

CubeSat Fase 4

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ingeniería




CubeSat Fase 4

Trabajo de graduación en modalidad de Megaproyecto presentado por Marvin Alexis Martínez Pérez y Daniel Alejandro García Orellana para optar al grado académico de Licenciados en Ingeniería Mecánica; Diego Alejandro Rodríguez García para optar al grado académico de Licenciado en Ingeniería Electrónica; y Luis Humberto Duarte Rodríguez y Daniel Alejandro Orozco Ramírez para optar al grado académico de Licenciados en Ciencia de la Computación y TI

Guatemala,


2017

Vo.Bo.:

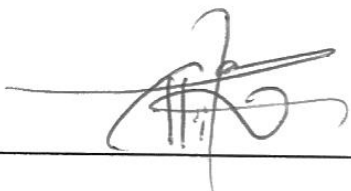
(f) 

Ing. Víctor Ayerdi


Aprobado por:

(f) 

Ing. Víctor Ayerdi

(f) 

MSc. Carlos Esquit

(f) 

Ing. Douglas Barrios

Fecha de aprobación: Guatemala, diciembre de 2017

CONTENIDO

Lista de cuadros	xi
Lista de figuras.....	xvi
Lista de acrónimos	xxii
Resumen	xxiii
I. Introducción	1
II. Objetivos.....	2
A. Objetivo general del megaproyecto.....	2
B. Objetivo general del módulo de ingeniería en sistemas y gestión de proyectos	2
C. Objetivos específicos del módulo de ingeniería en sistemas y gestión de proyectos.....	2
D. Objetivo general del módulo de potencia	2
E. Objetivos específicos del módulo de potencia	2
F. Objetivo general del módulo de manejo de comandos y datos.....	3
G. Objetivos específicos del módulo de manejo de comandos y datos	3
H. Objetivo general del módulo de control activo	3
I. Objetivos específicos del módulo de control activo	3
III. Justificación	4
IV. Marco teórico	5
A. Cubesat	5
B. Ingeniería de sistemas.....	6
C. Análisis de riesgos.....	7
D. Requerimientos de la misión.....	11
E. Verificación y validación.....	21
F. Materiales	22
G. Recolección de energía.....	23
1. Eficiencia.	24
2. Área.	24
3. Constante solar.	24
4. Ángulo de incidencia.	25

5.	Efectos de temperatura.....	25
6.	Simulaciones de generación de potencia.....	26
H.	Módulos dependientes de potencia.....	26
1.	Carga útil.....	26
2.	Manejo de datos y comandos.....	27
3.	Comunicaciones.	27
4.	Térmico.....	27
5.	Sistema de determinación y control de actitud.	27
I.	Almacenamiento de energía.....	28
1.	Tecnología de baterías.	28
2.	Familia de baterías de Litio.	28
3.	Capacidad requerida.	29
4.	Límites de profundidad de descarga.	30
5.	Modos de falla de las baterías.....	33
J.	Distribución de energía.....	35
K.	Sistema de manejo de comandos y datos.....	35
1.	Computadora a bordo.....	35
2.	Arquitectura de hardware.....	36
3.	Arquitectura de software.....	36
L.	I2C.....	37
M.	UART.....	39
N.	UML.....	39
1.	Diagrama de secuencia.....	40
2.	Diagrama de actividad.	40
O.	Interferencia electromagnética.....	40
P.	Cinemática.....	40
1.	Vectores físicos.	41
2.	Marco de referencia.	41
3.	Matrices de rotación.....	41

a.	Rotaciones principales.	42
b.	Ángulos de Euler.	42
c.	Cuaterniones.	43
4.	Derivaciones de vectores.	43
a.	Velocidad angular.	44
b.	Velocidad y aceleración.	45
Q.	Dinámica.	45
1.	Dinámica de un cuerpo rígido.	45
a.	Dinámica traslacional.	45
b.	Dinámica rotacional.	46
c.	Energía cinética.	47
R.	Perturbaciones en el espacio.	47
1.	Torque magnético.	48
2.	Torque por presión de radiación solar.	48
3.	Torque aerodinámico.	49
4.	Torque por gradiente de gravedad.	50
S.	Estabilización de posición de un satélite.	50
1.	Ruedas de inercia.	51
T.	Estabilidad en un sistema.	52
1.	Función de transferencia del satélite.	53
2.	Modelado del sistema.	54
3.	Teorema de estabilidad de Lyapunov.	55
U.	Diseño de control.	55
1.	Especificaciones en el dominio del tiempo.	57
a.	Especificaciones de la respuesta transitoria.	57
b.	Especificaciones del estado estable.	57
2.	Tipos de controladores.	57
a.	Proporcional integral.	57
b.	Proporcional derivativo.	58

c.	Lead.....	58
d.	Lag.....	58
e.	PID.....	59
f.	Regulador lineal cuadrático.....	59
V.	Metodología.....	60
A.	Análisis de riesgos.....	60
B.	Requerimientos relacionados a la misión.....	62
C.	Requerimientos de diseño.....	65
D.	Requerimientos operacionales.....	68
E.	Pruebas.....	70
F.	Prueba de integración.....	76
1.	C&DH.....	77
2.	Potencia.....	78
3.	ADCS.....	78
4.	Comunicación.....	79
5.	Carga útil.....	79
6.	Cronograma.....	79
G.	Información operacional de los componentes.....	80
H.	Factibilidad de diseño de potencia propuesto en fase 3.....	81
1.	Generación de potencia eléctrica.....	81
2.	Consumo de potencia eléctrica por módulo.....	84
3.	Consumo de potencia eléctrica por modo de operación.....	85
I.	Requerimientos funcionales del módulo de potencia.....	87
J.	Análisis de almacenamiento de energía.....	89
K.	Selección de componentes del módulo de potencia.....	89
L.	Interfaces del módulo de potencia.....	91
M.	Riesgos del módulo de potencia.....	94
N.	Factibilidad del diseño de potencia de fase 4.....	94
1.	Generación de potencia eléctrica.....	95

2.	Consumo de potencia eléctrica por módulo.....	99
3.	Consumo de potencia eléctrica por modo de operación.....	100
O.	Procedimientos de ensamblaje del módulo de potencia.....	101
1.	Paneles solares.....	101
2.	Baterías.....	103
P.	Pruebas del módulo de potencia.....	105
1.	Pruebas para paneles solares.....	105
2.	Pruebas para baterías.....	107
3.	Pruebas para sistema integrado.....	107
Q.	Módulo de manejo de comandos y datos.....	108
R.	Diseño Módulo de manejo de comandos y datos.....	110
1.	Arquitectura de hardware.....	110
a.	Arquitectura centralizada.....	110
b.	Selección OBC.....	111
c.	Interfaces eléctricas.....	112
1)	Térmico.....	112
2)	Energía.....	113
3)	ADCS.....	113
4)	Carga útil.....	114
5)	Comunicaciones.....	115
d.	Bus de datos.....	115
2.	Arquitectura de software.....	116
a.	Descripción de casos de uso.....	116
b.	Diagramas de flujo.....	118
c.	Diagramas de secuencia.....	122
d.	Diagramas de actividad.....	128
e.	Protocolo de comunicación.....	131
1)	Magnetómetro HMC5883L.....	132
2)	Convertidor analógico a digital ADC128D818.....	133

3.	Redundancia.....	135
a.	Diseño.....	135
b.	Implicaciones.....	137
c.	Desarrollo.....	138
d.	Implementación.....	138
e.	Pruebas.....	138
S.	Desarrollo e implementación del sistema de manejo de comandos y datos	139
T.	Pruebas de manejo de comandos y datos	139
1.	Componentes.....	139
a.	Pruebas de consumo energético.....	139
b.	Pruebas de interferencias electromagnéticas.....	141
2.	Funcionales.....	142
3.	Tiempo de ejecución de acciones.....	143
4.	Estrés.....	143
U.	Parámetros simulación del módulo de control activo	144
1.	Momentos de inercia.....	144
2.	Torque magnético.....	144
3.	Torque gravitacional.....	145
4.	Torque aerodinámico.....	146
5.	Torque de presión por radiación solar.....	146
6.	Resumen de magnitudes.....	147
V.	Controladores.....	148
W.	Simulación en MATLAB del módulo de control activo	149
X.	Interfaz de pruebas físicas.....	150
1.	Puente H.....	150
2.	Unidad de mediciones inerciales.....	151
3.	Microcontrolador.....	152
4.	Motor DC.....	152
5.	Estructura.....	153

VI.	Análisis de resultados	155
A.	Riesgos del proyecto	155
B.	Características operacionales de los componentes	157
C.	Factibilidad de diseño de potencia en fase 3	166
D.	Análisis de almacenamiento de energía.....	169
E.	Selección de componentes del módulo de potencia.....	170
F.	Sistema de distribución de energía.....	171
G.	Riesgos del módulo de potencia.....	173
H.	Factibilidad de diseño del módulo de potencia de fase 4.....	177
I.	Sistema de manejo de comandos y datos.....	180
J.	Simulación de orientación del satélite	211
K.	Estabilidad del sistema	212
L.	Controlador PID.....	213
M.	Controlador regulador lineal cuadrático (LQR)	214
N.	Perturbaciones.....	218
O.	Plataforma de pruebas físicas.....	221
VII.	Conclusiones	225
A.	Ingeniería de sistemas y gestión de proyectos.....	225
B.	Módulo de potencia.....	225
C.	Módulo de manejo de comandos y datos	227
D.	Módulo de control activo.....	228
VIII.	Recomendaciones	229
A.	Ingeniería de sistemas y gestión de proyectos.....	229
B.	Módulo de potencia.....	231
C.	Módulo de manejo de comandos y datos	232
D.	Módulo de control activo.....	232
IX.	Bibliografía	234
X.	Anexos	243
A.	Requerimientos de CAL POLY	243

B.	Planos e imágenes por JAXA y CAL POLY	245
C.	Código fuente de manejo de comandos y datos	247
1.	Código para lectura de datos de ADCS.....	247
2.	Código para lectura de datos del ADC128D818.....	248
3.	Código para lectura de datos del HMC5883L Driver.	253
4.	Código para manejo de la carga útil.....	256
5.	Código para manejo de la cámara.....	257
6.	Código para manejo del motor.	259
7.	Código para manejo del driver del motor.....	260
8.	Simulación HMC5883L Arduino	260
9.	Simulación ADC128D818 Arduino.....	264
10.	Código para lectura de componentes del módulo de energía y traspaso de información al módulo de comunicaciones.....	267
11.	Código para mecanismo de redundancia.....	284
12.	Código desarrollado para reiniciar computadoras para pruebas del mecanismo de redundancia	293
D.	Código de simulación del sistema no lineal y controladores.....	294
1.	Programa principal - controladores PID y LQR	294
2.	Función para el controlador PID.....	302
3.	Función del modelo no lineal del satélite	303
4.	Función de la perturbación aerodinámica.....	303
5.	Función de la perturbación gravitacional	303
6.	Función de la perturbación de presión por radiación solar	304
7.	Función de la perturbación magnética	304
8.	Función para animación del CubeSat.....	304
9.	Matriz skew 3D	305
10.	Matriz skew 4D	305
11.	Programa para desarrollo de la matriz del control LQR.....	305
E.	Código fuente de la plataforma física de control.....	307

LISTA DE CUADROS

	Página
Cuadro 1. Capacidad de baterías de Li-ion al almacenarse a diversas temperaturas.	32
Cuadro 2. Ciclos de carga/descarga en función de profundidad de descarga.	32
Cuadro 3. Tipos de arquitectura de hardware.	36
Cuadro 4. Cálculo de ponderación de los riesgos.	60
Cuadro 5. Requerimientos de la misión.	63
Cuadro 6. Requerimientos de diseño del satélite.	65
Cuadro 7. Requerimientos operacionales de la misión	68
Cuadro 8. Prueba para medición del momento magnético de la barra magnética.	71
Cuadro 9. Prueba de mediciones de la barra de histéresis.	71
Cuadro 10. Prueba de transferencia de imágenes.	71
Cuadro 11. Prueba de cambio entre computadoras.	72
Cuadro 12. Prueba de EPS.	72
Cuadro 13. Prueba de integración de sub-módulos EPS con baterías y celdas solares.	72
Cuadro 14. Prueba de ensamble de integración de EPS a las baterías y celdas solares.	72
Cuadro 15. Prueba de EPS y cargas integradas a la estructura del satélite.	73
Cuadro 16. Prueba a nivel de sistema.	73
Cuadro 17. Pruebas de termo-vacío y de vibraciones.	73
Cuadro 18. Pruebas con motor, eje y carrusel de filtros.	73
Cuadro 19. Prueba de integración de la carga útil.	74
Cuadro 20. Prueba de medición de patrones de radiación de la antena.	74
Cuadro 21. Prueba de medición de ganancia de la antena.	74
Cuadro 22. Prueba de medición de fase de la antena.	74
Cuadro 23. Prueba de medición de la polarización de la antena.	75
Cuadro 24. Pruebas de medición de impedancia de la antena.	75
Cuadro 25. Prueba con el módulo electrónico para comunicación.	75

Cuadro 26. Prueba de enlace de datos entre el módulo de comunicación y la estación en Tierra.	75
Cuadro 27. Duraciones de luz, penumbra y umbra obtenidos de simulación de STK.	82
Cuadro 28. Condiciones límite del perfil luz/sombra del satélite.	83
Cuadro 29. Características de celdas solares 3G30A de AZUR SPACE.	83
Cuadro 30. Características eléctricas de ADCS fase 3.....	84
Cuadro 31. Características eléctricas de PLD fase 3.....	84
Cuadro 32. Características eléctricas de C&DH fase 3.....	85
Cuadro 33. Características eléctricas de COMMS fase 3.....	85
Cuadro 34. Consumo de potencia durante vuelo con luz.	86
Cuadro 35. Consumo de potencia durante vuelo con luz sobre Guatemala.....	86
Cuadro 36. Consumo de potencia durante vuelo con luz sobre estación terrena.	86
Cuadro 37. Consumo de potencia durante vuelo con sombra.....	87
Cuadro 38. Requerimientos funcionales, sistema de recolección de energía.	87
Cuadro 39. Requerimientos funcionales, sistema de almacenamiento de energía.	88
Cuadro 40. Requerimientos funcionales, sistema de distribución de energía.....	88
Cuadro 41. Especificaciones de celdas solares.....	89
Cuadro 42. Normalización de variables de celdas solares.....	90
Cuadro 43. Especificaciones de baterías.....	90
Cuadro 44. Normalización de variables de baterías.....	91
Cuadro 45. Interfaces mecánicas.	91
Cuadro 46. Interfaces eléctricas.	92
Cuadro 47. Interfaces de programación.	93
Cuadro 48. Interfaces de servicios suministrados.....	93
Cuadro 49. Criterio de probabilidad.	94
Cuadro 50. Criterio de consecuencia.	94
Cuadro 51. Perfil de luz/sombra del satélite utilizado para análisis de fase 4.	98
Cuadro 52. Características eléctricas de ADCS fase 4.....	99
Cuadro 53. Características eléctricas de PLD fase 4.....	99
Cuadro 54. Características eléctricas de C&DH fase 4.....	99

Cuadro 55. Características eléctricas de COMMS fase 4.....	100
Cuadro 56. Consumo de potencia durante vuelo.	100
Cuadro 57. Consumo de potencia durante vuelo sobre objetivo.....	100
Cuadro 58. Consumo de potencia durante vuelo sobre Guatemala.....	100
Cuadro 59. Listado de materiales para ensamble de paneles solares.	101
Cuadro 60. Listado de materiales para ensamble de baterías.	104
Cuadro 61. Criterios de éxito para misión.....	108
Cuadro 62. Requerimientos de misión para manejo de comandos y datos.....	109
Cuadro 63. Requerimientos funcionales.	109
Cuadro 64. Caso de uso: recolectar información de fotodiodos.	116
Cuadro 65. Caso de uso: recolectar información de magnetómetro.	116
Cuadro 66. Caso de uso: validar y almacenar información de fotodiodos.	117
Cuadro 67. Caso de uso: validar y almacenar información de magnetómetro.....	117
Cuadro 68. Caso de uso: tomar fotografía.....	117
Cuadro 69. Caso de uso: mover carrusel.....	117
Cuadro 70. Lista de registros para magnetómetro HMC5883L.....	132
Cuadro 71. Especificación del registro de configuración A del chip HMC5883L.....	132
Cuadro 72. Especificación del registro de configuración B del chip HMC5883L.....	132
Cuadro 73. Especificación del registro de modo del chip HMC5883L.....	132
Cuadro 74. Lista de registros para el convertidor analógico a digital ADC128D818.	133
Cuadro 75. Especificación del registro de configuración avanzada del chip ADC128D818.....	134
Cuadro 76. Especificación del registro de tasa de conversión del chip ADC128D818.....	134
Cuadro 77. Especificación del registro deshabilitar canales del chip ADC128D818.	134
Cuadro 78. Especificación del registro de configuración del chip ADC128D818.....	134
Cuadro 79. Inercias de los ejes principales del CubeSat.....	144
Cuadro 80. Parámetros para calcular la magnitud de la perturbación de torque magnético.	145
Cuadro 81. Parámetros para calcular la magnitud de la perturbación de torque gravitacional.	146
Cuadro 82. Parámetros para calcular la magnitud de la perturbación de torque aerodinámica.	146

Cuadro 83. Parámetros para calcular la magnitud de la perturbación de torque de presión por radiación solar.....	147
Cuadro 84. Resumen de magnitudes de torques de las perturbaciones estudiadas.	147
Cuadro 85. Resumen de conexiones en el módulo L298N.....	151
Cuadro 86. Resumen de conexiones en el módulo BNO055.....	152
Cuadro 87. Riesgos del proyecto.....	155
Cuadro 88. Descripción de componentes y su masa.....	158
Cuadro 89. Rangos de temperatura y dimensiones de componentes.	161
Cuadro 90. Materiales de los componentes junto con sus cualidades de gasificación.	163
Cuadro 91. Evaluación de conceptos de celdas solares.....	170
Cuadro 92. Evaluación de conceptos de baterías.....	171
Cuadro 93. Riesgo 1, baterías fallan durante misión.	173
Cuadro 94. Plan de mitigación. riesgo 1, baterías fallan durante misión.	173
Cuadro 95. Riesgo 2, falla de sistema de distribución de energía eléctrica.....	174
Cuadro 96. Plan de mitigación. riesgo 2, falla de sistema de distribución de energía eléctrica.....	174
Cuadro 97. Riesgo 3, daño a celdas solares durante instalación.....	175
Cuadro 98. Plan de mitigación. riesgo 3, daño a celdas solares durante instalación.....	175
Cuadro 99. Riesgo 4, fallo de celdas solares durante misión.	176
Cuadro 100. Plan de mitigación. riesgo 4, fallo de celdas solares durante misión.	176
Cuadro 101. Ponderación de las variables esenciales para la selección de la computadora a bordo....	180
Cuadro 102. Selección de computadora a bordo utilizando el método de trade study.	181
Cuadro 103. Mediciones de energía de la Raspberry Pi sin modificaciones.....	183
Cuadro 104. Mediciones de energía de la Raspberry Pi sin modificaciones.....	183
Cuadro 105. Efectos de la interferencia electromagnética.	186
Cuadro 106. Tiempos promedio de ejecución para cada una de las actividades de carga útil.....	190
Cuadro 107. Desviación estándar para cada una de las actividades de carga útil.....	190
Cuadro 108. Tiempos de ejecución promedio para cada una de las actividades de ADCS.	194
Cuadro 109. Desviación estándar para cada una de las actividades de ADCS.	194
Cuadro 110. Tiempos de envío de información al módulo de comunicaciones.	199

Cuadro 111. Cantidad de mediciones del bus de datos durante 24 horas.....	204
Cuadro 112. Cantidad de reinicios de computadoras y tiempos de operación.....	207
Cuadro 113. Requerimientos de Cal Poly.	243

LISTA DE FIGURAS

	Página
Figura 1. Sistema de despliegue utilizado para poner en órbita a los CubeSats en el espacio.	6
Figura 2. Diagrama del desarrollo de los requerimientos a lo largo del proyecto.	14
Figura 3. Capacidad específica – densidad de energía por tecnología de baterías.	28
Figura 4. Características de las principales tecnologías dentro de la familia de baterías de litio.	29
Figura 5. Capacidad – ciclos carga/descarga, para distintos voltajes de fin de carga.	30
Figura 6. Capacidad – ciclos de carga/descarga, para diversas tasas de descarga.	31
Figura 7. Capacidad retenida – ciclos de carga/descarga con diferentes límites de carga.	32
Figura 8. Configuración física de baterías de litio.	33
Figura 9. Rango ideal operativo de estado de carga.	34
Figura 10. Ciclos de vida – temperatura.	34
Figura 11. Diagrama de conexión de I2C.	37
Figura 12. Ejemplo de funcionamiento del protocolo I2C.	39
Figura 13. Comunicación UART.	39
Figura 14. Vectores perpendiculares unitarios formando la base de un marco de referencia.	41
Figura 15. Acoplamiento y colocación de las ruedas de inercia en cada eje principal.	52
Figura 16. Diagrama descriptor de la función de un lazo de control cerrado para un satélite.	52
Figura 17. Diagrama de bloques del sistema de control del CubeSat.	55
Figura 18. Representación gráfica de la distribución de los módulos durante la prueba de integración.	77
Figura 19. Cronograma a seguir durante la prueba de integración.	80
Figura 20. Ajustes utilizados para simulación en STK fase 3.	81
Figura 21. Simulación en 3D de la órbita del satélite fase 3.	82
Figura 22. Órbita 2D del satélite, mostrado sobrevolando Guatemala fase 3.	82
Figura 23. Arreglo físico de celdas solares.	95
Figura 24. Ajustes utilizados para simulación en STK.	96
Figura 25. Simulación en 3D de la órbita del satélite.	96

Figura 26. Órbita 2D del satélite, con órbita de la ISS.....	96
Figura 27. Relación entre luz proveniente del Sol y ángulo de ubicación en órbita del CubeSat.	97
Figura 28. Marco de referencia del satélite.	98
Figura 29. Vista frontal de configuración de PCBs externas.....	101
Figura 30. Vista trasera de configuración de PCBs externas.....	102
Figura 31. Vista frontal de estructura interna para expansión de baterías.	104
Figura 32. Estructura interna para expansión de baterías.....	104
Figura 33. Prueba de electroluminiscencia a paneles solares.	106
Figura 34. Pruebas de termografía a paneles solares.	106
Figura 35. Pruebas de vibraciones a sistema integrado.....	107
Figura 36. Arquitectura centralizada.	111
Figura 37. Interfaces del módulo térmico.	113
Figura 38. Interfaces del módulo de potencia.	113
Figura 39. Interfaces del módulo ADCS.....	114
Figura 40. Interfaces del módulo de carga útil.	114
Figura 41. Interfaces del módulo de comunicaciones.	115
Figura 42. Diagrama de comunicación mediante el bus de datos I2C.	116
Figura 43. Diagrama de flujo para operación del módulo ADCS.....	118
Figura 44. Diagrama de flujo para operación del módulo de carga útil.	119
Figura 45. Diagrama de flujo para operación del módulo de energía.	120
Figura 46. Diagrama de flujo para operación del módulo de comunicaciones.	121
Figura 47. Diagrama de secuencia para la obtención de datos del ADCS sobre el magnetómetro.	122
Figura 48. Diagrama de secuencia para la obtención de datos del ADCS sobre los fotodiodos.	122
Figura 49. Diagrama de secuencia para la configuración y lectura de datos para el magnetómetro.	123
Figura 50. Diagrama de secuencia para la configuración y lectura de datos para el convertidor analógico a digital del cual proviene la información de los fotodiodos.	124
Figura 51. Diagrama de secuencia para el movimiento del carrusel de filtros.	125
Figura 52. Diagrama de secuencia para la toma de fotografías.	125
Figura 53. Diagrama de secuencia para obtener la temperatura de las baterías.....	126

Figura 54. Diagrama de secuencia para obtener el estado de carga de las baterías.....	126
Figura 55. Diagrama de secuencia para obtener el voltaje de las baterías.....	127
Figura 56. Diagrama de secuencia para transferir datos al módulo de comunicaciones.	127
Figura 57. Diagrama de actividad para obtener la temperatura de las baterías.....	128
Figura 58. Diagrama de actividad para obtener el estado de carga de las baterías.	129
Figura 59. Diagrama de actividad para obtener el voltaje de las baterías.	130
Figura 60. Diagrama de actividad para transferir datos al módulo de comunicaciones	131
Figura 61. Conexión UART.....	136
Figura 62. Interacción de redundancia.....	136
Figura 63. Estados de la redundancia.	137
Figura 64. Método para medir corriente eléctrica.....	140
Figura 65. Circuito que genera pulsos eléctricos.....	141
Figura 66. Mecanismo para generar interferencia electromagnética.	141
Figura 67. Frecuencímetro.	142
Figura 68. Analizador de espectro.....	142
Figura 69. Mapa de latitudes y longitudes de la Tierra con separación de grados.....	145
Figura 70. Diagrama de ingreso de perturbaciones a la planta del sistema.....	148
Figura 71. Módulo puente HL298N.	151
Figura 72. Módulo de mediciones inerciales BNO055.....	151
Figura 73. Microcontrolador Arduino UNO.	152
Figura 74. Motor DC 12V.....	152
Figura 75. Sujetador del motor.....	153
Figura 76. Estructura principal sujetador de pieza para sujeción de motor y de la IMU.....	153
Figura 77. Estructura para sujeción del cuerpo del motor.....	154
Figura 78. Rueda de inercia de 10cm de diámetro y de grosor 6.2mm.	154
Figura 79. Diagrama de criticidad de todos los riegos del proyecto.	157
Figura 80. Presupuesto de potencia fase 3– tiempo.	166
Figura 81. Energía disponible fase 3 – tiempo.	167
Figura 82. Presupuesto de potencia alternativa – tiempo.....	168

Figura 83. Energía disponible alternativa – tiempo.....	168
Figura 84. Esquema de sistema de distribución de energía preliminar.....	172
Figura 85. Matriz de riesgos del módulo de potencia.....	177
Figura 86. Presupuesto de potencia fase 4 – tiempo.....	178
Figura 87. Energía disponible fase 4 – tiempo.....	178
Figura 88. Medición de voltaje de la Raspberry Pi con multímetro.....	182
Figura 89. Medición de corriente de la Raspberry Pi.....	182
Figura 90. Prueba de interferencia electromagnética.....	183
Figura 91. Frecuencia de los pulsos eléctricos para la prueba de interferencia electromagnética.....	184
Figura 92. Medición del analizador de espectro sin actividad electromagnética.....	184
Figura 93. Medición del analizador de espectro con actividad electromagnética.....	185
Figura 94. Medición del analizador de espectro con actividad electromagnética.....	185
Figura 95. Conexión de servomotor MG90S a la Raspberry.....	187
Figura 96. Conexión de módulo de cámara a la Raspberry.....	187
Figura 97. Captura de imagen de prueba con el módulo de cámara de la Raspberry.....	188
Figura 98. Movimiento de carrusel a 0°.....	188
Figura 99. Movimiento de carrusel a 90°.....	189
Figura 100. Movimiento de carrusel a 180°.....	189
Figura 101. Comunicación de Raspberry con Arduino por medio de protocolo I2C.....	191
Figura 102. Verificación de comandos transmitidos por medio del protocolo de comunicación I2C para el chip HMC5883L (magnetómetro).....	191
Figura 103. Datos X, Y y Z obtenidos del magnetómetro.....	192
Figura 104. Datos X, Y y Z en gauss obtenidos del magnetómetro.....	192
Figura 105. Verificación de comandos transmitidos por medio del protocolo de comunicación I2C para el chip ADC128D818.....	193
Figura 106. Datos I0-I5 obtenidos del convertidor de señales analógica a digital.....	193
Figura 107. Pruebas unitarias del bus de datos I2C.....	195
Figura 108. Estructuras de datos.....	196
Figura 109. Mediciones durante 1 hora y 30 minutos de estado de carga de las baterías.....	196
Figura 110. Mediciones durante 24 horas de estado de carga de las baterías.....	197

Figura 111. Mediciones durante 1 hora y 30 minutos del voltaje disponible en las baterías.....	197
Figura 112. Mediciones durante 24 horas del voltaje disponible en las baterías.	198
Figura 113. Mediciones durante 1 hora y 30 minutos de la temperatura de las baterías.	198
Figura 114. Mediciones durante 24 horas de la temperatura de las baterías.	199
Figura 115. Tiempo de envío de datos del módulo de energía.	200
Figura 116. Tiempo de envío de imagen de 337.5 KBytes.	200
Figura 117. Tiempo de envío de imagen de 900 KBytes.	200
Figura 118. Mediciones con osciloscopio del bus de datos.....	201
Figura 119. Mediciones del bus de datos por 20 segundos.	201
Figura 120. Mediciones del bus de datos por 10 minutos.	202
Figura 121. Medición de una lectura del bus con escala de tiempo de 5 ms.	202
Figura 122. Medición de una lectura del bus de datos con escala de tiempo de 20 ms.....	203
Figura 123. Medición de tres lecturas del bus de datos.	203
Figura 124. Método de redundancia para la primera Raspberry Pi.....	205
Figura 125. Método de redundancia para la segunda Raspberry Pi.	205
Figura 126. Diagrama de flujo para computadora principal del método de redundancia.	206
Figura 127. Diagrama de flujo para computadora de respaldo del método de redundancia.	206
Figura 128. Pruebas unitarias del mecanismo de redundancia.	207
Figura 129. Tiempos de operación antes de reinicio.	207
Figura 130. Funcionamiento de computadoras en pruebas de tiempo prolongado.	208
Figura 131. Funcionamiento de redundancia en las primeras dos horas.	208
Figura 132. Funcionamiento de redundancia a mitad de la prueba de tiempo prolongado.....	209
Figura 133. Funcionamiento de redundancia en las últimas tres horas.	209
Figura 134. Funcionamiento de redundancia en las primeras doce horas.	210
Figura 135. Funcionamiento de redundancia en las últimas doce horas.	210
Figura 136. Resultado final de la simulación, animación del planeta con un satélite con cambios de orientación.	211
Figura 137. Estados de los cuaterniones del sistema con un controlador PID.	213
Figura 138. Velocidades angulares del sistema con un controlador PID.	213

Figura 139. Torques de control obtenidos para cada eje con un control PID.	214
Figura 140. Estado de cuaterniones con control LQR sin penalización.	215
Figura 141. Velocidades angulares del sistema con control LQR sin penalización.	216
Figura 142. Torques de control del sistema obtenidos con control LQR sin penalización.	216
Figura 143. Estado de cuaterniones con control LQR con penalización.	217
Figura 145. Torques de control del sistema obtenidos con control LQR con penalización.	218
Figura 146. Torques de perturbaciones totales del sistema.	219
Figura 147. Estados de cuaterniones con perturbaciones alteradas y control LQR penalizado.	219
Figura 148. Velocidades angulares con perturbaciones alteradas y control LQR penalizado.	220
Figura 149. Torques de control con perturbaciones alteradas y control LQR penalizado.	220
Figura 150. Torques de perturbaciones alteradas totales del sistema.	220
Figura 151. Plataforma de pruebas vista lateral.	221
Figura 152. Plataforma de pruebas vista aérea.	221
Figura 153. Plataforma integrada con caja de comando.	222
Figura 154. Componentes dentro de la caja de comando.	222
Figura 155. Prueba de varias vueltas de la plataforma sin acción de control.	223
Figura 156. Prueba de varias vueltas de la plataforma con acción de control.	223
Figura 157. Prueba de dos golpes en lados opuestos a la plataforma sin acción de control.	224
Figura 158. Prueba de dos golpes en lados opuestos a la plataforma con acción de control.	224
Figura 159. Puerto de acceso CubeSat 1U, estándar Cal Poly.	245
Figura 160. Puertos de acceso CubeSat 3U, estándar Cal Poly.	246
Figura 161. Ventana de acceso CubeSat 1U, estándar UNOOSA/JAXA.	246
Figura 162. Interfaces de Raspberry Pi Zero.	294

LISTA DE ACRÓNIMOS

ADC	Convertidor Analógico a Digital (<i>Analog to Digital Converter</i>)
ADCS	Sistema de Determinación y Control de Actitud (<i>Attitud Determination and Control System</i>)
C&DH	Manejo de Comandos y Datos (<i>Command and Data Handling</i>)
Cal Poly	Universidad Politécnica del Estado de California (<i>California Polytechnic State University</i>)
COMMS	Comunicación
CVCM	Materiales Volátiles Condensables Colectados (<i>Collected Volatile Condensable Materials</i>)
DITL	Prueba de Día-en-la-Vida (<i>Day-In-The-Life Test</i>)
EPS	Sistema de Potencia Eléctrica (<i>Electrical Power System</i>)
GCS	Estación de Control en Tierra (<i>Ground Control Station</i>)
ISS	Estación Espacial Internacional (<i>International Space Station</i>)
J-JSSOD	Desplegador Orbital de Satélites Pequeños de JEM (<i>JEM Small Satellite Orbital Deployer</i>)
JAXA	Agencia de Exploración Aeroespacial Japonesa (<i>Japan Aerospace Exploration Agency</i>)
JEM	Módulo de Experimentos Japonés (<i>Japanese Experiment Module</i>)
MAPTIS	Sistema de Información Técnica de Materiales y Procesos (<i>Materials and Processes Technical Information System</i>)
MUA	Acuerdo de Uso de Materiales (<i>Materials Usage Agreement</i>)
OBC	Computadora a Bordo (<i>On Board Computer</i>)
P-POD	Desplegador Orbital de Picosatélites de Poly (<i>Poly Picosatellite Orbital Deployer</i>)
PCB	Tabla de Circuitos Impresos (<i>Printed Circuit Board</i>)
PWM	Modulación por Ancho de Pulsos (<i>Pulse Width Modulation</i>)
PWR	Modulo de Energía (<i>Power</i>)
RAM	Memoria de Acceso Aleatorio (<i>Random Access Memory</i>)
RBF	Remover Antes del Vuelo (<i>Remove Before Flight</i>)
TED	Por ser Definido (<i>To be defined</i>)
TML	Pérdida Total de Masa (<i>Total Mass Loss</i>)
UART	Recibidor-Transmisor Universal Asíncrono (<i>Universal Asynchronous Receiver-Transmitter</i>)

RESUMEN

Un CubeSat es una clase de nano satélite que utiliza un estándar de tamaño y un factor de forma de 10x10x10 cm con una masa inferior a 1.33kg, comúnmente utilizado para realizar investigaciones en el espacio. Este proyecto consiste en el diseño, la elaboración y la puesta en órbita del primer CubeSat guatemalteco, siendo esta la cuarta fase del proyecto. En este trabajo se contempla los módulos de ingeniería de sistemas, potencia, manejo de comandos y datos, y el sistema de terminación y control de actitud.

Ingeniería de sistemas logró desarrollar un análisis de los principales riesgos que el proyecto enfrentará, una matriz para la verificación y validación de todos los riesgos, una base de datos con la información principal de los componentes del satélite, y la planificación de una Prueba de Integración. De esta manera se logró definir todos los requerimientos que el sistema diseñado debe tomar en cuenta, asegurar su implementación durante su desarrollo, disminuir la probabilidad de ocurrencia y/o consecuencia de ciertos riesgos sobre este, determinar cómo la elección de ciertos componentes puede afectarlo, y planificar una prueba en la cual se podrá observar cómo todos los módulos operan en conjunto como un solo sistema.

Para el manejo de comandos y datos se enfocó en la recepción, validación, decodificación y distribución de comandos a los sistemas de carga útil y ADCS, excluyendo el procesamiento de la información. Para poder cumplir con estos aspectos se realizó el diseño del módulo, definiendo la arquitectura de hardware, el de bus de datos y un primer acercamiento a una posible computadora a bordo.

Para diseñar el sistema de potencia del Megaproyecto CubeSat se llevó a cabo una investigación acerca de la recolección de energía mediante paneles solares, y los diversos factores que afectan este proceso. También se aprendió acerca del funcionamiento del resto de módulos y se analizó especialmente sus características en cuanto a consumo de potencia eléctrica. En cuanto al almacenamiento de energía se definió que se utilizará batería, además se plantearon, los parámetros de operación con el fin de que estas sirvan durante la duración esperada de la misión. Se planteó un sistema de distribución de energía eléctrica tomando en cuenta las características de las cargas.

El sistema de control activo desarrolló dos modelos de control para futuros satélites, estos hacían uso de teoría clásica como de moderna en el campo. Se hizo un control en base al controlador PID y otro se elaboró con el regulador lineal cuadrático (LQR). Estos demostraron ser capaces de estabilizar el satélite, pero especialmente fue el LQR el que integró la orientación como la reducción de las velocidades angulares del satélite de manera correcta. Se observó también las perturbaciones en el espacio y cómo estas afectan al CubeSat. Por último, se diseñó y construyó una plataforma de pruebas donde se comprobó la eficiencia y robustez del sistema de control desarrollado.

I. INTRODUCCIÓN

En el año 1999 la Universidad Politécnica del Estado de California (Cal Poly) desarrolla el estándar CubeSat el cual dictaba la especificación de este tipo de satélites. Los CubeSat son nano satélites programados para realizar una misión específica. El estándar del CubeSat define una unidad (1U) como un cubo de 10x10x10 cm que posee una masa inferior a 1,33 kg. En sus inicios su objetivo era el facilitar el acceso al espacio a los estudiantes universitarios permitiéndolos diseñar, construir, probar y operar un satélite en el espacio.

Desde los primeros CubeSats lanzados en el año 2003, se ha generado interés alrededor del estándar, ya que permite realizar investigación científica en el espacio y probar tecnologías en órbita baja. Además, este estándar ha permitido a varios países desarrollar las capacidades necesarias para elaborar sus primeros satélites y abrir el campo de la ingeniería aeroespacial. Se espera que este proyecto sea capaz de lograr esto para Guatemala.

Actualmente Guatemala está desarrollando un CubeSat cuya misión es evaluar e integrar los diferentes sistemas que componen un CubeSat, para desarrollar en Guatemala las capacidades requeridas para la operación de este tipo de satélites, al mismo tiempo que permita validar elementos necesarios para detección remota en futuros satélites. La construcción de este satélite se dividirá en distintos módulos, siendo: estructura, térmico, ADCS, comunicaciones, potencia, carga útil y manejo de comandos y datos. El módulo de carga útil es determinado por la misión a cumplir, en este caso el prototipo de sensor multi espectral.

Este proyecto consiste en el diseño, la elaboración y la puesta en órbita del primer CubeSat diseñado y ensamblado en Guatemala. La primera fase de este proyecto comenzó en el año 2014. Durante las primeras tres fases se realizaron importantes avances en todo lo concerniente a la estructura y el aislante térmico del satélite. También se seleccionaron algunos de los componentes que conformarán los distintos módulos; sin embargo, fue necesario reevaluar si estos componentes cumplen con los requerimientos más recientes del proyecto. Adicionalmente, durante la tercera fase se realizaron matrices de análisis de riesgos y de cumplimiento de requerimientos, las cuales deben ser modificadas para mostrar los riesgos específicos para cada módulo e incluir nuevos requerimientos para el sistema.

II. OBJETIVOS

A. OBJETIVO GENERAL DEL MEGAPROYECTO

Evaluar e integrar los diferentes sistemas que conforman un CubeSat para desarrollar en Guatemala las habilidades requeridas para la operación de satélites, a la vez que se validan elementos de teledetección.

B. OBJETIVO GENERAL DEL MÓDULO DE INGENIERÍA EN SISTEMAS Y GESTIÓN DE PROYECTOS

Asegurar que el proyecto cumplirá con todos sus objetivos de la manera más segura y efectiva, logrando así aumentar las probabilidades de éxito de este.

C. OBJETIVOS ESPECÍFICOS DEL MÓDULO DE INGENIERÍA EN SISTEMAS Y GESTIÓN DE PROYECTOS

1. Definir apropiadamente los requerimientos que el proyecto deberá cumplir para alcanzar sus objetivos.
2. Definir métodos mediante los cuales puede asegurarse que el proyecto toma en cuenta y cumple con todos los requerimientos definidos.
3. Identificar los principales riesgos que podrían afectar al proyecto y determinar las técnicas que podrían ser utilizadas para su mitigación.
4. Comprobar que los componentes seleccionados para el satélite poseen las características físicas y operacionales necesarias para poder ser utilizados en el ambiente en el cual operarán.
5. Planificar una prueba de integración, en la cual se pruebe que todos los subsistemas puedan cumplir con sus objetivos una vez integrados.

D. OBJETIVO GENERAL DEL MÓDULO DE POTENCIA

Diseñar un sistema de potencia preliminar para el satélite.

E. OBJETIVOS ESPECÍFICOS DEL MÓDULO DE POTENCIA

1. Definir apropiadamente los requerimientos que el proyecto deberá cumplir para alcanzar sus objetivos.
2. Definir métodos mediante los cuales puede asegurarse que el proyecto toma en cuenta y cumple con todos los requerimientos definidos.

3. Identificar los principales riesgos que podrían afectar al proyecto y determinar las técnicas que podrían ser utilizadas para su mitigación.
4. Comprobar que los componentes seleccionados para el satélite poseen las características físicas y operacionales necesarias para poder ser utilizados en el ambiente en el cual operarán.
5. Planificar una prueba de integración, en la cual se pruebe que todos los subsistemas puedan cumplir con sus objetivos una vez integrados.

F. OBJETIVO GENERAL DEL MÓDULO DE MANEJO DE COMANDOS Y DATOS

Diseñar e implementar el sistema de manejo de comandos y datos de acuerdo con las necesidades del primer CubeSat guatemalteco, capaz de integrar los módulos de carga útil, ADCS, energía y comunicaciones.

G. OBJETIVOS ESPECÍFICOS DEL MÓDULO DE MANEJO DE COMANDOS Y DATOS

1. Seleccionar una computadora de propósito general que cumpla con los requerimientos de la misión.
2. Implementar un protocolo de comunicación útil para la interacción entre módulos.
3. Implementar el software del bus de datos que garantice el intercambio de información entre módulos.
4. Distribuir comandos de acción a los módulos.

H. OBJETIVO GENERAL DEL MÓDULO DE CONTROL ACTIVO

Asegurar que el CubeSat sea capaz de posicionarse de manera adecuada en el espacio y reducir cualquier perturbación que afecte al sistema.

I. OBJETIVOS ESPECÍFICOS DEL MÓDULO DE CONTROL ACTIVO

1. Desarrollar el sistema de control activo de posición para un CubeSat.
2. Validar diferentes diseños de control activo a través de las herramientas de MATLAB y Simulink.
3. Desarrollar una plataforma para comprobar el funcionamiento del sistema de control activo desarrollado.

III. JUSTIFICACIÓN

Los CubeSat presentan una solución a tiempo y dinero de elaboración de satélites debido a que son soluciones prácticas que se apegan a diferentes problemáticas y soluciones para las mismas. Estos satélites normalmente están equipados con el equipo necesario para llevar a cabo una misión en específico. Para esto se les incluyen diferentes sensores que se encargarán de recopilar la información necesaria para monitorear un fenómeno que ocurra en la superficie terrestre.

Esto representa una oportunidad para que un país en vías de desarrollo como Guatemala pueda sacar provecho y abrirse nuevas fronteras. No obstante, esto presenta un reto bastante amplio para la nación y el grupo debido a que en Guatemala no se cuenta con experiencia respecto al tema. Hay poca información sobre la operación y desarrollo de sistemas como los satélites en el país y esto conlleva a realizar mucha investigación para dominar los temas necesarios para llevar a cabo una misión como esta. Estos son pasos a llevar a cabo para que Guatemala sea una más de las naciones con un satélite propio y que se utilice para beneficiar a la nación a través de la tecnología y el uso de sensores para monitorear procesos imprescindibles.

Adicionalmente, se ha identificado la necesidad de monitorear el progreso de cianobacteria en Lago Atitlán, ya que los métodos implementados hasta el momento no han sido efectivos y requieren de muchas mediciones. El propósito del primer satélite en desarrollo en Guatemala es el probar la tecnología que sería capaz de monitorear el lago de forma remota, para así en un satélite futuro tener la capacidad de monitorear este recurso natural, con la esperanza de poder ayudar a mejorar la situación actual en la cual se encuentra este.

IV. MARCO TEÓRICO

A. CUBESAT

En 1999 los estándares de diseño del CubeSat fueron creados por el profesor Jordi Puig Suari de la *California Polytechnic State University*, y el profesor Robert Twiggs de *Stanford University's Space Systems Development Lab*. El objetivo era facilitar el acceso al espacio a los estudiantes universitarios permitiéndolos diseñar, construir, probar y operar un satélite en el espacio. En sus inicios no se presentó como un estándar, más bien se convirtió con el paso de los años, debido a la adopción del diseño por cientos de organizaciones alrededor de todo el mundo. (Cal Poly, 2009)

Todo nació por la necesidad de tener un satélite con pequeñas dimensiones en 1998, cuando uno de los proyectos de la facultad de desarrollo de sistemas espaciales de Stanford específicamente el microsatélite OPAL se retrasó, buscando soluciones el profesor Twiggs buscó el apoyo de DARPA, que resultó en el rediseño del mecanismo de lanzamiento de satélites pequeños. Con la intención de disminuir el ciclo de desarrollo e inspirados por OPAL, Twiggs buscó la relación ideal entre tamaño y dificultad de desarrollo; lo cual lo llevó en pensar en un cubo de 10 cm de arista. En el verano de 1999 Twiggs presentó la idea a Puig Suari la cual se lanzó en la conferencia de *Science, Technology and Space Applications Program* de Japón-Estados Unidos en noviembre del mismo año. (Cal Poly, 2009).

Un CubeSat es una clase de nanosatélite que utiliza un estándar de tamaño y un factor de forma, comúnmente utilizado para realizar investigaciones en el espacio. El estándar del CubeSat define una unidad (1U) como un cubo de 10x10x10 cm con un posee un volumen de 1 litro y una masa inferior a 1,33 kg. Estas unidades pueden unirse para formar un CubeSat más grande 2U, 3U o 6U. (Loff, 2017)

Debido a su gran popularidad y gran funcionalidad en los últimos años se han incrementado sus implementaciones, su tamaño pequeño ayuda a rellenar espacios dentro de los cohetes, ya que son lanzados como carga secundaria dentro de cohetes, esto es hecho utilizando un sistema estandarizado para el despliegue creado por la Universidad Politécnica Estatal de California (Cal Poly, por sus siglas en inglés). Este sistema, el cual es conocido como P-POD, es puesto en el vehículo de lanzamiento que contiene 3 CubeSat y, por medio de resortes, realiza el despliegue de estos al espacio. (Krogh, *et al*, 2002)

Figura 1. Sistema de despliegue utilizado para poner en órbita a los CubeSats en el espacio.



El principal objetivo de la creación de éstos es para la creación de satélites universitarios, para introducir a los estudiantes, de una manera realista y práctica, al diseño de artefactos espaciales y para que tengan conocimientos del proceso de lanzamiento de una misión espacial. (Cote, *et al*, 2011)

B. INGENIERÍA DE SISTEMAS

De acuerdo con el Manual de Ingeniería de Sistemas (*Systems Engineering Handbook*) de la NASA (2007:3), la Ingeniería de Sistemas es la disciplina mediante la cual se desarrolla un sistema operable capaz de cumplir con ciertos requerimientos, a la vez que sortea distintos tipos de restricciones. Esta fuente explica también que la Ingeniería de Sistemas es una disciplina integrativa en la cual son necesarias las contribuciones de ingenieros e ingenieras con especializaciones en distintos campos, además de los de otros y otras profesionales. Estas contribuciones deben ser evaluadas y balanceadas, de manera que produzcan un sistema coherente que no esté dominado por la perspectiva de una sola disciplina.

La fuente mencionada anteriormente explica también que las personas encargadas del desarrollo de la disciplina de Ingeniería de Sistemas, denominadas comúnmente ingenieros de sistemas, deben desarrollar las habilidades necesarias para lograr optimizar el diseño completo de un sistema, y no favorecer solamente a un sistema/subsistema a expensa de otros. Para lograr esto, estas personas desempeñan distintas funciones en el sistema, las cuales dependen del proyecto en el cual se trabaja. Sin embargo, se afirma que independientemente de cuáles sean las funciones del ingeniero de sistemas, esta persona debe asegurar mediante su trabajo que el sistema cumple técnicamente con las necesidades y los requerimientos definidos para tal sistema, y que se está siguiendo el enfoque ingenieril apropiado en cada fase del proyecto.

