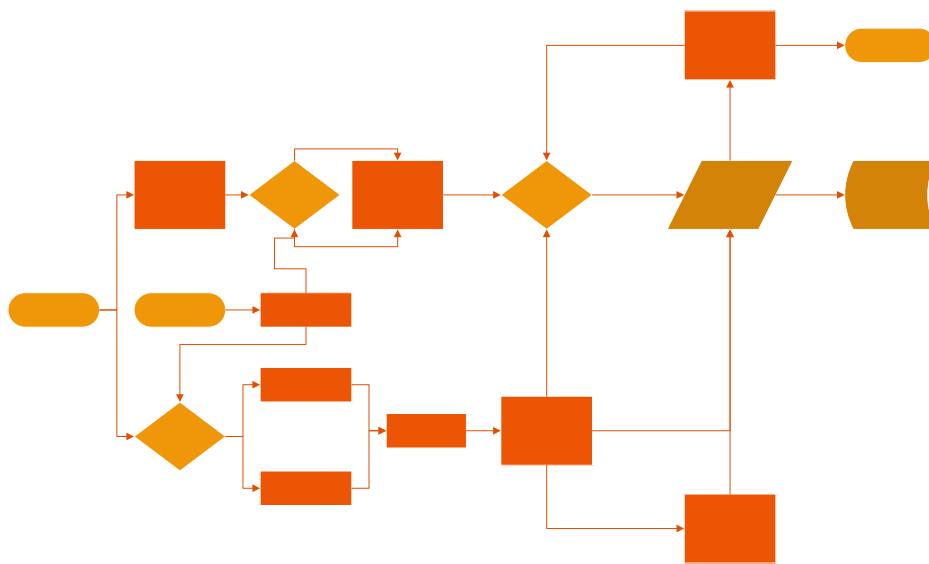


Figura 24. Diseño interno FPGA [11]



Este módulo maneja la comunicación USB y es el delegado a la selección de puntas del osciloscopio tanto para las entradas analógicas como las digitales. El detalle acerca de este módulo proviene Como se puede ver dentro de la Figura 23, este es el diseño físico que se puede encontrar, aquí se encuentra todo lo referente a las entradas de las señales y como estos van a funcionar. Explicando más a detalle: Aquí se encuentran 64 entradas en el extremo izquierdo, se encuentran todas las entradas digitales y todas las entradas analógicas. Estas entradas analógicas o digitales pueden ser escogidas desde la pantalla del Nanoboard 3000, el FPGA saca dos señales de control y una de su reloj interno, con el reloj interno está cambiando de señal de las primeras 16 entradas y luego cambia a las 16 entradas siguientes. Los multiplexores utilizados tienen la capacidad de cambiar las señales hasta una velocidad de 250 MHz, por esto la razón de seleccionar estos multiplexores (AD8174ARZ-ND). Son multiplexores de 4 a 1, con la idea de poder seleccionar entre dos entradas analógicas o dos entradas digitales. Debido a que se tienen en el Nanoboard 20 entradas/salidas, se tiene que 17 son entradas y 3 son salidas, estas salidas sirven para poder controlar los multiplexores externos, ya que estos no están incluidos en la arquitectura del Nanoboard 3000.

Antes de esto se tiene lo que se llama el FlashyD analog acquisition board el cual es capaz de poder adquirir datos de una señal y convertirlos en digital de una resolución de 8 bits y ponerlos en paralelo, con la intención de que este muestreando a 100 MHz, para poder obtener 4 entradas analógicas se necesitan entonces 32 bits totales, los cuales justifican 32 entradas.

Estas tienen sus propias puntas de alta frecuencia, para poder adquirir los datos que se requieren. Estas puntas traen una impedancia de $10M\Omega$ o de $1M\Omega$. El osciloscopio es capaz de elegir entre tres modos: modo digital, modo analógico y modo híbrido. El modo híbrido va a ser capaz de leer 2 entradas analógicas y 16 entradas digitales.

La razón por la que se eligió una resolución de 8 bits es para poder tener la opción de medir 4 entradas analógicas simultáneamente, ya que es mejor tener 4 entradas analógicas capaces de poder ver distorsiones en las señales, en lugar de solamente dos entradas de total resolución de 16 bits, configuración en la cual solo tendríamos dos entradas analógicas.

Con la pantalla del Nanoboard se puede seleccionar los tiempos de muestreo, tomando en cuenta la capacidad del registro. También se puede cambiar la frecuencia de muestreo y el porcentaje del trigger, ya sea se requiera el 30% de datos futuros con el 70% de datos pasados. El trigger puede ser seleccionado para cuando haya cambios de flanco, ya sean positivos o negativos, a selección del usuario. Luego esto se envía hacia el módulo de software, el encargado de almacenar los datos para poder ser graficados y desplegados con el debido programa.

Se debe de tomar en cuenta que los parámetros seleccionables se basan en el siguiente cuadro. Este cuadro enseña los rangos de visión, claramente el tiempo más grande va a ser muy difícil poderlo ver, por lo cual estará limitado únicamente por el usuario, el programa dejará elegir la máxima cantidad de tiempo, por fines de demostración. Este cuadro está realizada con la máxima cantidad de tamaño de registro, estos son datos almacenables que luego serán utilizados para enviarse al computador.

Cuadro 3. Tiempo Visible por Frecuencia de Muestreo

Frecuencia seleccionable	Frecuencia de muestreo (Hz)	Tamaño de registro	Tiempo visible (s)
100 MHz	100,000,000	32,000,000	0.32
10 MHz	10,000,000	32,000,000	3.2
1 MHz	1,000,000	32,000,000	32
100 kHz	100,000	32,000,000	320
10 kHz	10,000	32,000,000	3,200
1 kHz	1,000	32,000,000	32,000

Muy rara vez se puede denotar que los valores se alternan por pérdida de sincronización, pero esto se debe a que en el encendido ocurre una pequeña alteración la cual cambia el estado y provoca ciertos errores.

En la Figura 23, se puede ver los recursos utilizados en el Nanoboard, aquí se puede ver las conexiones internas de este, los recursos utilizados son los pines de entrada/salida, los cuales todos tienen una función asignada y se puede ver que se está conectado directamente a la pantalla táctil con la capacidad de poder cambiar los estados de este, al igual que las opciones previamente especificadas.

Para el funcionamiento directo del FPGA, el cual llevó más tiempo en su programación debido a que este se vuelve más complicado en la utilización de sus recursos, por ser una plataforma nueva, se configuró de la siguiente manera, como se muestra en la Figura 24. El FPGA es configurable en todo momento, por lo cual puede ser reconfigurado en cualquier momento para poderse arreglar errores de programación en cualquier momento. El FPGA está programado para que en su entrada capture las 16 entradas por medio de flip-flops D, los cuales capturan los datos en el cambio de flanco, de esta manera se asegura capturar los datos de una manera adecuada en los dos estados del multiplexor. Los relojes están siempre funcionando con el mismo del FPGA, asegurando la sincronización entre el Nanoboard y los multiplexores externos.