El Manual de Ingeniería de Sistemas de la NASA (2007:3) asegura también que el ingeniero de sistemas jugará un rol clave liderando el desarrollo de la arquitectura del sistema, definiendo y posicionando requerimientos, evaluando distintas opciones de diseño, balanceando riesgos técnicos entre sistemas, definiendo y asesorando interfaces, proveyendo supervisión en actividades de verificación y

validación, etc. En forma resumida, explica que un ingeniero de sistemas debe saber balancear interacciones organizacionales y técnicas en sistemas complejos. En otras palabras: deben poder ver “la imagen completa”.

Para abarcar de mejor manera el tema de Ingeniería de Sistemas, la fuente citada anteriormente afirma que también debe verse desde el contexto de la gestión del proyecto. Este último es definido por el documento NPR 7120.5, Programa Espacial de la NASA y Gestión de Proyecto de Procesos y Requerimientos (*NASA Space Flight Program and Project Management Processes and Requirements*, 2007) como la función de planear, supervisar y dirigir las distintas actividades necesarias para alcanzar los requerimientos, las metas y los objetivos del cliente y otros inversionistas a través de costos específicos, calidad y cronogramas. Así, se afirma que la función del gestor del proyecto puede dividirse en dos partes, ambas de la misma importancia. Estas dos partes son Ingeniería de Sistemas y control del proyecto. Ambas partes desempeñan distintas funciones; sin embargo, existen áreas en las cuales existirá un traslape entre estas. En estas áreas, la Ingeniería de Sistemas proporciona los aspectos técnicos necesarios, mientras que el control del proyecto proporciona los aspectos programáticos tales como costos y cronogramas.

C. ANÁLISIS DE RIESGOS

Cómo se mencionó anteriormente, la o las personas encargadas de aplicar la disciplina de Ingeniería de Sistemas en algún proyecto deben saber cómo balancear riesgos técnicos entre distintos subsistemas. De acuerdo con el documento Requerimientos de Procedimientos de Gestión de Riesgos de la Nasa (*NASA Risk Management Procedural Requirements*, 2008), un riesgo es el potencial que existe de que un sistema no se desempeñe como se espera. Según Brumbaugh y Lightsey (2013:147), estos desempeños deficientes pueden deberse a múltiples razones: falta de apoyo institucional, fallas en seguridad, fallas técnicas, cuestiones de costo o tiempo, etc. Como se puede observar, evitar estos desempeños deficientes que podrían presentarse en distintas fases de la vida del proyecto es importante para asegurar una mayor probabilidad de éxito para la misión. Para lograr esto, Brumbaugh y Lightsey (2013:147) resaltan la importancia de realizar el proceso de gestión de riesgos. Estos autores afirman que este es el proceso mediante el cual se identifican, analizan y mitigan los riesgos de un proyecto, a la vez que se monitorean las causas de los problemas y sus consecuencias sobre el sistema.

De acuerdo con los autores citados anteriormente, es importante mencionar también que actualmente los planes de gestión de riesgos se han utilizado solamente para proyectos que involucran satélites costosos, complejos y de enormes dimensiones; y por lo tanto, rara vez se han utilizado en satélites con masas menores a 10 kilogramos. Naturalmente, los riesgos asociados a los satélites más pequeños y baratos son distintos a los que se encontrarán cuando se trabaja con satélites más grandes y más complejos. Así, es

necesario que existan nuevos enfoques para mejorar los procesos de gestión de riesgos en satélites tipo CubeSat, especialmente aquellos desarrollados por grupos estudiantiles.

Brumbaugh y Lightsey (2013:148) afirman que los métodos para análisis de riesgos normalmente utilizados en el campo aeroespacial, como el Modos de Falla y Efectos, FMEA por sus siglas en inglés y la Evaluación de Riesgos Probabilísticos, PRA en inglés, no son viables para satélites universitarios debido a la gran cantidad de recursos que estos métodos requieren. Además, algunos de estos métodos requieren acceso a información sobre misiones pasadas, la cual rara vez se encuentra al alcance de grupos universitarios.

Es por estas razones, y muchas más, que los autores citados anteriormente desarrollaron un método de bajo costo para análisis de riesgos. Este método fue especialmente ideado para ser utilizado por estudiantes universitarios, por lo que solamente requiere de información del satélite a desarrollar, de la misión, y algunas herramientas computacionales accesibles para la mayoría de las personas.

Brumbaugh y Lightsey (2013:149) explican que todo método de gestión de riesgos está conformado por tres pasos, los cuales pueden dividirse en distintos sub-pasos. Estos tres pasos son: identificar los riesgos de la misión, determinar la mejor técnica para la mitigación de estos riesgos, y monitorear el progreso de los riesgos. Como puede observarse, ninguno de estos pasos requiere de herramientas de alto costo o de difícil obtención, por lo que el método logra cumplir con el principio de bajo costo y alta accesibilidad.

El primer paso del método presentado anteriormente es la identificación de los riesgos. Es recomendable comenzar este paso analizando las distintas fases de la misión y el funcionamiento de la carga primaria. Comúnmente, las primeras fases de la misión son la revisión previa al lanzamiento y el lanzamiento. Resaltan que los equipos universitarios no tienen ningún control sobre las fallas en el momento del lanzamiento; sin embargo, sí tienen control sobre las fechas de entrega de su satélite. Continuando con las siguientes fases de la misión, los autores recomiendan tomar en cuenta qué es lo que podría causar que la carga primaria no cumpla con su función. Blanchard y Fabrycky (2006:710) explican que todos los riesgos deben analizarse en términos de hardware, software y los problemas programáticos que causen.

Para poder continuar con el primer paso del método para el manejo de riesgos, Brumbaugh y Lightsey (2013:149) explican que, una vez identificados los riesgos de la misión, se debe determinar las causas de estos riesgos. Debe determinarse que hardware, software o problema programático causará el problema descrito por cada riesgo. Se afirma que en esta parte del método reside una de las mayores diferencias entre los métodos utilizados por los grupos que desarrollan satélites de mayores proporciones y el que describen. Aunque muchos de los riesgos que pueden encontrarse son similares tanto para satélites pequeños y simples

como para los grandes y complejos, las causas de estos riesgos pueden ser bastante distintos entre estos dos. Muchas de estas diferencias pueden provenir de las diferentes prácticas ingenieriles y el ambiente de trabajo. Además de esto, los proyectos universitarios tienden a padecer mayores riesgos en cuanto a recursos monetarios debido a que sus fondos son menores.

Brumbaugh y Lightsey (2013:150) recomiendan que para encontrar todas las causas de cada riesgo se examine la matriz de verificación de requerimientos, el cronograma del proyecto, el presupuesto y los documentos de revisión de la misión.

La fuente citada previamente explica que, aunque la responsabilidad principal del análisis de riesgos del sistema completo recae sobre el ingeniero de sistemas, todo el equipo debe participar en la mitigación de las causas de los riesgos. Es por esto que mencionan que es importante que se nombren responsables para cada causa encontrada. Los responsables deberán ser quienes tengan mayor conocimiento sobre el área técnica en la cual se presenta la causa del riesgo. Normalmente, esta responsabilidad recaerá sobre los encargados de los sub-sistemas.

Una vez identificados los riesgos, sus causas y nombrados los responsables Brumbaugh y Lightsey (2013:150) afirman que la siguiente parte de su método consiste en asignarle a cada causa de cada riesgo un valor para su probabilidad de ocurrencia y para la consecuencia que tendrá sobre el desempeño del sistema. Estos valores suelen darse en una escala de 1 a 5, en la cual 1 significa lo menos probable o crítico y 5 lo más. Los autores argumentan que para esta parte del método pueden utilizarse como guías las tablas de la Guía de Adquisición (*Guide of Acquisition*) de la Universidad de Adquisición de Defensa del Departamento de Defensa estadounidense. En estas tablas se califican las consecuencias dividiéndolas en tres grupos: técnicas, de presupuesto y de tiempo. Sin embargo, enfatizan en que la decisión final sobre el valor de la probabilidad y la consecuencia debe ser tomado por el encargado del sub-sistema, el ingeniero de sistemas y el gestor del proyecto en consenso.

Cada vez que se toman decisiones sobre los valores de la probabilidad y la consecuencia para cada caso, recomiendan que se presenten de forma sucinta las razones de por qué se dieron estos valores. Esto permitirá que todos los miembros del grupo estén al tanto de estas razones, y permite también que se observe mejor la evolución de los riesgos a lo largo del proyecto.

A continuación, Brumbaugh y Lightsey (2013:150) indican que se debe calcular cuantitativamente la prioridad en cuanto a recursos que debe dársele a cada causa. Esto puede lograrse una vez se haya encontrado el valor de la consecuencia y la probabilidad de cada causa. Para hacer esto, antes que nada, debe encontrarse el producto del valor de la consecuencia y la probabilidad de cada causa. Después deben ordenarse de mayor a menor las causas según el valor de su producto. Por último, debe dársele una

prioridad numérica de 1 a la causa con el producto más alto, de 2 a la segunda, y así hasta que se le haya dado una prioridad a todas las causas. Puede ocurrir, y sucede generalmente, que múltiples causas tengan el mismo valor numérico de prioridad.

Después de haber obtenido toda la información mencionada anteriormente, los autores citados previamente mencionan que con esta debe obtenerse el valor de la probabilidad y la consecuencia de cada riesgo. Estos valores se obtienen mediante el promedio ponderado de los valores de probabilidad y consecuencia de cada causa. El valor ponderado de cada causa se obtiene utilizando un método recíproco de rango. Este valor está dado por la siguiente ecuación:

$$w_i = \frac{\frac{1}{R_i}}{\sum_{j=1}^N \frac{1}{R_j}} \quad (1)$$

en donde:

- R_i es el valor del rango de prioridad de la causa i .
- N es el número total de causas para cierto riesgo.

Este factor de ponderación será un valor entre 0 y 1. A continuación, el valor de probabilidad y consecuencia de cada causa se debe multiplicar por el factor de ponderación, para después obtener los valores de consecuencia y probabilidad de cada riesgo sumando los productos de esta operación. Los autores afirman también que otros métodos de ponderación pueden utilizarse, pero explican que se recomienda utilizar el método descrito debido a que ha demostrado ser más útil en práctica.

Finalmente, en la fuente mencionada anteriormente se explica que la última parte de este primer paso es trazar un diagrama L-C. En este diagrama se colocan todos los riesgos de un sistema. Su posición en dicho diagrama depende del valor de probabilidad y consecuencia obtenidos. Este diagrama es un diagrama de 5X5 en el cual el eje horizontal es el eje de consecuencia, y el eje vertical es el eje de probabilidad. La parte superior derecha del diagrama es de color rojo, lo cual representa que los riesgos de esta área son críticos. La parte inferior izquierda es de color verde, lo cual significa que los riesgos en tal área son los menos críticos. En medio de estas dos áreas se encuentra una amarilla, la cual es el área para los riesgos que representan una amenaza media.

El segundo paso en este método, de acuerdo con Brumbaugh y Lightsey (2013:152), es la determinación de la técnica de mitigación más efectiva para cada riesgo. Según la Universidad de Adquisición de Defensa del Departamento de Defensa estadounidense (2006:3) un riesgo queda mitigado cuando se escoge la opción que provea el mejor balance entre desempeño y costo.

Brumbaugh y Lightsey (2013:152) afirman que existen cuatro formas de mitigar un riesgo: el riesgo puede evitarse eliminando sus causas y/o consecuencias, pueden controlarse sus causas o consecuencias, transferir el riesgo a otra persona o proyecto, o asumir el riesgo y continuar con el proyecto. Recomiendan que se escoja por lo menos uno de estos métodos para cada riesgo y cada una de las causas de estos riesgos. Explican también que escoger más de un método de mitigación para un riesgo o causa puede aumentar las probabilidades de mitigar tales.

Finalmente, los autores citados anteriormente mencionan que el último paso es el monitoreo de los riesgos. Esto puede lograrse reevaluando los valores de probabilidad y consecuencia de cada causa en ciertas fases del proyecto. Estas reevaluaciones deben llevarse a cabo por el ingeniero de sistemas, el gestor del proyecto, y la persona encargada del sub-sistema al que pertenece el riesgo. Recomiendan que se trace movimiento de cada riesgo en el diagrama L-C a lo largo del proyecto, de manera que sea más sencillo observar cómo es que ha evolucionado cada riesgo.

D. REQUERIMIENTOS DE LA MISIÓN

La definición y evaluación de los requerimientos de una misión es uno de los principales objetivos del método de ingeniería de sistemas, como se ha destacado anteriormente. Larson y Wertz (1999:15) explican que después de que se hayan definido los objetivos de una misión, se busca traducir estos en grupos de requerimientos y limitaciones preliminares sobre el desempeño y las operaciones de esta. Estos requerimientos deben reflejar lo mejor posible los objetivos de la misión ya que de ellos dependerán los conceptos operaciones que cumplirán con los objetivos. De acuerdo con los autores citados anteriormente, para lograr esto es necesario notar los siguientes tres conceptos:

1. Requerimientos funcionales. Definen qué tan bien se debe desempeñar el sistema diseñado para la misión para cumplir con los objetivos de esta.
2. Requerimientos operacionales. Definen cómo operará el sistema y cómo tendrán que interactuar con él los usuarios para poder cumplir con los objetivos.
3. Limitantes. Estos limitan los costos, el tiempo, y las técnicas de implementación disponibles para quienes diseñan el sistema.

De acuerdo con Larson y Wertz (1999:15) las necesidades, requerimientos y limitaciones de cualquier misión dependerán principalmente de esta misma, y de la manera en que será implementada. Debido a que la definición de los principales requerimientos de una misión es una tarea extremadamente difícil, tal y como se explica en la fuente recién citada, es una práctica común que estos pasen por varias iteraciones durante el proceso de diseño. Se recomienda realizar una primera aproximación de los requerimientos combinando las metas y los objetivos de la misión con observaciones sobre lo que es posible hacer.

Algunas veces es posible tomar los requerimientos de otra misión y modificarlos ligeramente para utilizarlos en alguna otra misión. Esto presenta la evidente ventaja de que se puede aprovechar la información obtenida por otras misiones. Los autores recalcan también la importancia de revisar constantemente los requerimientos de la misión, y modificarlos si es necesario.

Según Larson y Wertz (1999:15), es importante también tomar en cuenta que el desarrollo de una misión espacial también depende de temas políticos, económicos y legales, así como de tecnológicos. Así, en el momento de definir los requerimientos de una misión se debe tomar en cuenta tanto los aspectos técnicos, como los intereses políticos y económicos de los desarrolladores. Muchas veces, las consideraciones no-técnicas no se toman en cuenta, o se les da poca importancia en comparación las consideraciones técnicas. Sin embargo, los autores explican que una misión exitosa debe incluir todos los requerimientos y límites impuestos sobre un sistema.

En la fuente citada se explica también que muchas veces los diseñadores simplemente tratan de cumplir con todos los requerimientos establecidos preliminarmente ya que al no cumplir con estos consideran que la misión ha fallado. Así, los diseñadores nunca llegan a modificar los requerimientos preliminares, incluso cuando las modificaciones se traducen en reducciones de costo o mejoramientos en el desempeño.

Larson y Wertz (1999:73) afirman también que todos los requerimientos deben provenir de las necesidades de los usuarios y los clientes, las cuales deben estar escritas de la manera más concisa posible. Así mismo, explican que, durante el proceso de definición de requerimientos, quien los define debe enfocarse en todos aquellos requerimientos funcionales y operacionales que resulten críticos para la misión, sin limitar innecesariamente el diseño del sistema a desarrollar. Larson y Wertz (1999:73) mencionan que los temas que se discutirán a continuación sobre los requerimientos de una misión buscan que esta pueda cumplir con todos sus requerimientos al menor costo y riesgo.

El primer tema tratado por Larson y Wertz (1999:74) al hablar sobre la definición de los requerimientos de una misión es el rol de estos en el desarrollo de un sistema. De acuerdo con estos autores, los objetivos y los conceptos del sistema para alguna misión deben tomar en cuenta cinco aspectos: desempeño requerido, costo, cronograma de desarrollo y de ejecución, limitantes explícitas e implícitas, y los riesgos asociados. Se afirma que estos cinco aspectos deben tomarse en cuenta durante todo el proceso de Ingeniería de Sistemas de la misión. Durante este proceso, los requerimientos principales que guiarán el desarrollo del sistema diseñado deben ser descompuestos y distribuidos de manera que se formen segmentos individuales o elementos del sistema, interfaces entre estos y también interfaces externas al sistema. En este punto los autores recalcan también que para poder definir en su totalidad el sistema que se desarrollará, los usuarios, los clientes, el ingeniero de sistemas y los desarrolladores de cada segmento de este deben estar

comtamente interactuando. Facilitar estas interacciones es otra de las importantes funciones de la persona encargada del enfoque de Ingeniería de Sistemas.

Según la fuente ya mencionada, es común que los desarrolladores del sistema difieran con los usuarios en cuanto a los requerimientos del sistema. Explican que estas diferencias se deben a que los desarrolladores creen que los usuarios están sujetos a enfoques operacionales comunes e insensibilizados a cómo los requerimientos bastante específicos limitan los diseños. Por otro lado, los usuarios creen que los desarrolladores favorecen la utilización de nueva tecnología e ignoran las necesidades prácticas inherentes a la operación de un sistema y a la explotación de la información obtenida de una misión. Como consecuencia de esto, los desarrolladores pueden llegar a establecer requerimientos para la misión sin antes consultar a los usuarios, mientras que los usuarios pueden llegar a definir requerimientos rígidos y estáticos que serán tomados en cuenta a medida que el desarrollo del sistema avanza, afectándolo significativamente. Debido a que para este proceso es importante contar con el enfoque de ambos grupos, los autores explican que es importante que ambos cooperen desde el inicio de la formación de los requerimientos de la misión.

Sobre el rol de los requerimientos en el desarrollo de un sistema, Larson y Wertz (1999:74) explican también que otro problema que puede existir es que normalmente los desarrolladores, deseando comenzar a construir lo antes posible, definen prematuramente requerimientos y conceptos bastante detallados. Esto es causa de que muchas veces se les ponga poca atención a los objetivos principales de la misión. Otra posible complicación es que los clientes muchas veces limitan el desarrollo del sistema imponiendo requerimientos detallados antes de que se lleguen a definir los requerimientos críticos que determinan en su mayor parte los costos y riesgos de la misión. Así mismo, se afirma que estos requerimientos críticos deben estar siempre al frente durante el diseño, desarrollo y la validación del sistema, tomando en cuenta que el nivel de formalidad y detalle de estos puede depender de la madurez del sistema, de su complejidad y de su tamaño.

Larson y Wertz (1999:75) aseguran también que las limitantes de una misión son aquellos requerimientos para el sistema que no pueden modificarse bajo casi ninguna circunstancia. Estos requerimientos pueden ser requerimientos operacionales que indican que los niveles de capacidad de un sistema deben tener cierto valor. Algunas limitantes comunes son el costo, las fechas de entrega y la utilización de cierta tecnología para cumplir con algún objetivo.

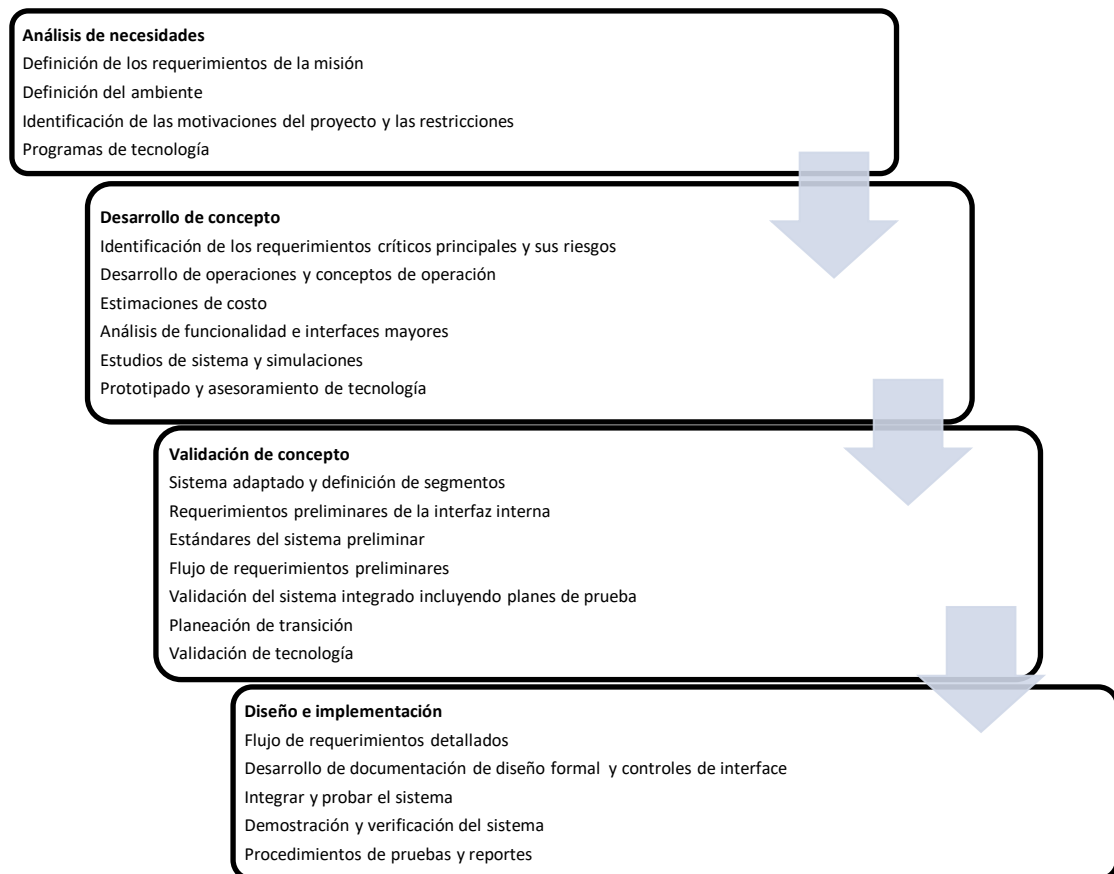
Estos autores explican a la vez que muchas veces sucede que los diseñadores se concentran principalmente en los requerimientos relacionados con el desempeño de algún sistema e ignoran completamente otros requerimientos que consideran menos importantes, como las necesidades del sistema respecto a su medio ambiente; y sin embargo, estos requerimientos pueden llegar a ser críticos en cuanto a

riesgo y costo. De igual forma, cuando se ignoran las interfaces externas de un sistema, este puede terminar como un diseño sin el soporte externo necesario para ser desplegado y operado durante la misión.

Sobre el tema que se ha venido discutiendo anteriormente, Larson y Wertz (1999:77) aseguran que, aunque el éxito de una misión recae en el desempeño, el tiempo y el costo, es este último el factor que representa la mayor limitante. Es por esto que una de las prácticas más utilizadas es la de diseño-a-costo (*design-to-cost*), práctica mediante la cual una cantidad fija de dólares afecta las posibles soluciones de diseño. Así, durante el desarrollo del sistema se podría requerir de revisiones de los requerimientos debido a limitaciones de costo, pesando el desempeño de este contra sus costos. Esta práctica puede llevarse a cabo de muchas formas.

Algunas veces, la estandarización puede reducir los costos de una misión y mejorar como esta es operada. Este proceso puede ser nombrado a veces como diseño basado en plataforma (*platform-based design*). Sin embargo, siempre se debe tomar en cuenta las consecuencias de aumentar los riesgos durante el desarrollo para reducir los costos.

Figura 2. Diagrama del desarrollo de los requerimientos a lo largo del proyecto.



Tal y como mencionan Larson y Wertz (1999:75), el rol y las características de los requerimientos cambia en cada fase del desarrollo de algún proyecto; lo cual debería notarse debido a que se debe tener un control de ellos a lo largo de todo el proyecto. Como consecuencia de esto, en las primeras fases del proyecto, los requerimientos deben especificarse utilizando un lenguaje y una estructura que no conlleve detalles prematuros. Mencionan que es importante que los requerimientos principales dirijan las primeras actividades del proyecto, pero a la vez se debe estar preparado para modificar estos requerimientos a medida que se avanza. En la Figura 2 se presenta un diagrama que muestra el desarrollo de los requerimientos durante un proyecto.

Se recomienda que durante el desarrollo del concepto de la misión se especifiquen y documenten los requerimientos críticos de manera flexible, debido a que durante esta fase se evalúan varias opciones de diseño. Comúnmente no se necesita de especificaciones formales que cumplan con algún estándar y que sirvan como guía del sistema hasta que se comience el desarrollo definitivo del sistema. Hasta entonces, no hay en el proyecto requerimientos fijos, más que aquellos con los cuales se pueda trabajar durante el desarrollo del sistema. Se resalta también que existen varias técnicas para el desarrollo de requerimientos de una misión. Estas técnicas pueden variar significativamente entre ellas, y conducir a distintas formas de trabajar.

Durante la fase del proyecto en el cual se comienza a desarrollar el sistema diseñado, Larson y Wertz (1999:77) explican que la mayoría de las actividades relacionadas con los requerimientos se centran en integrar las distintas interfaces del sistema y buscar maneras de cumplir con requerimientos específicos que se han venido acarreado. Resolver problemas significativos del sistema en este punto puede ser extremadamente riesgoso para la misión. Se afirma que comúnmente los requerimientos se congelan una vez se comienza con la producción del sistema. Es poco usual que un proyecto se dé el lujo de aceptar cambios en este punto, por lo que normalmente se acepta que el sistema diseñado tendrá ciertos límites o se dejan estos cambios como posibles mejoras futuras.

Otro punto importante, está relacionado con la tecnología utilizada en los sistemas. Afirman que es común escuchar que los requerimientos guían el desarrollo de la tecnología, pero en realidad es lo contrario, la tecnología más reciente normalmente permite la creación de ciertos sistemas. Sin embargo, ponen en manifiesto que suele ser riesgoso confiar en nuevas tecnologías o en nuevas técnicas de trabajo. Pero la utilización de esta tecnología también tiene sus ventajas, ya que muchas veces permiten reducciones significativas en costo, peso y volumen. Así, se debe mantener un monitoreo cercano sobre la tecnología utilizada y poseer planes de respaldo en caso de que se decida disminuir los riesgos de la misión reemplazando tecnología.

El desarrollo de una misión es un proceso altamente iterativo. Según Larson y Wertz (1999:77), cada fase de este desarrollo requiere de una buena cantidad de retroalimentación y de ajustes. Normalmente, la mayoría de la retroalimentación sucede entre fases adyacentes de desarrollo. Sin embargo, puede suceder que algunas situaciones requieran de retroalimentación a través de múltiples fases, como cuando un elemento de diseño no llega a alcanzar algún requerimiento y provoca un cambio en el diseño y en el concepto de operación.

Es importante mantener siempre un control sobre el desarrollo de los requerimientos y el costo durante todas las fases de la misión. A través de este control se puede observar que implementar soluciones de bajo costo en misiones con tecnología de punta puede dificultar bastante el desarrollo de estas; y, de hecho, muchas de estas acciones realizadas al inicio de la misión pueden causar crecimientos descontrolados en el costo durante las fases de diseño y operación. Sin embargo, se afirma que el crecimiento del costo obtenido de estimaciones tempranas durante la fase de desarrollo del concepto de la misión es causado muchas veces por algún problema controlable. Estos problemas en las etapas iniciales son causados por varias razones, de las cuales una de las principales es no tomar en cuenta todos los elementos que conformarán costos. Este error se debe muchas veces a una falta de comunicación con quienes manufacturarán y diseñarán el sistema y con quienes lo operarán. Otra causa de los problemas mencionados anteriormente se da cuando se especifica demasiado un sistema desde el inicio, lo cual provoca que no se puedan realizar cambios para reducir costos. Finalmente, la causa más probable de estos problemas surge de los grandes y descontrolados cambios que se les realizan a los requerimientos a medida que el sistema va madurando. Estos cambios pueden causar que los costos se disparen debido a constantes rediseños, los cuales conllevan a un desperdicio de tiempo y materiales. Esto último puede llevar incluso a la pérdida de un sistema bien conocido, lo cual eleva la probabilidad de que los costos se disparen a medida que el sistema diseñado se va desarrollando. Por lo tanto, a partir de todo lo mencionado se puede observar que el proceso de definir y desarrollar los requerimientos afecta el costo de una misión más que cualquier otra actividad. Los autores explican que es importante también recordar que aceptar sin mayor reflexión requerimientos del cliente pueden llevar a costos injustificables en el proyecto.

De acuerdo con Larson y Wertz (1999:78), una de las herramientas más utilizadas para desarrollar requerimientos a partir de las necesidades del usuario es el Despliegue de la Función de Calidad, o QFD por sus siglas en inglés. Esta herramienta utiliza una serie de matrices organizadas en niveles que definen las características del sistema. El primer nivel conecta las necesidades del usuario con las características técnicas del sistema. Este nivel es conocido comúnmente como Casa de Calidad.

Así, el QFD es una herramienta de estructuración que le permite a los grupos de diseñadores tomar en cuenta las necesidades del usuario y a la vez desarrollar las características del diseño que responderán estas necesidades. Se afirman que también puede utilizarse como una guía para la derivación de los

requerimientos y para analizar el impacto que ciertos cambios en los requerimientos tendrán sobre el sistema.

Larson y Wertz (1999:80) explican que otra parte importante de la definición de los requerimientos es el análisis y la medición del desempeño de estos. Para realizar esto, antes que nada, se deben descomponer todos los requerimientos del sistema a sus formas más simples, de manera que establezcan cómo deben realizarse las funciones más básicas del sistema. De igual forma, se afirma que se debe asignar la función y sus respectivos requerimientos de desempeño a un elemento de diseño más bajo.

La descomposición y distribución de los requerimientos, explicado anteriormente, comienza a nivel del sistema, en donde los requerimientos provienen de las necesidades de la misión, y continúa pasando a través de segmentos, subsistemas y finalmente diseño de componentes. Se menciona también que es importante que durante este proceso se asegure que al alcanzar requerimientos de bajo nivel se está cumpliendo también con el desempeño esperado para el próximo nivel, y que conozca en todo momento que necesidad de la misión satisface cada requerimiento. Como se podrá suponer, este es un proceso altamente iterativo.

Algo importante es definir adecuadamente las funciones que debe realizar un sistema antes de diseñarlo. Según afirman, la manera más sencilla de representar estas funciones es mediante un diagrama de bloques sobre el flujo de las funciones. Para realizar este diagrama, se debe comenzar con las funciones de primer nivel del sistema, colocándolas en el orden en que suceden. Descomponer después estas funciones permite observar cómo funciona un sistema en cada nivel. Una vez establecidas las funciones de más alto nivel es posible descomponer y analizar cada función a través del desarrollo del diagrama.

Se explica también que estos diagramas de bloques pueden ser utilizados también para presentar información sobre el sistema, así como también para controlar las secuencias de las funciones. La información que se presenta en estos diagramas puede ser información sobre las interfaces que se mueve entre las distintas funciones, información sobre el orden bajo el cual deben llevarse a cabo las funciones, o incluso información sobre fuentes y destinos de datos.

Según Larson y Wertz (1999:83), al realizarse un análisis de los requerimientos de una misión se llegará eventualmente a un medidor bien organizado y jerárquico del desempeño de los distintos segmentos del sistema y de su presupuesto. Esto se da como consecuencia de procesos iterativos que comienzan con medidores de desempeño derivados a partir de análisis, simulaciones, observación de datos y diseños conocidos y algo de experiencia en el campo. Los autores enfatizan la importancia que tiene la experiencia y las referencias pasadas en el desarrollo de los medidores iniciales del desempeño de una misión.

Sin embargo, la investigación afirma también que es importante entender la naturaleza de los medidores iniciales de desempeño. Comúnmente, estos medidores son desarrollados por ingenieros de sistemas, los cuales poseen un amplio conocimiento del sistema y sus elementos. Pero a la vez se debe tomar en cuenta que la aparición de nuevas tecnologías y de estudios de diseños de más bajo nivel puede y debería causar que los medidores de desempeño iniciales deban ser reajustados después de haber sido revisados por expertos familiarizados con subsistemas específicos. Por lo tanto, así como es importante incluir a los representantes de todos los niveles de diseño afectados en el diseño de los indicadores iniciales de desempeño, es importante también reconocer la naturaleza iterativa de estos proyectos, y que una opción que minimiza los riesgos y el costo total puede imponer restricciones más estrictas, a comparación de otras, en niveles de diseño más bajos. Encontrar un equilibrio entre los costos impuestos y los riesgos aceptados involucra un alto nivel de negociación.

Larson y Wertz (1999:88) recalcan que es imperativo que los ingenieros de sistemas comprendan profundamente cómo desarrollar y definir los requerimientos de una misión, para así lograr después distribuir y negociar los medidores de desempeño con su ayuda. Está de más decir que no lograr que un sistema alcance los valores de estos medidores puede llegar a ser un gran problema para la misión. Una definición temprana de estos medidores facilita el proceso iterativo de reajustar distribuciones, márgenes e incluso operaciones lo suficientemente temprano en el proyecto como para no causar mayores problemas de tiempo y costo.

Sin embargo, se menciona que antes de poder realizar cualquier proceso iterativo los parámetros de desempeño y sus respectivos requerimientos deben estar claramente definidos y deben poder ser rastreados fácilmente hasta la necesidad que los origina. Como se explicó anteriormente el método de Despliegue de la Función de Calidad, QFD por sus siglas en inglés, ayuda a hacer esto posible gracias a que documenta apropiadamente el enlace entre cierto requerimiento y su respectiva necesidad. Requerimientos vagos, inconsistentes o difíciles de medir llevan muchas veces a que se presente desinformación, malinterpretación, o incluso explotación. Los autores afirman que esto es especialmente cierto para áreas del sistema vitales para su correcto desempeño; áreas en las cuales la falta de pruebas tempranas utilizando prototipos puede poner en peligro toda la misión. Se reafirma nuevamente que el proceso iterativo ya mencionado requiere de negociaciones y re-negociaciones de los medidores de desempeño basados en evidencia obtenida del proceso de diseño y el descubrimiento de errores en los primeros intentos.

Otro punto importante a tomar en cuenta, de acuerdo con Larson y Wertz (1999:88), es el problema de enfrentar los aspectos de los requerimientos por nivel contra los aspectos del diseño por nivel. Explican que el diseño por nivel es un proceso integrativo lógico o una síntesis de distintos segmentos de diseño. Al definirse las funciones del sistema junto con sus respectivos requerimientos de desempeño, y al definirse

también las interfaces que requieren más atención se establece el marco de trabajo mediante el cual se decidirá cómo se diseñará cada segmento de este.

Los autores subrayan que normalmente las estimaciones iniciales de los medidores de desempeño del sistema casi nunca corresponden con las consideraciones del diseño en niveles bajos. Afirman que es claro que estas estimaciones son puntos de partida para llevar a cabo negociaciones y ajustes en los medidores de desempeño, y para reconciliar diseños iniciales del sistema con estimaciones de diseño de segmentos y de su desempeño. Los autores recomiendan que a medida que se van modificando los requerimientos se debe tener un control sobre estos mediante su documentación. Estos documentos solamente deben ser alterados cuando exista una completa trazabilidad y visibilidad del origen de la modificación para todos los interesados. Los medidores de desempeño también se irán cambiando a medida que se sigue avanzando en el desarrollo del sistema.

Larson y Wertz (1999:90) explican que, en cierto punto del desarrollo del sistema, el diseño de los márgenes de este puede causar problemas. Se afirma que comúnmente, los márgenes son obtenidos con ayuda de la estadística, por lo cual a medida que fluyen a través de varios niveles de diseño pueden causar problemas en cuanto a sobre-diseño y costos. En algunos casos sucede que los encargados del diseño pueden dificultar la realización de ajustes apropiados debido a que mantienen los márgenes en los niveles bajos del diseño, donde muchas veces pasan inadvertidos o no pueden moverse. Los autores citados mencionan que los márgenes pueden llegar a proveer ciertas contramedidas robustas contra las fallas en órbita, pero también pueden causar problemas. Así, es importante que los requerimientos principales del sistema también tengan márgenes, los cuales pueden cambiarse o derivarse hacia otro nivel, permitiendo alcanzar niveles de desempeño realistas con un mínimo riesgo.

La fuente citada previamente explica que una vez el primer ciclo de interacciones entre el sistema y sus segmentos ha establecido las mejores estimaciones de los parámetros de desempeño principales se debe continuar con las pruebas realizadas al sistema que está siendo desarrollado. La configuración del sistema debe ser validada a través de simulaciones o prototipos. Estas pruebas tempranas sobre la integración del sistema son de suma importancia ya que ayudan a desarrollar un consenso que se mantiene a través de las etapas iniciales de diseño.

De acuerdo con Larson y Wertz (1999:90) es importante que en todo momento una línea base de requerimientos comunes sirva como base de los procesos de analizar y estimar requerimientos de desempeño, interactuar y negociar con encargados de segmentos, y validar los motivos principales definidos tempranamente en la fase de diseño. Se explica que las técnicas de validación requieren de muchos escenarios específicos o situaciones puntuales para evaluar el desempeño. Satisfacer requerimientos de desempeño durante estas situaciones puntuales puede parecer suficiente, pero no lo es.

Esto se debe a que al utilizar escenarios diseñados para probar específicamente un aspecto del desempeño de un sistema puede llegar a ignorarse otros aspectos de este mismo. Se explica además que los requerimientos documentados correctamente durante la fase de diseño pueden ser la única referencia de las funciones de un sistema y de su desempeño. Así, la documentación sobre los requerimientos debe estar al mismo nivel que la fase en la cual se encuentra el diseño del sistema en cuanto a madurez, sin olvidar que también debe reflejar los resultados de análisis, negociaciones y pruebas de validación.

Los autores citados anteriormente afirman que para documentar los requerimientos efectivamente estos deben ser consistentes y completos, en relación con la madurez del sistema. Que sean consistentes significa que cada requerimiento de desempeño debe ser escrito solamente una vez, debe mantenerse en un lugar apropiado y, por último, debe ser posicionado apropiadamente en la jerarquía de los requerimientos ya establecidos. Que estén completos quiere decir que con cada nivel de requerimientos que se trabaje se debe asegurar que estos satisfacen también el siguiente nivel más alto.

Como se mencionó anteriormente, Larson y Wertz (1999:91) afirman que todos los requerimientos deben ser rastreables a medida que se van desarrollando, distribuyendo y descomponiendo. Debe recordarse que cada diseño y cada decisión tomada debe estar basada en los requerimientos, además de tomar en cuenta todos los demás requerimientos relacionados y el impacto de los cambios a través del desarrollo del sistema. Para esto mencionan los autores que cualquier método de anexión puede funcionar, siempre y cuando permita que los requerimientos puedan ser rastreados tanto hacia arriba como a través de todos los elementos. La documentación de los requerimientos debe especificar los métodos de rastreo utilizados y las formas cómo se derivaron estos. Los autores afirman también que cada ingreso de información en los documentos debe incluir información sobre el origen del requerimiento. Siempre que se produzca un requerimiento derivado deben incluirse las técnicas de análisis utilizadas. Debe tomarse en cuenta que según Larson y Wertz (1999:91) para que un requerimiento sea rastreable quiere decir que no presente ambigüedad y que sea verificable, de manera que se evite malinterpretaciones y sobreexplotaciones.

Estos autores también subrayan que las revisiones de requerimientos son necesarias durante las revisiones de diseño y que se les debe poner tanta atención como a las decisiones de diseño durante el desarrollo del sistema. La revisión de todos los requerimientos es una parte crítica de estos procesos. Durante estas revisiones es posible que se tengan que reexaminar los pilares de la misión tales como: énfasis en ciertas áreas del diseño, presupuestos de diseño, distribución de requerimientos, métodos de trabajo, etc.

Larson y Wertz (1999:91) aseguran que basados en las necesidades, los análisis, y los ejercicios de validación, los documentos de los requerimientos del sistema deben incluir cada aspecto relevante de la funcionalidad del sistema y de cómo debe realizar estas funciones. Además, ya que idealmente los

requerimientos del sistema forman la base de los requerimientos por sistema, los del sistema deben ser definidos primero. Sin embargo, una vez definidos los requerimientos por segmento es posible que los requerimientos del sistema deban ser modificados como consecuencia de cambios en el costo, problemas de interface, limitaciones de desempeño, o cambios en el cronograma. Mencionan que debe recordarse que las especificaciones de los requerimientos, a cualquier nivel, están expuestos a posibles cambios. Por lo tanto, deben ser designados preliminarmente antes de que se realicen revisiones en cada nivel de diseño. Además, durante las fases formales de diseño las especificaciones de los diseños deben estar sujetas a un riguroso control, sin importar si los requerimientos van cambiando a medida que se avanza.

Aseguran que es importante también notar que dentro de los planes del sistema derivados de los requerimientos se encuentran los planes para pruebas que reflejaran la verificación y validación de tales requerimientos. Esto es, según los autores, un reflejo de las especificaciones de las pruebas, las cuales identifican objetivos, ambientes y niveles de ensamblaje en los cuales se realizará las pruebas.

Otro punto importante a tomar en cuenta son las interfaces de un sistema. De acuerdo con Larson y Wertz (1999:92), los desarrolladores de estos muchas veces pasan por alto todas las interfaces, o solamente las asumen como interfaces externas durante las primeras fases de desarrollo del sistema. Sin embargo, afirman que es importante considerar cuidadosamente todas las interfaces, considerando la arquitectura del sistema.

E. VERIFICACIÓN Y VALIDACIÓN

Como se mencionó anteriormente, es responsabilidad de la persona encargada de ser el ingeniero de sistemas supervisar las actividades relacionadas con la verificación y la validación de los requerimientos de la misión. De acuerdo con Gebara y Spencer (2016:2), la verificación es un proceso mediante el cual se puede confirmar que un proyecto cumple con todos los requerimientos definidos al principio de este. Por otro lado, mencionan que el objetivo de la validación en un proyecto es confirmar que este puede completar todos los objetivos definidos originalmente.

Según JAXA, los requerimientos de una misión pueden verificarse a través de cuatro métodos. Estos métodos son los siguientes:

1. Análisis. En este método, la verificación del sistema creado para cumplir con los objetivos de la misión se realiza mediante cálculos utilizando modelos matemáticos cuya confiabilidad haya sido evaluada con ciertas técnicas o herramientas, como lo son reglas lógicas reconocidas en el ámbito académico. Estos modelos matemáticos pueden ser simulaciones computacionales. Este método es utilizado principalmente

cuando la verificación de algún requerimiento es difícil de realizar mediante pruebas o simples mediciones con instrumentos comunes.

2. Inspección. En este método, las verificaciones de los requerimientos pueden realizarse sin utilizar procedimientos, equipos o herramientas especiales. Esto significa que la verificación puede llevarse a cabo utilizando elementos comunes del ámbito en el que se trabaja. Un ejemplo de la utilización de este método es cuando se realiza una inspección visual de algún producto utilizando herramientas fáciles de conseguir y dibujos o documentos que especifican ciertas condiciones físicas o ciertos estándares.

3. Prueba. Con este método se verifican requerimientos de funcionalidad y resistencia al ambiente de operación utilizando hardware del sistema creado e información tomada durante algún proceso.

4. Revisión de diseño. Este método de verificación consiste en la revisión de los planos y documento de diseño del sistema creado.

Gebara y Spencer (2016:5) explican que varios de los requerimientos de un satélite tipo CubeSat pueden ser verificados y validados a través de pruebas realizadas a los distintos sub-sistemas de este, y al sistema completo integrado. Mencionan que una de las pruebas que puede utilizarse para verificar y validar al sistema ya integrado es la prueba Día-En-La-Vida, DITL por sus siglas en inglés.

De acuerdo con los autores citados anteriormente, la prueba DITL pretende ser una simulación de la misión entera del satélite. Esta simulación debe incluir el proceso de inicialización del satélite, la ejecución de sus distintas funciones, y la respuesta del satélite después de haber recibido un comando desde la estación terrestre que lo obligue a apagarse. Explicándolo más detalladamente, los autores afirman que esta prueba solamente debe incluir los comandos que normalmente se utilizarán durante la misión. Sin embargo, resaltan la importancia de probar completamente todos los modos y fases del satélite durante la prueba. Se menciona también que esta prueba debe ser cuidadosa al momento de simular la separación con el vehículo de lanzamiento y la iniciación del sistema; y que debe simular de la forma más realista posible la misión completa. Se debe mencionar también que, según la fuente citada anteriormente, la prueba DITL debe ser una versión acelerada de la misión real. Sin embargo, aseguran que la prueba no debe durar menos de 24 horas.

F. MATERIALES

De acuerdo con Fortescue, Swinerd y Stark (2011:40), uno de los problemas más comunes con los materiales utilizados en ambientes a presiones casi nulas es la gasificación o sublimación de estos. De acuerdo con estos autores, la gasificación, o sublimación, es cuando los átomos de la superficie de un

material se vaporizan cuando este se ve expuesto a una presión ambiental comparable con su propia presión de vaporización. Estas presiones comúnmente se encuentran entre los 10^{-11} y 10^{-15} Pascales, y son las presiones que pueden esperarse en las altitudes alcanzadas por los vehículos espaciales. Debe añadirse que la desgasificación aumenta a medida que la temperatura ambiente también lo hace.

Según los autores mencionados anteriormente, aunque estructuralmente la gasificación no causa ningún problema en la mayoría de los casos, la consecuente deposición de las partículas desprendidas de la superficie puede perjudicar el funcionamiento de componentes ópticos y electrónicos. Por otro lado, las capas delgadas de materiales plásticos y las capas de óxido pueden ser sensibles a la pérdida de masa, especialmente cuando tales materiales están siendo utilizados por sus propiedades térmicas. Esto último significa que si la pérdida de masa está asociada a alguna característica particular es posible que la emisividad de estos materiales cambie.

Fortescue, Swinerd y Stark (2011:40) explican también que, en los materiales plásticos, los componentes con altas presiones de vaporización tienden a evaporar más rápido. Sin embargo, la pérdida inicial de masa está asociada normalmente con la pérdida de gases absorbidos y vapor de agua.

Además, se afirma también que no es recomendable la utilización en el vacío de lubricantes utilizados en condiciones comunes por las razones mencionadas anteriormente. Comúnmente, los lubricantes utilizados normalmente poseen altas presiones de vaporización y por lo tanto sufrirían de gasificación rápidamente; empero, en este caso la pérdida de gases absorbidos y vapor de agua puede afectar su efecto como lubricante. A pesar de que pueden utilizarse aceites poco volátiles, es más común que se utilicen capas de lubricantes sólidos.

De acuerdo con JAXA, para que un material pueda ser utilizado en un satélite debe cumplir con tener una calificación A en cuanto a gasificación en el documento MSFC-HDBK-527F o en el Sistema de Información Técnica de Materiales y Procesos, MAPTIS por sus siglas en inglés, de la NASA. Para que a un material se le pueda dar tal calificación es necesario que cumpla con presentar una Pérdida Total de Masa, TML en inglés, menor o igual al 1.0% y un porcentaje de Materiales Volátiles Condensables Colectados, CVCM en inglés, menor o igual a 0.1% según los estándares presentados por el estándar ASTM-E595-84 de la NASA.

G.RECOLECCIÓN DE ENERGÍA

El sistema de recolección de energía más utilizado en satélites es compuesto por paneles solares que capturan la energía solar y convierten parte de ella en energía eléctrica. Otros tipos de sistemas de

recolección de energía eléctrica son mediante baterías cargadas, reactores nucleares o celdas de combustible (Larson, *et al*, 1999)

Las baterías son más utilizadas como un método de almacenamiento de energía que como fuente primaria de energía, ya que poseen una capacidad limitada, lo cual a su vez limita la vida útil del satélite y para que este modelo fuera útil se debería implementar una batería de alta capacidad, lo cual a día de hoy equivale a baterías de alto costo, tamaño y masa, lo cual va en contra de lo que el programa de pico-satélites tipo CubeSat representa.