En la función del Trigger existe una condición, dependiendo de cuál esté seleccionada, entra al método donde él decide si empieza a monitorear los cambios y activarse, ya sea en cambios de flanco negativos o de positivos. El trigger está monitoreando cualquier cambio entre las señales, ya sean de las primeras 16 entradas o las segundas. El trigger está implementado de esta manera, pero este igual puede ser actualizado por el usuario con conocimiento del programa. Al entrar cualquiera de las señales este pasa al módulo de habilitación. Este módulo activa una señal la cual le dice que empiece a escribir en la

memoria los datos, pero esta está limitada a la condición de cuando se está leyendo de la memoria para poderse enviar al computador del usuario.

Cuando se envía la señal de disparo (trigger), y dependiendo de la configuración de la pantalla, la señal es dirigida al control de señales. Este módulo determina la organización de los datos en la memoria, incluyendo la posición en la que se activó el trigger y la posición final de almacenamiento. Luego, la información se extrae en el mismo orden y se envía al módulo de armado de la trama, donde es codificada utilizando el código fuente de la memoria. Finalmente, los datos se almacenan en un archivo, que es empaquetado y transmitido por USB para su exportación a la computadora.

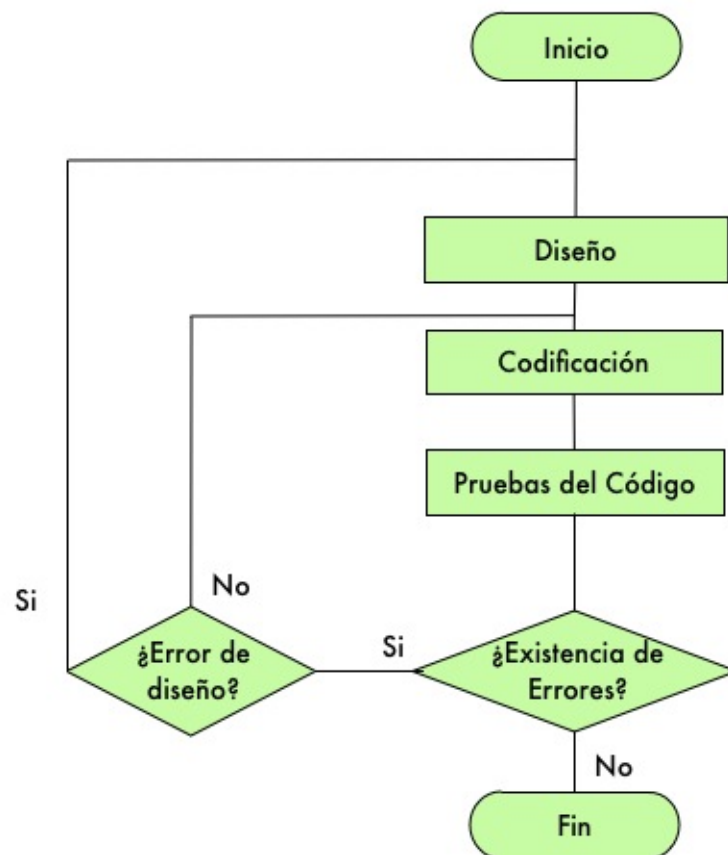
Las tasas de transferencias del USB, que posee el Nanoboard, son basadas en el USB 2.0, la cual tiene capacidad de mandar 450 Mb/s, lo que nos da un total de 56.25 MB/s pero al estar puesta en prueba obtenemos que la máxima cantidad de tasa de transferencia es de 20.8 MB/s. En un lapso de 2 segundos, la información se muestra en pantalla. Luego, el sistema reinicia el monitoreo del trigger y vuelve a enviar los datos de manera continua.

La fase de diseño de hardware fue creada por Juan Pablo Ruiz, para el trabajo de graduación Osciloscopio USB Multicanal módulo: Hardware. Las líneas muestran el inicio y el fin del texto e imágenes extraídos de ese documento.

F. DISEÑO SOFTWARE

El diseño del software fue realizado específicamente para ser utilizado con la arquitectura descrita en el módulo de hardware del presente trabajo de graduación. Se inicia este proceso con el análisis del módulo de hardware. Luego de revisar el hardware se inicia el proceso de desarrollo de software. La Figura 25 muestra el diagrama de flujo que demuestra este proceso.

Figura 25. Diagrama de flujo desarrollo de software



El diseño se inicia con la declaración de los estados de operación del software. El hardware cuenta con tres modos de operación generales, los cuales se muestran en el Cuadro 4. Las banderas de estado serán recibidas por medio serial.

Cuadro 4. Modos de operación del osciloscopio

Modo de operación	Descripción
Modo digital	Muestreo de señales digitales directamente con los puertos de entrada del FPGA, llegando hasta 32 canales simultáneamente.
Modo mixto	Este modo captura muestras de dos canales analógicos junto con 16 canales digitales.
Modo analógico	Se muestrean 4 canales analógicos simultáneamente.

Cuadro 5. Modos de operación del software

Estado	Descripción
Inicialización	Se realiza la inicialización de variables y se definen los valores iniciales del programa.
Elección de puerto	Se escoge el puerto de comunicación en el cual este conectado el Nanoboard 3000.
Espera de datos	Se espera a que el Nanoboard mande el paquete de información con los datos y la información de configuración.
Recepción de datos	En este modo se recibe la información de configuración que establece el modo de operación del osciloscopio al momento de tomar las muestras, junto con el paquete de muestras.
Interpretación de datos	Al saber en que modo se capturaron las muestras, se pueden interpretar y ordenar los datos recibidos. En este estado se separan los canales muestreados y se generan registros individuales por canal.
Despliegue de datos (Modo digital)	En este modo se despliegan secuencias de bits desde 1 hasta 32 canales a selección del usuario. Los 32 canales siempre son muestreados, el usuario solo decide cuales visualizar.
Despliegue de datos (Modo mixto)	El modo mixto es el más dinámico, ya que presenta hasta 16 canales digitales junto a 2 canales analógicos de alta frecuencia, lo cual sirve para identificación de integridad de señales.
Despliegue de datos (Modo analógico)	Se despliega desde 1 hasta 4 canales analógicos de alta frecuencia, en los cuales el usuario puede manipular la manera que visualiza la señal.

Estado	Descripción
Guardar onda	Modo en el cual se guardan los datos ordenados en el modo de interpretación, en donde se guarda la configuración actual y la hora en la cual el evento fue guardado.
Cargar onda	Carga un archivo con la información de la onda para su análisis.

Los posibles estados del software, junto con una breve descripción, se muestran en el Cuadro 5. El flujo del software se representa gráficamente en el diagrama de la Figura 26. Este es el ciclo principal del programa en el computador. Luego de este diagrama se muestra el diagrama para cada módulo principal de software.

Figura 26. Diagrama de flujo ciclo principal

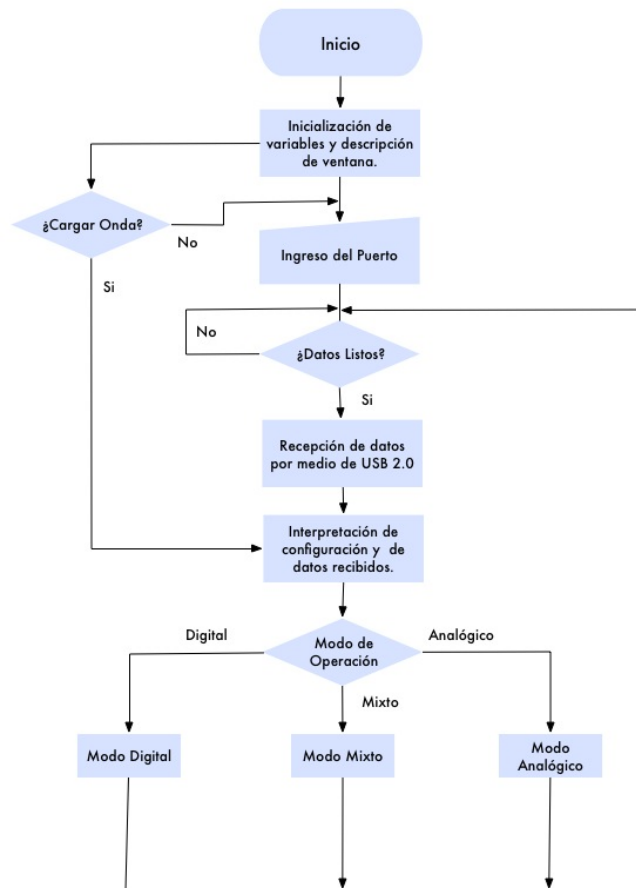


Figura 27. Diagrama de flujo módulo interpretador de control y datos.

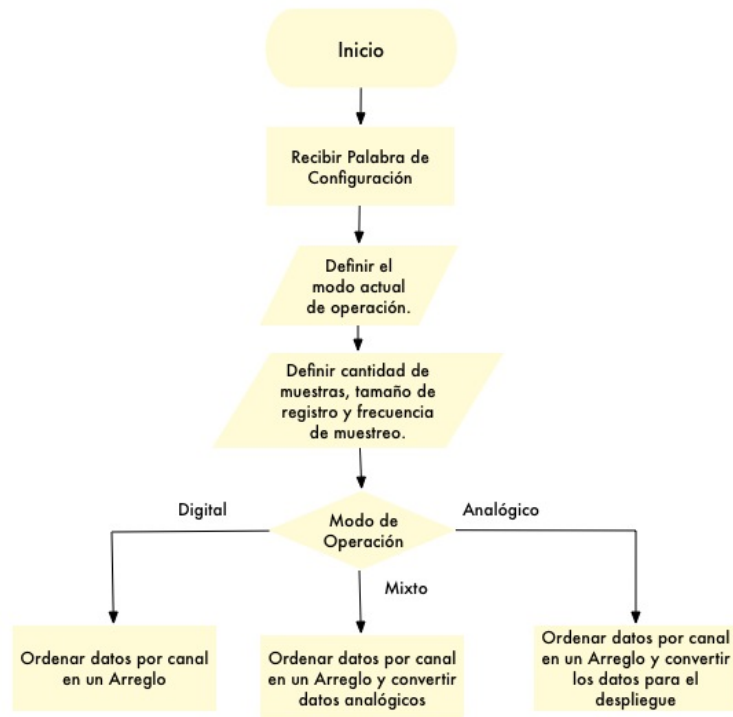


Figura 28. Diagrama de flujo módulo de muestras digitales.

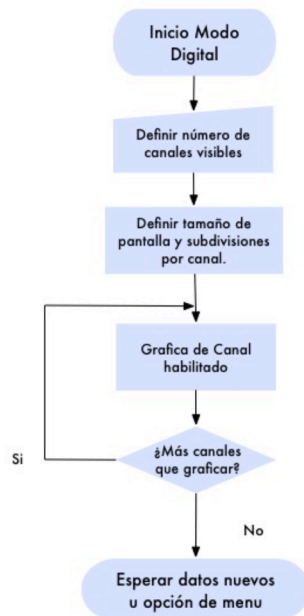


Figura 29. Diagrama de flujo módulo de muestras. mixtas

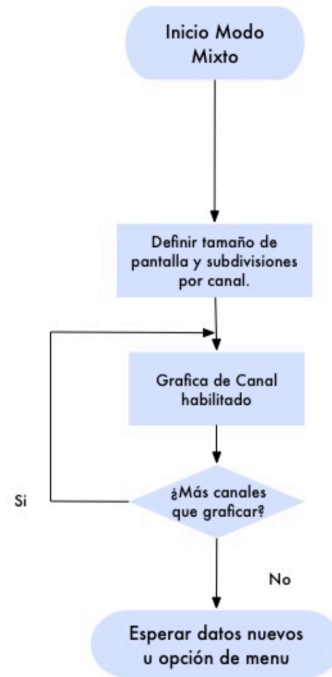
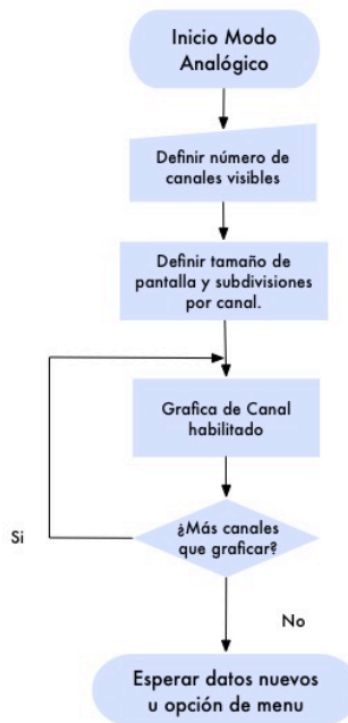


Figura 30. Diagrama de flujo módulo de muestras analógicas.



VII. DISCUSIÓN Y RESULTADOS

El programa resultante de los diagramas mostrados previamente logra representar las muestras digitales en pantalla. Las muestras obtenidas del módulo de hardware son interpretadas de manera diferente dependiendo del modo en las que se desean visualizar. Estas muestras provenientes del módulo de hardware pueden tener diferente origen. Dado al diseño de hardware, se pueden obtener datos digitales individuales por canal o datos en paralelo provenientes del módulo de ADC.