Los reactores nucleares no sólo suelen ser de dimensiones y costo elevado, sino que también requiere de un conocimiento técnico muy específico respecto al tema ya que son sistemas de mucha complejidad, por lo que esta no es una opción que haya sido utilizada con anterioridad en CubeSats. Así que la opción más viable, es a la vez la más utilizada que son los paneles solares. (Larson, *et al*, 1999)

La generación de potencia eléctrica de las celdas solares depende directamente de varios factores, entre los cuales se encuentran: eficiencia, área, constante solar, el ángulo de incidencia de los rayos solares y la temperatura. (Jordan, 2006)

1. **Eficiencia.** Los paneles solares son un conjunto de celdas solares conectadas entre sí. Estas celdas solares han continuado mejorando su eficiencia de conversión de energía con el paso de los años y a día de hoy es posible encontrar celdas solares comerciales para aplicaciones espaciales con eficiencias alrededor del 30%. Este tipo de celdas son formadas de tres capas de sustratos que en conjunto permiten transformar la energía transportada por los fotones provenientes del Sol en energía eléctrica. Actualmente se encuentran en desarrollo celdas solares de 40% o hasta 50% de eficiencia, sin embargo, no se encuentran disponibles en el mercado por el momento. (King, *et al*, 2012)

2. **Área.** El área de las celdas solares impacta directamente en el rendimiento de las mismas, por lo que, a mayor área, mayor generación de energía eléctrica. Los CubeSats se trabajan en base a unidades (U), donde 1U es un satélite de 10 cm x 10 cm x 10 cm, 2U es el doble y así sucesivamente. Es importante tomar esto en cuenta, ya que el área superficial disponible de un satélite CubeSat 1U es sumamente limitada, por lo que la generación de potencia eléctrica es escasa y, por ende, también lo debe ser el consumo, esta es una de las mayores limitantes de este tipo de satélites.

3. **Constante solar.** La constante solar es un tema que ha sido estudiado ampliamente en el pasado. Esta investigación acerca de la intensidad de los rayos del sol ha permitido conocer y predecir las condiciones de operación a las cuales se enfrentará el satélite en el espacio. La intensidad de los rayos solares varía inversamente proporcional a la distancia entre el objeto y el Sol. (Clawson, *et al*, 2002) Es

decir, a mayor distancia, menor es la intensidad de la radiación proveniente de los rayos Sol. Esto es importante tomando en cuenta las características de la órbita terrestre.

La órbita terrestre tiene una forma aproximadamente elíptica alrededor del Sol, lo cual implica que en momentos se encuentra más cerca o más lejos del Sol. Esto produce una variación en la “constante solar”, que en realidad no es constante. La intensidad solar más alta registrada hasta el momento es de 1,414 W/m², mientras la más baja es de 1,322 W/m². Para estimar el rendimiento operacional del satélite se suele utilizar un valor promedio de 1,367 W/m². (Clawson, *et al*, 2002)

4. **Ángulo de incidencia.** El ángulo de incidencia de los rayos solares influye en la generación de potencia eléctrica de forma cosenoidal, por lo que al minimizar el ángulo relativo entre la perpendicular a la superficie de la celda solar y los rayos solares se maximiza la generación de potencia eléctrica. El ángulo de incidencia depende del comportamiento del satélite en el espacio, específicamente del control que este posea sobre su actitud (orientación).

La actitud del satélite es determinada a su vez por el funcionamiento del módulo de determinación y control de actitud (ADCS, por sus siglas en inglés). Por lo general los sistemas de ADCS pasivos proveen menor control sobre la actitud que los sistemas activos. Los sistemas activos que utilizan volantes de reacción o electroimanes suelen ser más caros, masivos y ocupar más espacio, lo cual a su vez puede ser contraproducente para un CubeSat. (Larson, *et al*, 1999)

En cambio, los sistemas pasivos son más económicos, menos masivos, ocupan menos espacio y son más simples de implementar, por lo que se suelen utilizar en CubeSats, especialmente los 1U. Este tipo de sistemas funciona mediante un imán que se alinea con el campo electromagnético de la Tierra, su mayor desventaja es que el control sobre la actitud del satélite es limitado y el comportamiento del mismo es, hasta cierto punto, impredecible, lo cual incrementa la complejidad de los análisis de generación de potencia eléctrica.

Los CubeSats han sido desarrollados por una gran cantidad de países durante aproximadamente 20 años, hasta el momento. Esto provee de una base sólida de investigación a partir de la cual se puede modelar el sistema y obtener resultados que ayuden a comprender el funcionamiento del satélite.

5. **Efectos de temperatura.** Los paneles solares utilizados para recolectar energía están expuestos a un ambiente bastante extremo, donde las temperaturas pueden variar en un rango bastante amplio. Al igual que la mayoría de componentes eléctricos, los paneles solares se ven afectados por la temperatura bajo la cual operan.

Se han realizado mediciones experimentales del rendimiento de los paneles solares en el espacio a diferentes temperaturas con el fin de cuantificar el impacto de la temperatura en la recolección de energía. Por lo general, se incluye un factor de temperatura que puede oscilar entre 0 y 1 para las simulaciones de potencia satélites. Comúnmente se utiliza un factor entre 0.80 y 0.98, con un valor típico de 0.85. (Larson, *et al*, 2009)

6. Simulaciones de generación de potencia. Específicamente en Suiza se desarrolló un pico-satélite conocido como SwissCube, del cual existen documentos que ayudan a llevar a cabo simulaciones del sistema de potencia. En ellos se detalla como llevaron a cabo el análisis de generación de potencia utilizando herramientas computacionales para poder estimar la potencia eléctrica generada por el satélite dependiendo del ángulo de incidencia, este documento sirve como base para el análisis de un satélite con sistema ADCS pasivo. (Jordan, 2006)

El análisis llevado a cabo consistía de en lugar de hacer girar el satélite alrededor del Sol, dejar el satélite estático, mientras se hacía rotar el Sol alrededor de él. A partir de ello surgieron varios aspectos a considerar, como que la mayor cantidad de potencia generada se da cuando el Sol tiene incidencia sobre tres caras del satélite, claro si el satélite posee celdas solares en todas sus caras. Otro aspecto a considerar es que las celdas solares fueron modeladas de tal manera que poseían un ángulo límite a cuyo ángulo estas ya no eran capaces de generar energía eléctrica. (Jordan, 2006)

Esta estimación sirve como referencia de los intervalos de potencia generada que podría tener en un CubeSat de ADCS pasivo. La potencia promedio de los CubeSats se encuentra entre 1 y 2 W, esto se debe a la influencia de la tierra sobre la órbita del satélite, ya que impide el paso de la radiación solar, creando sombra en la cual no es posible generar potencia mediante paneles solares, lo que reduce las estimaciones realizadas en Suiza en aproximadamente un tercio, considerando que la órbita del satélite es una órbita baja, comúnmente conocida como LEO (Low Earth Orbit). (Selva, *et al*, 2012)

H. MÓDULOS DEPENDIENTES DE POTENCIA

1. Carga útil. La carga útil (PLD, acrónimo en inglés) del sistema es la razón por la cual se envía el satélite al espacio. Para este satélite en particular, la carga útil la representa el conjunto de elementos que se probarán para ver si se puede efectuar el monitoreo de cianobacteria en el Lago Atitlán con un satélite futuro.

Dichos componentes son: cámara monocromática, carrusel de filtros y motor piezoeléctrico. Estos componentes requieren de energía para operar por lo cual sus características eléctricas deben ser tomadas en cuenta para diseñar el sistema de distribución de energía.

2. **Manejo de datos y comandos.** Manejo de datos y comandos (C&DH, por sus siglas en inglés) es el módulo encargado de operar el satélite mediante envío de comandos que serán procesados y afectarán la distribución de energía. Para llevar a cabo esto es necesario de una computadora a bordo (OBC, por sus siglas en inglés).

3. **Comunicaciones.** El módulo de comunicaciones (COMMS, acrónimo en inglés) es el encargado de trasladar los datos que logre obtener el satélite en órbita hacia la estación terrestre donde se decodificarán los datos y se obtendrá la información que se busca.

El modo de operación de este módulo requiere que se implemente una antena desplegable en el satélite, lo que a su vez implica un mecanismo de despliegue que necesitará de energía eléctrica para accionarse. Además, el módulo empleará un transceptor de muy alta y ultra alta frecuencia (VHF y UHF, respectivamente, por sus siglas en inglés).

4. **Térmico.** El sistema térmico (THML, acrónimo en inglés) es el que lleva a cabo el control de las temperaturas interiores y exteriores del satélite. Debido a las restricciones de masa, volumen y el tiempo de vida esperado en este tipo de satélites se suele llevar a cabo un control pasivo de la temperatura ya sea utilizando aislante o en casos, incluso sin emplear aislante. (Selva, *et al*, 2012)

El emplear un sistema de control térmico activo incrementa la complejidad del sistema, ya que requiere sensores y energía eléctrica tanto para los sensores como para el sistema térmico. Sin embargo, el legado de pico-satélites anteriores ha indicado que en caso de implementar baterías es necesario tomar en cuenta que para que estas puedan ser recargadas se necesita que permanezcan arriba de los 0 °C. (Woodbank Communications Ltd., 2005) Por lo general, las características de órbitas bajas permiten que la temperatura descienda acercándose al límite, por lo que se requiere del uso de un sistema de control activo enfocado hacia las baterías que son el elemento crítico en cuanto a temperatura.

5. **Sistema de determinación y control de actitud.** El sistema de determinación y control de actitud (ADCS) es el responsable de orientar el satélite de manera correcta para la adquisición de datos, recolección de energía y/o comunicación con la estación terrena.

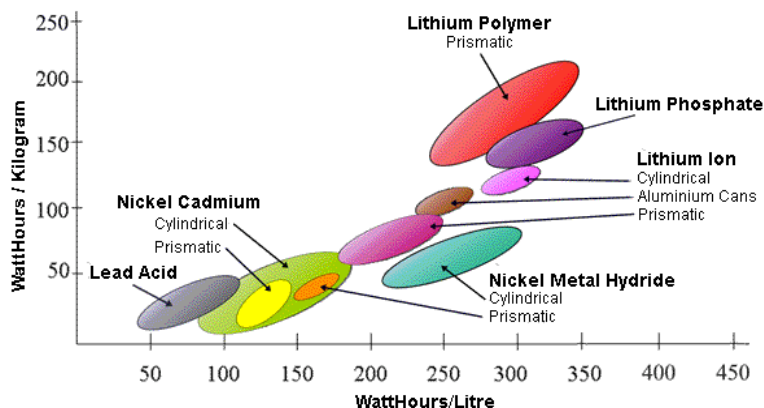
Existen dos variantes a este sistema que son: activo y pasivo. El sistema activo, requiere de un consumo de potencia más elevado que el pasivo, pero da un mejor control de la actitud del satélite. La variante pasiva suele ser más utilizada ya que esta consume menos potencia, la cual es limitada en un satélite de las dimensiones de un CubeSat 1U. Además, suele ser la alternativa más barata, de menos masa y de menor volumen, lo cual lo hace más atractivo para este tipo de satélites. No obstante, ambas variantes hacen uso de sensores como los fotodiodos y magnetómetros.

I. ALMACENAMIENTO DE ENERGÍA

El sistema de almacenamiento de energía por lo general está compuesto de un arreglo de baterías, conectadas en serie, paralelo o alguna combinación de ambas, para poder satisfacer las necesidades del satélite.

1. Tecnología de baterías. Dentro de las baterías disponibles en la actualidad existen diversas tecnologías, cada una con sus características particulares que les hacen útiles para aplicaciones a su vez particulares. Algunas de estas tecnologías son: Litio-ion, Litio-Polímero, Litio –Fosfato, Níquel-Cadmio, etc. Entre las propiedades de cada tecnología se destacan la capacidad específica (energía por unidad de masa) y densidad de energía (energía por unidad de volumen), por lo general las familias de baterías que poseen mejores propiedades también poseen un precio más elevado, por lo cual el costo se debe justificar.

Figura 3. Capacidad específica – densidad de energía por tecnología de baterías.



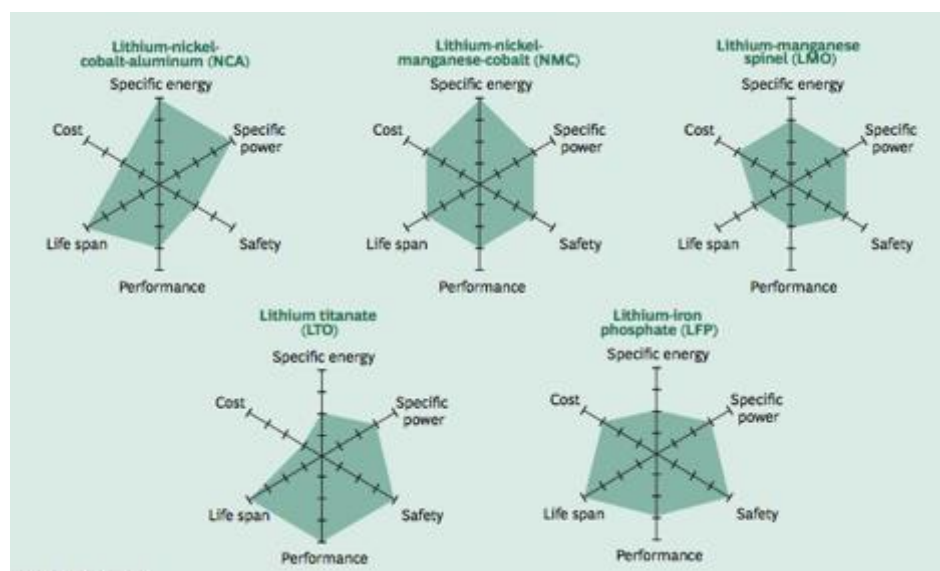
(Woodbank Communications Ltd., 2005)

El estándar para pico-satélites tipo CubeSat desarrollado por Cal Poly restringe la masa a 1.33 kg máximo y el volumen del satélite a aproximadamente 10 x 10 x 10 cm para un satélite 1U. (Cal Poly, 2009) Esto representa una gran limitación a la capacidad de las baterías lo que ha llevado a que la mayoría de CubeSats han utilizado baterías de la familia del Litio, debido a que poseen buena capacidad específica y densidad de energía a un costo accesible.

2. Familia de baterías de Litio. Dentro de la familia de baterías de litio existen cinco principales tecnologías que son: Litio-Níquel-Cobalto-Aluminio (NCA), Litio-Níquel-Manganeso-Cobalto (NMC), Litio-Manganeso (LMO), Litio-Titanio (LTO) y Litio-Hierro-Fosfato (LFP).

Las principales tecnologías de la familia del litio se pueden comparar en función de seis características: costo, capacidad específica, densidad de energía, seguridad, desempeño y tiempo de vida. (The Boston Consulting Group, 2010)

Figura 4. Características de las principales tecnologías dentro de la familia de baterías de litio.



(The Boston Consulting Group, 2010)

De acuerdo a las restricciones de volumen y masa que son de gran importancia bajo el estándar CubeSat se requiere de una alta densidad de energía y al mismo tiempo elevada capacidad específica, (Specific Power y Specific energy) de acuerdo a la imagen.

Es importante mencionar que a pesar que la mayoría de las baterías son capaces de operar en condiciones severas como las que se pueden dar dentro de un CubeSat, sí existen factores que pueden afectar su funcionamiento y la vida útil de las mismas. Esto a su vez podría significar el fin de la misión antes de lo esperado por lo que se deben tomar las precauciones pertinentes.

3. **Capacidad requerida.** La capacidad (energía almacenada) de las baterías depende del modo de operación del satélite el cual se ve afectado por el comportamiento de los demás módulos y la órbita en la cual traslada el satélite.

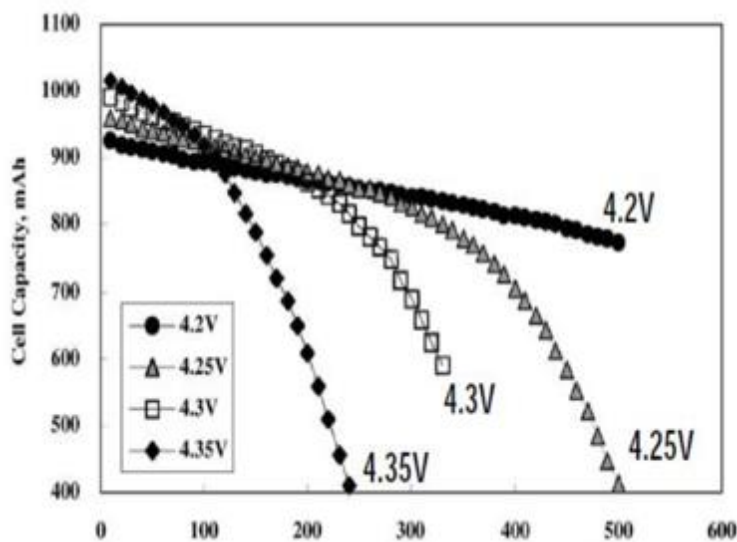
Los módulos dependientes pueden actuar ya sea en conjunto o por separado dependiendo de la función que se busca realizar. COMMS deberá estar accionado cuando el satélite se encuentre en rango de transmisión de datos con respecto a la estación terrena. PLD debe tomar fotografías y cambiar de filtro para permitir la adquisición de datos para distintas longitudes de onda. En el caso de C&DH este módulo siempre debe estar activo, ya que es el que controla la operación del resto del satélite, el elemento que requiere energía es la OBC. THML solo requerirá energía cuando la temperatura de la batería se acerque a los 0 °C.

La órbita también influye de manera considerable en la operación del satélite ya que dicta el momento y cantidad de tiempo que los distintos módulos requerirán energía, y también dicta la cantidad de energía disponible del Sol. Todos estos factores deben ser tomados en cuenta al llevar a cabo el cálculo de la capacidad requerida de la batería, de tal manera que se pueda tener certeza que el satélite tendrá energía disponible para operar incluso en la sombra (penumbra) o en las condiciones de consumo de potencia más extremas. Además, la órbita determina el perfil de temperatura del satélite por lo que a su vez determina en qué momento se requerirá energía para calentar la batería.

4. Límites de profundidad de descarga. Para definir los límites de profundidad de descarga permisibles en el sistema de almacenamiento de energía es necesario tomar en cuenta el comportamiento de las baterías ante temperatura, y uso repetitivo. Por lo general, se utilizan estos límites en el diseño de un sistema de almacenamiento de energía para prolongar la vida útil de la batería.

El llevar la batería muy cerca de su capacidad límite induce mayores esfuerzos en el ánodo y cátodo de la batería debido al incremento en el voltaje. Esto puede reducir la vida útil de las baterías, por lo cual no es recomendable el establecer un límite superior de carga cercano a la capacidad nominal de las baterías, especialmente para aplicaciones que requieren uso repetitivo de las mismas. (Battery University, 2016)

Figura 5. Capacidad – ciclos carga/descarga, para distintos voltajes de fin de carga.



(Richtek Technology Corporation, 2014)

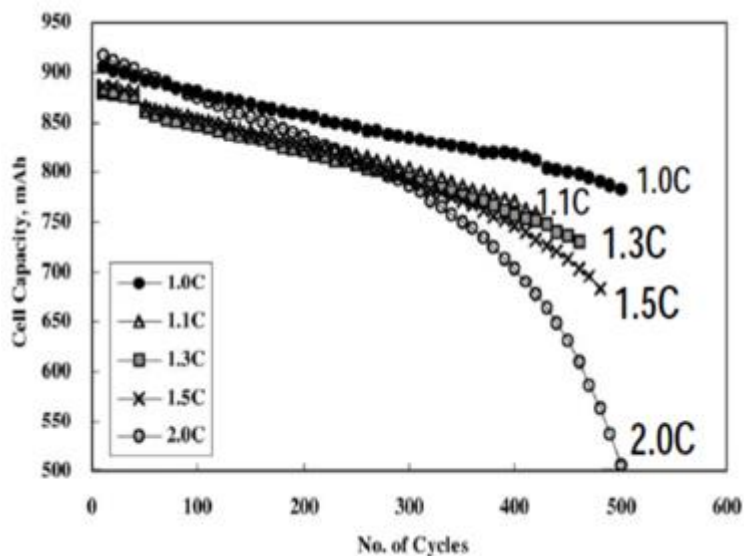
El voltaje de las celdas de las baterías está asociado con el estado de carga de las mismas. En la Figura 5, se muestra que para voltajes de fin de carga más elevados la capacidad de las celdas de las baterías se desvanece de forma más acelerada, resultado de los esfuerzos elevados en ánodo y cátodo de la batería. (Richtek Technology Corporation, 2014)

El límite inferior de carga también puede afectar de manera significativa la operación de la batería, especialmente en aplicaciones que requieren de una alta cantidad de ciclos de carga y descarga.

(Woodbank Communications Ltd., 2016)

La tasa de carga y descarga también tiene una influencia en la cantidad de ciclos que una batería puede soportar sin perder un alto porcentaje de su capacidad inicial, en lo que se conoce como desvanecimiento de capacidad. Además, se ha demostrado que el incrementar la temperatura de las baterías su capacidad tiende a desvanecerse de manera acelerada. Esto se debe a que a mayor temperatura se incrementan las reacciones químicas que pueden tener un efecto parasítico para las celdas de las baterías.

Figura 6. Capacidad – ciclos de carga/descarga, para diversas tasas de descarga.



(Richtek Technology Corporation, 2014)

La Figura 6 muestra que las baterías tienden a perder su capacidad a mayores tasas de descarga. Las tasas de descarga “C” relacionan la corriente de carga o descarga con la capacidad de las baterías. Una tasa de descarga 1C implica que la corriente de descarga causa que la batería se descargue por completo en un lapso de una hora, para 2C el lapso de tiempo es media hora, para C/2 el lapso es de 2 horas, etc.

(MIT Electric Vehicle Team, 2008)

Cuadro 1. Capacidad de baterías de Li-ion al almacenarse a diversas temperaturas.

Temperatura (°C)	40% de carga	100% de carga
0	98%	94%
25	96%	80%
40	85%	65%
60	75%	60%

(Battery University, 2016)

El cuadro anterior muestra que al almacenar las baterías a mayor temperatura su capacidad desvanece. Además, el almacenar las baterías con un 100% de carga generó un mayor desvanecimiento de la capacidad de las baterías en comparación con las baterías que fueron almacenadas con un 40% de carga.

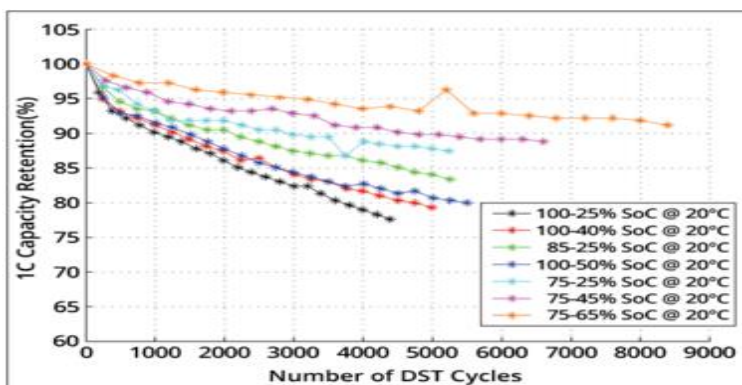
La profundidad de descarga, la diferencia porcentual de la capacidad total entre el límite superior e inferior de carga tiene una influencia elevada en el desvanecimiento de la capacidad de las baterías y, por ende, en sus ciclos útiles. En el siguiente cuadro se presenta una estimación de los ciclos de carga/descarga antes de que la capacidad de las baterías se desvanezca un 70%.

Cuadro 2. Ciclos de carga/descarga en función de profundidad de descarga.

Profundidad de descarga (%)	Cantidad de ciclos de carga/descarga
100	300 / 600
80	400 / 900
60	600 / 1,500
40	1,500 / 3,000
20	1,500 / 9,000
10	10,000 / 15,000

En el Cuadro 2 resulta evidente que una de las maneras más efectivas para incrementar la vida útil de las baterías es el reducir la profundidad de descarga de las mismas.

Figura 7. Capacidad retenida – ciclos de carga/descarga con diferentes límites de carga.

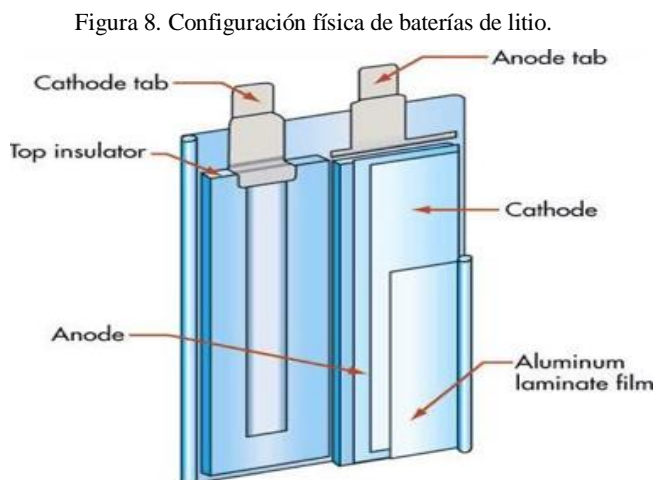


(Battery University, 2016)

A partir de la Figura 7 se observa que la capacidad de las baterías se retiene de mejor manera para una profundidad de descarga menor. Además, se observa que para profundidades de descarga iguales las baterías retienen más capacidad si oscilan alrededor del 50% del estado de carga.

5. Modos de falla de las baterías. Los modos de falla principales que se han identificado para baterías de Litio son por: voltaje, temperatura, no uniformidades y fatiga mecánica.

Si el voltaje en las celdas excede el límite operativo se pueden generar problemas. Uno de ellos es que los iones de Litio no se pueden acomodar en las capas de carbono del ánodo de manera correcta y como consecuencia se depositan como Litio metálico en la superficie del ánodo. Esto reduce la cantidad de iones de Litio libres y, por ende, la capacidad de las baterías. En casos extremos incluso se puede llegar a provocar un corto circuito entre los electrodos. Además, un voltaje muy elevado puede provocar calentamiento excesivo de las celdas. (Woodbank Communications Ltd, 2005)

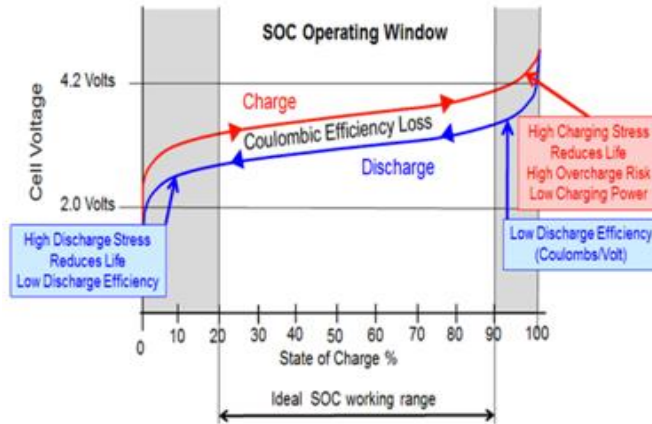


(Ebersole, 2016)

En el caso en el cual el voltaje de las celdas es bajo, ya sea por descarga excesiva o por almacenaje por tiempos prolongados, se produce un deterioro progresivo de los materiales de los electrodos. En el caso del ánodo, los iones de cobre se dispersan en el electrolito. Cuando se recarga la batería, estos iones se precipitan como cobre metálico en ubicaciones no deseadas, lo que puede causar cortocircuito entre los electrodos. Si las celdas permanecen tiempos extensos bajo condiciones de bajo voltaje se produce liberación de oxígeno y óxidos que resultan en pérdidas permanentes en la capacidad de las celdas. (Woodbank Communications Ltd, 2005)

Debido a que el voltaje de las celdas de Litio está relacionado con el estado de carga de las mismas se puede evitar estos problemas estableciendo de manera correcta los límites de carga de las baterías.

Figura 9. Rango ideal operativo de estado de carga.

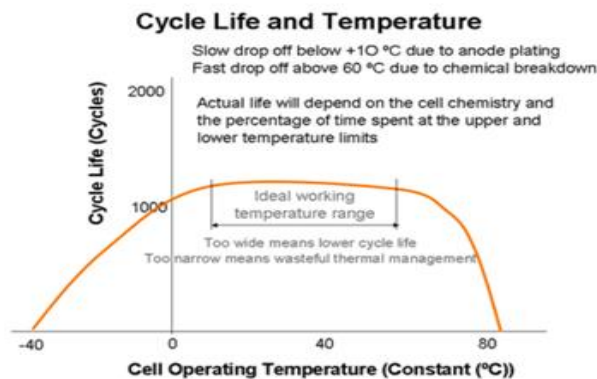


(Woodbank Communications Ltd, 2005)

La temperatura también puede causar el malfuncionamiento de las baterías y eventualmente la falla de las mismas. El exponer las baterías a bajas temperaturas reduce la tasa de carga que puede ser almacenada o extraída de las baterías. Además, las baterías de Litio a bajas temperaturas pueden sufrir deposición de Litio metálico que puede llevar al cortocircuito entre electrodos. (Woodbank Communications Ltd., 2005)

En el caso de altas temperaturas se provoca una reacción acelerada de las baterías lo que genera una alta disipación de calor, que tiende a calentar aún más las celdas. Si la temperatura es muy elevada se puede producir pasivación, luego la liberación de gases dentro de las celdas causa que se hinchen y eventualmente exploten. (Woodbank Communications Ltd., 2005)

Figura 10. Ciclos de vida – temperatura.



(Woodbank Communications Ltd. 2005)

Para evitar los problemas relacionados con temperatura en las baterías, se recomienda que las mismas se mantengan entre 10 °C y 60 °C.

Las no uniformidades pueden afectar al rendimiento de las baterías. Estas se producen como resultado de algún defecto en la manufactura de las celdas o bien por algún impacto físico significativo. En cuanto a la fatiga mecánica, esta se produce debido a la contracción y expansión que tienen naturalmente las baterías debido a su carga o descarga. En caso de tener un buen control de los límites de carga y recarga, la temperatura operacional y baterías sin no uniformidades, las baterías deberían fallar eventualmente por fatiga mecánica.

J. DISTRIBUCIÓN DE ENERGÍA

La distribución de energía se llevará a cabo con un circuito eléctrico. Este circuito eléctrico debe ser diseñado tomando en cuenta las características de los módulos dependientes, los sensores necesarios y las interfaces que tendrá el circuito con los demás módulos.

El circuito tendrá dos fuentes de energía, la principal que serán los paneles solares y la secundaria que será la batería. La batería suministrará la energía almacenada cuando no haya energía solar disponible o bien, cuando la energía solar recolectada no sea suficiente para proveer a los módulos la energía requerida.

Por lo general los pico-satélites CubeSat trabajan con componentes que requieren de 3.3 V o 5 V, además para poder llevar a cabo un mejor manejo de energía se requiere accionar los componentes solo cuando sea necesario. Esto implica que se utilicen interruptores (*switches*) que permitan o restrinjan el paso de corriente a los diversos componentes dependiendo del monitoreo y señales emitidas por la OBC.

En el caso del sistema de distribución de energía existen ciertos requerimientos específicos del estándar de Cal Poly y UNOOSA/JAXA que deben ser tomados en cuenta, de lo contrario el satélite no será aprobado para ser lanzado al espacio.

K. SISTEMA DE MANEJO DE COMANDOS Y DATOS

El sistema de manejo de comandos y datos es el "cerebro" del conjunto autónomo del CubeSat. Este sistema consta de una computadora a bordo, conocido como OBC el cual es el encargado de controlar el funcionamiento del satélite. La función de este sistema es recibir, validar, decodificar y distribuir comandos a otros subsistemas y recopilar, procesar y dar formato a los datos de funcionamiento y de misión. (Sabri, *et al*, 2016).

1. **Computadora a bordo.** Es considerado el orquestador del CubeSat y consiste esencialmente en un microcontrolador conectado a subsistemas a través de un bus de datos en serie y hardware periférico. Un sistema operativo gestiona todo el software de aplicación ejecutado en el microcontrolador y que constituye el software de vuelo de CubeSat (FSW, por sus siglas en inglés). La energía a bordo disponible es un

recurso escaso y requiere que el microcontrolador posea un consumo de energía promedio bajo y al mismo tiempo posea suficiente potencia de procesamiento para manejar todos los requerimientos de la misión. Las principales funciones de la OBC son las siguientes:

- Registro y almacenamiento de datos de telemetría y carga útil de satélites para su transmisión a la estación terrestre
- Codificación y decodificación de los paquetes de datos hacia y desde la estación terrestre.
- Seguimiento de subsistemas
- Implementación de funciones de vigilancia
- Restablecimiento de subsistemas en caso necesario. (ESA, 2014)

2. **Arquitectura de hardware.** La arquitectura del hardware se refiere a la forma en que la OBC está físicamente conectado a otros módulos de CubeSat. Hay tres tipos de arquitectura comúnmente utilizados en OBC:

Cuadro 3. Tipos de arquitectura de hardware.

Tipo	Ventajas	Desventajas
Centralizado: Tiene un microcontrolador principal que está conectado a cada módulo.	Simplicidad en la implementación. Menor consumo de espacio y potencia.	Microcontrolador principal debe poseer suficientes interfaces de comunicaciones. Si el microcontrolador principal falla, la misión termina.
Distribuido: Cada módulo tiene su propio microcontrolador.	Mayor abstracción Reducir el riesgo de falla total cuando falla un módulo	Microcontrolador principal debe poseer suficientes interfaces de comunicaciones. Si el microcontrolador principal falla, la misión termina.
Paralelo: Hay dos microcontroladores trabajando simultáneamente.	Multi tarea	Mayor consumo de energía y peso Implementación difícil.

(Sabri, *et al*, 2016)

3. **Arquitectura de software.** La arquitectura de software es un conjunto de patrones que proporcionan un marco de referencia indispensable para el desarrollo de software, permitiendo a todos los involucrados en el proyecto compartir unos mismos objetivos de trabajo y cumplir todos los requerimientos y restricciones de la aplicación. Ayuda a establecer la estructura, funcionamiento e interacción entre las partes del software. La arquitectura de software tiene la responsabilidad de:

- Definir los módulos principales
- Definir las responsabilidades que tendrá cada uno de estos módulos
- Definir la interacción que existirá entre dichos módulos
- Control y flujo de datos
- Protocolos de interacción y comunicación

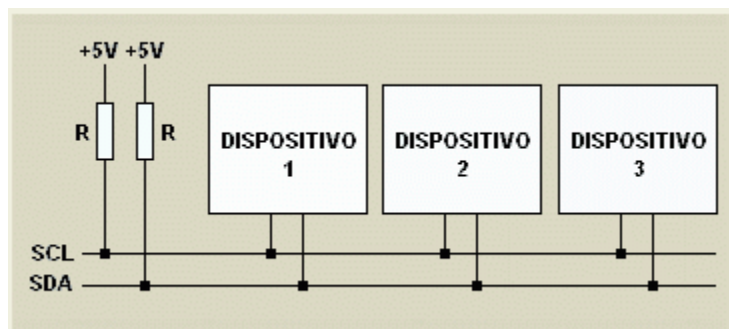
El diseño de la arquitectura de software ocurre inmediatamente después de la especificación de los requerimientos; debe considerarse una parte fundamental, crítica e imprescindible en el desarrollo de un sistema de software, es una de las fases más importantes en el desarrollo de software, pues es acá en donde se diseña la solución que cumplirá con los requerimientos funcionales y no funcionales. (Bass, *et al*, 2003)

L. I2C

Inter Integrated Circuits (I2C, por sus siglas en inglés) es un tipo de bus diseñado por *Philips Semiconductors* a inicios de los años 80's utilizado para conectar circuitos integrados. Este define un estándar que facilita la comunicación entre dispositivos electrónicos, solo requiere de dos líneas de señal, permite el intercambio de información entre muchos dispositivos a una velocidad aproximada de 100Kbits por segundos. La comunicación de datos del bus I2C es serie y sincrónica, es decir una de las señales del bus marca el tiempo y la otra se utiliza para intercambiar los datos (Robots A., 2017). A continuación, se presenta una descripción de las señales involucradas.

- SCL: (*System Clock*) es la línea de conexión de los pulsos de reloj que sincronizan el sistema.
- SDA: (*System Data*) es la línea de conexión por medio de la cual se mueven los datos entre los dispositivos.
- GND: (tierra): es una línea de conexión común entre todos los dispositivos conectados al bus.

Figura 11. Diagrama de conexión de I2C.



Es importante conocer un poco los aspectos de I2C antes de conocer su funcionamiento:

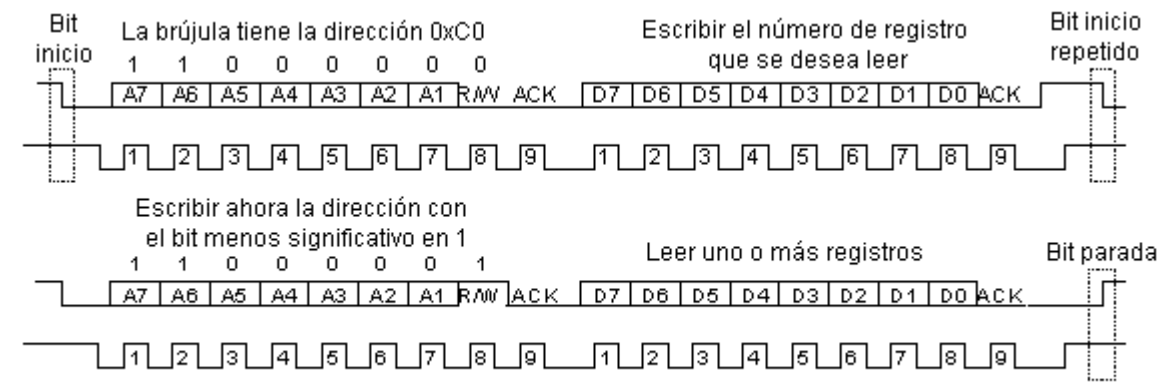
- Maestro: dispositivo que determina los tiempos y la dirección del tráfico. Es el encargado de

aplicar los pulsos de reloj de la línea SCL.

- Esclavo: cualquier dispositivo conectado al bus que no tiene la capacidad de generar los pulsos de reloj; reciben señales de comando y de reloj generadas por el maestro.
- Bus libre: estado en el que las líneas SDA y SCL están inactivas, es el momento en que un maestro puede utilizar el bus.
- Comienzo: es producido cuando un maestro ocupa el bus.
- Parada: condición generada por un maestro dejando libre el bus.
- Data valida: situación en la que un dato presente en la línea SDA es estable al tiempo que la línea SCL está a nivel lógico alto.
- Formato de Datos: la transmisión de datos a través de este bus se realiza por medio de 8 bits, a cada secuencia de bits transmitido le sigue un noveno pulso de reloj el cual pertenece al pulso de reconocimiento.
- Dirección: todo dispositivo diseñado para funcionar por medio de este bus posee su única y propia dirección de acceso.
- Lectura/Escritura: cada dispositivo tiene una dirección de 7 bits, el octavo que se envía durante la operación de direccionamiento indica el tipo de operación a realizar, si es un dispositivo maestro lee información, si es 0 el dispositivo maestro escribe información. (Robots A., 2017)

En el bus I2C el único que puede iniciar una comunicación es el maestro, la comunicación se realiza dispositivo por dispositivo, es decir, uno a la vez; la condición inicial de la comunicación es que el bus este libre, en ese momento cualquier maestro puede ocuparlo, estableciendo la condición de inicio. El primer byte de información que se transmite luego de la condición de inicio es la dirección del dispositivo con su respectiva operación (lectura/escritura). Si el dispositivo está en el bus, éste contesta con un bit en bajo ubicado después del octavo bit, este bit se conoce como el de reconocimiento (ACK) indica que se ha reconocido la solicitud y que está en condiciones de comunicarse. Si la instrucción fue de escritura, el maestro envía datos al dispositivo esclavo, manteniéndose mientras continúe recibiendo señales de reconocimiento, la conexión termina cuando se hayan transmitido todos los datos. En caso contrario, si la instrucción fue de lectura, el dispositivo maestro genera pulsos de reloj para que el esclavo pueda enviar los datos. Generando un pulso de reconocimiento luego de cada byte recibido. El maestro puede dejar libre el bus generando una condición de parada, si se desea seguir transmitiendo, el maestro puede generar otra condición de inicio en lugar de una condición de parada; esta nueva condición se puede emplear para direccionar a un esclavo diferente o para alterar el estado del bit de lectura/escritura. (Robots A., 2017).

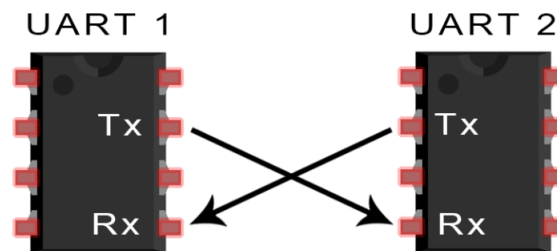
Figura 12. Ejemplo de funcionamiento del protocolo I2C.



M. UART

Sus siglas en inglés significan Transmisor-Receptor Asíncrono Universal. Utiliza transmisión de datos en serie y controla las interrupciones a los dispositivos conectados. Su objetivo principal es transmitir y recibir datos en serie. Solo se necesitan dos cables físicos para la transmisión de datos entre componentes.

Figura 13. Comunicación UART.



Como su nombre lo indica es un mecanismo asíncrono, por lo tanto, no necesita de una señal de reloj para la comunicación, en lugar de esto UART agrega bits de inicio y fin en cada paquete de datos. Cuando un dispositivo detecta el bit de inicio, este comienza a leer bits a una frecuencia determinada, esta frecuencia debe ser la misma en los dos componentes que se están comunicando. (Tomasi, 2003)

N.UML

Sus siglas en inglés significan lenguaje unificado de modelado. Es el más conocido y utilizado en la actualidad para el modelado de sistemas de software. Sirve para representar de distintas maneras gráficas un sistema, creando planos o guías para el mismo y son útiles en la documentación de un sistema. (UML-Diagrams, s.f.)

Existen tres tipos en los que se dividen los diagramas: estructurales, de comportamiento y de interacción.

1. **Diagrama de secuencia.** Corresponde al grupo de diagramas de interacción, su objetivo es darle una representación gráfica a todas las interacciones que se producen en un caso de uso y demostrar la secuencia que estas siguen. La representación más común de las interacciones son los mensajes o datos que tiene un sistema. (UML-Diagrams, s.f.)

2. **Diagrama de actividad.** Corresponde al grupo de diagramas de comportamiento, su objetivo es darle una representación gráfica a un flujo de ejecución. La secuencia y las condiciones que tiene el flujo deben representarse de forma clara. (UML-Diagrams, s.f.)

O. INTERFERENCIA ELECTROMAGNÉTICA

Señales de tipo electromagnético que provocan una perturbación en el funcionamiento normal de los equipos electrónicos o eléctricos. Las perturbaciones pueden no apreciarse externamente ya que depende de la magnitud de la interferencia y la susceptibilidad del equipo, algunas perturbaciones pueden causar daños permanentes. (Balcells, 1992)

El estudio de estas interferencias ha identificado tres partes para su interpretación:

- Origen: fuente de la interferencia.
- Medio de propagación: forma en la cual se propaga la interferencia, puede ser: por conducción e impedancia y por radiación magnética o eléctrica.
- Receptor: componente o sistema afectado.

Las interferencias electromagnéticas se pueden dar por fenómenos naturales como: carga y descarga de la atmósfera por rayos y precipitaciones estáticas, radiación del Sol, galaxias y otras fuentes cósmicas. (Tran, 1999)

P. CINEMÁTICA

Los satélites son cuerpos libres que poseen movimiento traslacional y rotacional. El movimiento traslacional depende de la dinámica de la órbita, mientras que el rotacional por la dinámica de posición del objeto. Estas dos componentes de movimiento se pueden trabajar por separado, siendo descritos por vectores, marcos de referencia y cálculos asociados.

1. **Vectores físicos.** Tienen magnitud y dirección, su concepto es independiente de un sistema de coordenadas. Se utilizan el producto escalar y cruz para operar dos vectores entre sí.

El producto escalar, o punto, es la proyección de un vector \vec{A} en un vector \vec{B} . Da como resultado un valor escalar. (Forbes, *et al*, 2013)

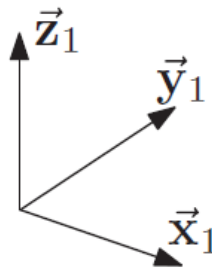
$$\vec{A} \cdot \vec{B} = |\vec{A}||\vec{B}| \cos \theta \quad (2)$$

El producto cruz, produce un vector con la magnitud dada por los vectores \vec{A} y \vec{B} , el cual es perpendicular a ambos vectores. Estos mismos vectores se pueden descomponer en vectores perpendiculares uno de otro en diferentes ejes. Es gobernado por la regla de la mano derecha.

$$|\vec{C}| = |\vec{A}||\vec{B}| \sin \theta \quad (3)$$

2. **Marco de referencia.** Al evaluar la cinemática de un satélite es necesario introducir un marco de referencia por razones computacionales, describir la orientación de un objeto y la formulación de ciertas dinámicas y cinemáticas. La base de esto son los vectores perpendiculares unitarios de los ejes \vec{x} , \vec{y} y \vec{z} .

Figura 14. Vectores perpendiculares unitarios formando la base de un marco de referencia.