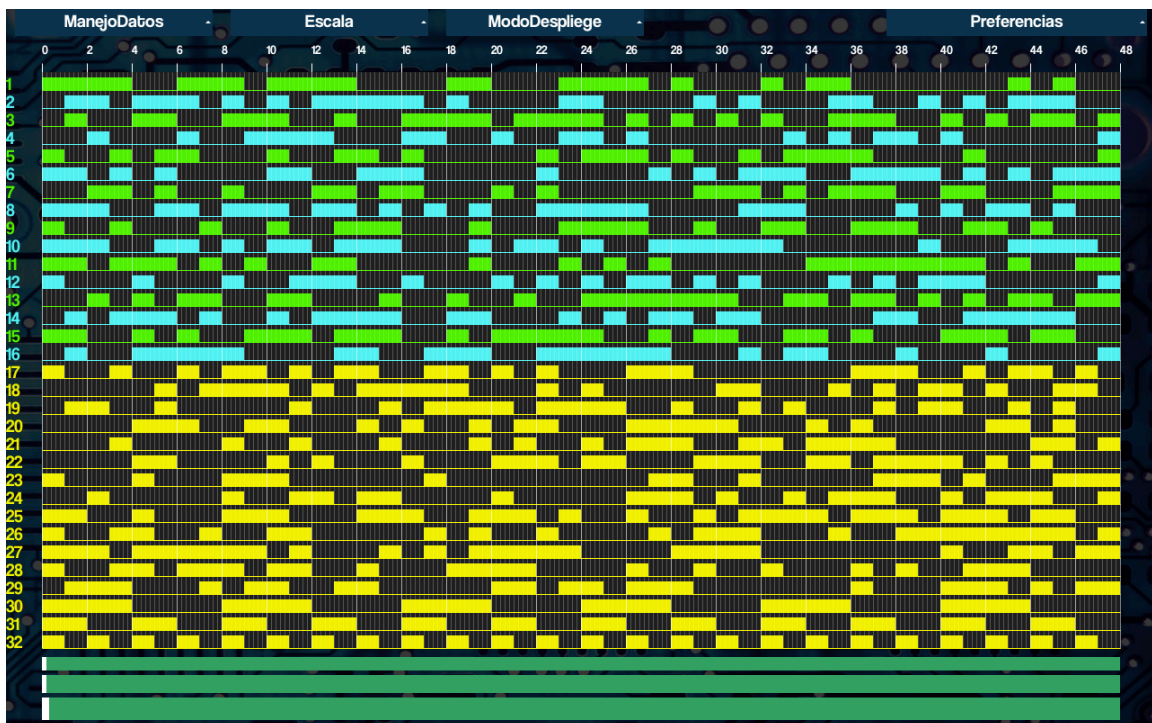
Figura 31. Modo inicial del software



En la Figura 31 se muestra el inicio de la aplicación al ser corrida, en la cual se puede esperar datos para graficar, cargar datos pasados o simular datos para aprender a utilizar el programa.

Cuando los datos provienen de canales digitales individuales muestreados en alta frecuencia, llamado modo digital, se logra tomar una fotografía a un momento en el tiempo iniciado por un trigger. Esta fotografía contiene el estado de los pines digitales muestreados a la frecuencia definida, lo cual aporta al usuario la posibilidad de realizar un análisis de tiempo para verificar sincronización y causalidad entre señales.

Figura 32. Modo digital del software



En la Figura 32 se muestra el modo digital de despliegue. Para este modo se genera una interfaz de usuario capaz de maximizar el espacio disponible de despliegue para datos en la pantalla.

Para representar un dato, se requiere al menos un píxel, lo que limita la cantidad de datos visibles simultáneamente por canal en la ventana de visualización. Definiremos el ancho de los píxeles utilizados para representar un dato digital como grosor de dato. Para observar el efecto del grosor de dato, en la Figura 33 se muestra el modo digital con un

grosor de 1 píxel por dato, mientras que en la Figura 34 se representa un grosor de 50 píxeles por dato.

Figura 33. Modo digital con grosor 1px

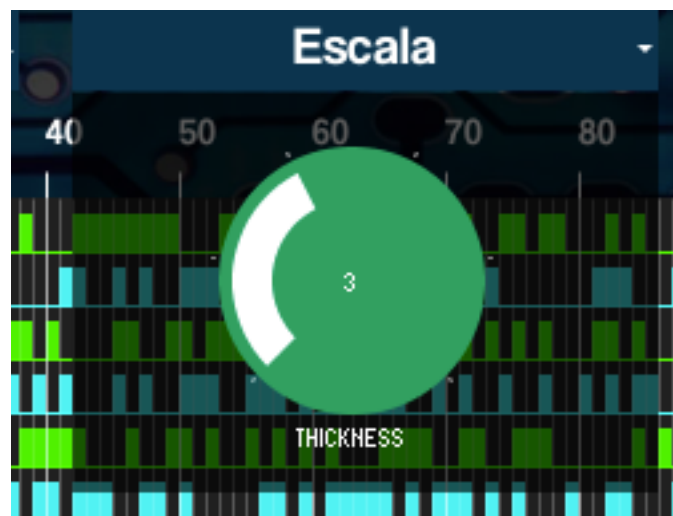


Figura 34. Modo digital con grosor 50px



El grosor de dato puede ser seleccionado por el usuario por medio del menú de escala, en el cual se presenta una perilla con valores para los distintos grosores. La selecciones varían desde un grosor de 1px hasta un grosor de 50 px por dato. Esto nos da la posibilidad de escoger entre ver una gran cantidad de datos a la vez o enfocarnos en una ventana de datos de menor tamaño para obtener con mayor detalle información de la señal. En la Figura 35 se muestra la perilla de selección de grosor.

Figura 35. Selector de grosor de dato



El tamaño de la ventana de datos visibles se define utilizando la resolución del monitor como referencia, de manera que siempre se logren visualizar la mayor cantidad de datos. Esto quiere decir que si se aumenta la resolución de la pantalla utilizada para desplegar los datos, se obtiene una mayor cantidad de datos visibles. Gracias a esto se logra ofrecer un modo de despliegue dinámico, que se adapta a la máquina en la cual se utiliza. En la Figura 36 se muestra el programa utilizado en una máquina con resolución de 1024x640px, mientras que en la Figura 37 se muestra con una resolución de 1440x900px.

Figura 36. Modo digital con resolución 1024x640px

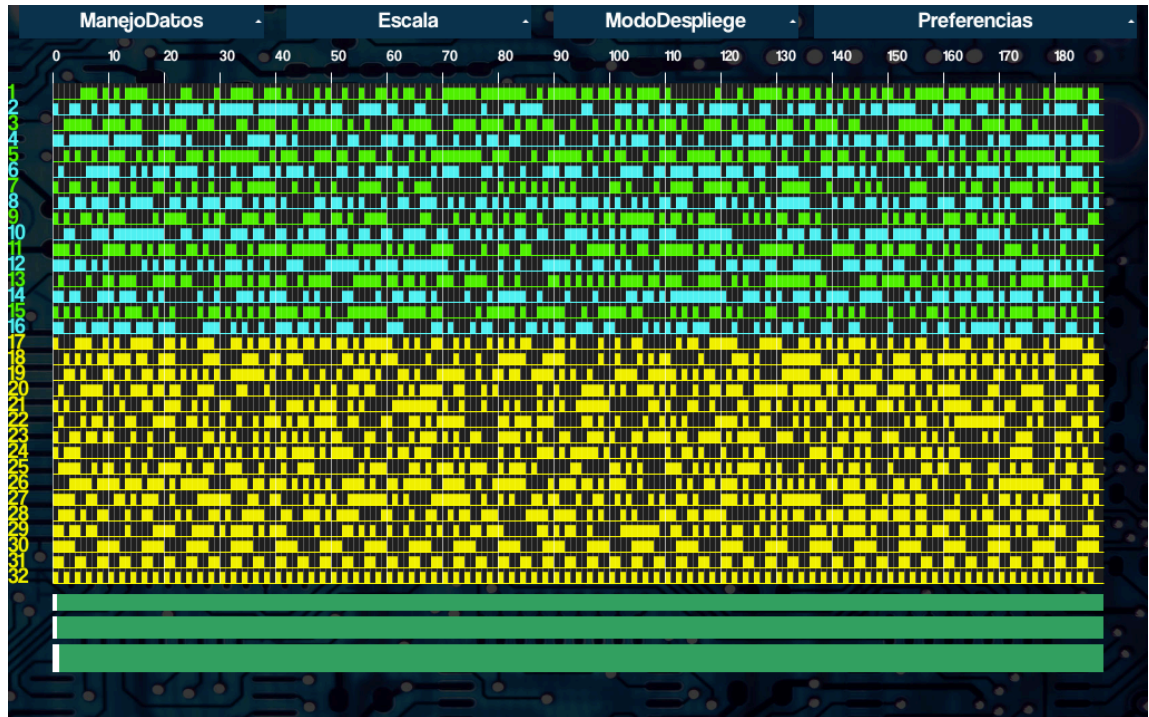
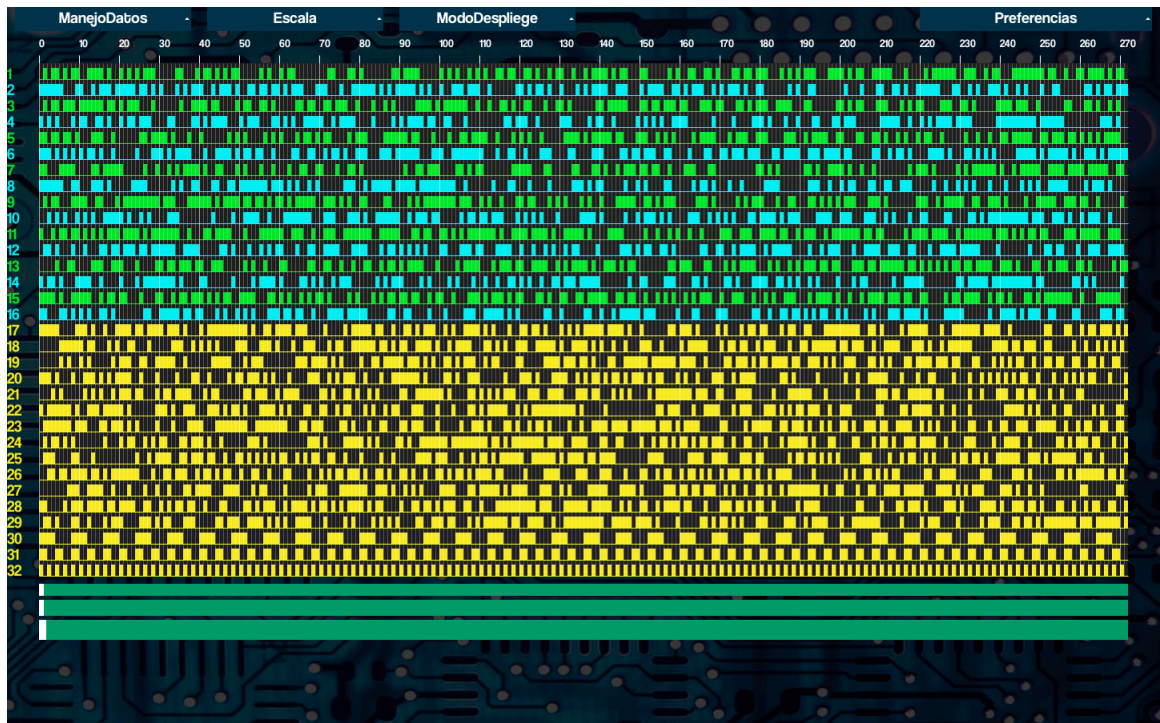


Figura 37. Modo digital con resolución 1440x900px



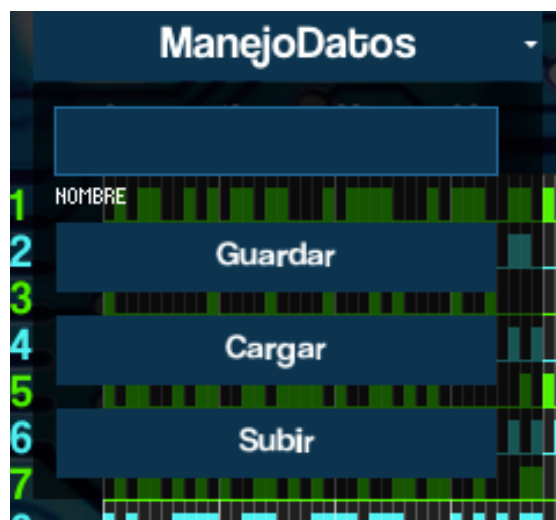
Para el control de la posición en la cual inicia la ventana se utilizan selectores de rango, con los cuales se realiza un barrido a diferentes tasas para escoger la posición inicial de las señales a desplegar.

Figura 38. Posición inicio de despliegue



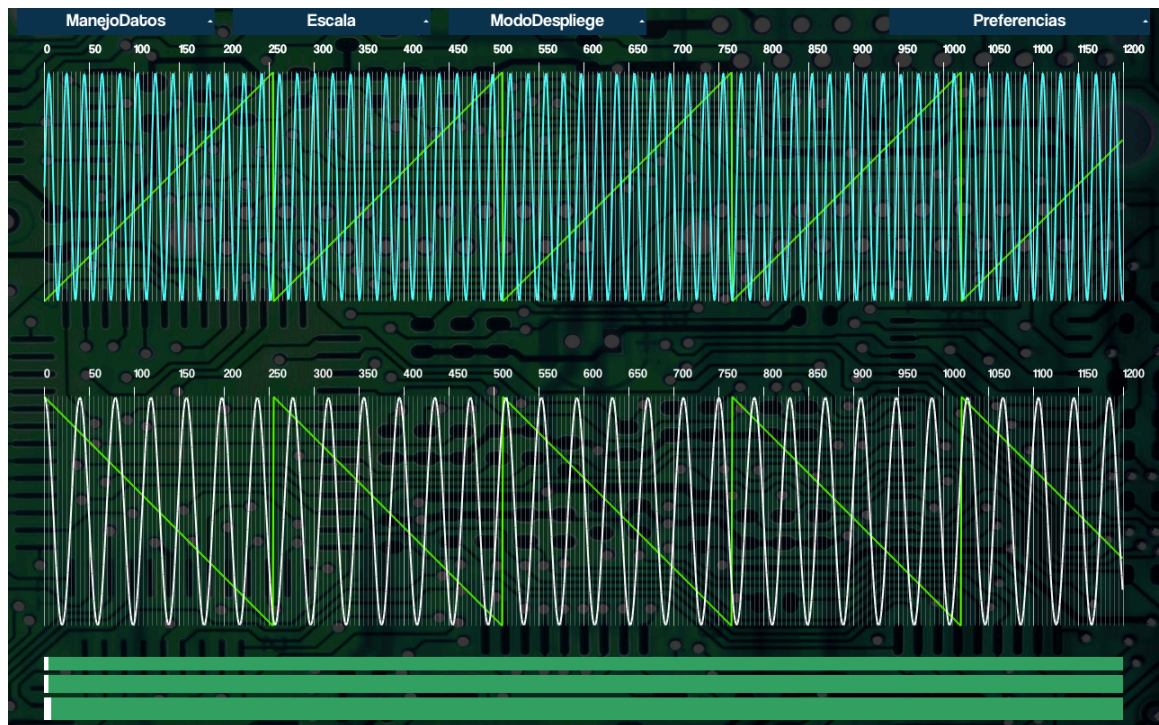
Si una onda es de interés y se desea guardar, se puede almacenar una copia local de las muestras en un archivo en formato CSV. Este archivo puede ser utilizado posteriormente para cargar los datos de las muestras al programa y analizar muestras pasadas. También se puede utilizar para ser compartido y que varios usuarios utilicen el programa para analizar las muestras en diferentes maquinas simultáneamente. Con esto en mente, se implementó una opción para subir el archivo a un servidor, para que compartir el archivo sea más fácil. La Figura 39 muestra el menú de manejo de datos, el cual nos pide el nombre del archivo a guardar o subir. También se puede cargar un archivo desde este menú.