(Forbes, *et al*, 2013)

Este plano nos permite describir un vector tridimensional como una combinación lineal del vector unitario, esto se denomina como vetric. Al ser un vector este se puede separar en componente de vector unitario y su componente de multiplicadores, esto permite que sea posible operar los diferentes productos. (Brathen, 2013)

$$\vec{F}_1 = \begin{bmatrix} \vec{x}_1 \\ \vec{y}_1 \\ \vec{z}_1 \end{bmatrix} \quad (4)$$

3. **Matrices de rotación.** Existen múltiples marcos de referencia para un satélite, desde su posición con respecto de la tierra, con respecto al sol y el mismo cuerpo libre. A estas referencias se les denomina

orientación o posicionamiento. De ser necesario se debe definir un marco con respecto a otro, cuya combinación es la que se denomina como matriz de rotación, la cual contiene toda la información de la referencia. Esta matriz tiene dos propiedades dominantes, ser invertible y ortogonal. (Brathen, 2013)

$$C_{21} = \vec{F}_2 \cdot \vec{F}_1^T = \begin{bmatrix} \vec{x}_2 \\ \vec{y}_2 \\ \vec{z}_2 \end{bmatrix} \cdot [\vec{x}_1 \quad \vec{y}_1 \quad \vec{z}_1] = \begin{bmatrix} \vec{x}_2 \cdot \vec{x}_1 & \vec{x}_2 \cdot \vec{y}_1 & \vec{x}_2 \cdot \vec{z}_1 \\ \vec{y}_2 \cdot \vec{x}_1 & \vec{y}_2 \cdot \vec{y}_1 & \vec{y}_2 \cdot \vec{z}_1 \\ \vec{z}_2 \cdot \vec{x}_1 & \vec{z}_2 \cdot \vec{y}_1 & \vec{z}_2 \cdot \vec{z}_1 \end{bmatrix} \quad (5)$$

a. Rotaciones principales. Estas rotaciones demuestran que, al ocurrir una rotación sobre un eje en específico, ocurre un cambio proporcional en los otros dos ejes perpendiculares. Esto forma un nuevo ángulo de referencia, el cual es el mismo que rotó el eje principal. (Brathen, 2013)

$$C_x(\theta_x) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_x & \sin \theta_x \\ 0 & -\sin \theta_x & \cos \theta_x \end{bmatrix} \quad (6)$$

$$C_y(\theta_y) = \begin{bmatrix} \cos \theta_y & 0 & -\sin \theta_y \\ 0 & 1 & 0 \\ \sin \theta_y & 0 & \cos \theta_y \end{bmatrix} \quad (7)$$

$$C_z(\theta_z) = \begin{bmatrix} \cos \theta_z & \sin \theta_z & 0 \\ -\sin \theta_z & \cos \theta_z & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (8)$$

b. Ángulos de Euler. Estos ángulos se denominan como la secuencia de posición 3-2-1. Su identificación es la siguiente:

- Ψ representa el eje z; recibe el nombre de rotación yaw.
- Θ representa el eje y; recibe el nombre de rotación pitch.
- Φ representa el eje x; recibe el nombre de rotación roll.

$$C_{21}(\phi, \theta, \psi) = C_x(\phi)C_y(\theta)C_z(\psi) = \begin{bmatrix} C_\theta C_\psi & C_\theta S_\psi & -S_\theta \\ S_\phi S_\theta C_\psi - C_\phi S_\psi & S_\phi S_\theta S_\psi + C_\phi C_\psi & S_\phi C_\theta \\ C_\phi S_\theta C_\psi + S_\phi S_\psi & C_\phi S_\theta S_\psi - S_\phi C_\psi & C_\phi C_\theta \end{bmatrix} \quad (9)$$

En la ecuación (9) dentro de la matriz de rotación se utiliza S para denominar al seno y C para el coseno. Utilizando esta secuencia, es importante especificar el orden de la rotación debido a que los resultados variaran ya que este método no es conmutativo. Una consecuencia directa es que el sistema permite que ocurran singularidades. Sin embargo, también permite encontrar los ángulos a través de los coeficientes de la matriz (9). (Fuglsang, 2010)

$$\mathbf{C} = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix}, \begin{aligned} \phi &= \tan^{-1}(C_{23}/C_{33}) \\ \theta &= -\sin^{-1}(C_{13}) \\ \psi &= \tan^{-1}(C_{12}/C_{11}) \end{aligned} \quad (10)$$

c. Cuaterniones. Debido a que los ángulos de Euler presentan singularidades se volvió necesario implementar un término redundante para la parametrización. Al volverse cuatro términos, estos se convirtieron en los parámetros de Euler o Cuaterniones. Los cuaterniones son términos con los que se pueden expresar matrices de rotación y cinemática con expresiones algebraicas puras, sin funciones trigonométricas. Estos valores son muy útiles y eficientes de utilizar para funciones computacionales. Estos tienen un eje principal y un ángulo de rotación lo que permite que haya diferentes propiedades las cuales cumplen los cuaterniones. Entre ellas está su matriz de rotación y su propiedad de magnitud.

$$\mathbf{C} = (2\eta^2 - 1) \mathbf{1} + 2 \boldsymbol{\epsilon} \boldsymbol{\epsilon}^T - 2\eta \boldsymbol{\epsilon}^x \quad (11)$$

$$\boldsymbol{\epsilon}^T \boldsymbol{\epsilon} + \eta^2 = 1 \quad (12)$$

Esta fórmula expresa que los cuaterniones son números positivos o negativos, también que son números entre 0-1. Otra propiedad útil es el cuaternión de error, la cual hace una transformación de la referencia del cuaternión. Esto quiere decir que crea un nuevo origen para la referencia del cuaternión, por lo que seguirá describiendo de la misma manera su rotación, pero ahora para una nueva orientación. Para esto se necesita el cuaternión actual y el de referencia y se utilizan de la siguiente manera. (Forbes, *et al*, 2013)

$$q_e = q_{referencia} \times q_{actual}^{-1} \quad (13)$$

4. Derivaciones de vectores. La derivación de vectores físicos es la forma de referenciarlos con respecto al tiempo. La derivada en el tiempo de un vector referenciado a un marco es la siguiente:

$$\dot{\vec{r}} \triangleq \lim_{\delta t \rightarrow 0} \frac{\delta \vec{r}}{\delta t} \quad (14)$$

Debido a esta definición, se encuentra que la derivada del producto de un vector con un escalar que varía es:

$$\overbrace{(\dot{a}\vec{r})} = \dot{a}\vec{r} + a\dot{\vec{r}} \quad (15)$$

La derivada de un vector unitario es $\vec{0}$. No obstante, la derivada de la suma de dos vectores resulta ser:

$$\overbrace{(\vec{a} + \vec{b})}^{\dot{}} = \dot{\vec{a}} + \dot{\vec{b}} \quad (16)$$

Por último, se obtiene la derivada del producto punto y el producto cruz de dos vectores. Estas fórmulas serán útiles al analizar la dinámica del satélite. (Forbes, *et al*, 2013)

$$\frac{d}{dt}(\vec{a} \cdot \vec{b}) = \dot{\vec{a}} \cdot \vec{b} + \vec{a} \cdot \dot{\vec{b}} \quad (17)$$

$$\overbrace{(\vec{a} \times \vec{b})}^{\dot{}} = \dot{\vec{a}} \times \vec{b} + \vec{a} \times \dot{\vec{b}} \quad (18)$$

a. Velocidad angular. La magnitud de la velocidad angular es la razón de rotación y su dirección es la rotación instantánea del eje. Si ocurre un ligero cambio en esta, se puede establecer la derivada de este vector.

$$\dot{\vec{v}} = \lim_{\delta t \rightarrow 0} \frac{\vec{v}(t + \delta t) - \vec{v}(t)}{\delta t} = \vec{\omega} \times \vec{v} \quad (19)$$

No obstante, este movimiento no será el mismo para cada marco de referencia, por lo que se debe obtener la derivada con respecto al marco de referencia original. Es de aquí que se obtiene el teorema de transporte, donde la derivada con respecto al marco original será $\dot{\vec{r}}$ y con respecto al marco de referencia 2 será $\dot{\vec{r}}^2$.

$$\dot{\vec{r}} = \dot{\vec{r}}^2 + \vec{\omega}_{21} \times \vec{r} \quad (20)$$

Se obtienen las velocidades angulares para cada eje de rotación.

$$\dot{\vec{x}}_2 = \vec{\omega}_{21} \times \vec{x}_2, \quad \dot{\vec{y}}_2 = \vec{\omega}_{21} \times \vec{y}_2, \quad \dot{\vec{z}}_2 = \vec{\omega}_{21} \times \vec{z}_2, \quad (21)$$

Esto es importante debido a que si \vec{r} es la posición del cuerpo, el marco de referencia original sería una referencia inercial, no una rotacional. Mientras que el marco de referencia 2 rota junto con el cuerpo en cuestión. Además, si la velocidad angular es $\vec{0}$ esto significa que los marcos de referencia no rotan con respecto al otro. En general se puede establecer que las velocidades angulares son aditivas. Consecuencia de esto la velocidad angular de un marco de referencia es la misma hacia la otra solo que con signo opuesto. (Forbes, *et al*, 2013)

$$\vec{\omega}_{31} = \vec{\omega}_{32} + \vec{\omega}_{21}, \quad \vec{\omega}_{21} = -\vec{\omega}_{12} \quad (22)$$

Se puede expresar la velocidad angular en términos de cuaterniones. Esto tendría siempre el término redundante, no obstante, facilita la forma de modelar el cambio en la orientación del satélite por lo cual es más conveniente utilizarlo.

$$\dot{q} = \frac{1}{2} \underline{\Omega}(\omega) q \quad (23)$$

, donde

$$\underline{\Omega}(\omega) = \begin{bmatrix} 0 & \omega_3 & -\omega_2 & \omega_1 \\ -\omega_3 & 0 & \omega_1 & \omega_2 \\ \omega_2 & -\omega_1 & 0 & \omega_3 \\ -\omega_1 & -\omega_2 & -\omega_3 & 0 \end{bmatrix} \quad (24)$$

b. Velocidad y aceleración. En la sección pasada se encontró que la derivada de la posición era (17), esto resulta ser también la velocidad del sistema. Este mismo proceso se puede repetir para la aceleración. $\ddot{\vec{R}}$ será la derivada de $\dot{\vec{r}}$, para la siguiente ecuación. (Forbes, *et al*, 2013)

$$\ddot{\vec{r}} = \dot{\vec{R}} + 2\vec{\omega}_{21} \times \dot{\vec{r}} + \dot{\vec{\omega}}_{21} \times \vec{r} + \vec{\omega}_{21} \times \vec{\omega}_{21} \times \vec{r} \quad (25)$$

A través de la ecuación (25) se obtienen tres términos de aceleración. El primero es la aceleración vista desde un segundo marco de referencia, el segundo término es la aceleración de Coriolis y el último la aceleración centrípeta.

Q. DINÁMICA

1. Dinámica de un cuerpo rígido. Un cuerpo rígido es un continuo en el cual la distancia entre dos puntos del mismo es siempre fija. Esto significa que el cuerpo se trabaja como un volumen sólido. (Forbes, *et al*, 2013)

a. Dinámica traslacional. La masa total sistema y su centro de masa pasan a ser resultado de una integral para encontrar un volumen.

$$m = \int_V dm, \quad \vec{r}_c = \frac{\int_V \vec{r} dm}{m} \quad (26)$$

Resultado de este se obtiene la dinámica traslacional de un cuerpo rígido dado por fuerzas externas.

$$m\ddot{\vec{r}}_c = \vec{F} = \int_V \vec{f} dV \quad (27)$$

b. Dinámica rotacional. Un requerimiento para esta dinámica es establecer un marco de referencia, este se denomina marco de cuerpo-fijo. Este marco tendrá el punto de origen al cual se le referencian cualquier cambio, es importante que el punto se encuentre dentro del cuerpo. A partir de este punto se encuentra el momento angular, dm es el elemento infinitesimal de masa.

$$\vec{h} = - \int_V \vec{\rho} \times (\vec{\rho} \times \vec{\omega}) dm \quad (28)$$

De la ecuación (28) se observa la matriz de momento de inercia del sistema:

$$J \triangleq - \int_V \rho^{\times} \rho^{\times} dm \quad (29)$$

Al reemplazar (29) en (28) se logra simplificar la ecuación de momento angular.

$$h = J\omega \quad (30)$$

En el caso de satélites, el centro de masa es un punto conveniente para colocar el origen. Esto permite que se pueda utilizar la ecuación de Euler para describir la totalidad de la dinámica de la velocidad angular de un sistema y la posición del sistema rígido. Al colocar el centro de masa como el origen del marco, se coloca una I en lugar de J para referirse a la inercia del cuerpo. (Li, *et al*, 2012)

$$I\dot{\omega} + \omega^{\times} I \omega = T_c \quad (31)$$

Las características más importantes de la matriz de inercia son:

- Es simétrica
- Es real
- Es definitiva positiva

Estas propiedades tienen como consecuencia que sus eigenvalores sean positivos y reales. Además, esta matriz permanece constante para el cuerpo rígido en cuestión. Esta matriz tiene ejes principales, los cuales son los ejes del marco de referencia lo que hacen que la matriz sea diagonal. Las coordenadas representadas son de tres vectores mutuamente perpendiculares, lo que forma una matriz de rotación. Debido a que esto se cumple para cualquier cuerpo rígido, los ejes principales permiten simplificar la dinámica rotacional considerablemente.

$$I = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix}, \omega = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, T_c = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix}$$

Sustituyendo estas matrices en (31) se obtienen las ecuaciones de Euler simplificadas.

$$\begin{aligned}
I_x \dot{\omega}_x + (I_z - I_y) \omega_y \omega_z &= T_x \\
I_y \dot{\omega}_y + (I_x - I_z) \omega_x \omega_z &= T_y \\
I_z \dot{\omega}_z + (I_y - I_x) \omega_x \omega_y &= T_z
\end{aligned}
\tag{32}$$

c. **Energía cinética.** Al analizar un cuerpo rígido usualmente se referencia cada elemento infinitesimal que lo forma tomando como referencia su origen. Este término de posición presenta una forma de encontrar la velocidad de cada elemento para posteriormente incorporarla en la energía cinética. (Forbes, *et al*, 2013)

$$\begin{aligned}
\vec{r}_{dm} &= \vec{r} + \vec{\rho} \\
\dot{\vec{r}}_{dm} &= \vec{v} + \vec{\omega} \times \vec{\rho} \\
dT &= \frac{1}{2} \dot{\vec{r}}_{dm} \cdot \dot{\vec{r}}_{dm} dm
\end{aligned}
\tag{33}$$

Para encontrar la energía cinética del cuerpo rígido como tal se debe integrar (32) con respecto todo su volumen. Obteniendo como resultado:

$$T = \frac{1}{2} \int_B \vec{v} \cdot \vec{v} dm + \int_B \vec{v} \cdot (\vec{\omega} \times \vec{\rho}) dm + \frac{1}{2} \int_B (\vec{\omega} \times \vec{\rho}) \cdot (\vec{\omega} \times \vec{\rho}) dm
\tag{34}$$

La ecuación (34) presenta tres términos, el primero es la energía cinética debido a la masa y la velocidad que la traslación le causa. El segundo es un término que desaparece debido a que el marco inercia está en el centro de masa del cuerpo. El tercero presenta la energía rotacional.

Primer término:

$$\frac{1}{2} m v^2, \quad v = |\vec{v}|
\tag{35}$$

Tercer término:

$$\frac{1}{2} \omega^T \mathbf{I} \omega
\tag{36}$$

R. PERTURBACIONES EN EL ESPACIO

Existen diferentes perturbaciones en el espacio que afecta a un satélite. Estas están presentes a través de la órbita y cambian con respecto a la misma. Estos se presentan como torques que desvían principalmente la orientación del cuerpo, no obstante, también afectan a cierto grado la posición del satélite. (Starin, 2010)

1. **Torque magnético.** Es causado por la interacción entre campo magnético de la Tierra \vec{B} y cualquier parte magnetizada del satélite \vec{m} , generalmente dipolos magnéticos por componentes electrónicos.

$$\vec{T}_m = \vec{m} \times \vec{B} \quad (37)$$

No obstante, existe un modelo simplificado de esta perturbación:

$$T_m = DB, \quad B = \frac{M}{R^3} \lambda \quad (38)$$

Donde:

- D es el momento de dipolo residual en $A \cdot m^2$, al ser la densidad de corriente en un componente electrónico.
- M es la fuerza del campo magnético de la Tierra en $T \cdot m^3$.
- R es la distancia entre el satélite y el centro de la Tierra en m.
- λ es una función adimensional que depende de los cambios de latitud magnética de la Tierra, es un multiplicador de 1 en el ecuador y 2 en los polos. Funciona como una estimación de como incrementa el campo magnético dependiendo de donde se encuentra el satélite.

2. **Torque por presión de radiación solar.** La radiación solar tiene presencia de fotones, los cuales tienen momento. Al entrar estos en contacto con una superficie de un satélite ejercen presión con un momento de $4.5 \times 10^{-6} \frac{N}{m^2}$. Es importante considerar que existen diferentes modos de contacto de esta radiación con una superficie. (Starin,2010)

- Absorción total
- Reflexión especulada
- Reflexión difusa

En un satélite están presentes ciertas cantidades de cada interacción. No obstante, la más predominante es la es la absorción total.

$$\vec{T}_s = \vec{C}_{ps} \times \vec{F}_s \quad (39)$$

Donde:

- \vec{F}_s es la presión que ejerce la radiación en la superficie del satélite.
- \vec{C}_{ps} es la interacción de la presión con relación al centro de masa.

Su modelo se simplifica de la siguiente manera:

$$T_s = \frac{\Phi}{c} A_s (1 + q)(cp_s - cm) \cos(\varphi) \quad (40)$$

Donde:

- Φ es la constante solar ajustada por la distancia del Sol en $\frac{W}{m^2}$.
- c es la velocidad de la luz ($3 \times 10^8 \frac{m}{s}$).
- A_s es el área superficial iluminada por la radiación en m^2 .
- q es el factor de reflexión, el cual es 0 para absorción total y 1 para reflexión especulada.
- cp_s es el centro de la presión de radiación solar.
- cm es el centro de masa.
- φ es el ángulo de incidencia del Sol.

3. Torque aerodinámico. Los CubeSats 1U normalmente se encuentran en orbitas bajas, alrededor de 200-2000km sobre la superficie de la tierra. Debido a que se encuentran a esa distancia existen perturbaciones a causa de residuos de atmósfera. No obstante, la densidad es tan baja que la mecánica de fluidos no aplica, es por eso que se toman un enfoque molecular bajo las siguientes asunciones: (Zagorski, s.f.)

- El momento de las partículas que tocan la superficie se pierde en la misma.
- El movimiento termal de la atmósfera es mucho menor que la del satélite.
- El satélite normalmente no gira.

Esto implica que las moléculas tendrán una velocidad orbital debido al satélite y que antes de la interacción estas no estaban en movimiento. Se forma un involucramiento de la densidad de la atmósfera, la superficie y velocidad de interacción entre el ambiente y el satélite.

$$\vec{T}_a = \vec{C}_{pa} \times \vec{F}_a \quad (41)$$

Donde:

- \vec{F}_a es la fuerza aerodinámica en la superficie del satélite.
- \vec{C}_{pa} es la zona de presión aerodinámica relación al centro de masa.

Su modelo se simplifica de la siguiente manera:

$$T_a = \frac{1}{2} \rho C_d A_r V^2 (cp_a - cm) \quad (42)$$

Donde:

- ρ es la densidad atmosférica en $\frac{kg}{m^3}$.
- C_d es el coeficiente de arrastre, es adimensional y se encuentra alrededor de 2.0 y 2.5 para un cuerpo aeroespacial.
- A_r es el área de contacto en m^2 .
- V es la velocidad del cuerpo en la órbita en $\frac{m}{s}$.
- cp_a es el centro de la presión aerodinámica.
- cm es el centro de masa.

4. Torque por gradiente de gravedad. Este torque se debe a que la fuerza gravitacional de la tierra no es constante a relación del centro de la tierra. Esta fuerza tiene un comportamiento cuadrático, lo que implica que entre más cerca de la superficie de la tierra se encuentre el objeto, mayor será la fuerza al igual que el torque. (Zagorski, s.f.)

$$\vec{T}_g = \frac{3\mu}{r^5} \int_V \vec{\rho} \times \vec{r} (\vec{r} \cdot \vec{\rho}) dm \quad (43)$$

Donde:

- μ es la constante gravitacional de la tierra. Cabe a resaltar no es la gravedad sino el producto de la masa del objeto que causa la atracción y la constante gravitatoria del mismo, es decir GM. Su valor aproximado es $3.986 \times 10^{14} \frac{m^3}{s^2}$.
- $\vec{\rho}$ es el vector de punto de masa afectado hacia el centro de masa.
- \vec{r} es el vector de posición orbital del centro de masa.
- r es la magnitud del vector \vec{r}

Su modelo se simplifica de la siguiente manera:

$$T_g = \frac{3\mu}{2r^3} |I_z - I_y| \sin(2\theta) \quad (44)$$

Donde:

- I son los momentos de inercia de los complementos del eje que roto.
- θ es el ángulo formado por el eje principal de la rotación.

S. ESTABILIZACIÓN DE POSICIÓN DE UN SATÉLITE

La razón principal para estabilizar un satélite es para direccionar un instrumento, o carga, hacia un punto específico para llevar a cabo una función. Una necesidad que surge es poder controlar de alguna manera la posición de un sistema. Podría ser de forma activa o pasiva, principalmente ligada a la aplicación

del satélite. La pasiva es utilizada para unirse y estabilizarse con la dinámica natural del satélite. En el caso de la activa utiliza actuadores para aplicar torque que corrige la posición.

Es importante tomar en cuenta que, al hablar de un control pasivo, los resultados pueden variar por $\pm 20^\circ$ y seguir siendo un resultado aceptable. No obstante, la mayoría de las misiones tienen requerimientos estrictos sobre la posición del sistema. Esto no quiere decir que se deje de lado técnicas pasivas, debido a que se debe evitar que la componente activa trabaje más de lo necesario. Este requerimiento se presenta al evaluar la energía que necesitan estos componentes a comparación de lo que se puede mantener almacenado dentro de las baterías del satélite. Los actuadores más comunes para el control activo son las ruedas de inercia y los electroimanes. (Brathen, 2013)

1. **Ruedas de inercia.** Las ruedas de inercia son actuadores le agregan cierta inercia a un satélite, no obstante, esta no es estática y se mantiene como un tipo de energía almacenada. Al ocurrir un evento donde se necesitará posicionar el satélite, estas comienzan incrementar su velocidad de giro para proporcionar el torque necesario al lado contrario de la perturbación. Una gran ventaja de este tipo de actuador es su facilidad para acoplarlo a un eje en específico. Es decir, pueden existir 3 o más ruedas de inercia en un mismo satélite y estar desacopladas totalmente una de la otra. En realidad, siempre habrá una parte de la inercia que se trasladará hacia otro eje, pero para los modelos no es un efecto significativo. (Froelich, 1959)

La rueda de inercia presenta la siguiente ecuación de movimiento:

$$J_s (\dot{\Omega} + s\Theta) = -T_c \quad (45)$$

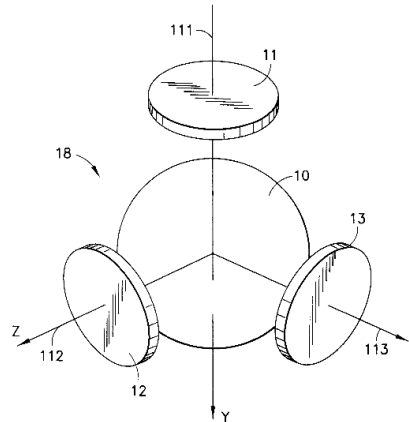
La ecuación (45) establece que el torque brindado por la rueda es igual a la suma entre la aceleración ($\dot{\Theta}$) y velocidad angular (Ω) de la misma, multiplicada por su momento de inercia (J). Este actuador también presenta la particularidad de tener un modelo simplificado para expresar su dinámica.

$$\frac{1}{\tau s + 1} \quad (46)$$

Este modelo representa la dinámica que le aportará la rueda de inercia al sistema. Es importante tomar en cuenta que τ es la constante de tiempo del sistema, es decir su tiempo de reacción. Este, por lo general, es un valor pequeño debido a que debe reaccionar muy rápido para cumplir con ciertos criterios de diseño, siendo menor 1 segundo. Cabe a resaltar que todos los actuadores tienen un límite de actuación. Esto ocurre cuando el actuador llega al límite en la cantidad de torque que puede suministrarle al sistema. La razón de la saturación de las ruedas de inercia es el hecho que algunas perturbaciones tienen una presencia constante. Al irse acumulando estos torques, llega un momento donde la rueda no tiene la capacidad de girar más

rápido, lo que se traslada a no poder controlar la posición. Es por esto que se debe diseñar un sistema de control que evite que las ruedas entreguen su torque máximo de manera abrupta. Por lo general, para eliminar la saturación de las ruedas de inercia se hace una combinación de control con electroimanes. Estos se encargan de ciertas perturbaciones y liberan a las ruedas para continuar con perturbaciones más rápidas. (Forbes, *et al*, 2013)

Figura 15. Acoplamiento y colocación de las ruedas de inercia en cada eje principal.

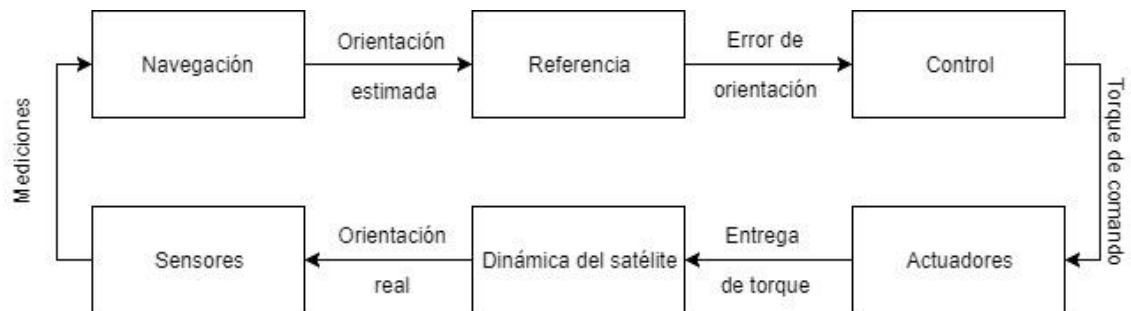


(Tilley, *et al*, 2001)

T. ESTABILIDAD EN UN SISTEMA

Todo sistema tiene una estabilidad que representa cómo reaccionará ante cierta entrada. En el caso de un CubeSat existen diferentes componentes que le agregan dinámica al sistema y esto facilite obtener una mejor estabilidad junto con los sensores, actuadores y control. (Westfall, 2015)

Figura 16. Diagrama descriptor de la función de un lazo de control cerrado para un satélite.



Al analizar la Figura 16 se observa de manera más directa que hay distintos componentes que le agregan una dinámica al sistema. Estas afectarán al comportamiento asintótico del sistema cuando el tiempo tiende al infinito, en el caso de sistemas lineales e invariantes en el tiempo. Al convertir estas dinámicas a Laplace, como la ecuación (46) estos van generando un polinomio que contienen los polos y

ceros del sistema. Estos valores son muy importantes al estudiar un sistema debido a que contienen la estabilidad del mismo y si se podrá llevar al cuerpo a la orientación se desea. (NASA, 2015)

$$X(s) = \frac{a_1 s^{n-1} + a_2 s^{n-2} + \dots + a_n}{s^n + b_1 s^{n-1} + b_2 s^{n-2} + \dots + b_n} \quad (47)$$

$$X(s) = \frac{c_1}{s - p_1} + \frac{c_2}{s - p_2} + \dots + \frac{c_n}{s - p_n} \quad (48)$$

$$x(t) = c_1 e^{p_1 t} + c_2 e^{p_2 t} + \dots + c_n e^{p_n t} \quad (49)$$

A partir de las ecuaciones (47) y (48) se puede encontrar la estabilidad del sistema:

- El sistema es inestable si alguno de sus polos es $Re(pi) < 0$, debido a que $x(t)$ crecería desmesuradamente, teniendo al infinito.
- El sistema es asintóticamente estable si sus polos son $Re(pi) > 0$, en este caso $x(t)$ converge a 0.
- El sistema es estable si, ninguno de sus polos tiene parte real positiva, pero hay distintos polos en el eje imaginario. Esto hará que $x(t)$ tenga un comportamiento oscilatorio restringido y sea estable, en caso de haber polos repetidos esto podría hacer que el sistema se vuelva inestable. (Forbes, *et al*, 2013)

1. **Función de transferencia del satélite.** Al retomar la ecuación (32) se establecían las ecuaciones de cinemática y dinámica del satélite. Estas ecuaciones son la base para obtener la función de transferencia del sistema. Si se linealiza el sistema con variaciones en ángulos muy pequeñas estas ofrecen una simplificación considerable al sistema.

$$\theta \lll 1; \phi \lll 1; \psi \lll 1;$$

$$\dot{\theta} \lll 1; \dot{\phi} \lll 1; \dot{\psi} \lll 1;$$

$$\begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}$$

(50)

Al considerar esta igualdad, se puede modificar la ecuación (32).

$$I_x \ddot{\phi} + (I_z - I_y) \dot{\theta} \cdot \dot{\psi} = T_x$$

$$I_y \ddot{\theta} + (I_x - I_z) \dot{\phi} \cdot \dot{\psi} = T_y$$

$$I_z \ddot{\psi} + (I_y - I_x) \dot{\phi} \cdot \dot{\theta} = T_z$$

(51)

Debido a que el producto punto de ángulos perpendiculares es 0, se simplifica aún más la ecuación. Es importante recordar que el torque del sistema tiene una componente controlada y otra agregada por la perturbación. (Hao, 2013)

$$\begin{aligned} I_x \ddot{\phi} &= T_{xc} + T_{xp} \\ I_y \ddot{\theta} &= T_{yc} + T_{yp} \\ I_z \ddot{\psi} &= T_{zc} + T_{zp} \end{aligned} \quad (52)$$

Un aspecto que debe tomarse en cuenta es que los ángulos de orientación del satélite son la salida que se desea, y el torque de control es la entrada que el control tiene para manejar la orientación. Por lo tanto, $\theta(t) = y(t)$, salida del sistema, $T_c(t) = u(t)$, entrada del sistema. Al convertir esta ecuación diferencial a Laplace se obtiene:

$$Y(s) = \frac{1}{I_s^2} (U(s) + T_d(s)) \quad (53)$$

Para encontrar la función de transferencia se debe hacer una de las dos componentes de entrada 0, y se evalúa la relación $\frac{\text{Salida}}{\text{Entrada}}$ de cada una. Al llevar a cabo este proceso se obtiene la función de transferencia de la planta del sistema:

$$G_p(s) = \frac{1}{I_s^2} \quad (54)$$

La ecuación (54) demuestra que los polos del sistema están en el margen de ser inestables o estables. Por lo que será tarea del control convertir este sistema en uno estable que pueda combatir perturbaciones externas. (Forbes, *et al*, 2013)

2. **Modelado del sistema.** En la mayoría de casos las funciones de transferencia no son suficientes para encontrar el comportamiento correcto de todo un sistema. Por esto mismo es importante modelar el sistema ya sea de forma lineal o no lineal. En el caso del CubeSat su modelo es no lineal, por lo que las matrices que lo representan no son valores constantes sino resultado de funciones. El CubeSat tiene una parte que depende de la derivada de su orientación y de la derivada de sus velocidades angulares. Esto se logra a través de la descripción de la velocidad angular a través de cuaterniones y también por las ecuaciones de Euler.

$$\begin{bmatrix} \dot{q} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} \underline{\Omega}(\omega) \mathbf{q} \\ \underline{I}_{sat}^{-1} [\underline{S}(\omega) (\underline{I}_{sat} \omega + \mathbf{T}_{ctrl} + \mathbf{T}_{dist})] \end{bmatrix} \quad (55)$$

donde

$$\underline{\mathbf{S}}(\omega) = \begin{bmatrix} 0 & \omega_3 & -\omega_2 \\ \omega_3 & 0 & \omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \quad (56)$$

3. Teorema de estabilidad de Lyapunov. Este teorema es una herramienta que se utiliza para encontrar la estabilidad de un sistema. Esto ocurre al haber una región donde el sistema es estable y cómo este se comporta al no iniciar en su punto de equilibrio. Para comprobar el teorema se debe proponer un candidato de Lyapunov, la cual es una función que al evaluar las variables de estado del sistema dentro de la misma se obtengan los siguientes resultados.

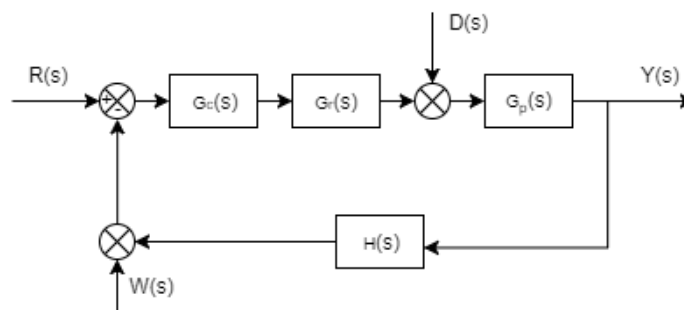
- $V(x) > 0$, si $x \neq 0$
- $V(x) = 0$, si $x = 0$
- $\dot{V}(x) = \frac{dV}{dt} f(x) \leq 0$

Si se cumple con todas estas especificaciones, entonces el sistema es globalmente asintóticamente estable. Lo que significa que no importa la condición inicial del sistema, siempre llegará al equilibrio. De no cumplirse las dos ecuaciones, no se puede decir nada de la estabilidad del sistema.

U. DISEÑO DE CONTROL

El primer paso en un sistema de control es desarrollar su diagrama de bloques que muestre la dinámica del sistema completo.

Figura 17. Diagrama de bloques del sistema de control del CubeSat.



El diagrama de bloques es importante debido a que facilita la forma de visualizar la forma en la que afecta cada entrada del sistema al satélite. El efecto de cada entrada se debe obtener asumiendo que el resto de entradas son 0 y la simplificación de los diagramas de bloque a través de técnicas se obtiene: (Forbes, *et al*, 2013)

$$Y_r(s) = \frac{G_p(s)G_r(s)G_c(s)H(s)}{1 + G_p(s)G_r(s)G_c(s)H(s)}R(s) \quad (57)$$

$$Y_d(s) = \frac{G_p(s)}{1 + G_p(s)G_r(s)G_c(s)H(s)}D(s) \quad (58)$$

$$Y_w(s) = \frac{G_p(s)G_r(s)G_c(s)H(s)}{1 + G_p(s)G_r(s)G_c(s)H(s)}W(s) \quad (59)$$

Utilizando el teorema de superposición se obtiene:

$$Y(s) = Y_r(s) + Y_d(s) + Y_w(s) \quad (60)$$

Junto con esta ecuación viene la necesidad de que la salida sea igual a la referencia, esto está directamente ligado a la función de transferencia de la planta y la del sistema de control. Debido a que no se puede cambiar la ecuación de la planta, el control debe ser modificado a discreción del diseño. En este caso en particular se desea que el control tienda al infinito para obtener las siguientes propiedades. (Li, *et al*, 2012)

$$\frac{G_p(s)G_r(s)G_c(s)H(s)}{1 + G_p(s)G_r(s)G_c(s)H(s)} \rightarrow 1, \frac{G_p(s)}{1 + G_p(s)G_r(s)G_c(s)H(s)} \rightarrow 0$$

De esta manera se puede hacer desaparecer las perturbaciones del sistema y maximizar la referencia. El mayor problema con esto es que no se puede reducir las perturbaciones sin agrandar la presencia de ruido en el sistema. Este ruido afecta directamente a los componentes electrónicos tales como la computadora a bordo y los sensores, Este es un problema que se resuelve con técnicas de filtrado, no obstante no se pueden reducir con control. Realmente este es un sistema no lineal, no obstante, la base del control permite diseñar para un sistema lineal como primer paso para luego desarrollar soluciones no lineales. Es por esto que la posición y orientación se linealizan con respecto a un punto específico.

1. Especificaciones en el dominio del tiempo. En todo sistema es importante cumplir con ciertas especificaciones para evitar que el sistema se convierta en uno inestable. La respuesta del sistema a cualquier entrada debería estar dada por una estabilidad asintótica de lazo cerrado. Esto se representa en dos estados de la respuesta; la transitoria y el estado estable. (Forbes, *et al*, 2013)

a. Especificaciones de la respuesta transitoria. Debe mostrar un comportamiento idealmente críticamente amortiguado. Si este fuera sub-amortiguado, representaría que no se está logrando alcanzar el valor deseado y que el error no se disminuye de manera rápida. En el caso del sobre-amortiguado, significa que se alcanzará más rápido el valor deseado, pero a costo de una reacción más abrupta del sistema. Esto al largo plazo puede representar que el sistema se salga de su región de estabilidad y no poder reducir el error. El resultado de un sistema pasarse del valor deseado de la respuesta es denominado sobre-alcance. También se toma en cuenta el tiempo de pico, que expresa en que tiempo se llegó al valor máximo y empezó a reducirse el sobre-alcance del sistema.

Otra especificación de la respuesta transitoria es su tiempo de subida. Esta parte está ligada a como el sistema es capaz de reaccionar ante un cambio. Expresa la velocidad con la que se va a corregir el error hasta un punto en específico.

b. Especificaciones del estado estable. El estado estable presenta el tiempo de asentamiento, este parámetro especifica el tiempo que le tarda al sistema acentuarse dentro de ciertos valores que se consideran ya es una oscilación aceptable. Este valor es muy importante porque entre más rápido se logre alcanzar, mayor será la posibilidad que se vuelva a apagar el actuador y reservar una mayor cantidad de energía. (Forbes, *et al*, 2013)

2. Tipos de controladores. Existen distintos tipos de controladores en la teoría de control para sistemas lineal e invariantes en el tiempo. La facilidad de estos tipos de control es que se pueden crear combinaciones de los mismos para ir obteniendo mejores respuestas del sistema. Para todos estos casos T es un parámetro que ayuda a posicionar los polos y ceros en puntos específicos del plano imaginario. Los principales se explican a continuación. (Forbes, *et al*, 2013) (Quadrino, *et al*, 2014)

a. Proporcional integral. Un control integral proporcional se puede describir como:

$$G_c(s) = K_{pi} \frac{s + \frac{1}{T}}{s} \quad (61)$$

Este control le agrega un cero muy cercano al eje positivo y un polo en el origen. Se encarga de reducir el error en estado estable subiendo el nivel del sistema sin afectar su eficiencia.

b. Proporcional derivativo. Un control derivativo proporcional se puede describir como:

$$G_c(s) = K_{pd} \left(s + \frac{1}{T} \right) \quad (62)$$

Este control le agrega solamente un cero al sistema por lo que reduce el número de asíntotas en la dinámica. Se encarga de mejorar la eficiencia de la parte transitoria de la respuesta.

c. Lead. Su función de transferencia se define como:

$$G_c(s) = K_{lead} \frac{s + \frac{1}{T}}{s + \frac{1}{\alpha T}} \quad (63)$$

Este controlado tiene una parte proporcional y una diferencial como el controlador PD. No obstante, su dinámica agrega un cero y un polo al sistema en lazo abierto, esto representa lo siguiente:

- No hay cambio en el número de asíntotas
- El polo está a la izquierda del cero
- Mueve el sistema más hacia el lado negativo del plano imaginario.
- Mejora la respuesta transitoria
- Puede cambiar los polos dominantes de localización

La constante α es la encargada que el polo se encuentre del lado izquierdo del cero. Este parámetro se encuentra entre $0 < \alpha < 1$.

d. Lag. Su función de transferencia se define como:

$$G_c(s) = K_{lag} \frac{s + \frac{1}{T}}{s + \frac{1}{\beta T}} \quad (64)$$

Este controlador añade un cero y un polo al sistema en lazo abierto, el polo se encuentra antes del cero. Esto implica:

- No cambia el número de asíntotas
- Puede usarse para mejorar el error en estado estable sin afectar la respuesta transitoria.

El controlador lag tiene dos restricciones:

- $\beta > 1$
- $K_{lag} \approx 1$

e. PID. Este controlador sería muy acertado para este sistema debido a que los requerimientos hacen que el sistema dependa de los polos dominantes para cumplirlos, lo cual se cumple con un PD. Además, se debe tener un error en estado estable igual a 0, lo cual se controla con un PI.

Su función de transferencia es la siguiente:

$$G_c(s) = K \left(s + \frac{1}{T_{pd}} \right) \left(\frac{s + \frac{1}{T_{pi}}}{s} \right) \quad (65)$$

f. Regulador lineal cuadrático. Existe un controlador que ofrece una solución óptima para el problema de control para los sistemas no lineales como el descrito anteriormente. Esta solución resuelve el problema de horizonte infinito solamente si se cumplen las siguientes características: (Onojo, 2016)

- $R > 0$
- $Q > 0$
- $R^{-1}B^T P x = k$
- El par (A, B) es completamente controlable y está dada por $\dot{x} = -kx$ donde $P = P^T > 0$ es la solución de la Ecuación Algebraica de Ricatti.

Esta ecuación tiene la siguiente forma:

$$0 = -A^T P - PA - Q + PBR^{-1}B^T P \quad (66)$$

V. METODOLOGÍA

A. ANÁLISIS DE RIESGOS

A continuación, se presentan las causas de cada riesgo, junto con el método utilizado para darle un valor a la probabilidad y a la consecuencia de cada riesgo utilizando estas causas. Para esto se utilizará la ecuación de ponderación presentada en la sección de Análisis de Riesgos del Marco Teórico. Los riesgos serán presentados más adelante, en la sección de Resultados.

A cada riesgo se le dio una identificación, la cual está formada por las primeras letras del nombre en inglés del módulo al que pertenecen y un número. A los riesgos del sistema en general se les dio las letras SYS; a los de Estructura, ST; a los de Control Térmico; TH, a los de Manejo de Comandos y Datos, CDH; a los de Potencia, PWR; a los de Carga Útil, PLD; a los de Determinación y control de actitud, ACS; a los de Comunicación, COMM; y a los de Gestión del Proyecto, PRG.

Las columnas “P” y “C” representan la probabilidad y la consecuencia de cada causa. La columna “Pon” representa la ponderación calculada para cada caso. Para calcular la ponderación de cada riesgo se utilizan las causas de este. Las columnas “PxPon” y “CxPon” representan el producto entre la probabilidad y la ponderación y la consecuencia con la ponderación, respectivamente.

Cuadro 4. Cálculo de ponderación de los riesgos.

Código	Causa	P	C	Prioridad	Pon	PxPon	CxPon
SYS-01	El material de la estructura no cumple con los requerimientos.	1	5	3	0.13	0.13	0.67
SYS-01	La estructura no soporta la cargas durante el lanzamiento.	3	5	1	0.4	1.20	2
SYS-01	El centro de gravedad del satélite no se encuentra en la posición necesaria.	2	5	2	0.2	0.4	1
SYS-01	La masa del satélite no cumple con los requerimientos.	1	5	3	0.13	0.13	0.67
SYS-01	Las dimensiones del satélite no cumplen con los requerimientos.	1	5	3	0.13	0.13	0.67
SYS-02	Las dimensiones del satélite no cumplen con los requerimientos.	1	5	2	0.21	0.21	1.07
SYS-02	Pérdida de conexión entre componentes.	1	5	2	0.21	0.21	1.07
SYS-02	Los componentes no permaneces fijos.	4	4	1	0.43	1.71	1.71
SYS-02	Las dimensiones de los componentes no son las que se esperaban.	1	3	3	0.14	0.14	0.43
SYS-03	Los componentes no poseen los mismos protocolos de comunicación.	1	4	2	0.27	0.27	1.09
SYS-03	Los módulos no pueden comunicarse con la computadora a bordo.	4	5	1	0.55	2.18	2.73
SYS-03	El EPS no puede producir el voltaje que los componentes necesitan.	1	3	3	0.18	0.18	0.55

Continuación Cuadro 4

Código	Causa	P	C	Prioridad	Pon	PxPon	CxPor
SYS-04	Colisión con partículas en órbita.	1	3	5	0.08	0.08	0.24
SYS-04	Las temperaturas dentro del satélite son más altas o bajas que lo esperado.	2	4	2	0.2	0.4	0.81
SYS-04	Componentes electrónicos dañados durante la misión.	1	5	4	0.1	0.1	0.5
SYS-04	Daño debido a radiación.	1	3	5	0.08	0.08	0.24
SYS-04	Daño al sensor de la cámara debido a exposición solar directa.	3	5	1	0.4	1.21	2.01
SYS-04	Los componentes o la estructura son dañados antes del lanzamiento.	2	3	3	0.13	0.27	0.4
SYS-05	Falla del vehículo de lanzamiento.	1	5	1	1	1	5
ST-01	Manejo incorrecto de la estructura.	2	3	1	1	2	3
ST-02	Los componentes no fueron fijados adecuadamente.	4	4	1	1	4	4
ST-03	El diseño de la estructura no es el adecuado para el ambiente.	3	5	1	1	3	5
ST-04	Los proveedores no suministran el material adecuado.	1	5	1	1	1	5
ST-05	Basura espacial en la órbita.	1	3	1	1	1	3
TH-01	El análisis térmico no fue realizado adecuadamente.	2	4	1	1	2	4
TH-02	El análisis térmico no fue realizado adecuadamente.	2	4	1	1	2	4
TH-03	Manejo incorrecto.	3	1	1	1	3	1
TH-04	El personal no está entrenado.	1	3	1	1	1	3
TH-05	Basura espacial en la órbita.	1	3	1	1	1	3
CDH-01	Error de programación.	3	2	1	1	3	2
CDH-02	Falla en el programa o en la comunicación.	2	3	1	1	2	3
CDH-03	Error del software.	1	2	1	1	1	2
CDH-04	Activación falsa.	2	3	1	1	2	3
CDH-05	Error en la indexación de la memoria.	1	3	1	1	1	3
CDH-06	Falta de protección por parte de la estructura.	1	3	1	1	1	3
CDH-07	Pérdida de datos de origen desconocida.	2	2	1	1	2	2
PLD-01	El movimiento del motor no es el indicado.	2	5	1	1	2	5
PLD-02	Cálculos incorrectos de la resolución espacial, la distancia focal y el espacio entre estructuras.	3	2	1	1	3	2
PLD-03	El sensor no está protegido cuando apunta directamente al Sol.	3	5	1	1	3	5
PLD-04	Falta de protección contra la radiación.	3	4	1	1	3	4
PLD-05	Elección incorrecta de cables.	1	5	1	1	1	5
PLD-06	Número-F escogido es el incorrecto para la órbita.	3	4	1	1	3	4
PLD-07	Imágenes borrosas debido a la naturaleza de la tecnología del sensor.	5	1	1	1	5	1
PLD-08	Utilización de conexiones USB con ciertos PCB.	5	5	1	1	5	5
ACS-01	Los imanes no son sujetos de la forma correcta.	2	5	1	1	2	5
ACS-02	El método de control pasivo no es el adecuado.	3	4	1	1	3	4
ACS-03	Posicionamiento incorrecto.	1	5	1	1	1	5

Continuación Cuadro 4

Código	Causa	P	C	Prioridad	Pon	PxPon	CxPor
ACS-04	Personal insuficiente.	2	5	1	1	2	5
ACS-05	La cantidad de movimiento angular del carrusel es comparable a la del satélite.	4	3	1	1	4	3
ACS-06	El sistema pasivo no permite que la cara del satélite que contiene la cámara quede frente a la Tierra.	2	4	1	1	2	4
COMM-01	No hay suficiente potencia para alimentar la radio.	5	5	1	0.48	2.4	2.4
COMM-01	La antena no se desplegó.	4	4	2	0.24	0.96	0.96
COMM-01	No hay comunicación entre la radio y la computadora a bordo.	3	4	3	0.16	0.48	0.64
COMM-01	Falla en los electrónicos de la radio.	2	5	4	0.12	0.24	0.6
COMM-02	Fallas mecánicas del mecanismo de despliegue.	5	4	1	0.67	3.33	2.67
COMM-02	Falla de la computadora a bordo.	4	4	2	0.33	1.33	1.33
COMM-03	Falla de la computadora a bordo o del radio de comunicación.	3	4	1	0.67	2	2.67
COMM-03	Falta de comunicación apropiada entre la radio y la computadora a bordo.	3	3	2	0.33	1	1
COMM-04	No se realizó un endurecimiento contra radiación.	3	3	1	0.4	1.2	1.2
COMM-04	Corto circuito o sobre voltaje en la radio.	3	3	1	0.4	1.2	1.2
COMM-04	Desconexión de equipos electrónicos.	2	4	2	0.2	0.4	0.8
COMM-05	Las instalaciones de prueba en la UVG o el equipo no son los adecuados.	5	4	1	1	5	4
PRG-01	Malos diseños.	-	-	-	-	-	-
PRG-01	Mala identificación de riesgos.	-	-	-	-	-	-
PRG-01	Alta curva de aprendizaje.	-	-	-	-	-	-
PRG-01	Falta de transferencia de conocimientos.	-	-	-	-	-	-
PRG-01	Falta de entendimiento del proyecto.	-	-	-	-	-	-
PRG-02	No hay fondos al inicio del proyecto.	-	-	-	-	-	-
PRG-02	La falta de fondos afecta el cronograma.	-	-	-	-	-	-
PRG-02	La falta de fondos afecta el diseño.	-	-	-	-	-	-
PRG-03	Pérdida de información.	-	-	-	-	-	-
PRG-03	Falta de entrega de información a otros miembros del grupo.	-	-	-	-	-	-
PRG-04	Falta de tiempo debido a proyectos de clase.	-	-	-	-	-	-
PRG-04	Dificultad de realizar un cronograma.	-	-	-	-	-	-
PRG-04	Dificultad en apegarse al cronograma.	-	-	-	-	-	-
PRG-05	Tiempo perdido durante reclutamiento.	-	-	-	-	-	-
PRG-05	Dificultad para atraer estudiantes más experimentados.	-	-	-	-	-	-
PRG-05	Demasiado tiempo consumiendo en entrenamientos.	-	-	-	-	-	-
PRG-06	Varias fuentes de requerimientos.	-	-	-	-	-	-
PRG-06	Metas inalcanzables.	-	-	-	-	-	-

B. REQUERIMIENTOS RELACIONADOS A LA MISIÓN

Estos son todos los requerimientos propios al objetivo de la misión. Este objetivo, según ha sido definido por coordinadores, es evaluar e integrar los diferentes sistemas que forman un CubeSat para desarrollar en Guatemala las habilidades requeridas para la operación de un satélite, a la vez que se validan

elementos de teledetección. Cada uno de estos requerimientos posee un código, el cual está compuesto por las letras MI y un número. Los requerimientos con un número que comienza en 0 pertenecen al módulo de carga útil; los que comienzan con un 1, a manejo de comandos y datos; los que comienzan con un 2 o con un 3, a potencia; los que comienzan con un 4 a control térmico; los que comienzan con 5, a comunicación; y los que comienzan con 6, control y determinación de orientación.

Se utilizarán las siguientes iniciales para describir de qué forman se verificarán los siguientes requerimientos: “A” cuando la verificación sea por análisis, “P” cuando es por pruebas, “R” cuando es por revisión de diseños e “I” cuando es por inspección. Para cada requerimiento puede utilizarse más de un método de verificación.

Cuadro 5. Requerimientos de la misión.

Código	Requerimiento	Verificación
MI-010	El satélite debe poseer un disco rotatorio con al menos tres filtros distintos.	R
MI-011	El carrusel de filtros debe estar hecho de un material transparente.	R, I
MI-012	El carrusel de filtro debe poder mover filtros de 25 milímetros de diámetro.	R, I
MI-013	El carrusel de filtros debe poseer un punto de unión a un eje para poder ser integrado a un motor.	R
MI-020	El satélite debe recopilar información en longitudes de onda de 443, 555 y 670 nanómetros.	R, P
MI-030	Los filtros deben poseer un Ancho-Medio Completo Máximo, FWHM por sus siglas en inglés, de 10 ± 2 nanómetros.	A
MI-040	El satélite debe poseer un motor para poder rotar el carrusel.	R
MI-050	El tiempo máximo de transición entre los filtros debe ser de 2 segundos.	P
MI-060	El lente de la cámara debe estar protegido contra rayaduras.	R
MI-070	El satélite debe tomar fotografías con todos los filtros en menos de 10 segundos.	P
MI-100	La computadora a bordo debe inicializar los módulos de Carga Útil y Comunicación cuando el satélite pase sobre Guatemala.	R
MI-110	La computadora a bordo debe revisar que todas las fotografías han sido tomadas.	P
MI-120	La computadora a bordo debe enviar las fotografías al sistema de comunicación.	R, P
MI-130	La computadora a bordo debe comparar las coordenadas globales del satélite con las de Guatemala.	R
MI-140	El protocolo de comunicación entre todos los componentes debe ser I2C.	R
MI-150	La computadora a bordo debe poseer un microprocesador redundante.	R
MI-160	El satélite debe poseer un sistema que le permita cambiar entre las dos computadoras a bordo que llevará.	R, P
MI-200	El satélite debe poseer un sistema de obtención de energía compuesto por celdas solares.	R
MI-210	El satélite debe poseer 11 celdas solares para la recolección de energía.	R
MI-211	El área de cada celda solar no debe superar los 4 000 milímetros cuadrados.	R, I
MI-220	El sistema de recolección de energía debe proveer energía suficiente como para permitir que el satélite opere durante toda la misión.	A, P

Continuación Cuadro 5

Código	Requerimiento	Verificación
MI-221	El sistema de recolección de energía debe proveerle a las baterías energía suficiente como para que el satélite pueda operar durante el periodo sin luz solar.	A, P
MI-222	Las celdas solares deben ser capaces de operar a temperaturas entre -45 y 60 °C.	R
MI-223	Las celdas solares deben poseer un diodo integrado que impida la circulación de corriente en la dirección contraria.	R
MI-224	El satélite debe poseer suficientes celdas solares como para permitir su operación en caso de que una falle.	A, P
MI-230	El ancho de cada celda solar no debe exceder los 6.5 milímetros.	I, R
MI-240	El sistema de almacenamiento debe poseer baterías como una fuente de potencia secundaria.	R
MI-250	El sistema de almacenamiento de energía debe ser capaz de almacenar al menos 0.95 Watt-Hora.	A, P
MI-260	El sistema de almacenamiento de energía debe ser capaz de operar a través de toda la misión.	A, P
MI-261	El sistema de almacenamiento debe poseer protección contra voltajes demasiado altos o bajos.	R, P
MI-262	Las baterías deben ser capaces de proveer energía junto con el sistema de recolección para cubrir la demanda de energía durante cada fase de la misión.	A, P
MI-263	Las baterías deben ser capaces de funcionar en rangos de temperatura de 10 a 50 °C.	P, R
MI-264	El sistema de almacenamiento debe poseer una profundidad de descarga (<i>depth of discharge</i>) admisible de 15%.	A, P
MI-265	Las baterías deben ser capaces de operar a través de 4 500 ciclos de cargas/descarga.	A, P
MI-270	La masa de cada batería no debe sobrepasar los 50 gramos; y la de cada celda, los 20 gramos.	I
MI-280	El tamaño del sistema de almacenamiento de energía no debe superar los 115 000 milímetros cúbicos.	I, R
MI-290	Las baterías deben ser ensambladas de manera que se permita su expansión natural durante su operación.	R, P
MI-300	Los voltajes y las corrientes de los sistemas de almacenamiento y recolección de energía deben ser monitoreados, y ambas variables deben ser reportadas por la computadora a bordo.	R, P
MI-310	Interruptores eléctricos deben ser incluidos para encender y apagar componentes a través de la computadora a bordo.	R, P
MI-320	Un dispositivo para calentar las baterías debe ser incluido de manera que estas operen dentro de sus temperaturas límites.	R, P
MI-330	La potencia de salida de las baterías debe ser regulada para poder cumplir con los requerimientos de voltaje de los componentes eléctricos.	R, P
MI-331	El sistema de almacenamiento debe poseer convertidores integrados para proveer diferentes voltajes a los componentes que lo requieran.	R, P
MI-340	El estado de carga de las baterías debe ser medido con una precisión de \pm 5% y reportado por la computadora a bordo.	P
MI-350	La eficiencia entrada-salida del sistema de regulación y distribución de potencia eléctrica debe ser superior al 75%.	A, P
MI-400	La temperatura interna del satélite debe estar entre los 0 y los 60 °C.	A
MI-500	El módulo de comunicación debe proveer un enlace para transmitir datos y enviar comandos desde y hacia el satélite.	R, P
MI-510	El módulo de comunicación debe transmitir y enviar datos a la estación en Tierra utilizando una frecuencia entre 395 y 440 MHz.	R, P
MI-520	El satélite debe utilizar una tasa de descarga de datos de por lo menos 9.6 kbps.	R, P
MI-530	El satélite debe utilizar el protocolo AX.25.	R, P

Continuación Cuadro 5

Código	Requerimiento	Verificación
MI-540	El satélite debe poseer y utilizar una identificación para satélites única en la telemetría de descarga.	R
MI-550	El satélite debe transmitir los valores actuales importantes de sus módulos y su identificación única satelital por lo menos una vez cada 30 segundos si existe la suficiente energía como para hacerlo.	R
MI-560	El satélite debe poseer una antena omnidireccional desplegable.	R, P
MI-600	El sistema de Control y Determinación de Posición debe asegurar la alineación con el campo magnético de la Tierra con una precisión de $\pm 15^\circ$.	P
MI-610	La alineación con el campo magnético de la Tierra debe ocurrir dentro de un periodo de 7 días después de haber sido expulsado de la Estación Espacial Internacional.	P

C. REQUERIMIENTOS DE DISEÑO

El siguiente grupo de requerimientos está formado por los requerimientos concernientes al diseño de todos los componentes del satélite. La mayoría de estos requerimientos fueron obtenidos del manual para la acomodación de la carga útil de la Agencia de Exploración Espacial Japonesa.

Para los códigos de estos requerimientos se utilizará el mismo principio explicado anteriormente, con la diferencia de que las letras utilizadas serán “DE” en lugar de “MI”. Se incluirá también una columna para detallar a qué módulo pertenece cada uno de los requerimientos; la columna utilizará el mismo código de letras que se presentó en la sección de Análisis de Riesgos para nombrar a cada módulo. La numeración de estos requerimientos no dependerá del módulo al que pertenece, como sucedió con los anteriores, sino que dependerá del grupo al cual pertenece según características que compartan.

En cuanto a la verificación de los requerimientos, se realizará utilizando el mismo principio que el que se explicó anteriormente.

Cuadro 6. Requerimientos de diseño del satélite.

Código	Requerimiento	Módulo	Verificación
DE-010	Todos los ejes del sistema de coordenadas del satélite deben estar alineados con los del Contenedor de Instalación de Satélites del J-SSOD.	ST	R
DE-020	El eje +Z debe apuntar hacia la dirección del despliegue, el eje -Z hacia la dirección de instalación del contenedor, y el eje +Y hacia el punto-base del contenedor.	ST	R
DE-030	El satélite debe ser de 100 ± 0.1 milímetros de ancho en los ejes X y Y.	ST	R, I
DE-040	El satélite debe ser de 113.5 ± 0.1 milímetros de alto en el eje Z.	ST	R, I
DE-050	El satélite debe poseer 4 rieles en cada esquina a lo largo del eje Z.	ST	R
DE-060	Los rieles deben poseer un espesor mínimo de 8.5 milímetros.	ST	R, I
DE-070	Los rieles no deben poseer una dureza superficial mayor a Ra1.6 micrómetros.	ST	R
DE-080	Los bordes de los rieles deben ser redondeados con un radio de por lo menos 1 milímetro.	ST	R, I
DE-090	Los bordes de los rieles en la cara +Z deben poseer un área superficial mínima de 6.5×6.5 milímetros.	ST	R, I
DE-100	Por lo menos 75% de superficie de los rieles, con excepción de las caras del eje Z, deben estar en contacto con las guías de los rieles del Contenedor de Instalación del Satélite del J-SSOD. Esto significa que por lo menos 85.1 milímetros de los rieles deben estar en contacto con las guías.	ST	I

Continuación Cuadro 6

Código	Requerimiento	Módulo	Verificación
DE-110	La superficie de los rieles que está en contacto con las guías y los que contactan con satélites adyacentes deben ser de aluminio anodizado duro después del maquinado. El espesor de la capa del anodizado duro debe ser mayor a 10 micrómetros.	ST	R
DE-120	El envoltorio dinámico del satélite deber cumplir con los requisitos dados por JAXA.	ST	R
DE-130	La estructura principal del satélite debe estar separada por más de 7 milímetros de los bordes de los ejes en la dirección Z+. Todos los componentes deben estar separados por más de 0.5 milímetros de los bordes de los ejes en la misma dirección.	ST	R, I
DE-140	La estructura principal de satélite debe estar separada por más de 6.5 milímetros de los bordes del riel en la dirección -Z. Todos los componentes deben estar separados de los bordes de los rieles en la misma dirección.	ST	R, I
DE-150	La estructura principal del satélite no debe sobrepasar las superficies laterales de los rieles en las direcciones $\pm X$ y $\pm Y$. Cualquier componente en estas superficies no debe exceder 6.5 milímetros normales a las superficies laterales de los rieles incluyendo el pin RBF.	ST	R, I
DE-160	Todos los componentes desplegables deben estar sostenidos por el satélite.	COMM	R
DE-170	Si algún componente desplegable pudiera hacer contacto con una de las paredes internas del Contenedor de Instalación del Satélite al desplegarse accidentalmente, la superficie de contacto de tal componente debe poseer un espesor mayor a 1 milímetro.	COMM	R, I
DE-180	La masa del satélite debe ser mayor a 0.13 kilogramos y menor a 1.33 kilogramos para satélites 1U.	Todos	I
DE-190	El número balístico del satélite en la configuración en la cual es colocado dentro del Contenedor de Instalación del Satélite no debe ser mayor a 100 kilogramos por metro cuadrado.	ST	A
DE-200	El centro de gravedad del satélite debe estar posicionado dentro de una esfera de 20 milímetros desde su centro geométrico.	ST	A
DE-210	A modo de resortes de separación, el satélite debe poseer dos émbolos de resorte (proveídos por JAXA) colocados en la superficie diagonal de un par de rieles seleccionados. Estos émbolos de resorte deben estar colocados firmemente a la superficie mencionada.	ST	R
DE-220	Todos los componentes que deban ser manipulados después de la instalación del satélite dentro del Contenedor de Instalación del Satélite deben estar localizados en el área de la ventana de acceso.	Todos	R
DE-230	El satélite debe poseer suficiente resistencia estructural, con un margen de seguridad prudente, como para operar a través de las operaciones en Tierra, las pruebas, el manejo en Tierra, el lanzamiento, y la operación en órbita.	ST	A
DE-240	Cada riel debe poseer la resistencia estructural suficiente como para soportar una fuerza de compresión de 46.6 N.	ST	A
DE-250	La frecuencia fundamental mínima del satélite no debe ser menor a 100 Hz bajo una condición en la cual las cuatro bases de los rieles están rígidamente fijados. En caso contrario, se necesita contactar a JAXA.	ST	A
DE-300	El satélite debe poseer dos o más interruptores de despliegue en las bases de los rieles en la dirección -Z. Estos interruptores de despliegue pueden estar instalados a los lados de los rieles (en la dirección X o/y Y) si no hay impacto en las condiciones de despliegue.	PWR	R

Continuación Cuadro 6

Código	Requerimiento	Módulo	Verificación
DE-310	Cuando uno de los interruptores de despliegue permanece despresionado el satélite no debe activarse. La definición de despresionado es que los interruptores estén como máximo a 0.75 milímetros de la superficie de las bases de los rieles. Cuando los interruptores de despliegue estén localizado a los lados de los rieles, estos interruptores no deben ser activados antes del despliegue.	PWR	R, I, P
DE-320	Si es necesario, el cargador de la batería debe ser deshabilitado cuando los interruptores de despliegue están despresionados.	PWR	R
DE-330	El recorrido del interruptor de despliegue debe ser menor a 2 milímetros desde la superficie de la base del riel.	PWR	R, I
DE-340	La fuerza generada por el interruptor de despliegue no debe ser mayor a 3 N para cada uno.	PWR	P
DE-350	El satélite debe poseer por lo menos tres inhibidores para su activación por una celda solar o una batería. Uno de los inhibidores debe estar localizado en el regreso del circuito a tierra.	PWR	R
DE-360	Si es imposible colocar tres interruptores de despliegue en el satélite un pin RBF puede ser utilizado.	PWR	R
DE-370	El pin RBF debe ser accesible desde la ventana de acceso.	PWR	R
DE-380	El pin RBF debe cortar toda la alimentación de energía una vez insertado en el satélite.	PWR	R, P
DE-390	El pin RBF debe estar dentro del envoltorio mostrado por JAXA una vez completamente insertado.	PWR	R
DE-400	Un cordón debe ser atado al pin RBF. Este cordón no está sujeto a los requerimientos del envoltorio, pero el satélite debe poder ser cargado dentro del Contenedor de Instalación del Satélite con el cordón atado.	PWR	R
DE-410	El satélite debe poseer una interface de conexión al lado de la ventana de acceso en caso se requiera de un acceso cuando ya esté instalado en el Contenedor de Instalación del Satélite.	ST	R
DE-420	Si una frecuencia de descarga menor a 110 MHz es utilizada, la corriente máxima en los circuitos no debe exceder los 50 mA.	COMM	R, I
DE-430	Los niveles de radiación de las frecuencias de radio no deben exceder los valores presentados por JAXA.	COMM	R, I, A
DE-500	Materiales con calificación "A" deben ser utilizados para el satélite.	Todos	R
DE-510	Si existe la posibilidad de que algún material se rompa, este debe estar contenido o debe utilizarse algún método para evitar que se rompa.	Todos	R, P
DE-520	Para los materiales que no cumplan con el estándar SSP 30233 un informe MUA debe ser enviado.	Todos	R
DE-530	Materiales permitidos de acuerdo con el país en el cual se hará el lanzamiento deben ser utilizados.	Todos	R
DE-540	Todos los PCB utilizados en el satélite deben haber sido utilizados previamente en misiones espaciales.	ST	R
DE-600	Todos los bordes deben estar redondeados o aplanados en más de 0.7 milímetros.	ST	R, I
DE-610	Agujeros sin cobertura deben tener un diámetro igual o mayor a 25 milímetros, o igual o menor a 10 milímetros.	ST	R, I
DE-620	Las baterías deben cumplir con los requerimientos de la forma JSC 1230, 9.	PWR	R, I
DE-630	Las partes del satélite que roten deben estar contenidas en un espacio cerrado.	PLD	R
DE-640	Ninguno de los componentes que roten deben superar los 200 milímetros de diámetro ni las 8 000 RPM de velocidad.	PLD	R, P
DE-650	La sección transversal mínima del satélite no debe ser menor a los 100 centímetros cuadrados.	ST	A, I, R
DE-700	El campo magnético afuera del envoltorio estático del satélite debe ser menor al de la Tierra más 0.5 Gauss.	ACS	R

D. REQUERIMIENTOS OPERACIONALES

El último grupo de requerimientos son los operacionales. Estos requerimientos son aquellos con los cuales el satélite debe cumplir una vez terminada la fase de diseño y construcción. Se utilizará el mismo principio para los códigos, con la diferencia de que las letras utilizadas serán “OP”. Estos requerimientos fueron obtenidos de los documentos proveídos por JAXA y por la Oficina del Programa de la Estación Espacial de la NASA.

Cuadro 7. Requerimientos operacionales de la misión

Código	Requerimiento	Módulo	Verificación
OP-010	El proveedor del satélite debe asumir que el tiempo máximo de almacenamiento después de entregado puede ser de un año.	Todos	R
OP-020	El proveedor del satélite no debe programar ninguna activación, revisión o mantenimiento después de que este haya sido instalado en el Contenedor de Instalación del Satélite.	Todos	R
OP-030	El satélite debe poseer la capacidad de sobrevivir en el ambiente frío durante el lanzamiento.	TH	R, A
OP-040	El satélite debe mantenerse desactivado desde la instalación en el Contenedor de Instalación del Satélite en Tierra hasta su despliegue.	PWR	R, P
OP-050	Todos los componentes desplegables deben esperar para ser desplegados por lo menos 30 minutos después de que los interruptores de despliegue hayan sido activados durante la eyección del satélite del contenedor. Cuando cualquiera de los dos interruptores vuelva a des-presionarse el contador debe reiniciarse.	CDH	R, P
OP-060	Se debe esperar por lo menos 30 minutos desde que los interruptores son activados durante la eyección para transmitir frecuencias de radio. Si alguno de los interruptores vuelve a ser des-presionado el contador debe reiniciarse.	CDH	R, P
OP-100	La vida del satélite hasta la re-entrada debe ser igual o menor a 25 años.	Todos	A
OP-110	Una destrucción intencional no debe ser planeada ni conducida.	Todos	R
OP-120	Todos los módulos deben ser diseñados para prevenir explosiones y/o rupturas accidentales al final de la misión.	Todos	R
OP-130	No debe existir una probabilidad mayor 1/10 000 de que alguien resulte herido en Tierra.	Todos	A
OP-140	Cualquier componente desplegable solamente debe ser desplegado después de que el satélite ha alcanzado una distancia mayor o igual a 500 km de la ISS.	CDH	A
OP-150	Cualquier componente desplegable solamente debe ser desplegado después de que el apogeo del satélite sea menor que el perigeo de la ISS.	CDH	A, P
OP-160	El satélite debe ser diseñado de tal forma que no libere desechos durante su operación.	Todos	R
OP-170	La probabilidad de rupturas durante la operación debe ser minimizada.	Todos	R, P
OP-200	El satélite debe soportar aceleraciones cuasi estáticas en cualquier dirección máxima de 18.1 g.	ST	A
OP-210	El satélite debe soportar una serie de vibraciones aleatorias especificadas por JAXA.	ST	P
OP-220	El satélite debe soportar presiones tan altas como 104.8 kPa y tan bajas como 0 Pa.	ST	P
OP-230	El satélite debe soportar tasas de despresurización de por lo menos 1.33 kPa/segundo.	ST	P
OP-231	Se necesita de un análisis estructural si la relación entre el volumen interno del satélite y el área de los puestos de escape del satélite es mayor a 50.8 metros.	ST	R, I
OP-240	El satélite debe soportar temperaturas tan bajas como -15 °C y tan altas como 60 °C.	TH	R, P

Continuación Cuadro 7

Código	Requerimiento	Módulo	Verificación
OP-250	El satélite debe soportar el ambiente húmedo presente durante el transporte, el almacenamiento y su estadía en la ISS.	Todos	R
OP-300	Las conexiones eléctricas en el satélite deben realizarse a manera de minimizar las diferencias de potencial.	PWR	R
OP-310	Cuando se utilice una conexión simple para dos o más clases de aplicaciones, el diseño debe cumplir con los requerimientos más estrictos para conexiones.	Todos	R
OP-320	Conductores que contenga cableados eléctricos deben poseer una conexión de baja resistencia, menor a 0.1 ohm, con las estructuras conductoras en cada terminación y punto de ruptura.	PWR	R, I
OP-330	Los marcos conductores expuestos o las partes eléctricas o de equipos electrónicos deben tener un enlace de baja resistencia, menor a 0.1 ohm, con una estructura conductora.	PWR	R, I
OP-340	Todas las unidades eléctricas y electrónicas o componentes que utilicen o produzcan energía electromagnética deben instalarse de manera que generen un camino continuo de baja impedancia desde el compartimento del equipo hasta la estructura conductiva.	PWR	R, A
OP-341	El método de conexión propuesta debe resultar en una resistencia DC menor a 2.5 miliohms a través de cada superficie en el camino de la conexión desde el compartimento a la estructura y una impedancia menor a 100 miliohms hasta una frecuencia de 1 MHz.	PWR	I, A
OP-350	En todos los casos en los que la Clase S es aplicable la metodología de la conexión debe asegurar que ninguna superficie conductora con un área mayor a 200 centímetros cuadrados carezca de una conexión desde la capa de conducción hasta la estructura conductora.	Todos	R, I
OP-351	La resistencia de la conexión desde el punto de conexión hasta la estructura conductora debe ser menor a 1 ohm.	Todos	A, I
OP-360	Las capas de conducción deben estar conectadas entre sí en por menos dos puntos.	Todos	R
OP-361	La resistencia de la junta desde estos puntos hasta la estructura conductora debe ser menor a 1 ohm.	Todos	A, I
OP-370	Debe evitarse la utilización de cables tipo <i>jumpers</i> .	Todos	R
OP-380	Los cables tipo <i>jumpers</i> que pasen a través de una junta vibratoria móvil o con aislamiento térmico deben cumplir con los requerimientos apropiados de resistencia/impedancia para conexiones.	Todos	R
OP-390	Los cables tipo <i>jumpers</i> deben mantenerse tan cortos como sea posible y tener una relación largo/ancho menor a 5.	Todos	R, I
OP-400	Los diseños que requieren más de un cable tipo <i>jumper</i> en serie para proveer el largo total requerido no serán permitidos.	Todos	R
OP-410	El hardware utilizado debe permitir que la conexión cumpla con sus requerimientos durante la vida útil del satélite en el ambiente especificado.	Todos	R, P
OP-420	Las conexiones deben ser instaladas de manera que las vibraciones, la expansión, la contracción o el movimiento relativo presentado durante la operación no rompa o afloje tales conexiones al punto de variar la resistencia eléctrica.	Todos	P
OP-430	Las conexiones deben estar localizadas en un área protegida, aislada tanto como sea posible, y tan cerca como el diseño lo permita a un agujero, compuerta de inspección o algún otro acceso.	Todos	R
OP-440	El equipo debe estar conectado directamente a la estructura conductora básica o a través de partes permanentemente conectadas.	Todos	R
OP-450	Los cables tipo <i>jumper</i> deben instalarse de manera que no se restrinja la operación de los componentes móviles.	Todos	R
OP-460	Las conexiones no deben ser sujetas por compresión con materiales no metálicos.	Todos	R

Continuación Cuadro 7

Código	Requerimiento	Módulo	Verificación
OP-470	Los materiales con baja conductividad deben ser conectados de manera que se permite la descarga de estática eléctrica.	Todos	R
OP-471	Las resistencias de la conexión deben cumplir con los requerimientos de una conexión clase S.	Todos	R, I
OP-480	Los cables tipo <i>junper</i> y el resto del hardware utilizado en las conexiones entre metales disímiles debe ser seleccionado para minimizar la posibilidad de corrosión.	Todos	R
OP-481	Las arandelas no deben poseer tratados de superficie o capas que afecten la conductividad eléctrica en ninguna manera.	ST	R
OP-482	No se debe utilizar acero inoxidable sin protección como arandela.	ST	R
OP-490	El contacto eléctrico intermitente entre superficies conductoras debe prevenirse, ya sea con conexiones o con aislamiento.	Todos	R
OP-500	Los cables y los alambres deben ser colocados de manera que produzcan una pérdida en la junta de 20 dB de cable a cable usando separación física o eléctrica y considerando el peor escenario de estado estacionario o condiciones transitorias.	PWR	A
OP-510	Los diseños de los cables deben incluir disposiciones para la terminación de cables blindados.	PWR	R
OP-520	Los conectores utilizados para transportar cables con blindajes completos deben tener un acabado conductor y debe utilizar una carcasa trasera que permita la unión periférica de los blindajes.	PWR	R
OP-600	El equipo y los componentes electrónicos no energizados no deberán ser dañados por descargas electroestáticas iguales o menores a 4 000 voltios.	Todos	R
OP-610	Los equipos que pudieran dañarse por descargas electroestáticas entre 4 000 y 15 000 voltios deben tener una etiqueta pegada a la carcasa en un lugar claramente visible en la posición en la que se instalará.	Todos	R
OP-700	Las emisiones electromagnéticas no deben aparecer en conductores DC en valores que exceden los límites de emisión CE01 y CE03 del documento SSP 30237.	PWR	R, A
OP-710	Los transitorios de encendido y apagado y de conmutación de modo no deben exceder la envolvente definida por los valores de la envolvente de transición de conmutación del modelo CE07 del documento SSP 30237.	PWR	R, A
OP-720	El subsistema del equipo no debe presentar ningún fallo, degradación de rendimiento o desviación de las indicaciones especificadas más allá de la tolerancia indicada en el equipo individual o la especificación del subsistema cuando se someta a 1 Vrms de una fuente de 50 ohmios.	PWR	R, P
OP-730	Las emisiones del campo E no deben ser radiadas en exceso a los valores de los especificados para los límites RE02 del documento SSP 30237.	PWR	R, P
OP-740	Las emisiones de banda estrecha del campo E no deben ser radiadas en exceso a los valores de los límites para las emisiones del campo.	PWR	A
OP-800	El satélite debe cumplir con el estándar de seguridad de la ISS NSTS 1700.7B-ISS.	Todos	R
OP-810	La UVG debe transportar el satélite a al lugar especificado por JAXA para la entrega.	Todos	R, I
OP-820	La UVG debe operar el satélite incluyendo el seguimiento y la adquisición de datos después del despliegue desde la ISS.	COMM	R
OP-830	El satélite debe ser registrado por el equipo guatemalteco como un satélite guatemalteco.	COMM	R

E. PRUEBAS

Las pruebas son una parte vital de la verificación y validación de los requerimientos. A través de estas puede obtenerse información concluyente sobre si se cumple o no con los requerimientos de la forma que se tenía prevista. Por esta razón se presentará un resumen de cada prueba que se realizará para verificar y validar ciertos requerimientos que solamente pueden ser verificados y validados a través de estas.

Cuadro 8. Prueba para medición del momento magnético de la barra magnética.

Nombre/tipo de prueba	Medición del momento magnético de la barra magnética
Lugar donde se realizará	Aún no establecida
Componente probado	Barra magnética
Procedimiento	Se utilizará una jaula de Helmholtz y un magnetómetro para medir el campo magnético local dentro de la jaula mientras la barra magnética se encuentra adentro también y alejada de la jaula. El campo B generado por la barra magnética debe ser calculado por la sustracción de los datos de la no-barra magnética a los datos de la barra magnética. El momento magnético de la barra puede ser calculado utilizando un ajuste de mínimos cuadrados no lineal en el modelo de la magnitud del dipolo magnético. El momento de la barra magnética debe estar dentro de los márgenes aceptables de error con respecto al mínimo calculado.
Duración de la prueba	Aún no establecida
Fecha de realización	Aún no establecida
Requerimientos verificados	MI-600, MI-610
Estado de la prueba	Aún no realizada

Cuadro 9. Prueba de mediciones de la barra de histéresis.

Nombre/tipo de prueba	Mediciones de la barra de histéresis
Lugar donde se realizará	Aún no establecida
Componente probado	Barra de histéresis
Procedimiento	Utilizar un generador de funciones para crear un campo H cambiante dentro de una jaula de Helmholtz. Una bobina de sensación debe utilizarse para posicionar la barra de histéresis. La configuración debe ser capaz de medir el campo H producido por la jaula y el campo B en la bobina de sensación mientras la corriente a través de la jaula de Helmholtz varía de acuerdo con el generador de funciones. Un par de medidas deben ser hechas para cada punto: con y sin las barras de histéresis dentro de la bobina de sensación. La sustracción de estos grupos de datos ayudará a encontrar el campo B. Los grupos de datos finales (campos H y B) pueden ser mapeados a un modelo para un ciclo de histéresis. Este modelo será utilizado para calcular la cantidad de barras de histéresis necesarias para el sistema.
Duración de la prueba	Aún no establecida
Fecha de realización	Aún no establecida
Requerimientos verificados	MI-600, MI-610
Estado de la prueba	Aún no realizada

Cuadro 10. Prueba de transferencia de imágenes.

Nombre/tipo de prueba	Transferencia de imágenes entre la OBC y COMM
Lugar donde se realizará	Laboratorios de la UVG
Componente probado	OBC y módulo de comunicaciones
Procedimiento	Se utilizarán imágenes grabadas en la memoria. Las imágenes serán fraccionadas en fracciones de datos basados en las limitaciones del módulo de comunicaciones. A continuación se establecerá un enlace entre ambos componentes y se comenzará a enviar los fragmentos de datos que contienen a la imagen. El tiempo que tardará el envío de las imágenes será programado.
Duración de la prueba	Aún no establecida
Fecha de realización	Aún no establecida
Requerimientos verificados	MI-120
Estado de la prueba	Aún no realizada

Cuadro 11. Prueba de cambio entre computadoras.

Nombre/tipo de prueba	Cambio entre OBC principal y OBC secundaria
Lugar donde se realizará	Laboratorios de la UVG
Componente probado	OBC
Procedimiento	Después de que se ha simulado la falla de la OBC principal, se verificará que se haya realizado el cambio entre computadoras. La OBC de respaldo debe tomar el estado de un módulo para verificar que el cambio ha sido exitoso.
Duración de la prueba	Aún no establecida
Fecha de realización	Aún no establecida
Requerimientos verificados	MI-160
Estado de la prueba	Aún no realizada

Cuadro 12. Prueba de EPS.

Nombre/tipo de prueba	Prueba del sub-módulo EPS
Lugar donde se realizará	Laboratorios electrónicos de la UVG
Componente probado	Convertidores de potencia, controles de carga, protección electrónica de la batería, RBF e interruptores de despliegue, interruptores de potencia, baterías y calentadores de baterías
Procedimiento	Voltajes, niveles de corriente, márgenes, formas de onda y las señales de entrada/salida de cada sub-módulo EPS serán medidos y verificados. Se harán ciclos de carga/descarga en la baterías y estimaciones de degradación de capacidad para 4 500 ciclos. Se probará también el desempeño de los calentadores de baterías.
Duración de la prueba	2 semanas
Fecha de realización	Aún no establecida
Requerimientos verificados	MI-250, MI-260, MI-261, MI-262, MI-265, MI-300, MI-310, MI-320, MI-330, MI-331, MI-340
Estado de la prueba	Aún no realizada

Cuadro 13. Prueba de integración de sub-módulos EPS con baterías y celdas solares.

Nombre/tipo de prueba	Simulación de integración de sub-módulos EPS con baterías y celdas solares
Lugar donde se realizará	Laboratorios electrónicos de la UVG
Componente probado	EPS y baterías
Procedimiento	El EPS será integrado en su configuración final. Voltajes, corrientes, formas de onda, límites de márgenes de entrada/salida serán medidos y verificados.
Duración de la prueba	2 semanas
Fecha de realización	Aún no establecida
Requerimientos verificados	MI-224, MI-262, MI-300, MI-350
Estado de la prueba	Aún no realizada

Cuadro 14. Prueba de ensamble de integración de EPS a las baterías y celdas solares.

Nombre/tipo de prueba	Ensamble de integración de EPS a las baterías y celdas solares
Lugar donde se realizará	Laboratorios electrónicos de la UVG
Componente probado	Arreglo de baterías y celdas solares
Procedimiento	El EPS junto con el arreglo de baterías será conectado a las celdas solares. Ciclos de carga y descarga serán simulados utilizando las celdas solares. El desempeño del EPS en cuanto a la recolección de energía de las celdas solares será verificado, junto con el desempeño total de las cargas.
Duración de la prueba	4 semanas
Fecha de realización	Aún no establecida
Requerimientos verificados	MI-221, MI-224, MI-262, MI-264, MI-350
Estado de la prueba	Aún no realizada

Cuadro 15. Prueba de EPS y cargas integradas a la estructura del satélite.

Nombre/tipo de prueba	EPS y cargas integradas a la estructura del satélite.
Lugar donde se realizará	Laboratorios de ingeniería mecánica de la UVG
Componente probado	Satélite completo
Procedimiento	Todos los componentes electrónicos y de soporte serán integrados a la estructura del satélite. Pruebas y procedimientos de despliegue se realizarán para verificar que se cumple con requerimientos de operación y lanzamiento.
Duración de la prueba	8 semanas
Fecha de realización	Aún no establecida
Requerimientos verificados	MI-220, MI-221, MI-224, MI-260, MI-262, MI-350, DE-310, DE-380
Estado de la prueba	Aún no realizada

Cuadro 16. Prueba a nivel de sistema.

Nombre/tipo de prueba	Prueba a nivel de sistema
Lugar donde se realizará	Laboratorios de la UVG
Componente probado	Satélite completo
Procedimiento	Se probará el funcionamiento de los sub-sistemas de potencia externa, mecánicos e interfaces de comunicación, comunicación con el satélite, almacenaje de energía, la compatibilidad electromagnética y el control de la OBC.
Duración de la prueba	12 semanas
Fecha de realización	3 de marzo de 2018
Requerimientos verificados	OP-050, OP-060, OP-420, OP-720, OP-730
Estado de la prueba	Aún no realizada

Cuadro 17. Pruebas de termo-vacío y de vibraciones.

Nombre/tipo de prueba	Pruebas de termo-vacío y de vibraciones
Lugar donde se realizará	Universidad de Wurzburg, Alemania
Componente probado	Satélite completo
Procedimiento	Para la prueba de vibraciones el satélite será montado en un adaptador para la prueba siguiendo especificaciones del vehículo de lanzamiento. Oscilaciones armónicas de baja intensidad, vibraciones de alta intensidad asociadas con vibraciones aleatorias y cargas cuasi-estáticas son utilizadas comúnmente. El ambiente dentro del vehículo de lanzamiento y el perfil de oscilaciones serán seguidos durante la prueba. La prueba de termo-vacío comenzará con ciclos térmicos. Usualmente se llevan a cabo 4 ciclos en un vacío de 5×10^{-5} mBar. El tiempo de exposición será de entre 30 minutos y 2 horas.
Duración de la prueba	3 semanas
Fecha de realización	15 de mayo de 2018
Requerimientos verificados	MI-263, MI-290, MI-320, DE-510, OP-170, OP-410
Estado de la prueba	Aún no realizada

Cuadro 18. Pruebas con motor, eje y carrusel de filtros.

Nombre/tipo de prueba	Pruebas con el motor, el eje y el carrusel de filtros
Lugar donde se realizará	Laboratorios de la UVG
Componente probado	Motor, eje, carrusel de filtros y OBC
Procedimiento	El motor será conectado al eje, y este al carrusel de filtros. Se validará el sistema haciendo que el motor rote el carrusel de filtros. Una rotación completa del filtro debería durar menos de 10 segundos.
Duración de la prueba	1 semana
Fecha de realización	15 de noviembre de 2018
Requerimientos verificados	MI-050, MI-070
Estado de la prueba	Aún no realizada

Cuadro 19. Prueba de integración de la carga útil.

Nombre/tipo de prueba	Prueba de integración de carga útil
Lugar donde se realizará	La prueba será realizada mientras se sobrevuela el lago de Atitlán
Componente probado	Módulos de Carga Útil, Potencia y C&DH
Procedimiento	Los componentes de los módulos de Potencia, Carga Útil y C&DH serán sujetos a un avión que sobrevolará el lago de Atitlán mientras los componentes capturan imágenes del cuerpo de agua utilizando los tres filtros ópticos. La calidad y utilidad de las imágenes capturadas serán revisadas después.
Duración de la prueba	2 días
Fecha de realización	Aún no establecida
Requerimientos verificados	MI-020, MI-110
Estado de la prueba	Aún no realizada

Cuadro 20. Prueba de medición de patrones de radiación de la antena.

Nombre/tipo de prueba	Medición de patrones de radiación de la antena
Lugar donde se realizará	Laboratorios electrónicos de la UVG
Componente probado	Antena y módulo para comunicación
Procedimiento	Una antena fuente se utilizará para cubrir a la antena del satélite con una onda plana de una dirección específica. La antena del satélite estará rotando utilizando una sistema de posicionamiento; la potencia recibida será registrada.
Duración de la prueba	1 semana
Fecha de realización	Aún no establecida
Requerimientos verificados	MI-560
Estado de la prueba	Aún no realizada

Cuadro 21. Prueba de medición de ganancia de la antena.

Nombre/tipo de prueba	Medición de ganancia de la antena
Lugar donde se realizará	Laboratorios electrónicos de la UVG
Componente probado	Antena y módulo para comunicación
Procedimiento	Utilizando la ecuación de transmisión de Friis y una antena de ganancia estándar que transmite una cantidad fija de potencia, el pico de ganancia de la antena del satélite será medido.
Duración de la prueba	1 semana
Fecha de realización	Aún no establecida
Requerimientos verificados	MI-560
Estado de la prueba	Aún no realizada

Cuadro 22. Prueba de medición de fase de la antena.

Nombre/tipo de prueba	Medición de fase de la antena
Lugar donde se realizará	Laboratorios electrónicos de la UVG
Componente probado	Antena y módulo para comunicación
Procedimiento	La antena del satélite será utilizada para transmitir señales, y otra antena será utilizada para recibir señales. Las posiciones de los picos y los valles de la señal recibida será comparada para determinar la fase relativa de la antena.
Duración de la prueba	1 semana
Fecha de realización	Aún no establecida
Requerimientos verificados	MI-560
Estado de la prueba	Aún no realizada

Cuadro 23. Prueba de medición de la polarización de la antena.

Nombre/tipo de prueba	Medición de la polarización de la antena
Lugar donde se realizará	Laboratorios electrónicos de la UVG
Componente probado	Antena y módulo para comunicación
Procedimiento	La antena del satélite se utilizará para transmitir una señal, mientras que otra antena linealmente polarizada será utilizada para recibir la señal. Esta segunda antena será rotada y la potencia recibida será grabada como una función del ángulo de la antena que recibe la señal.
Duración de la prueba	1 semana
Fecha de realización	Aún no establecida
Requerimientos verificados	MI-560
Estado de la prueba	Aún no realizada

Cuadro 24. Pruebas de medición de impedancia de la antena.

Nombre/tipo de prueba	Medición de impedancia de la antena.
Lugar donde se realizará	Laboratorios electrónicos de la UVG
Componente probado	Antena y módulo para comunicación
Procedimiento	Utilizando un Analizador de Red de Vectores, la impedancia de la antena del satélite puede ser medida como una función de la frecuencia.
Duración de la prueba	1 semana
Fecha de realización	Aún no establecida
Requerimientos verificados	MI-560
Estado de la prueba	Aún no realizada

Cuadro 25. Prueba con el módulo electrónico para comunicación.

Nombre/tipo de prueba	Prueba con el módulo electrónico para comunicación.
Lugar donde se realizará	Laboratorios electrónicos de la UVG
Componente probado	Antena y módulo para comunicación
Procedimiento	Las características principales del módulo electrónico para comunicación serán probadas.
Duración de la prueba	2 semanas
Fecha de realización	Aún no establecida
Requerimientos verificados	MI-500, MI-520, MI-530, MI-560
Estado de la prueba	Aún no realizada

Cuadro 26. Prueba de enlace de datos entre el módulo de comunicación y la estación en Tierra.

Nombre/tipo de prueba	Enlace de datos entre el módulo de comunicación y la estación en Tierra
Lugar donde se realizará	Laboratorios electrónicos de la UVG
Componente probado	Antena y módulo para comunicación
Procedimiento	Un enlace de datos entre el módulo de comunicación y la estación en Tierra será establecido con el mayor distanciamiento físico posible.
Duración de la prueba	2 semanas
Fecha de realización	Aún no establecida
Requerimientos verificados	MI-500, MI-510, MI-520
Estado de la prueba	Aún no realizada

Debe tomarse en cuenta que a medida que el proyecto continúe madurando, y que se obtenga mayor asesoramiento por parte de JAXA, nuevos requerimientos aparecerán. Por otro lado, es bastante probable que algunos requerimientos deban ser modificados debido a distintas razones. Esto significa que el proceso

de definición de requerimientos para el proyecto aún no ha finalizado; sin embargo, ahora se cuenta con una base a partir de la cual seguir trabajando.

F. PRUEBA DE INTEGRACIÓN

La Prueba de Integración será una prueba a nivel de sistema en la cual los módulos de potencia, comunicación, manejo de comandos y datos, carga útil y determinación y control de orientación operaran simultáneamente ejecutando las mismas operaciones que ejecutarían durante la misión. La prueba recreará la operación del satélite después de que este haya sido lanzado por el mecanismo de despliegue y de que haya dejado de girar sin control. Esto quiere decir que el sistema pasará a través de la fase de iniciación antes de operar como lo hará durante la mayoría del tiempo de la misión. A continuación, se presentan como referencia algunos de los aspectos más importantes a tomar en cuenta durante la realización de esta prueba. Aún no es posible definirla formalmente debido a que no todos los componentes del satélite han sido seleccionados todavía. Cuando el proceso de selección haya acabado, la metodología de esta prueba deberá ser actualizada.

El objetivo de la prueba de integración es verificar que los módulos mencionados anteriormente pueden operar apropiadamente como un sistema completa que cumple con los objetivos de la misión. Para lograr esto, los requerimientos de cada módulo deben ser validados y verificados mientras operan como un solo sistema unificado, y no como componentes independientes.

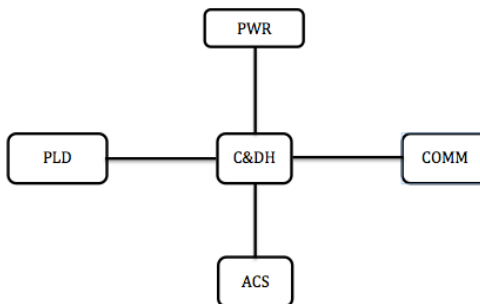
Esta prueba será llevada a cabo a través de la DITL. Como se mencionó anteriormente, esta prueba busca ser una simulación de la misión entera del satélite. Para lograrlo, la simulación debe pasar por el proceso entero de iniciación antes de comenzar la operación nominal, y solamente deben incluirse comandos requeridos normalmente para esta operación. Sin embargo, es importante que se incluya cada fase y modo del satélite en la prueba.

Debido a que no se cuenta con el tiempo necesario para simular el periodo entero de la misión, la DITL debe realizarse en un periodo bien establecido. Para asegurar que la prueba pueda recrear las condiciones lo mejor posible, sin consumir demasiado tiempo, la prueba debería tener una duración de 24 horas. Debe ser realizada dentro de las instalaciones de la universidad, en un área especialmente acomodada.

Para esta prueba, puede notarse que tener todos los componentes del satélite apilados de la misma forma como lo estarán una vez instalados en la estructura dificultará los procesos de medición o la manipulación de los elementos. Esta dificultad proviene del poco espacio que queda entre cada elemento. Una posible solución a este problema es posicionar a todos los elementos en una configuración en la cual se colocan sobre un banco de pruebas con cierta separación uno del otro y conectados mediante cables extensos.

Es importante mencionar que, aunque este método no dará una verdadera representación de las condiciones en las cuales operará el satélite, sí ayudará a encontrar simples y posibles errores que podrían poner en peligro el éxito de la misión. En otras palabras, si algo no funciona en esta configuración, probablemente tampoco lo hará una vez en la misión.

Figura 18. Representación gráfica de la distribución de los módulos durante la prueba de integración.



1. **C&DH.** Dos de los componentes principales de este módulo serán incluidos en la prueba: la computadora principal, y la computadora de redundancia. La computadora principal debe probar que es capaz de enviar todos los comandos necesarios para inicializar y terminar cada función del sistema. Al mismo tiempo, la computadora principal debe procesar la información enviada desde los varios sensores en el sistema, y enviarla al módulo de comunicación. En algún momento de la prueba, la computadora principal será desconectada y la redundante será inicializada y deberá tomar las funciones de la otra computadora. Este cambio de computadora representará una situación en la cual la computadora principal fallará y la secundaria deberá tomar su lugar automáticamente.

Durante la prueba, el módulo de C&DH será responsable de:

- Inicializar el movimiento del motor que controlará al carrusel.
- Inicializar la secuencia de captura de imágenes.
- Procesar la información proveniente del sensor monocromático y enviarla al módulo de Comunicación.
- Inicializar la transmisión de datos desde el módulo de Comunicación.
- Cambiar la fuente de suministro de energía. La computadora debe cambiar estas fuentes dependiendo de cuál se necesite.
- Procesar la información proveniente del magnetómetro y los fotodiodos.

La computadora a bordo será el enlace directo entre quienes realizan la prueba y el sistema durante esta. Esto significa que, durante la mayoría del tiempo, la computadora a bordo debe tener comunicación con

una computadora personal. A través de esta conexión se podrá enviar comandos que la computadora a bordo enviará al resto de los componentes, y recibir información de estos. Los comandos que se enviarán serán aquellos que inicializarán la función que se quiere probar. Esto quiere decir que nuevos comandos serán enviados constantemente a la computadora a bordo, dependiendo de lo que se quiera probar. A través de la conexión entre la computadora a bordo y la computadora personal se podrá obtener parte de la información de los sensores, y del sistema en sí.

2. **Potencia.** Durante la prueba, el módulo de potencia será responsable de proveer al resto de los componentes con la energía que necesitan para poder operar. Parte de la prueba será realizada utilizando la potencia generada por los paneles solares y la energía almacenada en las baterías, y otra parte utilizando solamente la energía de las baterías. Esto ayudará a probar las dos condiciones en las que operará el satélite: cuando recibe luz solar, y cuando esta le es bloqueada por la Tierra.

Es importante mencionar que las baterías no serán capaces de mantener al satélite en operación durante toda la noche que dure la prueba. Así que, para llevar a cabo se requerirá de una fuente auxiliar de potencia. Esta fuente auxiliar mantendrá al sistema en operación después de que las pruebas con las baterías y las celdas solares hayan finalizado. Esta fuente de potencia podría ser una fuente de voltaje programable.

El cambio entre las tres fuentes de energía se dará dependiendo de qué se necesita probar. La prueba comenzará utilizando la energía generada por las celdas solares y las baterías. A continuación, las celdas solares se desconectarán y solamente las baterías permanecerán como fuente de energía. Las baterías necesitaran ser cargadas constantemente durante la segunda parte de la prueba.

3. **ADCS.** Para la prueba, tanto el magnetómetro como los fotodiodos deben ser incluidos y probados. Estos componentes serán encendidos después de que el sistema se inicialice y acumule la suficiente energía como para comenzar su operación. Los componentes se mantendrán encendidos durante toda la prueba, simulando de esa forma las condiciones reales de la misión.

Debe notarse que esta prueba no pretende probar la exactitud de las medidas del magnetómetro y de los fotodiodos; este módulo solamente será incluido para observar cómo el resto de los módulos se comportan cuando operan en un sistema integrado. Esto significa que la información enviada por los componentes del módulo de ADCS solamente se utilizará para confirmar que están funcionando, y no para medir su efectividad.

4. **Comunicación.** Para poder probar los efectos de este módulo sobre todos los demás el elemento electrónico que permitirá la comunicación, la antena y el mecanismo para desplegar esta deben ser incluidos en la prueba. Durante la prueba, la antena será desplegada después de que el sistema sea inicializado siguiendo los requerimientos de la misión. Después de esto, el módulo comenzará a enviar la información que recibe de la computadora a bordo usando la frecuencia seleccionada. El módulo enviará información cada vez que el sistema deba enviar información sobre su estado, o las imágenes que acaba de capturar. Por esta razón, este módulo solamente será utilizado en ciertos momentos de la prueba.