Figura 39. Posición inicio de despliegue



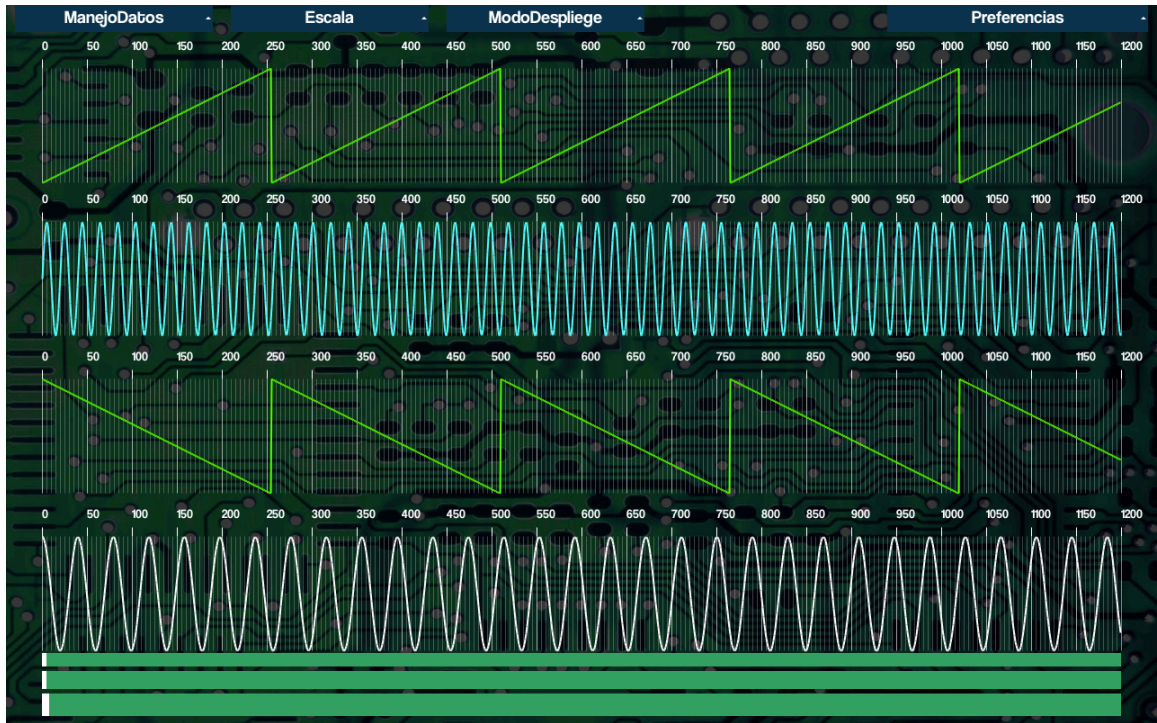
En el modo de despliegue analógico, el usuario puede visualizar 4 canales analógicos con resolución de 8 bits por canal. Al igual que en el modo digital, se aprovecha la mayor cantidad de tamaño de ventana. Se grafica la magnitud de la muestra en su equivalente decimal. Los 32 canales que antes teníamos disponibles para visualizar datos individuales ahora se utilizan para obtener 4 representaciones de señales convertidas. Estas señales se muestran en la Figura 40, en la cual se muestran 4 canales. El canal 1 y 2 se muestran en la parte superior y los canales 3 y 4 se muestran en la ventana de onda inferior.

Figura 40. Modo analógico



Este modo sirve para comparar señales provenientes de dos canales entre sí, pero también es posible representar los canales en ventanas individuales como se muestra en la Figura 41.

Figura 41. Modo Analógico 4 canales



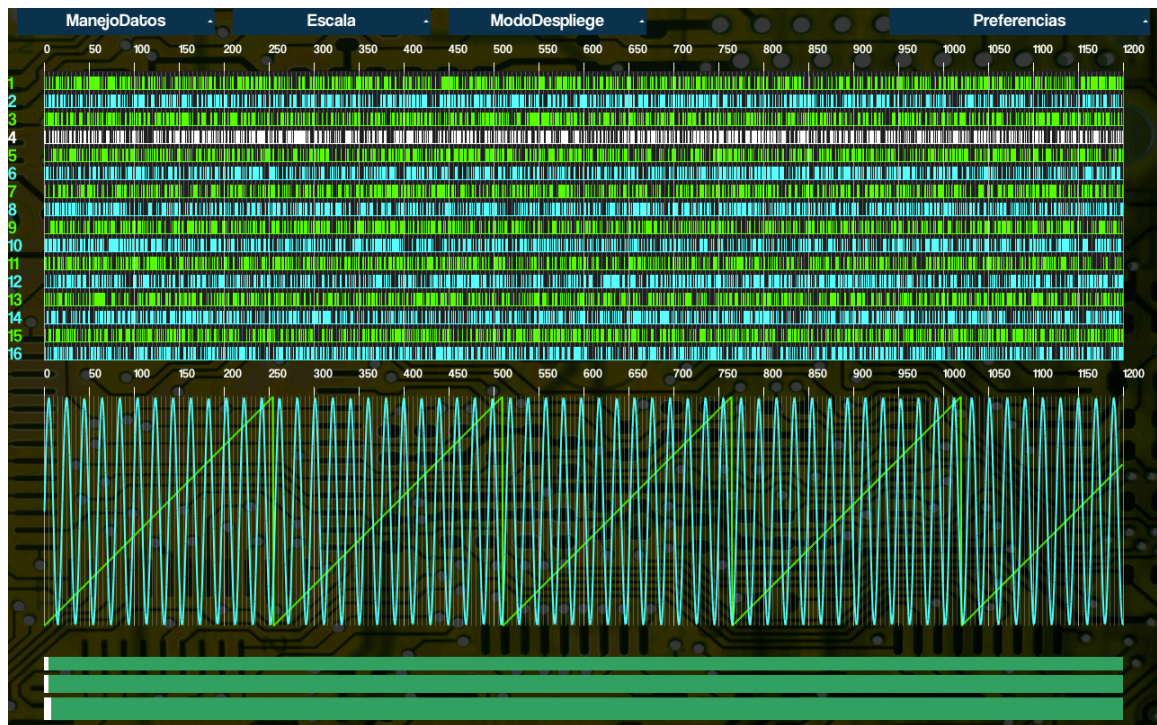
La capacidad de muestrear cuatro canales de forma simultánea permite analizar con alta precisión eventos transitorios que serían imperceptibles para el ojo humano. Esta funcionalidad es especialmente útil en el estudio de señales de alta frecuencia, detección de pulsos breves y análisis de comportamientos anómalos en circuitos electrónicos. Además, este módulo puede emplearse para la representación gráfica de datos provenientes de sensores, facilitando la visualización e interpretación de señales analógicas en aplicaciones de monitoreo y control.

Los datos analógicos capturados poseen una resolución de 8 bits, lo que permite representar hasta 256 niveles discretos (de 0 a 255) dentro del rango de medición. El sistema es capaz de muestrear señales con voltajes comprendidos entre -3 V y 10 V, lo que posibilita la evaluación de diversos parámetros, como el valor máximo de una señal o la duración temporal de eventos específicos dentro de la adquisición de datos.

Al unir el modo digital junto con el modo analógico obtenemos el modo de señal mixta. Este modo nos permite visualizar al mismo tiempo 16 canales digitales junto con 2 canales analógicos. Teniendo ambos tipos de datos visibles se puede obtener mayor información acerca de los niveles de voltaje que causan cambios en una señal.

En la Figura 42 se pueden observar 2 canales analógicos graficados en conjunto con 16 canales digitales, desplegados sincronizando las muestras de los datos para que su análisis en tiempo se facilite.

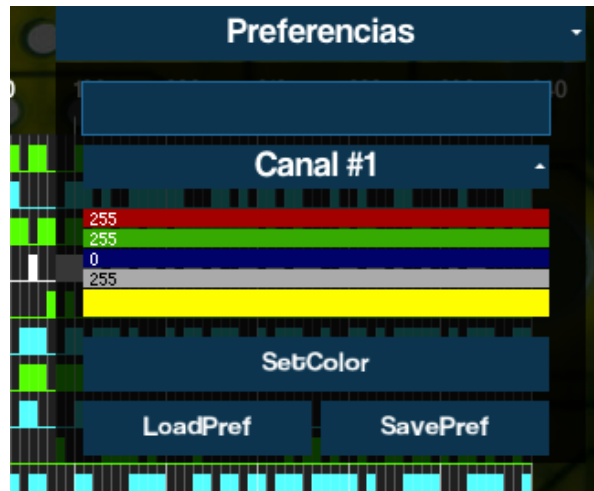
Figura 42. Modo mixto



Adicionalmente a desplegar y ordenar los datos de una manera que fueran entendibles para el usuario, se agregó un modo de preferencias. En este modo el usuario puede crear o cargar un archivo que contiene sus preferencias de despliegue. Entre estas preferencias se tiene el nombre de canal, y el color de canal individual. Esto facilita el entendimiento de los datos si es que estos son cargados en el futuro, o si estos son compartidos.

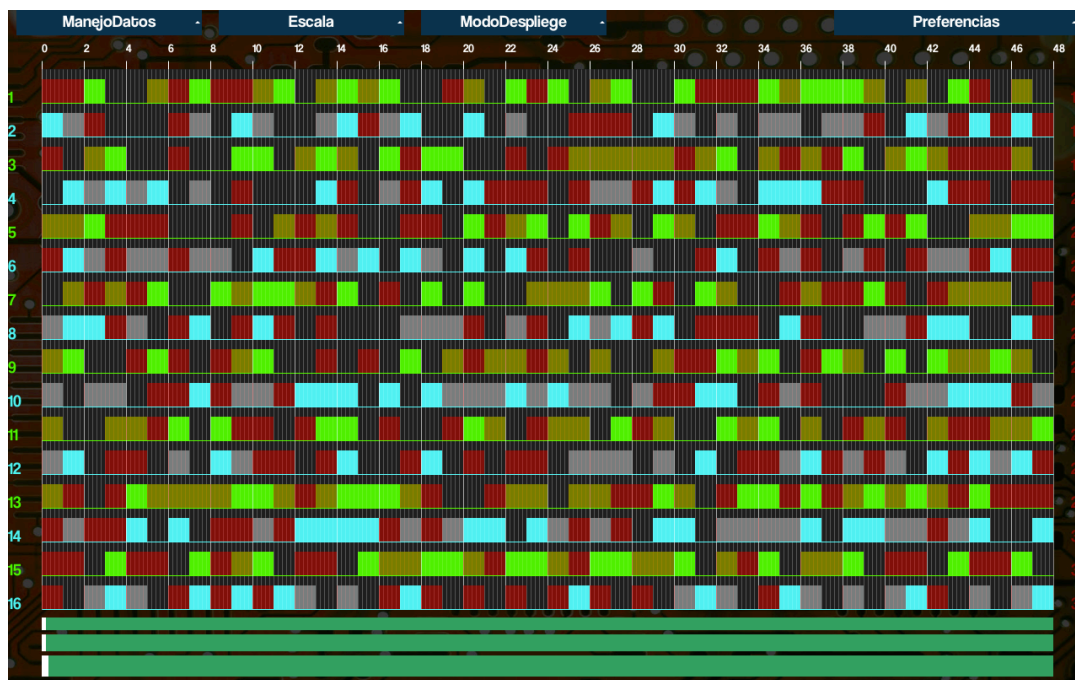
En la Figura 43 se muestra este modo, en el cual se puede escoger el color de cada canal por medio de 4 barras, las cuales definen la cantidad de color rojo, verde y azul junto con la transparencia.

Figura 43. Modo mixto



Como opción extra el usuario puede comparar los datos de distintos canales digitales, como se muestra en la Figura 44. En la cual se compara el dato por medio del uso de colores con transparencia, mostrándose de diferente color cuando el dato existe en ambos canales.

Figura 44. Modo mixto



El usuario tiene la opción de guardar una imagen de lo desplegado en pantalla, como segunda propuesta para compartir la información muestreada. Se puede definir el nombre de la imagen a guardar para su mejor entendimiento y orden. Esto facilita la adquisición de imágenes de datos muestreados, lo cual es comúnmente utilizado en reportes técnicos.

Para el control de eventos se utilizó la librería para Processing, CP5. Esta librería nos auxilia en la creación de objetos para el control de la aplicación por medio de la interfaz gráfica. Adicionalmente se utilizó FTPClient para poder subir la tabla de datos al servidor remoto.

El programa es independiente del hardware, lo que permite modificar su código para adaptarlo a distintos dispositivos de adquisición de datos. Esto facilita su uso como una herramienta flexible, multiplataforma y de código abierto, compatible con diversos sistemas.