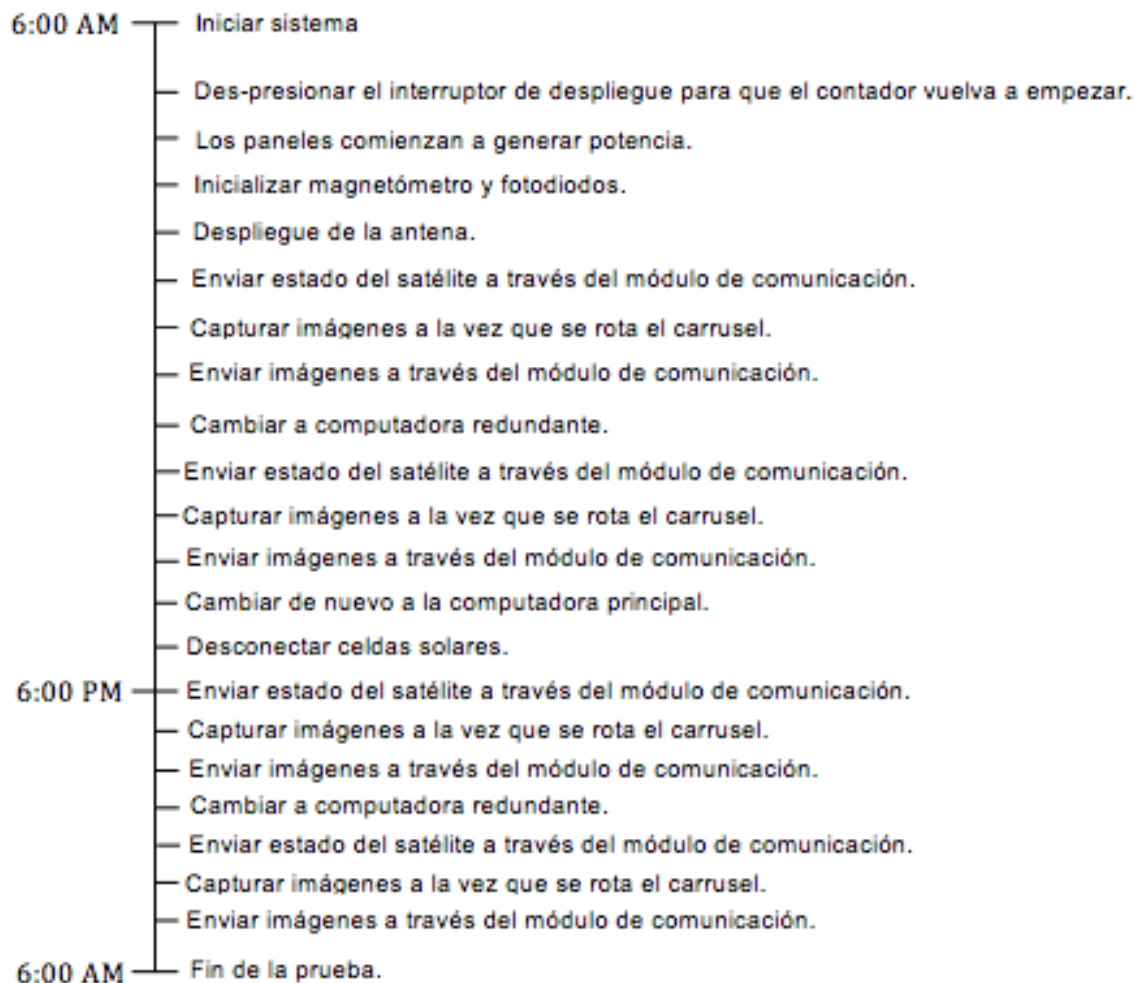
Ya que la prueba será realizada en un espacio cerrado, no se podrá comprobar que este módulo transmite la información. Sin embargo, así como con el módulo ADCS, la razón para incluir el módulo de comunicación en la prueba es para observar el comportamiento de todo el sistema en una misión simulada, y no para probar el módulo en sí.

5. **Carga útil.** El propósito de este módulo durante la prueba será capturar fotografías y rotar el mecanismo de cambio de los filtros hasta la posición requerida. Esto quiere decir que para la prueba los componentes de este módulo que deben ser incluidos son el sensor monocromático, el motor, y el mecanismo de cambio de filtros. En algún momento de la prueba un comando será enviado al motor por la computadora a bordo. Cuando esto suceda, el sensor monocromático comenzará a capturar imágenes; mientras tanto, el motor comenzará a rotar el carrusel a la posición necesaria.

Ya que el objetivo de la prueba no es verificar la calidad de las imágenes capturadas, los filtros ópticos no serán necesarios. Sin embargo, el motor y el sensor monocromático deben estar sincronizados de la misma forma como lo estarán durante la misión. Esto, para que la prueba sea una representación lo más fiel posible de la misión real.

6. **Cronograma.** Para poder aprovechar de la mejor manera posible el limitado tiempo con el que se contará al momento de realizar la prueba de integración, se ha elaborado un cronograma que muestra las funciones principales que marcan las etapas más importantes de la operación del satélite.

Figura 19. Cronograma a seguir durante la prueba de integración.



G.INFORMACIÓN OPERACIONAL DE LOS COMPONENTES

Una forma apropiada de observar cómo la elección de componentes individuales en cada módulo afecta al sistema completo es a través de llevar un riguroso control de la información operacional y las características físicas de estos componentes. Algunas de las características que pueden afectar en gran medida al sistema total son la masa y las dimensiones de los componentes; el satélite debe ser diseñado bajo ciertos límites de masa y volumen, por lo tanto, es importante que se conozcan estos para cada componente. En cuanto a la información operacional, entre los aspectos más importantes están la potencia consumida, el voltaje necesario y las temperaturas de operación de los componentes. Es importante tomar en cuenta estas características para asegurar la operación correcta de los componentes.

H. FACTIBILIDAD DE DISEÑO DE POTENCIA PROPUESTO EN FASE 3

Se analizó una de las variantes del módulo de potencia propuestas en la fase 3 del Megaproyecto. Esta preveía el uso de un panel solar, conformado de 6 celdas solares. (Barrios, *et al*, 2016) Este panel requería que todas las celdas actuaran en un mismo plano, por lo que era necesario utilizar caras desplegables, esto agrega complejidad al diseño de estructura y al control de la orientación del satélite debido a un cambio en el centro de gravedad e inercia rotacional a partir del momento de despliegue.

Para que el funcionamiento de este sistema fuera el correcto se requería un sistema ADCS activo que fuera capaz de apuntar de manera efectiva los paneles al Sol y generar la mayor cantidad de energía eléctrica posible. Una característica de este diseño era que utilizaba celdas 3G30A de AZUR SPACE, con una eficiencia nominal de 30% y un área superficial de 30.18 cm². (AZUR SPACE, 2016)

1. **Generación de potencia eléctrica.** Para llevar a cabo el análisis de factibilidad del diseño propuesto por la fase 3 del Megaproyecto se asumió la órbita con la cual se estuvo trabajando durante la fase que era una órbita circular (excentricidad = 0) inclinación de 20°, una altura sobre la superficie de la Tierra de 450 km y una ascensión recta del nodo ascendente (RAAN, por sus siglas en inglés) de 185°.

Se llevó a cabo una simulación utilizando el software STK de AGI para poder obtener el perfil de luz al cual se vería expuesto el satélite en órbita. Debido a la incertidumbre en la fecha de lanzamiento se trabajó bajo el supuesto que el lanzamiento sería en el año 2018, por lo que se simuló desde enero 1, 2018 hasta enero 1, 2019.

Figura 20. Ajustes utilizados para simulación en STK fase 3.

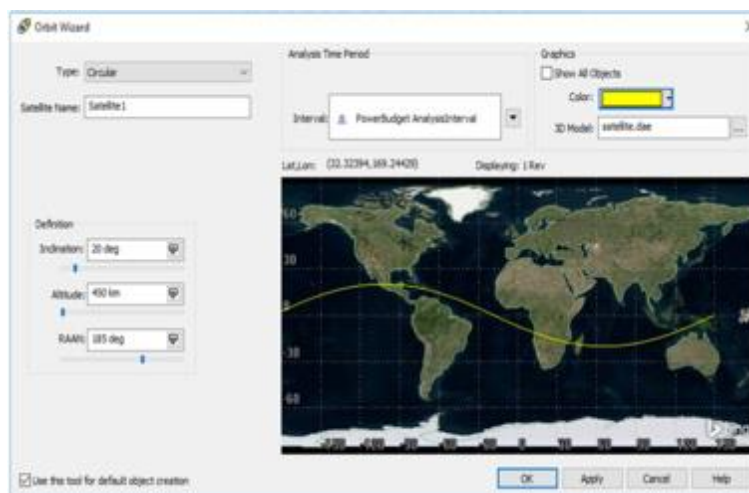


Figura 21. Simulación en 3D de la órbita del satélite fase 3.



Figura 22. Órbita 2D del satélite, mostrado sobrevolando Guatemala fase 3.



A partir de la simulación se obtuvo el perfil de luz al cual estaría expuesto el satélite a través del año 2018. El siguiente cuadro resume la información más relevante.

Cuadro 27. Duraciones de luz, penumbra y umbra obtenidos de simulación de STK.

	Duración mínima (s)	Duración máxima (s)	Duración promedio (s)
Luz	3,393.082	3,704.085	3,500.073
Penu mbra	8.159	12.398	9.091
Umbr a	1,876.950	2,142.083	2,085.59

Del Cuadro 27 se observa que el tiempo en penumbra del satélite es despreciable en comparación con el tiempo en el cual se encuentra expuesto a luz o en la sombra (umbra). Para llevar a cabo el análisis de factibilidad de diseño se debe tomar en cuenta las condiciones más críticas que son cuando hay menos tiempo de luz disponible y un mayor tiempo de sombra, ya que la producción de energía eléctrica es

proporcional al tiempo que el satélite está expuesto a la luz solar. Sin embargo, la duración mínima de luz del Cuadro 5 no es representativa ya que ocurría en el momento en el cual se inició el análisis, por lo que podría ser el resultado de que se asumió que el satélite empezó a orbitar en un punto arbitrario en el cual no le era posible recibir la totalidad de luz que tendría disponible en un periodo o una revolución real.

Fue necesario identificar la segunda duración más corta de luz, lo cual se logró al exportar los datos a una hoja electrónica de Microsoft Excel y analizar los datos. A partir de este análisis se obtuvo la duración límite de exposición a la luz, las condiciones límites se presentan en el siguiente cuadro.

Cuadro 28. Condiciones límite del perfil luz/sombra del satélite.

	Tiempo de luz (s)	Tiempo de sombra (s)	Periodo (s)
Perfil luz/sombra	3,444.131	2,142.083	5,586.174

Para estimar la producción de potencia eléctrica se tomaron en cuenta las características de las celdas solares en su inicio de vida (BOL, por sus siglas en inglés). Es importante mencionar que las celdas solares se ven expuestas al medio ambiente espacial lo que incluye radiación cósmica, cambios de temperatura drásticos y, dependiendo de la altitud de la órbita, arrastre. (Larson, *et al*, 1999) Es por esto que las celdas se deterioran con el paso de tiempo y disminuyen su capacidad de producción de energía eléctrica hasta llegar al final de su vida útil (EOL, por sus siglas en inglés). Sin embargo, la duración estimada de la misión es de seis meses, lo cual es poco tiempo para notar un deterioro significativo en el desempeño de las celdas solares.

Cuadro 29. Características de celdas solares 3G30A de AZUR SPACE.

	Cantidad de celdas	Voltaje (V)	Corriente (mA)	Potencia unitaria (W)	Potencia total (W)
B OL	6	2.409	502.9	1.211	7.266
E OL	6	2.191	477.6	1.046	6.276

(Dahir, *et al*, 2017)

Se asumió que el ángulo de incidencia era de 0° en todo momento, incluso cuando se tomaba la fotografía. Este tipo de control es posible a través de un sistema ADCS activo de gran precisión.

2. Consumo de potencia eléctrica por módulo. El consumo de potencia eléctrica para la fase 3 del Megaproyecto se encuentra detallado, por módulo, en los cuadros 30 - 33.

Cuadro 30. Características eléctricas de ADCS fase 3.

Componente	Cantidad	Voltaje (V)	Corriente (mA)	Potencia unitaria (W)	Potencia total (W)
Motor ^a	4	6.000	250.000	1.500	6.000
IMU ^b	2	5.000	50.000	0.250	0.500
Sensores de luz ^c	12	6.000	1.600	0.010	0.120
Magnetorquers ^d	3	5.000	28.000	0.140	0.900

a = (Faulhaber, 2017)

b = (CH Robotics, 2014)

c = (ams AG, 2016)

d = (ZARM Technik AG, 2013)

La potencia total requerida para el funcionamiento del módulo de ADCS trabajado en la fase 3 del Megaproyecto es de 7.520 W. Tomando en cuenta la capacidad de generación de potencia eléctrica y el modo de operación del satélite se logra evidenciar que este consumo es excesivo, esta fue la base de análisis posterior.

Debido a los elevados consumos de potencia del módulo ADCS se decidió buscar soluciones alternativas comerciales (*commercial off-the-shelf, COTS*). La alternativa ADCS era un módulo “pasivo”, a pesar que si consume potencia eléctrica, conocido como Cube ADCS, que no tenía tanto consumo de energía. Sin embargo, tampoco garantizaba un control tan preciso en la actitud del satélite. La alternativa tiene un consumo de potencia nominal de 1.000 W. (CubeSatShop, 2016)

Cuadro 31. Características eléctricas de PLD fase 3.

Componente	Cantidad	Voltaje (V)	Corriente (mA)	Potencia unitaria (W)	Potencia total (W)
Cámara ¹	1	2.800	142.857	0.400	0.400
GPS ²	1	3.300	151.151	0.500	0.500
Motor stepper ³	1	6.000	120.000	0.720	0.720

1 = (e-con Systems Pvt. Ltd., 2015)

2 = (Swift Navigation Inc., 2016)

3 = (FeeTech, s.f.)

El consumo total de PLD para la fase 3 constituye 1.620 W. Es importante notar que el sistema de posicionamiento global (GPS) debería estar encendido en todo momento para saber la ubicación del satélite y así indicar en qué momento tomar fotos o enviar la información a la base terrestre.

Cuadro 32. Características eléctricas de C&DH fase 3.

Componente	Cantidad	Voltaje (V)	Corriente (mA)	Potencia unitaria (W)	Potencia total (W)
OBC ¹	1	5.000	160.000	0.800	0.800

1 = (Pi Supply, 2017)

Cuadro 33. Características eléctricas de COMMS fase 3.

Componente	Cantidad	Voltaje (V)	Corriente (mA)	Potencia unitaria (W)	Potencia total (W)
Transceptor ¹	1	3.000	21.500	0.065	0.065
Antena Banda S ²	1	IND	IND	10.000	10.000
Controlador de nodo terminal ³	1	5.00	300.000	1.500	1.500
Amplificador RF ⁴	1	5.000	200.000	1.000	1.000

RF = radio frecuencia

IND = información no disponible

1 = (Texas Instruments, 2017)

2 = (CubeSatShop, 2016)

3 = (Microchip Technology Inc., 2011)

4 = (Qorvo, 2017)

Tomando en cuenta todos los elementos del módulo de COMMS la potencia total requerida por el módulo es de 12.565 W. Este consumo es excesivo, sin embargo, tiene un tiempo de activación relativamente corto, por lo cual fue necesario realizar una simulación para determinar la factibilidad del diseño realizado hasta el momento y, en caso de no ser factible, buscar soluciones.

3. Consumo de potencia eléctrica por modo de operación. El satélite posee 4 modos de operación en órbita. El modo de operación depende de la disponibilidad de luz y la ubicación geográfica que determina la adquisición de datos (captura de fotografías) o comunicación de datos con la estación terrena. Debido a que el objetivo inicial de la misión era tomar fotografías del Lago Atitlán para el monitoreo de cianobacteria solo se tomaban fotografías cuando se orbitaba sobre Guatemala. Además, se

estaban contemplando diversas alternativas para la estación terrena, desde una desarrollada en Guatemala hasta una alianza estratégica con otros países de la región.

Los cuatro modos de operación son: vuelo con luz, vuelo con luz sobre Guatemala, vuelo con luz sobre estación terrena y vuelo con sombra. La información acerca del consumo de potencia del satélite diseñado por la fase 3 y con la alternativa de ADCS “Cube ADCS” por modo de operación se resume en los cuadros 34 - 37.

Cuadro 34. Consumo de potencia durante vuelo con luz.

	Potencia ADCS (W)	Potencia C&DH (W)	Potencia GPS (W)	Potencia total (W)
Fase 3	7.040	0.800	0.500	8.340
Alternativa	1.000	0.800	0.500	2.300

Cuadro 35. Consumo de potencia durante vuelo con luz sobre Guatemala.

	Potencia ADCS (W)	Potencia C&DH (W)	Potencia PLD (W)	Potencia total (W)
Fase 3	7.040	0.800	1.620	9.460
Alternativa	1.000	0.800	1.620	3.420

Cuadro 36. Consumo de potencia durante vuelo con luz sobre estación terrena.

	Potencia ADCS (W)	Potencia C&DH (W)	Potencia GPS (W)	Potencia COMMS (W)	Potencia total (W)
Fase 3	7.040	0.800	0.500	12.565	20.905
Alternativa	1.000	0.800	0.500	12.565	14.865

Es importante mencionar que no es necesario logísticamente que el satélite esté expuesto a la luz para transmitir y recibir información mediante su comunicación con la estación. Sin embargo, debido al alto consumo de potencia del módulo de COMMS no era recomendable activarlo sin acceso a la energía proveniente de los paneles solares, ya que podría llevar al agotamiento de las baterías y, consecuentemente, al fin de la misión.

Cuadro 37. Consumo de potencia durante vuelo con sombra.

	Potencia sensores de luz (W)	Potencia C&DH (W)	Potencia total (W)
Fase 3	0.180	0.800	0.920

En la fase 3 del Megaproyecto se calculó que el tiempo disponible para la comunicación entre la estación terrena y el satélite era de 7 minutos y 30 segundos, este tiempo fue asumido como válido para la captura de imágenes para poder estimar el consumo de energía del satélite, también se asumió en qué momento se energizaba y desenergizaban los módulos de acuerdo al concepto de operaciones.

I. REQUERIMIENTOS FUNCIONALES DEL MÓDULO DE POTENCIA

Los requerimientos funcionales se dividieron en tres secciones: recolección de energía, almacenamiento de energía y distribución de energía.

Cuadro 38. Requerimientos funcionales, sistema de recolección de energía.

Identificación	Descripción
1.1	El satélite utilizará paneles solares como fuente de energía primaria.
1.1.1	El satélite deberá utilizar once celdas solares para recolectar energía.
1.1.2	El área de cada celda solar no deberá ser superior a 4,000 mm ² .
1.1.3	Las celdas solares deberán ser sujetadas a las caras externas del satélite.
1.2	El sistema de recolección de energía deberá proveer suficiente energía al satélite para que este opere durante toda la misión.
1.2.1	El sistema de recolección de energía deberá suministrar los excedentes de energía a las baterías, para permitir la operación en la sombra.
1.2.2	Las celdas solares deberán ser capaces de operar desde -45 °C hasta 60 °C.
1.3	Las celdas solares deberán tener un diodo integrado para evitar circulación reversa.
1.4	El espesor de las celdas solares deberá ser inferior a 6.5 mm.
1.5	La masa de cada celda solar no deberá exceder 20 g.
1.6	Los materiales utilizados en el sistema de recolección de energía deberán estar en la lista de materiales de baja gasificación aprobada por la NASA.

Cuadro 39. Requerimientos funcionales, sistema de almacenamiento de energía.

Identificación	Descripción
2.1	El satélite utilizará baterías como fuente secundaria de energía.
2.2	La batería deberá ser capaz de almacenar la energía suficiente para permitir la operación del satélite durante toda la misión.
2.2.1	El sistema de almacenaje de energía deberá ser capaz de almacenar al menos 9.5 Wh.
2.2.2	El sistema de almacenaje de energía tendrá protecciones contra voltajes fuera de límites.
2.2.3	Las baterías deberán proveer energía al mismo tiempo que los paneles solares, de ser necesario.
2.2.4	Las baterías deberán ser capaces de operar entre 0 °C y 50 °C.
2.2.5	El sistema de almacenaje de energía deberá tener una profundidad de descarga permisible de 10%.
2.2.6	Las baterías deberán ser capaces de operar durante 4,500 ciclos de carga/descarga.
2.3	La masa de cada celda de la batería no deberá ser superior a 50 g.
2.4	El tamaño de las baterías no deberá exceder 115,000 mm ³ .
2.5	El sistema de recolección de energía deberá proteger a las baterías de hinchazón excesiva por vacío y permitir la expansión durante su operación normal.
2.6	Los materiales utilizados en el sistema de recolección de energía deberán estar en la lista de materiales de baja gasificación aprobada por la NASA

Cuadro 40. Requerimientos funcionales, sistema de distribución de energía.

Identificación	Descripción
3.1	El voltaje y la corriente de los sistemas de recolección y almacenaje de energía deberán ser monitoreados y reportados a OBC.
3.2	Se deberán utilizar interruptores eléctricos para activar/desactivar los componentes del satélite cuando OBC lo requiera.
3.3	Se deberán incluir calentadores para evitar que las baterías sean expuestas a temperaturas fuera del rango operacional de las mismas.
3.4	Se deberá regular la potencia proveniente de las baterías para asegurar que los requerimientos de voltaje de los componentes eléctricos sean satisfechos.
3.4.1	Se utilizarán convertidores integrados para proveer los diferentes voltajes que requieren las cargas.
3.5	Las mediciones del estado de carga de las baterías, que se reportará a OBC, deberán tener una precisión de +/- 5%.
3.6	Los materiales utilizados en el sistema de recolección de energía deberán estar en la lista de materiales de baja gasificación aprobada por la NASA

J. ANÁLISIS DE ALMACENAMIENTO DE ENERGÍA

Para estimar la capacidad requerida de las baterías es necesario calcular la energía que deberá ser almacenada por ellas. Para esto es necesario tomar en cuenta la operación del satélite y las características eléctricas de las cargas. Tomando en cuenta la operación del satélite en la sombra:

$$E_{sombra} = P_{sombra} * T_{sombra} = 0.92 W * \frac{2,142.083 s}{3,600 \frac{s}{h}} = 0.55Wh \quad (67)$$

Tomando en cuenta que las baterías también deben contribuir a proporcionar energía al satélite para permitir la operación cuando el satélite se comuniquen con la estación en tierra, también es necesario analizar este caso.

$$P_{baterías} = P_{ET} - P_{PS} = 14.865W - 7.266W = 7.60W \quad (68)$$

Donde (ET) significa estación terrena y (PS) significa paneles solares. Es importante mencionar que al momento de realizar este análisis se contempló un sistema activo de ADCS (iACS200) con un consumo máximo de 1W. (Hyperion Technologies 2016)

$$E_{ET} = P_{baterías} * T_{ET} = 7.60 W * \frac{450 s}{3,600 \frac{s}{h}} = 0.95Wh \quad (69)$$

K. SELECCIÓN DE COMPONENTES DEL MÓDULO DE POTENCIA

Se llevaron a cabo evaluaciones de conceptos (*trade studies*) para seleccionar las celdas solares y las baterías que se utilizarán en el módulo de potencia del CubeSat que está siendo desarrollado en la Universidad el Valle de Guatemala. Esta es una forma objetiva de seleccionar entre diversas opciones, y se ha utilizado desde los inicios del proyecto.

Cuadro 41. Especificaciones de celdas solares.

Modelo	Voltaje (V)	Corriente (mA)	Potencia (W)	Masa (g)	Precio por celda (\$)	dV/dT (mV/°C)	Legado espacial
AZUR SPACE SCA 3G30A ¹	2.409	502.9	1.211	3.561	321	-6.7	Sí
Spectrolab 30.7% XTJ Solar Cells ²	2.390	473.6	1.132	2.286	329	-6.5	Sí
Spectrolab 28.3% UTJ Solar Cells ³	2.390	443.7	1.042	2.286	319	-6.5	Sí

1 = (AZUR SPACE, 2016)

2 = (Spectrolab, 2016)

3 = (Spectrolab, 2010)

Es importante notar que también se tomaron en cuenta otros factores, como temperaturas máximas y mínimas que soportaban las celdas, así como su espesor con el fin de ver si las celdas a analizar si podían operar bajo las restricciones del sistema. Todas las celdas cumplían con el requerimiento funcional 1. 4. El precio por celda y el legado espacial son datos que fueron proveídos por los fabricantes de las celdas solares.

Cuadro 42. Normalización de variables de celdas solares.

Variable	1	2	3	4	5	6	7	8	9	10
Potencia (W)	1.043	1.061	1.080	1.099	1.118	1.136	1.155	1.174	1.193	1.211
Masa (g)	3.561	3.419	3.278	3.136	2.995	2.853	2.711	2.570	2.428	2.286
dV/dT (mV/°C)	-6.70	-6.68	-6.66	-6.63	-6.61	-6.59	-6.57	-6.54	-6.52	-6.50
Precio por celda (\$)	329	328	327	326	325	323	322	321	320	319
Legado espacial	No	-	-	-	-	-	-	-	-	Sí

Se tomó en cuenta que se espera que las celdas solares se encuentren a temperaturas elevadas la mayoría del tiempo que estén en órbita. Ya que el parámetro dV/dT utiliza como referencia 25 °C mientras la magnitud de este parámetro sea mayor, se tendrá un mayor detrimento en el desempeño de las celdas solares.

Cuadro 43. Especificaciones de baterías

Modelo	% Volumen	% Masa	Costo (\$)	Capacidad (Wh)	Misiones exitosas	Temp. mínima (°C)	Temp. máxima (°C)
Sparkfun 2Ah PRT-08483 ¹	4.6%	5.6%	26	16.8	1	0	40
Clyde Space CubeSat Battery ²	10.6%	9.5%	1,900	10	2	0	40
EXA BA01X Battery Array ³	7.3%	8.6%	3,200	26.6	2	-30	80
Gomspace NanoPower Cells ⁴	4.3%	7.2%	1,250	19.2	3	-5	45

1 = (Wang, 2006)

2 = (Clyde Space, 2016)

3 = (EXA, 2016)

4 = (GomSpace, 2016)

Se obtuvieron los datos de masa y volumen de las baterías de sus hojas técnicas. Posteriormente, se calculó el porcentaje de volumen interno que ocuparían las baterías ya que esto es más representativo tomando en cuenta que el volumen interno es de 96 mm x 94 mm x 90 mm. De igual manera se llevó a cabo el cálculo del porcentaje de masa, solo que en este caso se tomó en cuenta el límite de masa de un CubeSat 1U que es de 1.33 kg.

Cuadro 44. Normalización de variables de baterías.

Variable	1	2	3	4	5	6	7	8	9	10
% Volumen	12.5%	11.7%	10.8%	10%	9.2%	8.4%	7.5%	6.7%	5.9%	5.1%
% Masa	9.5%	9.1%	8.7%	8.2%	7.8%	7.3%	6.9%	6.4%	6%	5.6%
Costo (\$)	3,200	2,847	2,495	2,142	1,789	1,437	1,084	731	379	26
Capacidad (Wh)	10	11.8	13.7	15.5	17.4	19.2	21.1	22.9	24.8	26.6
Misiones exitosas	0	-	-	1	-	-	2	-	-	3
Temp. mínima (°C)	0	-3.3	-6.7	-10	-13.3	-16.7	-20	-23.3	-26.7	-30
Temp. máxima (°C)	40	44.4	48.9	53.3	57.8	62.2	66.7	71.1	75.6	80

L. INTERFACES DEL MÓDULO DE POTENCIA

A continuación, se identificarán las interfaces mecánicas, eléctricas, de programación y de servicios suministrados. Además, se definirá como se planea definir la interfaz de tal manera que la integración sea lo más fácil posible. Ver cuadros 45 – 48.

Cuadro 45. Interfaces mecánicas.

Identificación	Interfaz	Descripción
IM-1	Sujeción de PCBs externas a estructura.	Se prevé la sujeción mediante tornillos M2.
IM-2	Sujeción de PCBs internas a estructura.	Se utilizará un estándar como el de UNISEC o PC104, para la unión física de las PCBs internas y dependiendo del estándar seleccionado se definirá la forma de sujeción de las PCBs internas a la estructura.
IM-3	Sujeción de celdas solares a PCBs externas.	Se utilizará una combinación de cinta adhesiva de doble lado y soldadura para efectuar la sujeción, el método se detalla más adelante.

Continuación Cuadro 45

Identificación	Interfaz	Descripción
IM-4	Orientación de celdas solares.	Las celdas solares deberán ser colocadas con la misma orientación para facilitar las conexiones eléctricas.
IM-5	Sujeción de celdas de batería.	Se utilizará el compuesto Arathane 5753 como adhesivo.
IM-6	Control térmico de batería.	Se utilizarán calentadores para suministrar calor a las baterías y evitar que su temperatura salga del rango operacional. Se utilizará una cinta adhesiva para transferencia de calor entre las celdas de la batería para homogeneizar la temperatura. La PCB sobre la cual irá montada la batería no tendrá una capa de cobre con el fin de aislar térmicamente las baterías del resto del satélite ¹
IM-7	Expansión térmica de batería.	Se realizará un ensamblaje mecánico para controlar la expansión térmica y la fatiga mecánica asociada.
IM-8	Interruptores	Los interruptores de despliegue y el pin RBF deberán ser sujetos a la estructura.

¹ = (Mason, *et al*, 2015)

Cuadro 46. Interfaces eléctricas.

Identificación	Interfaz	Descripción
IE-1	Interconexión de celdas solares.	Las celdas solares de cada cara externa del satélite se deberán conectar en serie, se hará a través de las PCB externas.
IE-2	Interconexión entre caras externas.	Esta conexión se llevará a cabo a través de una PCB que estará dedicada a la conexión que formará los paneles solares.
IE-3	Paneles solares – circuito interno	Se utilizarán convertidores de voltaje para poder interconectar los paneles solares con el resto del sistema.
IE-4	Batería – circuito interno	Se utilizarán protecciones ante voltaje alto y bajo.
IE-5	Componentes - circuito interno	Se utilizarán convertidores de voltaje e interruptores para proveer el voltaje necesario cuando sea necesario.
IE-6	Celdas de baterías	Se conectarán dos celdas en serie para formar la batería con el voltaje necesario.

Cuadro 47. Interfaces de programación.

Identificación	Interfaz	Descripción
IP-1	Estado de carga	OBC estimará el estado de carga de la batería mediante medición de corriente de carga y de descarga.
IP-2	Temperatura de batería	Se medirá la temperatura de las baterías para que OBC accione los calentadores de ser necesario.
IP-3	Interruptores de cargas	OBC tendrá control de los interruptores que restringen el paso de potencia hacia las cargas.
IP-4	Monitoreo	Se monitorearán parámetros relevantes como corriente y voltaje, para tener una idea de la salud del sistema y dejar aprendizaje para misiones futuras.

Cuadro 48. Interfaces de servicios suministrados.

Identificación	Interfaz	Descripción
IS-1	Vehículo de lanzamiento	Los componentes deberán ser capaces de soportar las vibraciones del despegue, se harán pruebas de vibraciones de acuerdo al vehículo de lanzamiento.
IS-2	Gasificación ¹	Los componentes deberán cumplir con un criterio de TML < 0.1%.
IS-3	Dimensiones externas	Ningún componente deberá exceder 6.5 mm normales a las superficies exteriores del satélite.
IS-4	Diagnóstico y carga de batería ¹	Para permitir diagnósticos y carga de batería previo al lanzamiento, todos los conectores deberán estar dentro del puerto de acceso, ver anexo 1.
IS-5	Interruptores de despliegue ^{1,2}	Los interruptores de despliegue deberán estar colocados en los rieles en dirección X y/o Y y permanecer oprimidos para que su altura sea de máximo 1.25 mm dentro del mecanismo de despliegue. La carrera de los mismos deberá ser de un máximo de 2 mm, y su fuerza no superior a 3 N.
IS-6	Pin RBF	Los pines RBF deberán estar colocados dentro de los puertos de acceso, ver figuras 159-161. Se deberá ajustar una correa al pin para que pueda ser removido con facilidad.

1 = (Cal Poly, 2005)

2 = (JAXA, 2009)

M. RIESGOS DEL MÓDULO DE POTENCIA

Se llevó a cabo la clasificación de riesgos con su respectiva probabilidad y consecuencia tomando en cuenta las posibles causas de los riesgos. Estas causas debían ser clasificadas de acuerdo a su probabilidad y consecuencia. Para clasificar la probabilidad se siguió un criterio de probabilidad, el cual se muestra en el Cuadro 49. En cuanto a la consecuencia, se siguió el criterio de consecuencia que se encuentra en el Cuadro 50.

Cuadro 49. Criterio de probabilidad.

Nivel	Probabilidad	Probabilidad de ocurrencia
1	No es probable	10%
2	Baja probabilidad	30%
3	Probable	50%
4	Alta probabilidad	70%
5	Casi seguro	90%

(Brumbaugh, *et al*, 2013)

Cuadro 50. Criterio de consecuencia.

Nivel	Consecuencia
1	Mínima o nula consecuencia en desempeño
2	Leve reducción en desempeño o duración de misión, tolerable con pequeño impacto al proyecto.
3	Reducción moderada en desempeño o duración de misión, impacto limitado al proyecto.
4	Reducción significativa en desempeño o duración de la misión, puede afectar el éxito del proyecto.
5	Reducción severa en desempeño, afectará el éxito de la misión.

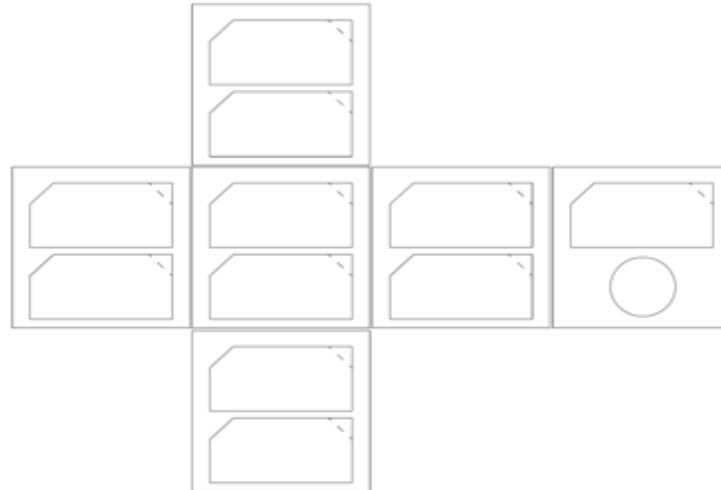
Se analizaron los riesgos identificados para el módulo de potencia y se planteó un plan de mitigación de los riesgos en relación a las causas de los riesgos. Además, se clasificaron las causas con sus respectivas técnicas de mitigación, acciones a tomar y responsables de llevar a cabo las acciones. Las técnicas de mitigación (TM) son eliminar la causa o consecuencia (E), controlar las causas o consecuencias (C), transferir la responsabilidad (T) o asumir el riesgo (A).

N. FACTIBILIDAD DEL DISEÑO DE POTENCIA DE FASE 4

El módulo de potencia desarrollado durante la fase 4 prevé la utilización de dos paneles solares, uno de ellos conformado por 6 celdas solares y el otro con las cinco celdas restantes, se utilizarán once celdas solares. Debido a que el sistema ADCS que se utilizará es pasivo, no se tiene un control lo suficientemente capaz de la actitud del satélite como para ubicar todas las celdas en un plano, como se planteó en la fase 3 del Megaproyecto.

Tomando esto en cuenta se llegó a la conclusión que la mejor opción es ubicar las celdas solares en las caras externas del satélite. Debido a que se deberá dejar un espacio libre para adquirir información a través de la captura de imágenes, no es posible utilizar la totalidad de una de las caras externas. Tomando en cuenta las dimensiones de las celdas solares de AZUR SPACE 3G30A que se utilizarán en el satélite, se definió que solo dos celdas podrían ser ubicadas en cada una de las caras externas, con la excepción de la cara con la apertura para la cámara, en la cual solo se pondrá una celda solar.

Figura 23. Arreglo físico de celdas solares.



1. **Generación de potencia eléctrica.** Para llevar a cabo el análisis de factibilidad del diseño propuesto por la fase 4 del Megaproyecto se asumió la órbita de la Estación Espacial Internacional (ISS, por sus siglas en inglés). Esto es debido a que se desconoce la información precisa de la órbita que tendrá el satélite al ser lanzado, pero se sabe que será lanzado desde la ISS.

Esta órbita tiene una excentricidad de 0.0004087, por lo que es casi circular, tiene una altura de perigeo de 401 km y una altura de apogeo de 407 km sobre la superficie de la Tierra. Además, tiene una ascensión recta del nodo ascendente de 302.9462°. (Heavens Above, 2017)

Se llevó a cabo una simulación utilizando el software STK de AGI para poder obtener el perfil de luz al cual se vería expuesto el satélite en órbita. Ya que se prevé que el lanzamiento del satélite a la ISS sea a finales del año 2018 y su operación sea de seis meses se decidió realizar la simulación en STK desde el 1 de enero de 2019 hasta el 1 de enero de 2020. Es importante notar que por lo general el tiempo entre el arribo a la ISS y el despliegue el espacio es de uno a seis meses.

Figura 24. Ajustes utilizados para simulación en STK.

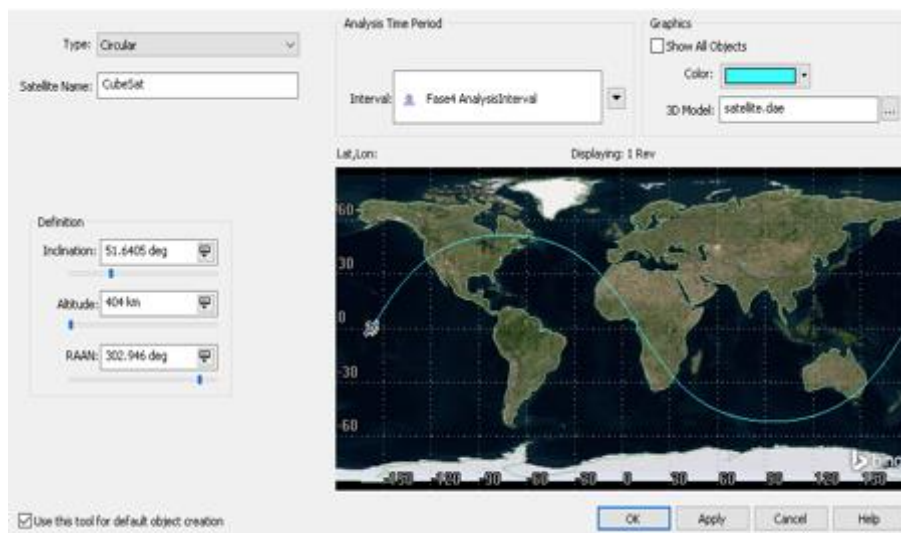


Figura 25. Simulación en 3D de la órbita del satélite.



Figura 26. Órbita 2D del satélite, con órbita de la ISS.

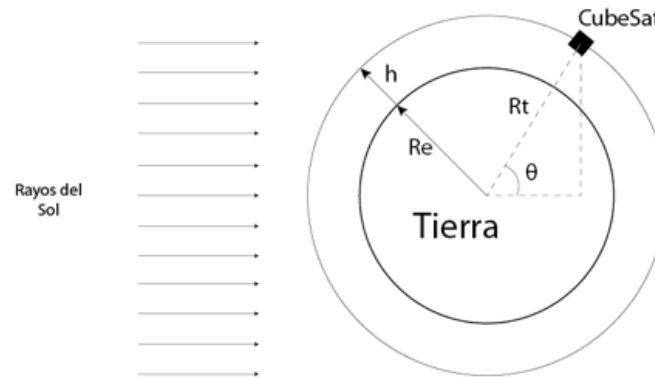


A partir de la simulación se obtuvo el perfil de luz al cual estaría expuesto el satélite a través del año 2019. Se encontraron muchos datos atípicos acerca de la duración de luz, penumbra y umbra, por lo que se decidió realizar el análisis de forma analítica, sin la ayuda del software STK.

Para llevar esto a cabo se trabajó con varios supuestos, como el que el tiempo de penumbra era despreciable a comparación del tiempo de luz o sombra. También que la órbita era circular, considerando que la altitud del perigeo es de 401 km y la del apogeo es de 407 km, se trabajó con un promedio de 404 km. También se despreció la inclinación de la órbita debido a que se desconoce el momento del lanzamiento.

Tomando en cuenta que la inclinación de la órbita se da con respecto a la línea del Ecuador de la Tierra y que la Tierra varía su inclinación con respecto al Sol desde -23.5° hasta 23.5° en el transcurso de un año, se consideró que aumentaba mucho la complejidad del análisis y que además podría afectar los resultados al no reflejar a cabalidad la realidad.

Figura 27. Relación entre luz proveniente del Sol y ángulo de ubicación en órbita del CubeSat.



La Figura 27 se utilizó para calcular el tiempo de luz y sombra del satélite con la órbita de la ISS. Se tomaron en cuenta el radio de la Tierra (R_e), la altura de la órbita (h), la suma de ambos (R_t) y la posición angular del CubeSat en la órbita (θ). Al igualar el lado opuesto del ángulo (θ) del triángulo con hipotenusa (R_t) al radio de la Tierra de 6,371 km. Se obtuvo el ángulo a partir del cual el CubeSat recibiría luz solar, este fue de 70.11° .

Tomando en cuenta que la órbita es circular y es simétrica se obtuvo que el mismo ángulo delimitaba cuando el CubeSat ya no recibiría más luz solar. Al restar este ángulo a la revolución completa (360°) se obtuvo que el ángulo en el cual iniciaba la sombra era 289.89° . Se calculó la velocidad del satélite al utilizar la siguiente ecuación.

$$V = \sqrt{\frac{\mu}{Rt}} = \sqrt{\frac{398600 \frac{km^3}{s}}{6775 km}} = 7.67 \frac{km}{s}$$

(70)

Donde (μ) es la constante gravitacional de la Tierra que es constante y (V) es la velocidad tangencial del satélite. Luego se obtuvo la duración del periodo para finalmente obtener la duración de sombra y la luz al relacionar el ángulo (θ), longitud de la órbita y la velocidad tangencial (V).

$$T_{\text{periodo}} = \frac{2\pi R_t}{V} = 5,549.77 \text{ s} \quad (71)$$

$$T_{i.\text{sombra}} = \frac{360 - \theta}{360} T_{\text{periodo}} = 4,468.95 \text{ s} \quad (72)$$

$$T_{f.\text{sombra}} = \frac{\theta}{360} T_{\text{periodo}} = 1,080.81 \text{ s} \quad (73)$$

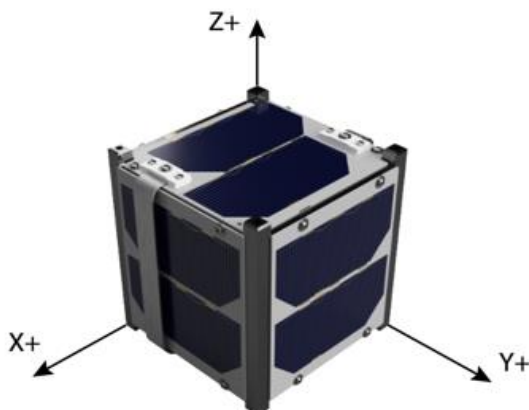
Es importante notar que el tiempo final de sombra ocurre ya en la siguiente revolución. Tomando en cuenta el lapso ocurrido entre el inicio y el final de la sombra se calculó la duración de la sombra y la duración de la luz solar sobre el satélite.

Cuadro 51. Perfil de luz/sombra del satélite utilizado para análisis de fase 4.

	Tiempo de luz (s)	Tiempo de sombra (s)	Periodo (s)
Perfil luz/sombra	3,388.14	2,161.63	5,549.77

A diferencia que el caso analizado para la fase 3, el ángulo de incidencia de los rayos del Sol con respecto a la normal a la superficie de las celdas solares si varía. El sistema de ADCS está siendo trabajado en base al que se utilizó en el satélite tipo CubeSat CSSWE, este presentó una rotación durante la misión de aproximadamente 3° por segundo para el primer mes en órbita.

Figura 28. Marco de referencia del satélite.



La rotación de 3° por segundo era con respecto al eje Z. En cuanto a la rotación en los ejes X y Y, esta se logró controlar a través del sistema ADCS pasivo y su rotación era prácticamente nula.

2. Consumo de potencia eléctrica por módulo. El consumo de potencia eléctrica para la fase 4 del Megaproyecto se encuentra detallado, por módulo, en los cuadros 52 - 55.

Cuadro 52. Características eléctricas de ADCS fase 4.

Componente	Cantidad	Voltaje (V)	Corriente (mA)	Potencia unitaria (W)	Potencia total (W)
Magnetometro ¹	1	3.300	0.100	3.30 E-4	3.30 E-4
Fotodiodos ²	6	3.300	0.100	3.30 E-4	0.002

1 = (Honeywell, 2010)

2 = (Vishay, 2017)

El consumo de potencia de este módulo es bajo, por lo que es considerado un sistema pasivo. Además, requiere de otros componentes como circuitos integrados y barras de histéresis con un consumo nominal de corriente en el orden de los micro amperios o en el caso de las barras de histéresis, nulo.

Cuadro 53. Características eléctricas de PLD fase 4.

Componente	Cantidad	Voltaje (V)	Corriente (mA)	Potencia unitaria (W)	Potencia total (W)
Cámara ¹	1	3.300	84.800	0.280	0.280
Motor piezoeléctrico ²	1	5.000 ³	2000.000	10.000	10.000

1 = (Kuuste, *et al*, 2014)

2 = (Teckeleo, 2017)

3 = (Telemaq, 2017)

El consumo total de PLD para la fase 4 constituye 10.280 W. Es importante notar que el motor piezoeléctrico actúa mediante pulsos eléctricos, por lo que la potencia se requeriría en instantes específicos.

Cuadro 54. Características eléctricas de C&DH fase 4.

Componente	Cantidad	Voltaje (V)	Corriente (mA)	Potencia unitaria (W)	Potencia total (W)
OBC ¹	1	3.300	51.515	0.170	0.170

1 = (Gomspace, 2016)

Cuadro 55. Características eléctricas de COMMS fase 4.

Componente	Cantidad	Voltaje (V)	Corriente (mA)	Potencia unitaria (W)	Potencia total (W)
Módulo integrado ¹	1	3.300	1200.000	3.960	3.960

¹ = (Gomspace, 2016)

3. Consumo de potencia eléctrica por modo de operación. A diferencia de la fase 3, al implementar un sistema pasivo de ADCS se tienen tres modos de operación en lugar de cuatro. Estos son: vuelo, vuelo sobre Guatemala, vuelo sobre objetivo. Debido a que no existe un control activo de la actitud del satélite se alteró el modo de operación del mismo, de tal manera que las fotos se tomarían cuando se tuvieran los suficientes recursos para llevarlo a cabo, en lugar de hacerlo cuando se sobrevolara Guatemala. Además, para aplicar al programa KiboCube de UNOOSA/JAXA era necesario registrar el satélite como originario de Guatemala, para lo cual era necesario tener la estación terrena en Guatemala, por lo que la comunicación entre el satélite y la estación terrena se llevará a cabo cuando se sobrevuele Guatemala.

Cuadro 56. Consumo de potencia durante vuelo.

	Potencia ADCS (W)	Potencia C&DH (W)	Potencia total (W)
Fase 4	0.002	0.170	0.172

Cuadro 57. Consumo de potencia durante vuelo sobre objetivo.

	Potencia ADCS (W)	Potencia C&DH (W)	Potencia PLD (W)	Potencia total (W)
Fase 4	0.002	0.170	10.280	10.452

Cuadro 58. Consumo de potencia durante vuelo sobre Guatemala.

	Potencia ADCS (W)	Potencia C&DH (W)	Potencia COMMS (W)	Potencia total (W)
Fase 4	0.002	0.170	3.960	4.132

En la fase 3 del Megaproyecto se calculó que el tiempo disponible para la comunicación entre la estación terrena y el satélite era de 7 minutos y 30 segundos, este tiempo fue asumido como válido para la captura de imágenes para poder estimar el consumo de energía del satélite, también se asumió en qué momento se energizaba y desenergizaban los módulos de acuerdo al concepto de operaciones.

O. PROCEDIMIENTOS DE ENSAMBLAJE DEL MÓDULO DE POTENCIA

1. Paneles solares. Los paneles solares serán ensamblados con base en los procedimientos de ensamblaje desarrollados por el personal de la Universidad de Colorado, Boulder, que trabajó en el CubeSat MinXSS.

Cuadro 59. Listado de materiales para ensamble de paneles solares.

Materiales
Celdas solares 3G30A de AZUR SPACE
PCBs externos
Material de aporte para soldadura Sn96.5/Ag 3.5%
Cinta adhesiva de doble lado para alta temperatura
Guantes de látex
Pinzas
Pinzas de vacío
Bisturí
Pulseras anti estáticas
Alcohol
Pluma de flujo limpio (no clean flux pen)

(Dahir, *et al*, 2017)

Es importante notar que las PCBs externas deberán ser diseñadas específicamente para la aplicación. Además, se requerirá acceso a equipo como: cámara de vacío, *cleanbox* con una superficie antiestática y un ionizador. Se recomienda mucho cuidado ya que las celdas solares y las PCBs pueden llegar a ser bastante frágiles. A continuación, se detallan los pasos a seguir para el ensamblaje de los paneles solares.

Figura 29. Vista frontal de configuración de PCBs externas.

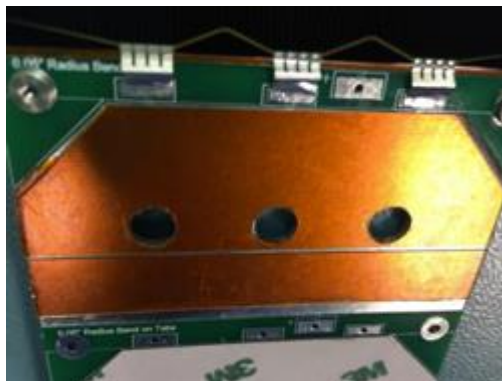
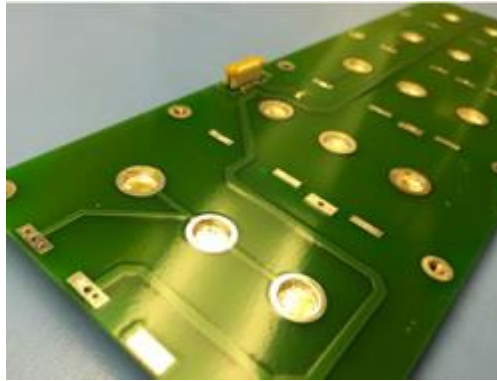


Figura 30. Vista trasera de configuración de PCBs externas.



Pasos: (Dahir, *et al*, 2017)

- Limpiar superficie de trabajo. Para ello se deberán poner las pulseras anti estáticas y conectarlas a la superficie anti estática de la *cleanbox*. Luego, se deberá encender el ionizador para despejar las partículas de la superficie de trabajo.
- Para montar los electrónicos que irán en la superficie exterior de los PCBs se deberá limpiar la superficie sobre la cual se colocarán utilizando una pluma de flujo limpio. Luego se procederá a soldar los componentes como fotodiodos y sensores de temperatura.
- Se deberán limpiar las superficies donde se montarán las celdas solares con alcohol. Para luego colocar la cinta adhesiva de doble lado para alta temperatura en tres tiras. La primera de ellas debe ser la superior, la segunda será la inferior y la última será la tira del medio que cubrirá los tres agujeros para soldadura. Se dejará excedente de cinta adhesiva que luego será removida. No se debe remover la protección del adhesivo de la cara sobre la cual irán las celdas solares.
- Para remover el excedente de cinta adhesiva se deberá utilizar un bisturí. Primero se dará vuelta a la PCB y se removerá el excedente. Luego, se dará vuelta de nuevo a la PCB y se removerá la cinta adhesiva de los agujeros para soldadura. Para llevar esto a cabo se recomienda posicionar la PCB sobre dos apoyos ubicados en los extremos de la misma.
- Para preparar las PCBs se deberá remover la capa protectora de la cinta adhesiva. Se recomienda llevar esto a cabo con mucho cuidado y haciendo uso de dos pinzas, una para sujeción de la cinta y otra para remoción de la capa protectora. Se deberán eliminar todas las arrugas de la cinta adhesiva, para ello se deberán aplanar, cortar o empujar hacia los agujeros si se considera apropiado.

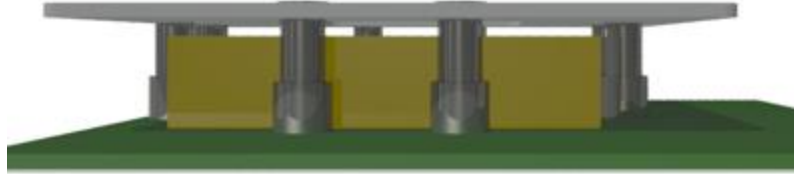
- Se deberán seleccionar las celdas solares con la corriente más cercana entre sí, esto basado en las especificaciones del proveedor al momento de la entrega. Para colocar las celdas solares se deberá utilizar las pinzas de vacío, tomar en cuenta la orientación de las mismas y las posiciones de los interconectores integrados. En caso de mal posicionamiento se podrá utilizar una cuchilla para remover la celda, esto requiere de extremo cuidado por la fragilidad de las celdas.
- Se deberá colocar un cilindro de 1/16 de pulgada debajo de los interconectores para ello se deberán doblar los interconectores utilizando pinzas de manera cuidadosa rodeando al rodillo para formar un lazo. Luego, se deberán soldar los interconectores a las PCBs, uno de los interconectores es un diodo de protección y los demás son el polo positivo de la celda solar.
- Para uniformizar la adhesión de las celdas solares se deberá colocar las PCBs dentro de una bolsa para vacío. Evitar que cables se encuentren sobre las celdas solares durante este proceso. Se deberá colocar una superficie plana debajo de las PCBs para levantar las mismas, dejando el mínimo espacio libre para el conector ubicado debajo de las PCBs. Luego, se deberá accionar la cámara de vacío por a un máximo de 24 horas, la presión deberá ser la dictada por el proveedor de las celdas solares.
- Existe un método alternativo que utiliza un bloque o peso con superficie inferior lisa para ejercer presión sobre la PCB, sin embargo, se considera más apropiada la cámara de vacío.
- Se colocará la PCB viendo hacia abajo y se colocará el material de aporte dentro de los agujeros para conectar la cara trasera, que es el polo negativo, de las celdas solares a la PCB. Se deberá colocar la PCB dentro de un horno a 80 °C durante 3 horas. Luego se deberá verificar el estado del material de aporte, si este está duro, entonces el proceso ha sido finalizado.
- Finalmente se realizarán las pruebas pertinentes para comprobar que la instalación no hay perjudicado el desempeño de las celdas solares.

Se prevé que los paneles solares sean formados al conectar las caras con las celdas solares ensambladas a través de una PCB dedicada a la interconexión de las caras. Además, esta PCB servirá para interconectar los paneles solares con el resto del sistema de potencia del satélite.

2. **Baterías.** Para el caso de las baterías se tendrá una estructura interna para controlar hasta cierto punto la expansión de las baterías en condiciones de temperatura y vacío y así controlar la fatiga mecánica

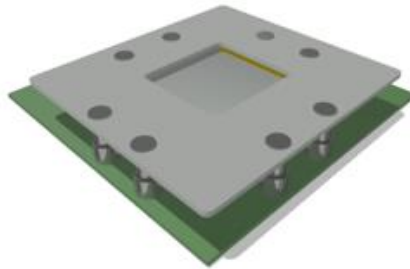
asociada. Esta estructura interna y el procedimiento de ensamblaje están basados en el ensamblaje de las baterías utilizado en el CubeSat 3U MinXSS-1 de la Universidad de Colorado, Boulder.

Figura 31. Vista frontal de estructura interna para expansión de baterías.



Se recomienda llevar a cabo un diseño y análisis estructural y térmico de esta estructura para verificar que cumpla con los requerimientos de vibraciones, masa, volumen y temperatura. Las figuras 31 y 32 son solo con fines representativos.

Figura 32. Estructura interna para expansión de baterías.



Cuadro 60. Listado de materiales para ensamble de baterías.

Materiales
Baterías Li-ion 2Ah de Sparkfun
Arathane 5753
Calentadores
Sensores de temperatura
Cinta adhesiva de transferencia de calor
PCB
Placa de aluminio
Separadores mecánicos
Guantes de látex

Pasos: (Mason, *et al*, 2015)

- El primer paso será adherir la celda de la batería inferior a la PCB utilizando Arathane 5753, se deberá tomar en cuenta la orientación de las celdas para facilitar la conexión. Se debe dejar curar este adhesivo para garantizar una buena sujeción.

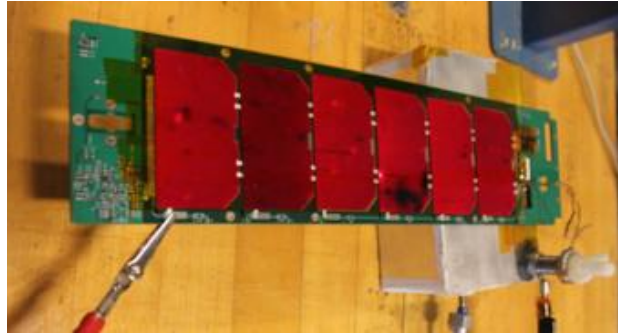
- Luego, se colocará la cinta adhesiva para transferencia de calor en la superficie de una de las caras de las celdas de la batería y se colocarán los calentadores y los sensores de temperatura en esta superficie dejando libre sus cables de conexión.
- Se posicionará una celda de la batería sobre la otra tomando en cuenta la orientación de las mismas para la conexión. La cinta adhesiva para transferencia de calor ayudará a que estas queden ensambladas. Luego, se deberán conectar las celdas en serie para formar la batería.
- Se deberán insertar y ajustar los separadores mecánicos a la PCB para posteriormente colocar la placa superior de aluminio y ajustar la placa a los separadores mecánicos.
- Luego, se llevarán a cabo las pruebas necesarias para comprobar que el funcionamiento de las baterías sea el adecuado y que estas no hayan sido dañadas durante el proceso de ensamblaje.
- Finalmente, se conectarán los sensores de temperatura, calentadores y batería al resto del sistema de potencia.

P. PRUEBAS DEL MÓDULO DE POTENCIA

Para comprobar el correcto funcionamiento de los componentes es necesario llevar a cabo pruebas, como un plan de verificación. Las pruebas deberán probar el funcionamiento de las baterías, los paneles solares y el sistema integrado.

1. Pruebas para paneles solares. Las pruebas de electroluminiscencia pueden revelar daños a las celdas solares. Estas pruebas se pueden realizar antes y después de la instalación, de esta manera se revela el estado de las celdas solares al momento de ser recibidas y revelar no conformidades en el producto. Para llevar estas pruebas a cabo es necesario utilizar una fuente de potencia. Se debe calibrar el voltaje para que corresponda el de las celdas solares o el panel solar dependiendo de a lo que se le esté haciendo pruebas. También se deberá calibrar el límite de la corriente baja, se recomienda 0.01 A. Se deberán conectar las celdas solares, o el panel solar a la fuente de voltaje y las celdas solares se deberían iluminar. Para ello es necesario incrementar de manera gradual la corriente. (Dahir, *et al*, 2017)

Figura 33. Prueba de electroluminiscencia a paneles solares.



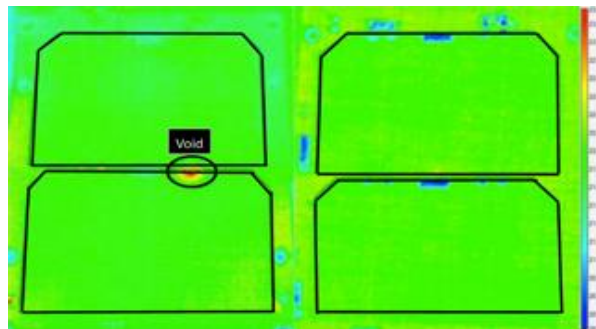
(Sandberg, 2016)

Al momento de iluminarse las celdas solares se deberían iluminar por completo si estas se encuentran en buen estado. En caso de algún daño se verían áreas negras que no se iluminan.

Otras pruebas que se pueden realizar a los paneles solares son de termografía. Para llevar estas pruebas a cabo se necesitará una cámara termográfica y una fuente de calor.

Los paneles solares se deberán exponer a calor y luego se deberá evaluar la disipación de calor utilizando la cámara termográfica. El objetivo de realizar esta prueba es identificar la formación de burbujas que impidan la disipación de calor. La disipación de calor puede afectar la temperatura operativa de las celdas solares afectando su desempeño e incluso causando que estas fallen. (Sandberg, 2016)

Figura 34. Pruebas de termografía a paneles solares.



(Sandberg, 2016)

Además, se recomienda realizar pruebas para probar el desempeño de las celdas solares, para ello se necesita una fuente de luz y un multímetro para medir la potencia eléctrica generada por los paneles solares. Se puede realizar con el Sol como fuente de luz, que es más económico, sin embargo, se depende de las condiciones climáticas y claridad bajo las cuales sean evaluadas. Para tener medidas más precisas se recomienda realizar con una fuente de iluminación artificial de la cual se conozcan sus características.

2. **Pruebas para baterías.** Las pruebas para las baterías deberían incluir inspección visual. En la inspección visual se deberán observar posibles golpes o fugas. También se debería medir el voltaje de la batería ensamblada y medir voltaje, corriente y potencia durante al menos 3 ciclos de carga/descarga. (Mason, *et al*, 2015)

Se deberán probar las protecciones y su funcionamiento al alcanzar los límites superiores e inferiores de carga. También se deberá medir la masa antes y después de pruebas en vacío para medir la pérdida total de masa (TML). (Mason, *et al*, 2015)

3. **Pruebas para sistema integrado.** Se deberán realizar pruebas de termo vacío para probar el sistema en un ambiente que refleje en cierta medida las condiciones de operación que se tendrán en el espacio. Para ello se requiere de una cámara de termo vacío especializada. Se deberán efectuar varios ciclos de cambios de temperatura para comprobar el correcto funcionamiento de los componentes. Se recomiendan realizar estas pruebas para todo el sistema incluyendo los módulos que componen al satélite.

Las pruebas deben ser con el sistema en funcionamiento para validar los límites operacionales de los componentes del satélite. Además, este ambiente permite la expansión y contracción que los componentes tendrán en el espacio y esta prueba puede revelar posibles problemas.

Adicionalmente a estas pruebas se deberán realizar pruebas de vibraciones para el satélite integrado para comprobar que todos los elementos y las uniones realizadas resistan el ambiente de despegue. Para ello será necesario someter el sistema a vibraciones aleatorias y vibraciones cosenoidales que se ajusten a los parámetros del vehículo de lanzamiento. (Steiner, 2007)

Figura 35. Pruebas de vibraciones a sistema integrado.



(Steiner, 2007)

Las pruebas podrán ser modeladas a través de software computacional, esto para dar cierta validación preliminar e identificar posibles modos de falla. Sin embargo, las pruebas son necesarias ya que el sistema integrado podría no ser representado de manera fiel en el programa computacional afectando los resultados obtenidos del análisis.

Q. MÓDULO DE MANEJO DE COMANDOS Y DATOS

La misión contempla distintos niveles de éxito desde un criterio mínimo hasta un criterio completo de éxito, estos niveles se presentan a continuación:

Cuadro 61. Criterios de éxito para misión.

ID	Objetivo	Criterio de evaluación
Criterios mínimos de éxito		
1	Para <i>detumble</i> y estabilizar el CubeSat después del despliegue.	Adquisición de imágenes (sensor orientado a la tierra)
2	Para desplegar la antena	Adquisición de enlace de comunicación
3	Para recolectar energía y carga de baterías	Sensores a bordo
4	Para mandar información de estado para Tierra	Recibió paquetes de datos en el suelo
5	Para recibir y procesar paquetes de datos de Tierra	Datos brutos recibidos presentados a los usuarios en la Interfaz gráfica de usuario (GUI)
6	Para desorbitar	Seguimiento satelital
Criterios significantes de éxito		
7	Para adquirir imágenes a una longitud de onda	Paquete de datos recibidos que indica el almacenamiento a bordo de los datos de imagen
8	Para transmitir imágenes a Tierra	Paquete de datos recibido incluyendo la imagen
9	Para cambiar la posición del carrusel de filtro	Paquete de datos recibido indicando posición del carrusel
Criterios completos de éxito		
10	Para adquirir imágenes en tres longitudes de onda diferentes	Paquete de datos recibidos que incluye imágenes en tres longitudes de onda diferentes
11	Para procesar las imágenes en la Tierra	Datos de concentración de clorofila calculados a partir de imágenes

Basado en estos, se definieron de los requerimientos de misión, los cuales definen los aspectos necesarios a implementar para lograr el criterio completo de éxito. Estos requerimientos se presentan a continuación:

Cuadro 62. Requerimientos de misión para manejo de comandos y datos.

ID	Requerimiento
100	La OBC activará e inicializará los subsistemas de carga útil y COMM cuando el satélite esté sobre el GCS
110	La OBC verificará que carga útil haya tomado todas las imágenes.
120	La OBC mandará imágenes al módulo de COMM.
130	La OBC deberá comparar las coordenadas de CubeSat y las coordenadas globales de Guatemala.
140	El protocolo de comunicación entre todos los componentes será I2C.
150	La OBC tendrá redundancia.
160	El satélite tendrá un sistema de <i>watchdog</i> para cambiar entre OBCs.

Con los requerimientos de misión definidos, se continuó definiendo los requerimientos funcionales requeridos para la incorporación de los módulos de carga útil, ADCS, energía y comunicaciones en base a la delimitación de este trabajo.

Cuadro 63. Requerimientos funcionales.

ID	Módulo	Requerimiento
1	ADCS	Recolectar la información sobre los fotodiodos.
2	ADCS	Almacenar información sobre sobre los fotodiodos.
3	ADCS	Validar que la información sobre los fotodiodos se encuentre dentro del rango correcto.
4	ADCS	Recolectar la información sobre el magnetómetro.
5	ADCS	Almacenar la información sobre el magnetómetro.
6	ADCS	Validar que la información sobre el magnetómetro se encuentre dentro del rango correcto.
7	PLD	Tomar fotografías.
8	PLD	Almacenar fotografías.
9	PLD	Mover el carrusel de filtros.
10	PLD	Cuando se mueva el carrusel de filtros, actualizar el registro con el filtro utilizado.
11	PWR	Recolectar la información sobre el estado de carga de las baterías cada 120 segundos.
12	PWR	Almacenar al menos una órbita de información del satélite sobre el estado de carga de las baterías.
13	PWR	Validar que la información del estado de carga de las baterías esté dentro del rango 0, 65535 mAh.
14	PWR	Recolectar la información sobre el voltaje disponible en las baterías cada 120 segundos.

Continuación Cuadro 63

ID	Módulo	Requerimiento
15	PWR	Almacenar al menos una órbita de información del satélite sobre el voltaje disponible en las baterías.
16	PWR	Validar que la información del voltaje disponible en las baterías esté dentro del rango -16.4, 16.4 Volts.
17	PWR	Recolectar la información sobre la temperatura de las baterías cada 60 segundos.
18	PWR	Almacenar al menos una órbita de información del satélite sobre la temperatura de las baterías.
19	PWR	Validar que la información de la temperatura de las baterías esté dentro del rango -128, 128°C.
20	COMMS	Enviar imágenes al módulo de comunicaciones.
21	COMMS	Enviar información sobre los sensores del módulo de energía al módulo de comunicaciones.
22	C&DH	El tiempo de respuesta de los componentes del módulo de ADCS no debe ser superior a 300 milisegundos.
23	C&DH	El tiempo de respuesta de la cámara no debe ser superior a 1 segundo.
24	C&DH	El tiempo de respuesta del motor del carrusel no debe ser superior a 1 segundo.
25	C&DH	La temperatura mínima de operación para la computadora a bordo debe ser menor o igual -20°C.
26	C&DH	La temperatura máxima de operación para la computadora a bordo debe ser mayor o igual a 60°C.
27	C&DH	El tamaño de la computadora a bordo no debe exceder los 100x100x113 mm.
28	C&DH	La masa de la computadora a bordo no debe exceder los 100 g.
29	C&DH	La computadora a bordo no debe consumir más de 0.4 W.

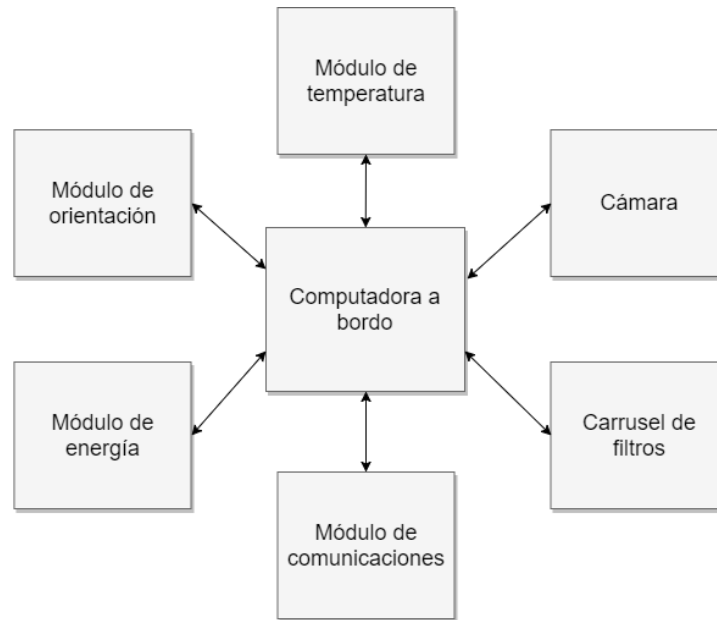
R. DISEÑO MÓDULO DE MANEJO DE COMANDOS Y DATOS

1. Arquitectura de hardware

a. Arquitectura centralizada. La arquitectura define la manera como la OBC estará físicamente conectada a los otros módulos del CubeSat. De acuerdo con los requerimientos de nuestra misión, se determinó que lo mejor era utilizar una arquitectura centralizada, en donde se tendría una OBC principal conectada a todos los demás módulos del CubeSat que necesitaran algún tipo de control. Esta decisión se tomó con base en los siguientes aspectos:

- **Simplicidad:** el diseño de una arquitectura centralizada es más sencillo y fácil de aplicar debido a que contempla únicamente un componente para el control de todos los módulos.
- **Espacio:** la incorporación de un único elemento para el control de los demás módulos ayuda a minimizar el espacio utilizado dentro del CubeSat y los problemas que puede provocar su colocación, esto es un aspecto muy importante debido a que el espacio es crítico en este tipo de satélites debido a que su volumen es muy reducido.
- **Peso:** debido a los estándares establecidos el peso es uno de los factores importantes, la incorporación de la arquitectura centralizada disminuye la cantidad de componentes para el control y en consecuencia el peso.
- **Consumo de potencia:** el consumo de potencia es un aspecto crítico durante la operación de un satélite debido a que limita las acciones que puedan realizarse, con esta arquitectura se disminuye en consumo debido a la utilización de un componente para el control.

Figura 36. Arquitectura centralizada.



La figura muestra la manera como estará diseñada la arquitectura de hardware, en donde todos los demás módulos serán controlados por una OBC central.

b. **Selección OBC.** Luego de la definición de la arquitectura, se definió cual sería la OBC encargada de controlar los demás módulos. Para seleccionarla se consideran los siguientes aspectos:

- **Masa:** debido a las restricciones de peso que define los estándares del CubeSat, la OBC seleccionada debe tener una masa mínima.

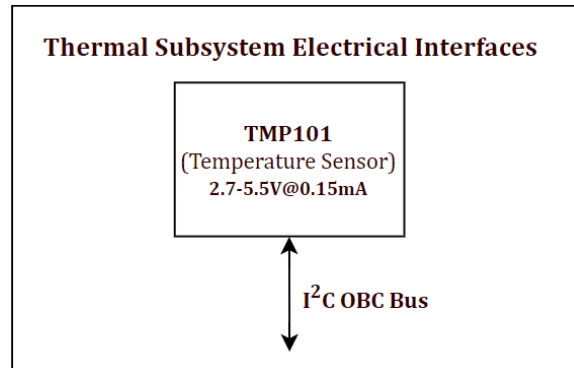
- Volumen: las restricciones de volumen de $10 \times 10 \times 10 \text{ cm}^3$ limitan las opciones de OBC a dispositivos con dimensiones menores a los 10 cm, pero más allá de las dimensiones también es importante que la OBC posea un volumen pequeño para que no se dificulte su incorporación dentro del CubeSat.
- Consumo de Potencia: el consumo de potencia es un factor fundamental debido a que la energía almacenada por los CubeSat suele ser muy pequeña, por lo tanto, mientras menos potencia consuma la OBC será mejor.
- Velocidad de reloj: los tiempos de procesamiento son críticos en este tipo de sistemas, debido a que una tardía reacción de procesamiento podría significar el fin de la misión.
- RAM: este aspecto va de la mano con la velocidad de reloj, de igual manera es importante para manejar los tiempos de procesamiento, debido a que maneja las instrucciones que son enviadas al procesador.
- Memoria: el satélite se encargará de recolectar información, entre lo cual están incluidas imágenes por lo tanto la OBC debe tener memoria de almacenamiento suficiente para poder guardarlas.
- Interfaces de comunicación: como ya se definió en la arquitectura de hardware la OBC controla los demás módulos, por lo cual necesita un canal de comunicación con cada uno de los módulos, esta es la razón por la cual deben evaluarse que la OBC posea distintas interfaces de comunicación.
- Curva de aprendizaje: cualquier dispositivo tiene su curva de aprendizaje para conocer a fondo su funcionamiento, el desconocimiento sobre el dispositivo expone una serie de riesgos extras para el proyecto debido a esto es importante evaluar si es posible llegar a tener un conocimiento profundo.
- Misiones: se debe asegurar que la OBC utilizada puede soportar las condiciones extremas que presenta el espacio, y no hay mejor manera que evaluando si la OBC ha sido previamente utilizada para una misión espacial.
- Costo: aunque no es un factor determinante en el éxito de la misión, cuando se está trabajando con pocos recursos económicos como es el caso de este proyecto es importante conocer si se puede ser adquirido o no.

Estos aspectos fueron evaluados en un *trade study* para siete dispositivos (Raspberry Pi Zero, STM32F407, LCP 3250, Stamp9G20, Mega 2560, MSP420F67791A, Prometheus PC-104), resaltando entre todos la Raspberry Pi Zero debido a lo cual fue seleccionada (ver sección resultados).

c. Interfaces eléctricas. Se definieron los requisitos eléctricos de cada componente y la interfaz de comunicación necesaria para su control, con esta información se determinó la interfaz de comunicación para el bus de datos.

1) Térmico. El sistema térmico tiene un sensor de temperatura, para verificar y validar la temperatura dentro del CubeSat, el chip es el TMP101, tiene un bus de comunicación I2C y necesita 2.7V - 5.5V para operar.

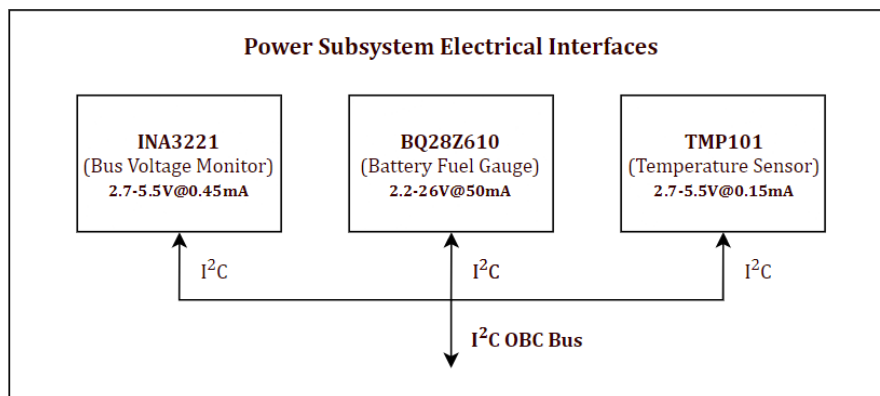
Figura 37. Interfaces del módulo térmico.



2) **Energía.** Este se compone de tres circuitos integrados que se encargan de medir el estado de las baterías del CubeSat y comunicarlo a OBC. Las fichas son:

- El INA3221 es responsable de medir tensiones y corrientes del panel solar, la batería y otros subsistemas, este chip tiene un bus de comunicación I2C y necesita 2.7V - 5.5V para operar.
- El BQ28Z610 es responsable de controlar, proteger y medir el estado de carga de las baterías, este chip tiene un bus de comunicación I2C y necesita 2.2V - 26V para operar.
- El TMP101 es responsable de medir la temperatura de las baterías CubeSat, este chip tiene un bus de comunicación I2C y necesita 2.7V - 5.5V para operar.

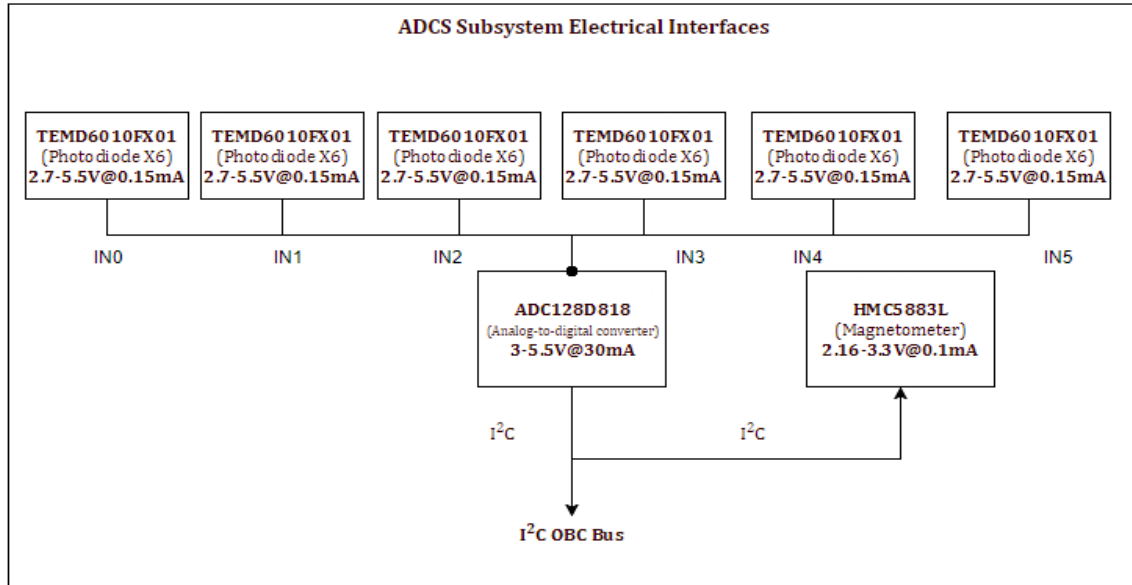
Figura 38. Interfaces del módulo de potencia.



3) **ADCS.** Este módulo consiste en un imán de barra y varillas de histéresis que trabajarán en conjunto para alinear el satélite al campo magnético de la Tierra y para proporcionar amortiguación para reducir las oscilaciones. Coordinado a través de un conjunto de fotodiodos y un magnetómetro responsables de recopilar información y transmitirla al OBC para su procesamiento.

Los fotodiodos Vishay TEMD6010FX01 requieren un convertidor analógico a digital para poder procesar la señal, el ADC128D818 cumple esta función, el ADC tiene un bus de comunicación (I2C); y ambos fotodiodos y ADC necesitan 3.3 V para operar. Mientras que el magnetómetro es el HMC5883L que ya contiene un ADC, tiene un bus de comunicación I2C y funciona con 3.3V.

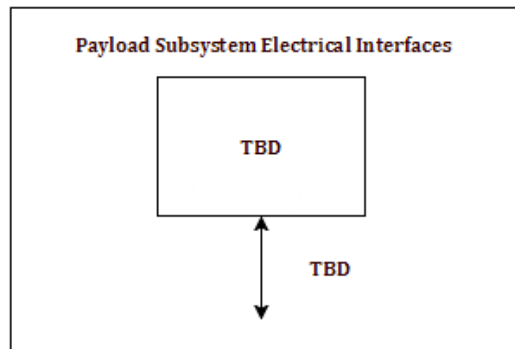
Figura 39. Interfaces del módulo ADCS.



4) Carga útil. Este se compone de cuatro elementos: una cámara monocromática, un motor, lentes y un carrusel de filtro. La OBC debe controlar el carrusel del filtro a través del motor y tomar capturas a través de la cámara monocroma.

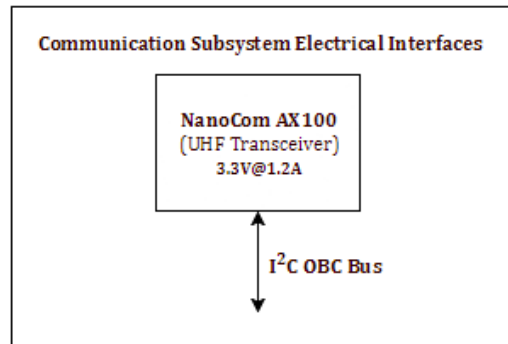
El motor en revisión es el motor piezoeléctrico Tekceleo WLG-30 que se puede controlar a través de los puertos GPIO. La cámara monocromática aún no se ha seleccionado, pero debería funcionar a través de una de las siguientes interfaces de comunicación: CAN, RS-485, RS-422, UART, I2C o SPI. Por lo tanto, las interfaces deben ser definidas.

Figura 40. Interfaces del módulo de carga útil.



5) **Comunicaciones.** El módulo se compone de dos componentes, un transceptor VHF / UHF (Nanocom AX100 de GOMSpace) y una antena omnidireccional basada en cinta de medir, construida internamente. El único componente que necesita comunicación con la OBC es el Nanocom AX100, el AX100 es un módulo COTS que tiene tres buses de comunicación diferentes que se pueden usar para la comunicación con OBC (I2C, UART). Además, el AX100 también tiene un canal UART específico para fines de depuración. Finalmente, el AX100 necesita 3.3V para operar.

Figura 41. Interfaces del módulo de comunicaciones.

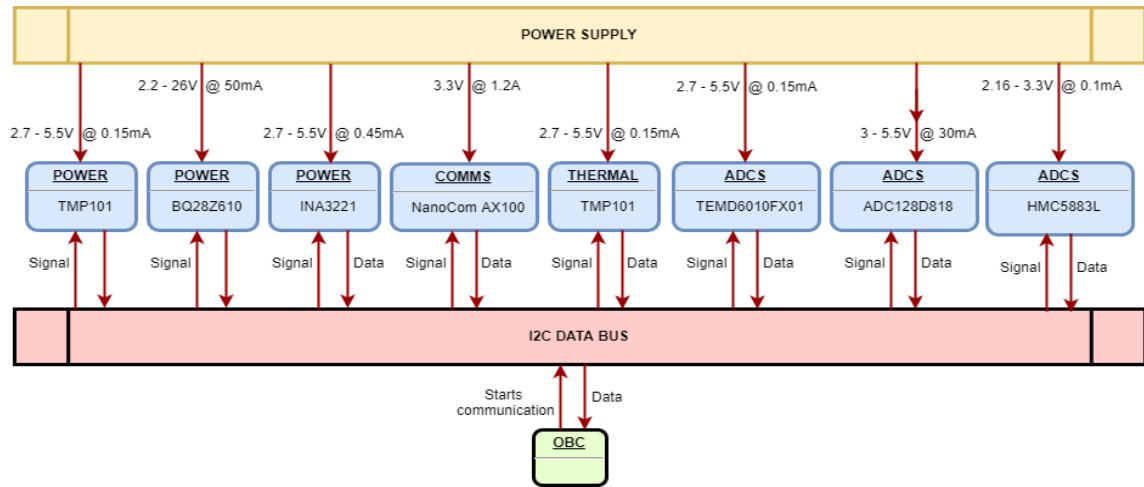


d. **Bus de datos.** El bus de datos determina el conjunto de conexiones físicas que compartirán los múltiples componentes de hardware que tendrá el satélite, con el fin de lograr la comunicación entre sí. Su propósito es reducir el número de rutas necesarias para su comunicación. Luego de identificar las interfaces de comunicación, se determinó que se utilizaría el protocolo I2C para el bus de datos, esto nos da una serie de ventajas.

- **Simple:** este protocolo utiliza únicamente dos cables de comunicación, por lo cual su conexión es muy simple y se reduce el espacio utilizado.
- **Multi-dispositivo:** este protocolo puede soportar múltiples maestros y esclavos, lo cual ayuda a poder incorporar todos los componentes necesarios.
- **Confiable:** utiliza una confirmación de información recibida con éxito.
- **Renombre:** es un protocolo ampliamente conocido y utilizado, incluso hay una gran cantidad de misiones espaciales exitosas que han utilizado este protocolo.

En esta configuración el dispositivo maestro será la OBC y cada uno de los componentes listados anteriormente para cada módulo serán esclavos, la comunicación es establecida un esclavo a la vez según el funcionamiento del bus de datos I2C, cada dispositivo esclavo tiene una dirección específica y es utilizada cada vez que se quiere establecer comunicación con el dispositivo, todas las direcciones son de 7 bits por lo cual se pueden tener hasta 128 dispositivos conectados, el siguiente diagrama muestra la manera como interactuarán los dispositivos.

Figura 42. Diagrama de comunicación mediante el bus de datos I2C.



2. Arquitectura de software. La arquitectura de software se desarrolló para la delimitación del proyecto, incorporando los módulos de Carga Útil, ADCS, Energía y Comunicaciones.

a. Descripción de casos de uso

Cuadro 64. Caso de uso: recolectar información de fotodiodos.

Nombre	Recolectar información de fotodiodos
Propósito	Recolectar la información proveniente del chip ADC128D818 que es el convertidor de señales a digital de las señales provenientes de los fotodiodos TEMD6010FX01.
Resumen	Se debe establecer conexión por medio del bus de datos I2C a la dirección del chip ADC128D818 y mandar la secuencia de instrucciones necesarias para obtener la información de los 6 canales requeridos por los fotodiodos.

Cuadro 65. Caso de uso: recolectar información de magnetómetro.

Nombre	Recolectar información del magnetómetro
Propósito	Recolectar la información proveniente del chip HMC5883L las cuales nos da la información de X, Y, Z.
Resumen	Se debe establecer conexión por medio del bus de datos I2C a la dirección del chip HMC5883L y mandar la secuencia de instrucciones necesarias para su configuración y la obtención de los valores necesarios.

Cuadro 66. Caso de uso: validar y almacenar información de fotodiodos.

Nombre	Validar y almacenar información de fotodiodos
Propósito	Validar que los datos obtenidos no estén corrompidos y guardar información para un log.
Resumen	Luego de obtener la información de los fotodiodos se debe verificar que se encuentre entre el rango de valores aceptados, si es el caso estos son almacenados en el log.

Cuadro 67. Caso de uso: validar y almacenar información de magnetómetro.

Nombre	Validar y almacenar información de magnetómetro
Propósito	Validar que los datos obtenidos no estén corrompidos y guardar información para un log.
Resumen	Luego de obtener la información del magnetómetro se debe verificar que se encuentre entre el rango de valores aceptados, si es el caso estos son almacenados en el log.

Cuadro 68. Caso de uso: tomar fotografía.

Nombre	Tomar fotografía
Propósito	Realizar la captura de las tres imágenes bajo los tres distintos filtros.
Resumen	Se debe mandar el comando a la cámara para capturar una fotografía, luego debe ser almacenada en un lugar específico para poder ser accesada después.

Cuadro 69. Caso de uso: mover carrusel.

Nombre	Mover carrusel
Propósito	Mover el carrusel de filtros para cambiar el filtro actual a uno nuevo, con el fin de poder tomar las fotos con los tres filtros distintos.
Resumen	Se debe mandar el comando de mover filtro al motor de acuerdo con el filtro correspondiente.

b. Diagramas de flujo

Figura 43. Diagrama de flujo para operación del módulo ADCS.

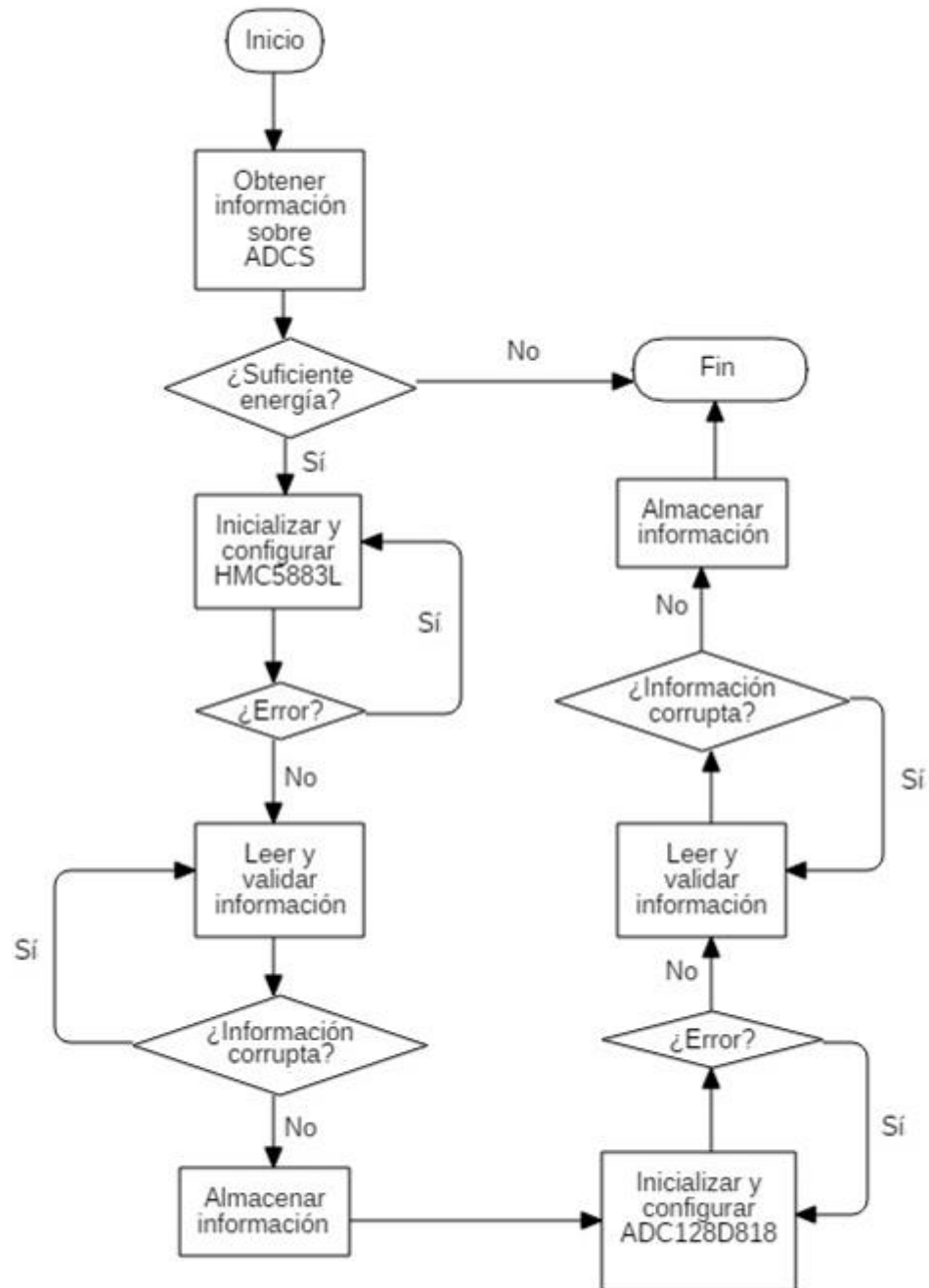


Figura 44. Diagrama de flujo para operación del módulo de carga útil.

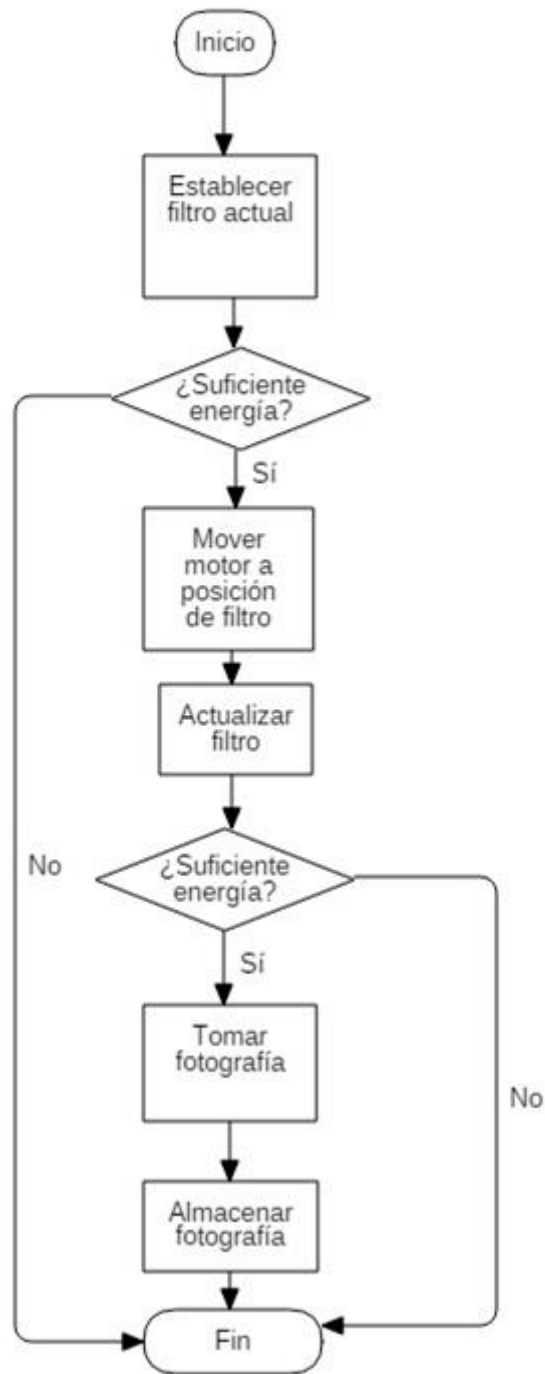


Figura 45. Diagrama de flujo para operación del módulo de energía.