VIII. CONCLUSIONES

- Se desarrolló un software capaz de visualizar hasta 32 canales digitales de forma simultánea, permitiendo un análisis detallado de señales de alta velocidad.
- Se implementó una solución que permite el despliegue de 4 canales analógicos, facilitando la observación y análisis de señales eléctricas con variaciones continuas.
- Se diseñó una interfaz de usuario intuitiva y fácil de utilizar, optimizada para la visualización y manipulación de señales mixtas en tiempo real.
- Se utilizó el Nanoboard 3000 como interfaz principal, permitiendo la conversión y transmisión eficiente de señales de alta frecuencia hacia la computadora para su procesamiento y análisis.
- El sistema desarrollado permite al usuario definir el tiempo de visualización y la frecuencia de muestreo directamente desde el Nanoboard 3000, ofreciendo flexibilidad en la captura y análisis de señales.
- Se aportó una herramienta funcional al grupo educativo de la Universidad del Valle de Guatemala, proporcionando un dispositivo útil para el estudio y resolución de problemas en proyectos que requieren la visualización simultánea de múltiples secuencias de bits.

IX. RECOMENDACIONES

- Para verificar el ancho de banda del osciloscopio, se recomienda el uso de un generador de funciones, permitiendo evaluar su respuesta en diferentes frecuencias.
- Se sugiere utilizar una computadora con una resolución mínima de 1280×800 píxeles para garantizar una visualización clara y detallada de las señales en pantalla.
- Para asegurar una comunicación de datos estable y de alta velocidad, se recomienda emplear un puerto USB 2.0 o superior.
- En el muestreo de señales de alta frecuencia, es fundamental el uso de una sonda de medición adecuada para minimizar pérdidas y garantizar la precisión en la captura de datos.
- Se aconseja realizar pruebas extensivas sobre el sistema con diferentes configuraciones y condiciones operativas, con el fin de detectar posibles mejoras y optimizaciones.
- Dado que el software desarrollado es de código abierto y libre de licencias, se recomienda su estudio y posible expansión para futuras mejoras o adaptaciones a nuevas aplicaciones.
- El software podría optimizarse para ser compatible con otros dispositivos de medición, ampliando su aplicabilidad en diferentes entornos de ingeniería.

X. BIBLIOGRAFÍA

1. Altium Designer. Altium, *Overview Altium Designer Software*, 2014. [Online]. Disponible: <http://www.altium.com/en/products/altium-designer>
2. Altium Live. Altium, *Online Community*, 2014. [Online]. Disponible: <https://live.altium.com>
3. Oppenheim, A. y Willsky, A. (1997). *Señales y Sistemas*. 2da ed. Prentice Hall, pp. 1-56.
4. Analyzer Software. Picoscope, *Oscilloscope and Spectrum Analyzer Software*, 2013. [Online]. Disponible: <http://www.picotech.com/picoscope-oscilloscope-software.html>
5. *Arduino Multi-Channel Oscilloscope (Poor Man's Oscilloscope)*. Instructables, Rui Santos, 2013. [Online]. Disponible: <http://www.instructables.com/id/Arduino-Multi-Channel-Oscilloscope-Poor-Mans-O/>
6. *Clasificación y Propiedades de las Señales*. Connexions, Melissa Selik, Richard Baraniuk, Michael Haag, Ricardo von Borries, 2006. [Online]. Disponible: <http://cnx.org/content/m12818/latest/>
7. *DAQ32 – 32 Channel Oscilloscope & Data Logger*. X-io Technologies, 2013. [Online]. Disponible en: <http://www.x-io.co.uk/daq32/>
8. *Digital Oscilloscope*. FPGA4Fun.com, Digital Oscilloscope, 2013. [Online]. Disponible en: <http://www.fpga4fun.com/digitalscope.html>

9. KNJN, Flashy acquisition boards. KNJN, *Flashy Boards*, 2013. [Online]. Disponible en: <http://www.knjn.com/docs/KNJN%20Flashy%20boards.pdf>
10. *Girino – Fast Arduino Oscilloscope*. Instructables, Girino – Fast Arduino Oscilloscope, 2012. [Online]. Disponible en: <http://www.instructables.com/id/Girino-Fast-Arduino-Oscilloscope/#step0>
11. J.P. Ruiz, “*Osciloscopio USB Multicanal módulo: Hardware*”, Universidad del Valle de Guatemala, Trabajo de Graduación, 2014, pp. 4-36.
12. *Oscilloscope Fundamentals*. Tektronix, 2009. [Online]. Disponible en: <http://www.tek.com/learning/oscilloscope-tutorial>
13. *Physics III: Vibrations and Waves*. MIT, Nelson Thornes, 2004. [Online]. Disponible en: <http://ocw.mit.edu/courses/physics/8-03-physics-iii-vibrations-and-waves-fall-2004/>
14. *Processing*. Processing, B. Fry, C. Reas, D. Shiffman, 2014 [Online]. Disponible en: <http://www.processing.org/overview/>
15. R. Priemer, “*Introductory Signal Processing*”, por World Scientific Publishing Co, 1991, p. 1.
16. *Signal Integrity*. Fluke, Support, Signal Integrity, 2004. [Online]. Disponible en: <http://www.fluke.com/fluke/sgen/support/library/signal-integrity.htm>
17. University of Maryland. (2004). *Signal Integrity Analysis*. [Online]. Disponible en: http://www.csee.umbc.edu/csee/research/vlsi/reports/si_chapter.pdf

18. Tektronix. (2014). *Digital Oscilloscope*. [Online]. Disponible en: <http://www.tek.com/digital-oscilloscope>

19. National Instruments. (2011). *Top 10 Things to Consider When Selecting a Digitizer/Oscilloscope*. [Online]. Disponible en: <http://www.ni.com/white-paper/4333/en/>

20. Xoscillo. Google Code, *xoscillo*, 2014. [Online]. Disponible en: <https://code.google.com/p/xoscillo/>

XI. GLOSARIO

Algoritmo: conjunto ordenado y finito de operaciones que permite hallar la solución de un problema.

Ancho de banda: velocidad de muestreo, dada en hz por segundo.

Amplificación: incremento de amplitud de una señal durante la transmisión de esta de un punto a otro.

Amplitud: la cantidad de magnitud o la fuerza de una señal. En electrónica, amplitud generalmente se refiere al voltaje o potencia.

Atenuación: decremento de amplitud de una señal durante la transmisión de esta de un punto a otro.

Ciclo principal: se refiere al ciclo en el cual el programa se mantiene durante el periodo de operación, luego de ser inicializado.

Diagrama: representación gráfica de un proceso.

Fase: la cantidad de veces que pasa sobre un determinado tiempo un ciclo en el inicio del siguiente ciclo, medido en grados.

Fpga: acrónimo en ingles para Field Programmable Gate Array, el cual es un circuito integrado utilizado para describir hardware.

Frecuencia: número de veces que la señal se repite en un segundo. Medido en Hertz.

Osciloscopio: instrumento electrónico de visualización de voltaje sobre un tiempo determinado.

Señal analógica: una señal con voltajes variables continuos.

Señal digital: una señal que sus muestras son representadas por números binarios discretos.