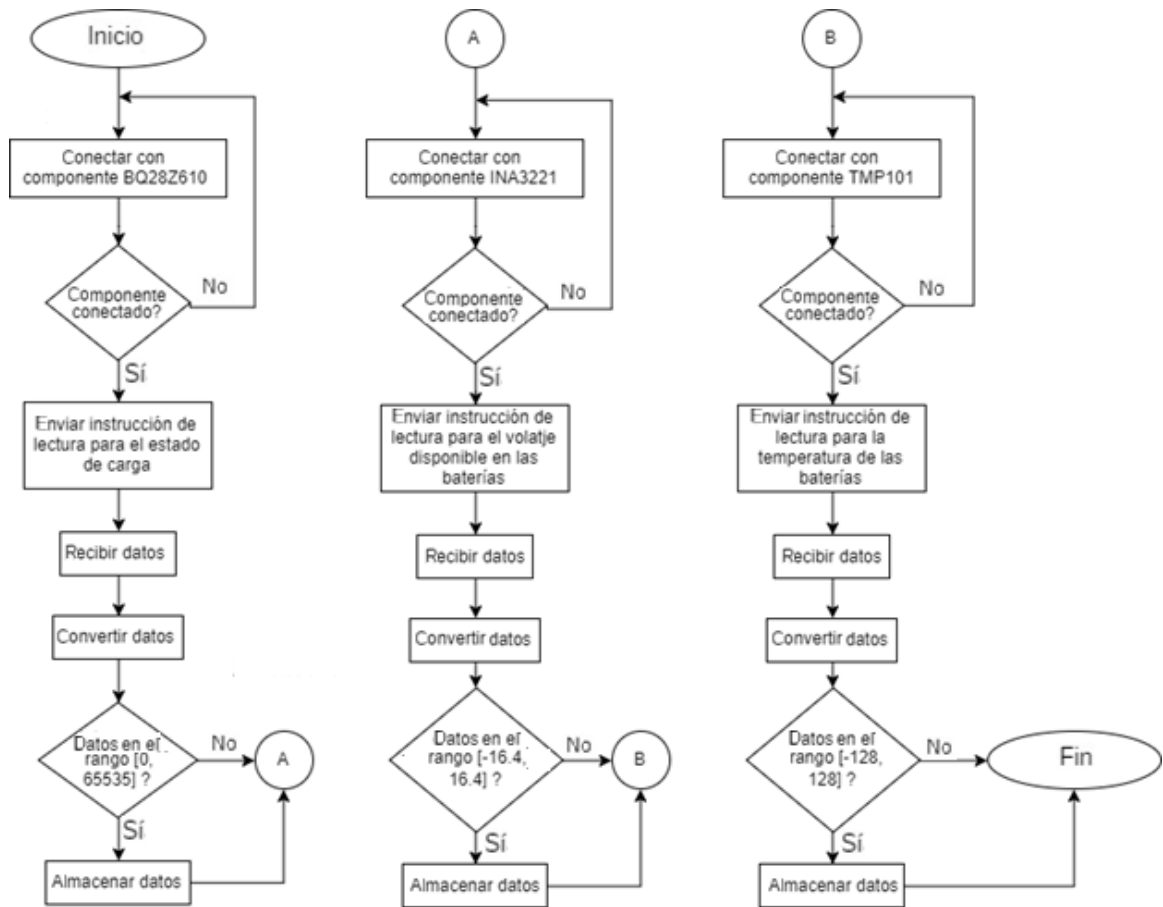
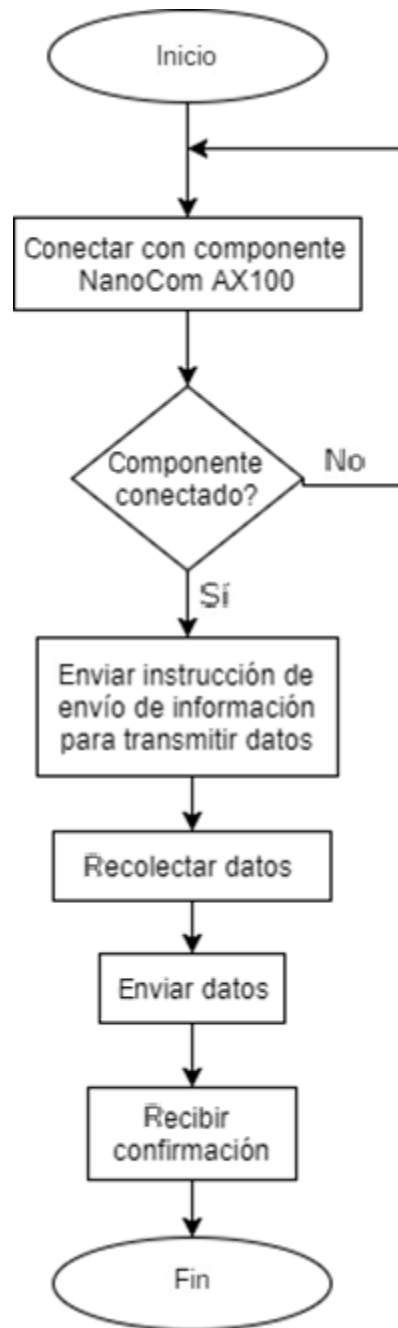


Figura 46. Diagrama de flujo para operación del módulo de comunicaciones.



c. Diagramas de secuencia

Figura 47. Diagrama de secuencia para la obtención de datos del ADCS sobre el magnetómetro.

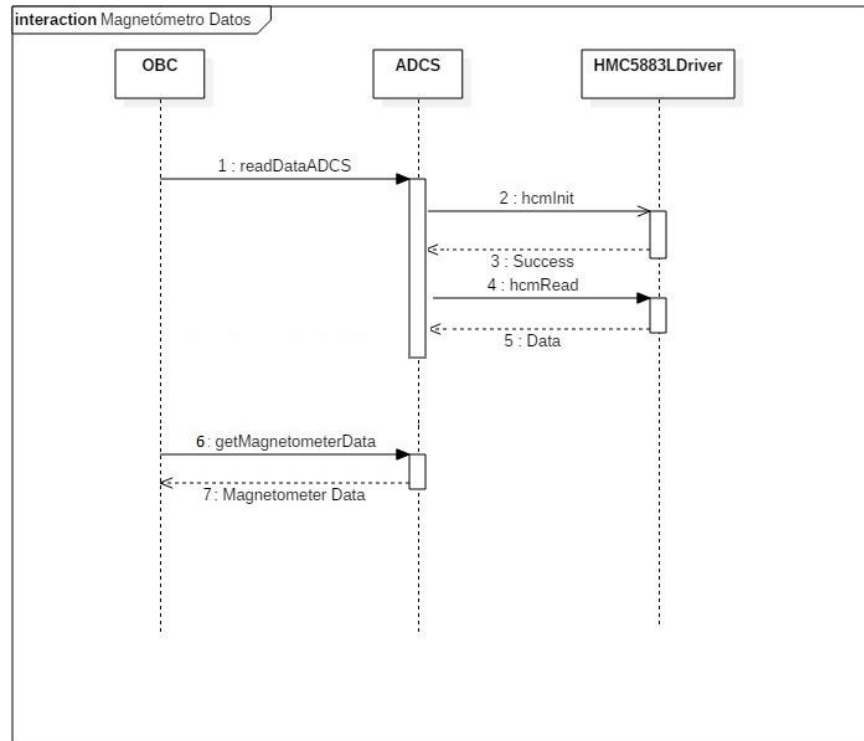


Figura 48. Diagrama de secuencia para la obtención de datos del ADCS sobre los fotodiodos.

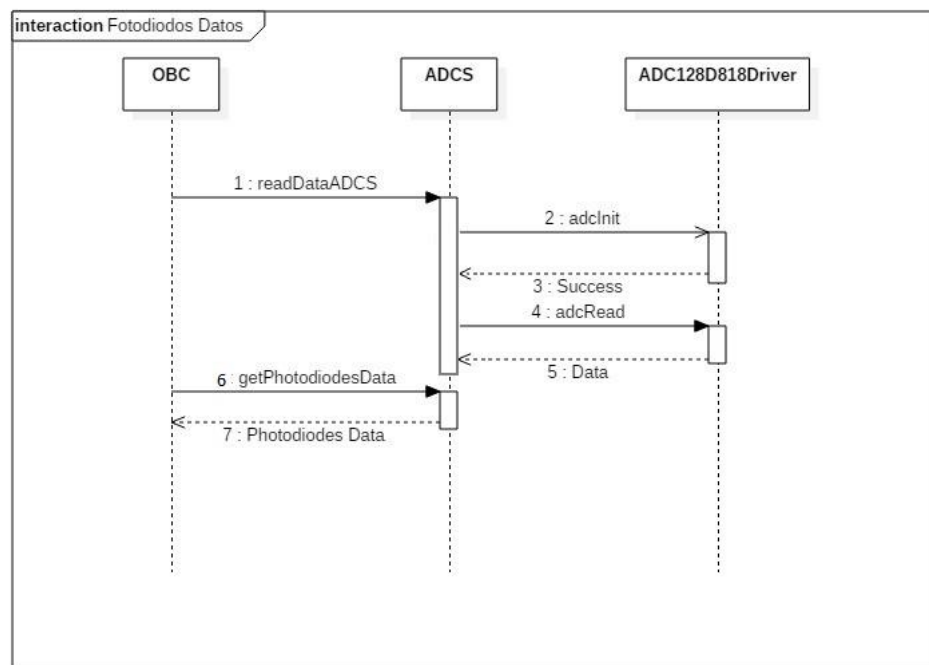


Figura 49. Diagrama de secuencia para la configuración y lectura de datos para el magnetómetro.

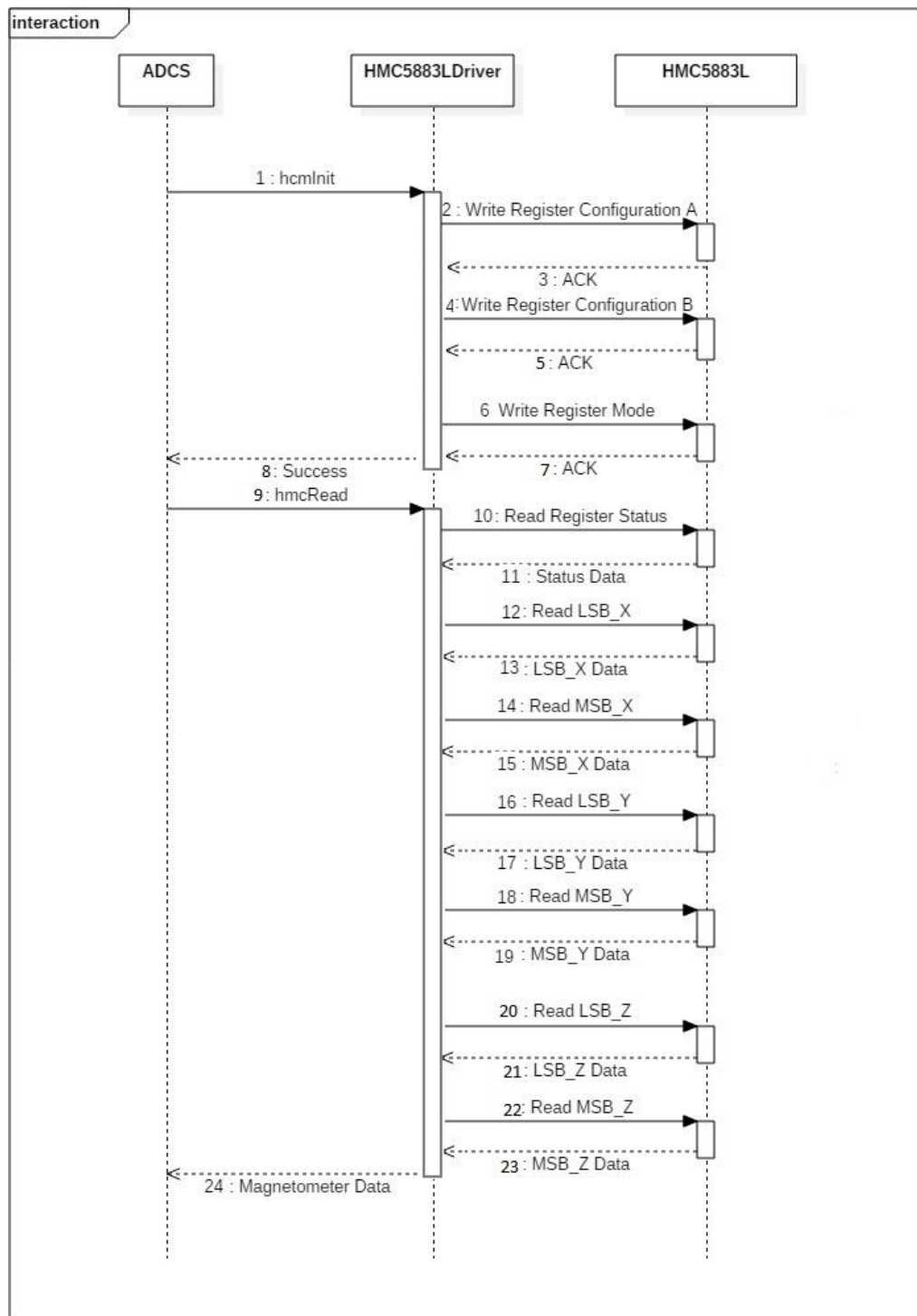


Figura 50. Diagrama de secuencia para la configuración y lectura de datos para el convertidor analógico a digital del cual proviene la información de los fotodiodos.

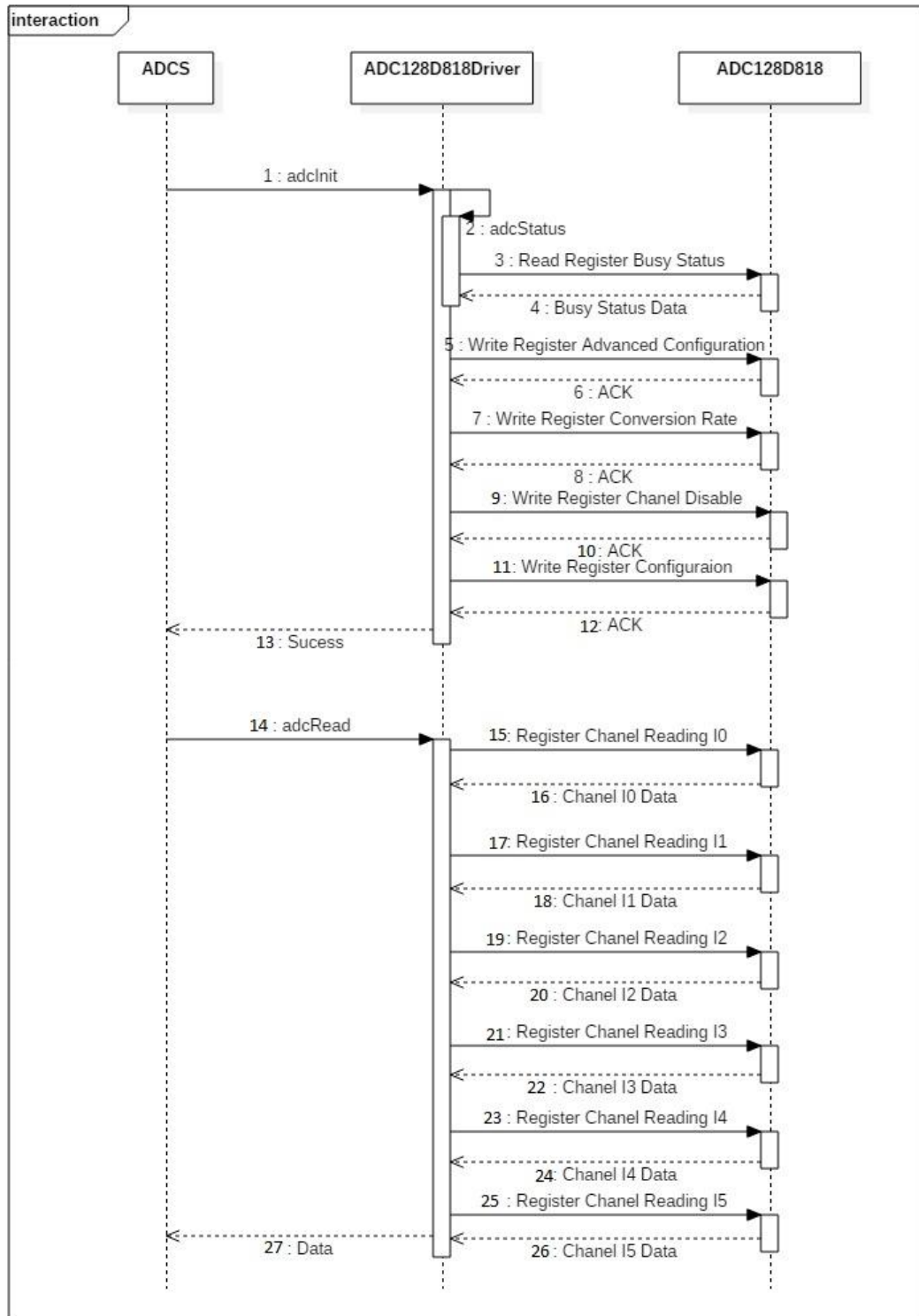


Figura 51. Diagrama de secuencia para el movimiento del carrusel de filtros.

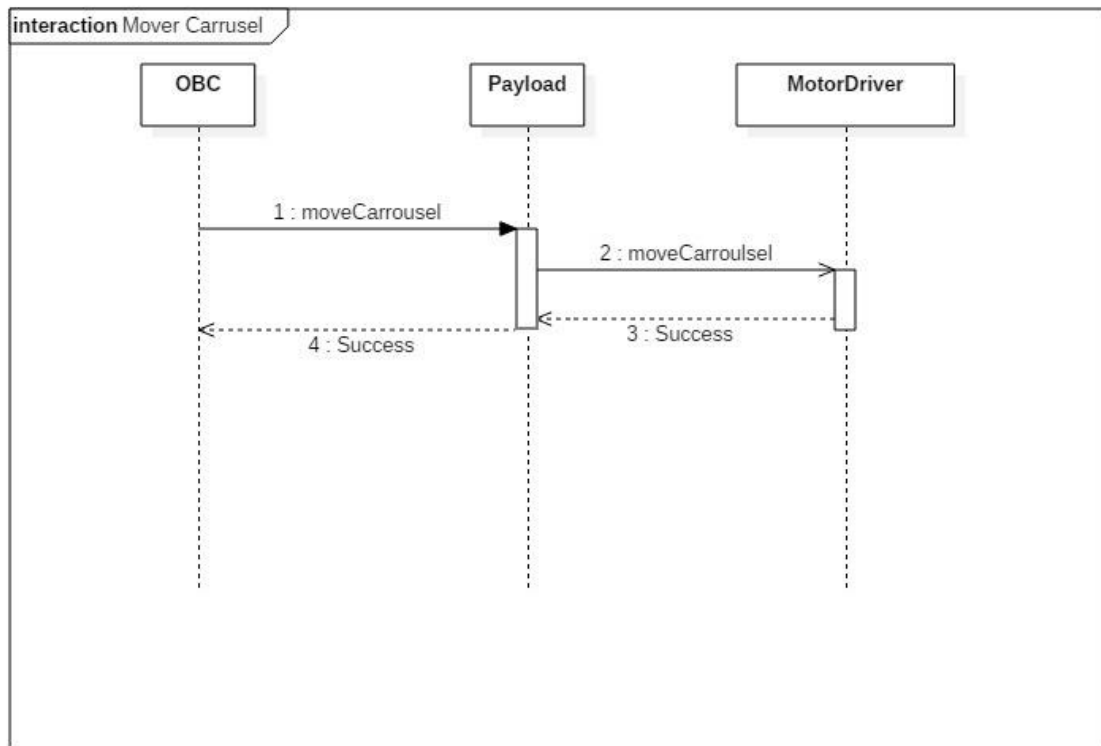


Figura 52. Diagrama de secuencia para la toma de fotografías.

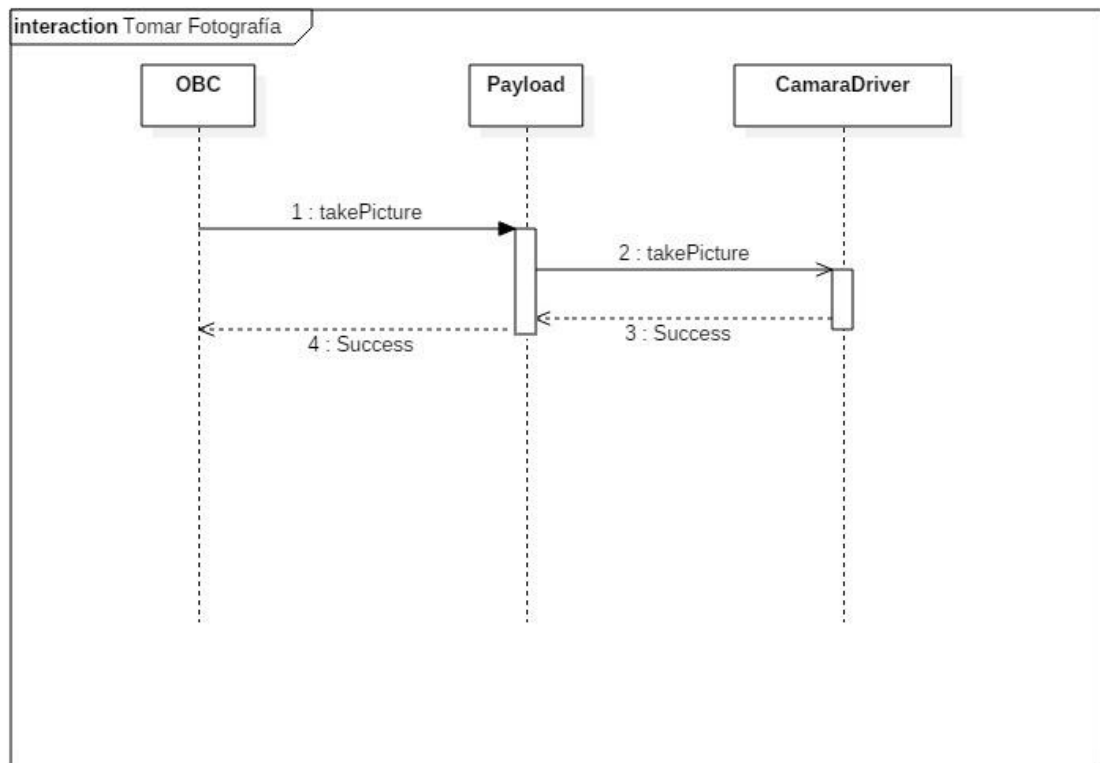


Figura 53. Diagrama de secuencia para obtener la temperatura de las baterías.

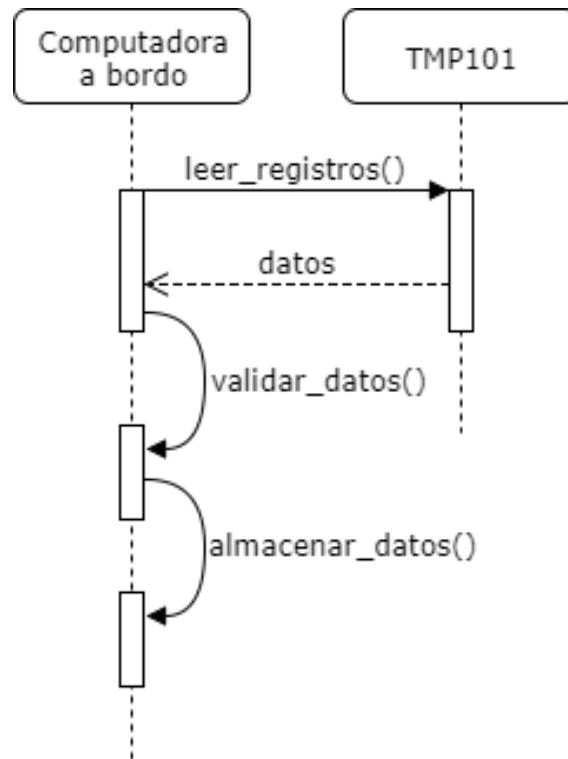


Figura 54. Diagrama de secuencia para obtener el estado de carga de las baterías.

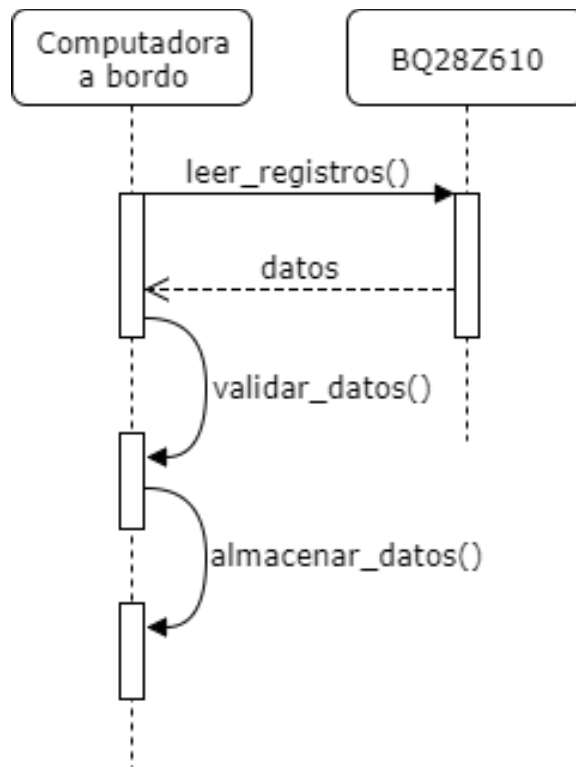


Figura 55. Diagrama de secuencia para obtener el voltaje de las baterías.

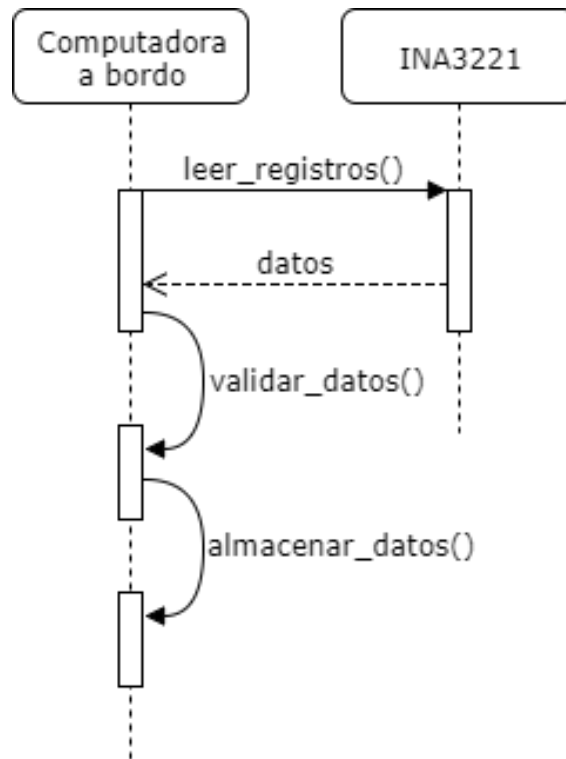
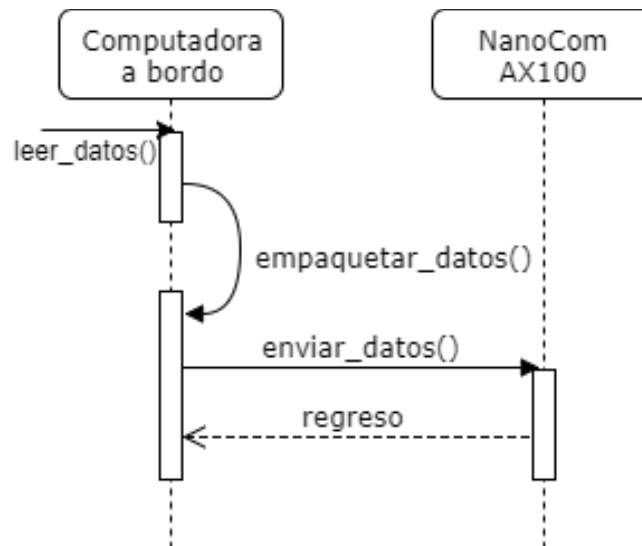


Figura 56. Diagrama de secuencia para transferir datos al módulo de comunicaciones.



d. Diagramas de actividad

Figura 57. Diagrama de actividad para obtener la temperatura de las baterías.

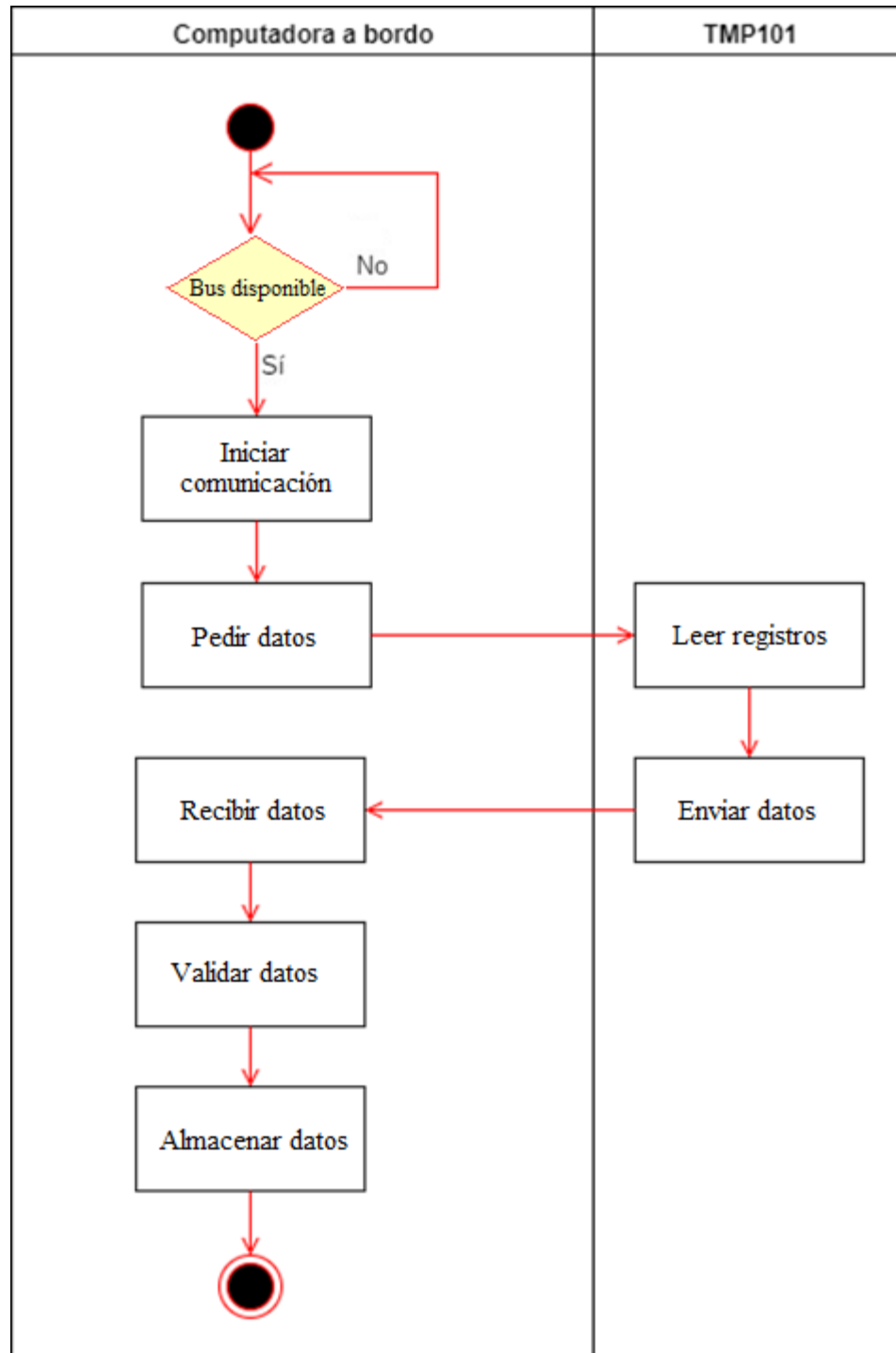


Figura 58. Diagrama de actividad para obtener el estado de carga de las baterías.

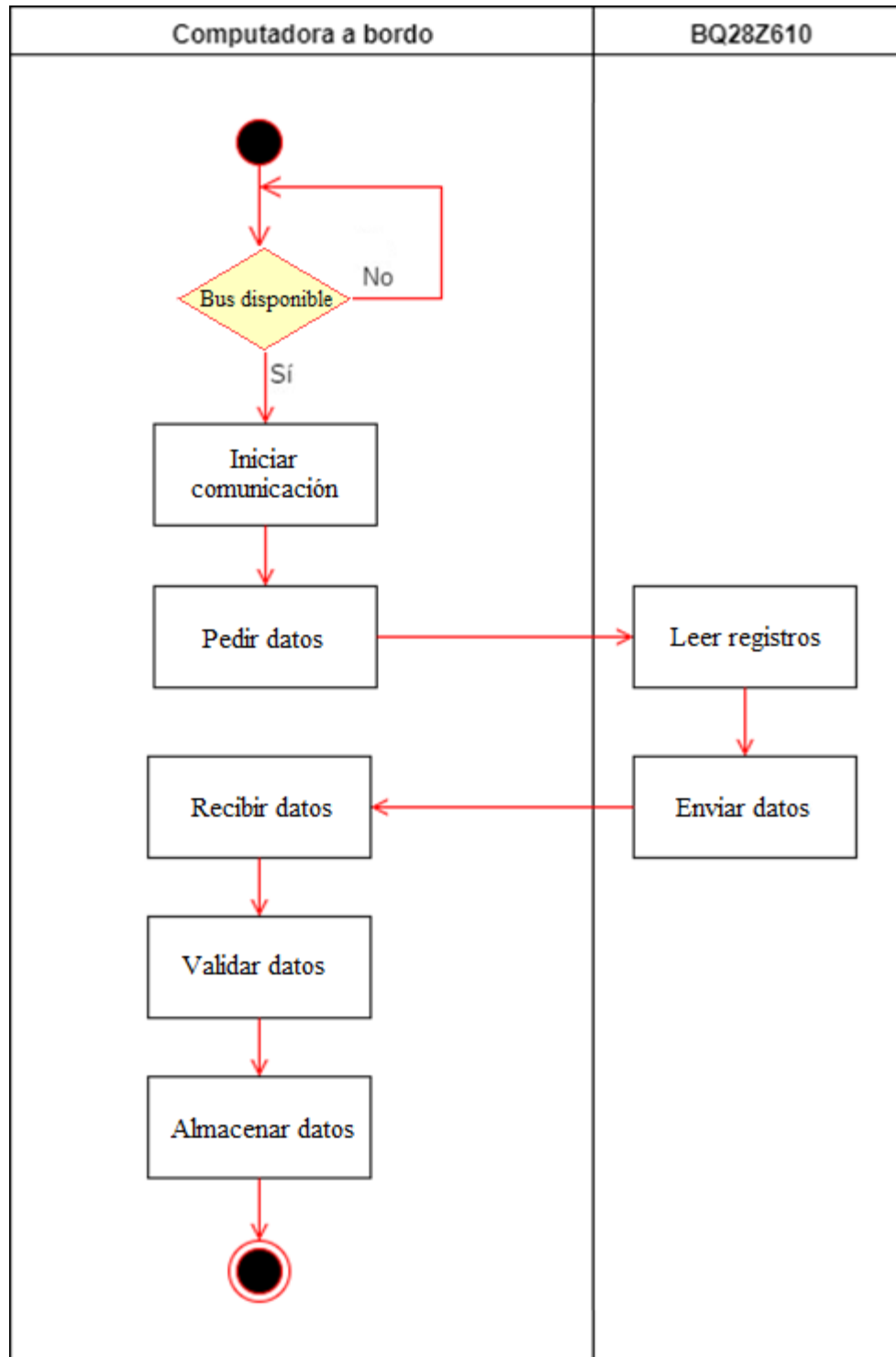


Figura 59. Diagrama de actividad para obtener el voltaje de las baterías.

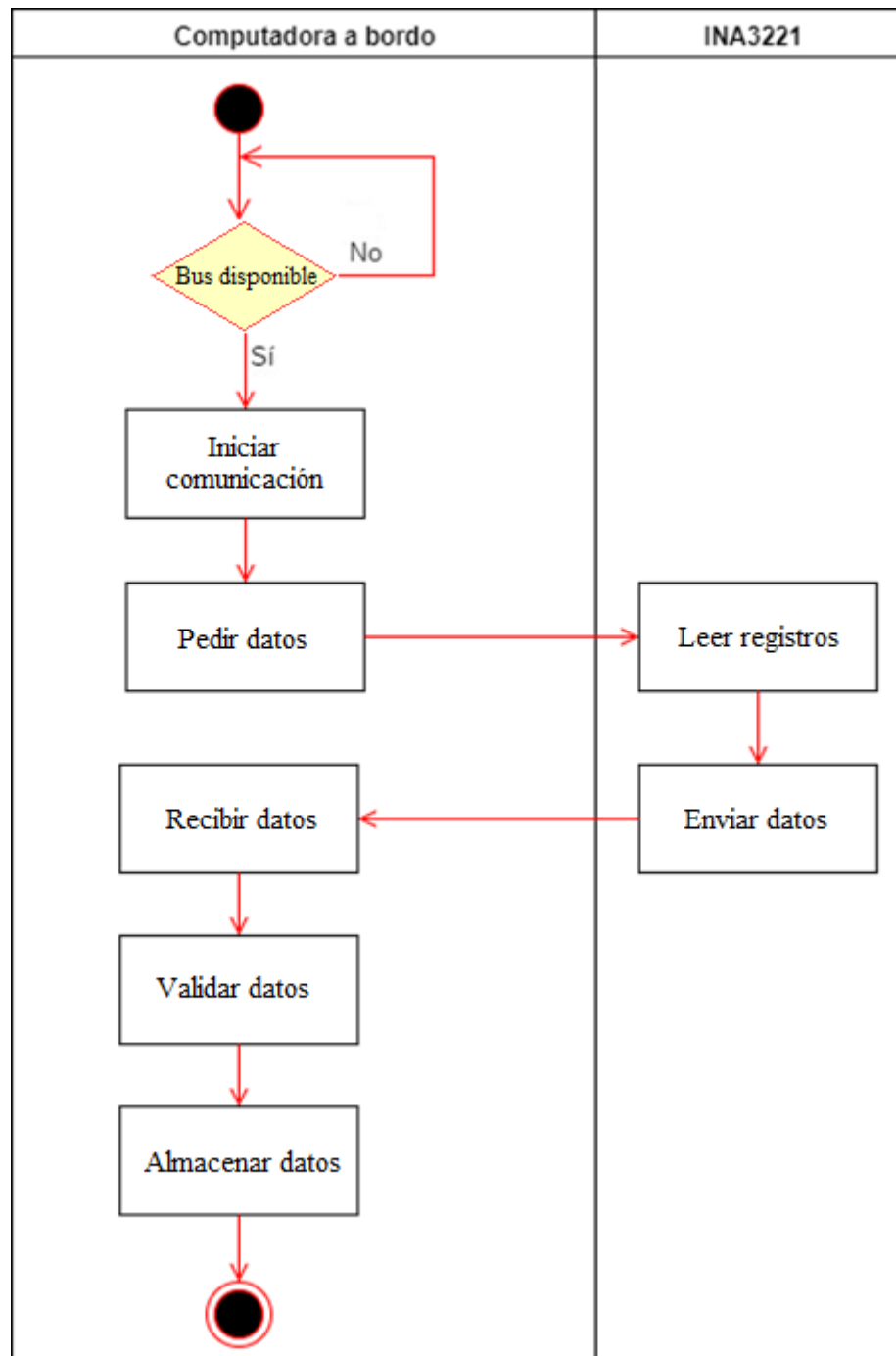
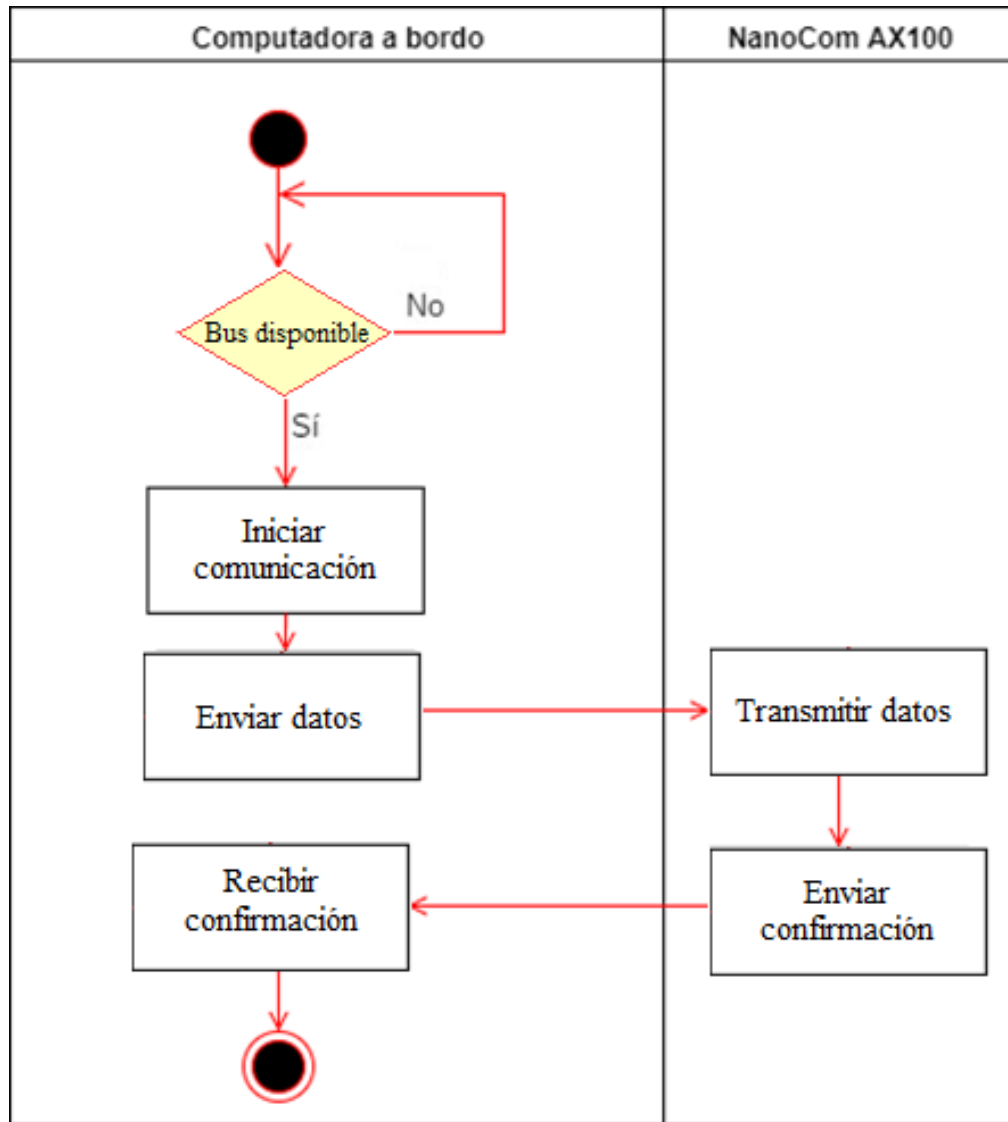


Figura 60. Diagrama de actividad para transferir datos al módulo de comunicaciones



e. Protocolo de comunicación. Es importante definir el conjunto de instrucciones utilizables en los componentes, estas son enviadas por medio del bus de datos I2C a la dirección específica del dispositivo, estas instrucciones son utilizadas para configurar los dispositivos y hacer la lectura de los datos. A continuación, se presentan la lista para los componentes de ADCS que son los que requieren este tipo de comunicación:

1) Magnetómetro HMC5883L

Cuadro 70. Lista de registros para magnetómetro HMC5883L.

Dirección	Nombre	Acceso
00	Configuration Register A	Leer/escribir
01	Configuration Register B	Leer/escribir
02	Mode Register	Leer/escribir
03	Data Output X MSB Register	Leer
04	Data Output X LSB Register	Leer
05	Data Output Z MSB Register	Leer
06	Data Output Z LSB Register	Leer
07	Data Output Y MSB Register	Leer
08	Data Output Y LSB Register	Leer
09	Status Register	Leer
10	Identification Register A	Leer
11	Identification Register B	Leer
12	Identification Register C	Leer

De igual manera, se describe la especificación de cada uno de los bits para los registros de configuración necesarios para inicializar el dispositivo, estos registros determinan el modo de funcionamiento del dispositivo.

Cuadro 71. Especificación del registro de configuración A del chip HMC5883L.

Configuration Register A								
Posición de bit en la secuencia	7	6	5	4	3	2	1	0
Función	Reservado	Define el número de muestras por salida en una medición.	Define la velocidad de salida de los datos.			Definen el flujo de medición del dispositivo.		

Cuadro 72. Especificación del registro de configuración B del chip HMC5883L.

Configuration Register B								
Posición de bit en la secuencia	7	6	5	4	3	2	1	0
Función	Definen la ganancia para el dispositivo, para todos los canales.			Estos bits deben ser autorizados para el correcto funcionamiento del dispositivo.				

Cuadro 73. Especificación del registro de modo del chip HMC5883L.

Mode Register								
Posición de bit en la secuencia	7	6	5	4	3	2	1	0
Función	Estos pines establecen I2C en alta velocidad, 3400kHz.					Definen el modo de operación del dispositivo.		

Luego de verificar las posibles configuraciones para el chip HMC5883L se estableció la configuración siguiente:

- Configuración de medición: se estableció la normal la cual es valor por defecto, en donde la medición del dispositivo sigue el flujo de medición normal, también existe la polarización positiva y la polarización negativa.
- Promedio de muestreo: se definió un muestro de ocho por medición, lo cual asegura una mayor exactitud en la medición de los valores.
- Velocidad de datos: establecida en 15 Hz es la configuración por defecto, es utilizada cuando el magnetómetro está en modo operación continua, pero en este caso que se utilizará una sola medición no toma mucha relevancia.
- Ganancia: se estableció la ganancia por defecto, la ganancia estable el rango del campo magnético reconocido por el sensor, esta configuración se establece más por experimentación por lo cual en este momento se establece por defecto.
- Modo de operación: de una sola medición se estableció esta configuración debido a que el CubeSat no estará únicamente leyendo datos del magnetómetro, lo hará solo cuando sea necesario, además esta configuración disminuye el consumo de potencia.

2) Convertidor analógico a digital ADC128D818

Cuadro 74. Lista de registros para el convertidor analógico a digital ADC128D818.

Dirección	Nombre	Acceso
00h	Configuration Register	Leer/escribir
01h	Interrupt Status Register	Leer
03h	Interrupt Mask Register	Leer/escribir
07h	Conversion Rate Register	Leer/escribir
08h	Channel Disable Register	Leer/escribir
09h	One-Shot Register	Escribir
0Ah	Deep Shutdown Register	Leer/escribir
0Bh	Advanced Configuration Register	Leer/escribir
0Ch	Busy Status Register	Leer
20h – 27h	Chanel Readings Registers	Leer
2Ah – 39h	Limit Register	Leer/escribir
3Eh	Manufacturer ID Register	Leer
3Fh	Revision ID Register	Leer

También, se describe la especificación de cada uno de los bits para los registros de configuración necesarios para inicializar el dispositivo, estos registros determinan el modo de funcionamiento del dispositivo.

Cuadro 75. Especificación del registro de configuración avanzada del chip ADC128D818.

Advanced Configuration Register								
Posición de bit en la secuencia	7	6	5	4	3	2	1	0
Función	Reservados			Define el modo de operación.			Define el voltaje de referencia a utilizar.	

Cuadro 76. Especificación del registro de tasa de conversión del chip ADC128D818.

Conversion Rate Register								
Posición de bit en la secuencia	7	6	5	4	3	2	1	0
Función	Reservados						Controla el modo de conversión del chip.	

Cuadro 77. Especificación del registro deshabilitar canales del chip ADC128D818.

Channel Disable Register								
Posición de bit en la secuencia	7	6	5	4	3	2	1	0
Función	Deshabilitar Temperatura			Deshabilitar IN#				

Cuadro 78. Especificación del registro de configuración del chip ADC128D818.

Configuration Register								
Posición de bit en la secuencia	7	6	5	4	3	2	1	0
Función	Restaura a los valores por defecto.	Reservado		Limpia la salida de interrupción INT.	Reservado		Activa la interrupción de la salida INT.	Activa el inicio de las operaciones

Luego de verificar las posibles configuraciones para el chip ADC128D818 se estableció la configuración siguiente:

- Modo de operación: este chip define cuatro modos de operación, se estableció el modo 0, en este modo se establecen ocho canales de salida, los cuales funcionan para perfectamente para las salidas de los fotodiodos.
- Voltaje de referencia: se definió un voltaje de referencia interno establecido por defecto, aunque también puede configurarse para un voltaje de referencia externo.
- Tasa de conversión: se estableció la tasa de conversión low power, como su nombre lo indica se utiliza para ahorrar energía estableciendo una sola conversión y no una conversión continua.

- Deshabilitar canales: como son 6 canales de salida para los fotodiodos, se utilizarán los canales de I0 a I5, por lo tanto, fueron deshabilitados los canales I6 y I7.

3. **Redundancia.** Una de las principales limitaciones de CubeSat 1U es el espacio disponible, esto limita la aplicación de redundancia de sistemas, el módulo de manejo de comandos y datos es un módulo fundamental dentro del funcionamiento del CubeSat, podría decirse que es el cerebro del satélite controlando y manejando la información de los demás módulos; debido a lo cual es razonable considerar redundancia para un módulo tan importante.

La incorporación de redundancia para el módulo de manejo de comandos y datos considera tener dos computadoras a bordo, con posibilidad de tener redundancia similar o redundancia disímil. Una redundancia similar implica tener dos computadoras del mismo modelo. La redundancia disímil implica tener dos computadoras de diferentes modelos. La redundancia disímil tiene una mayor tolerancia a fallas en comparación con redundancia similar, pero también tiene serias desventajas para nuestro CubeSat. Las desventajas son: los otros componentes del subsistema también deben tener interfaces compatibles con los dos modelos de computadora, esto reduce mucho las computadoras que se pueden seleccionar. También tomaría el doble del tiempo su implementación, debido a que se debe realizar desarrollo de software para otra computadora.

Aunque la redundancia disímil representa mayores ventajas, en este proyecto se tiene limitaciones por espacio y el tiempo de desarrollo por lo cual tomando en consideración los requerimientos de la misión la mejor opción es la redundancia similar. Ahora bien, la redundancia similar presentamos dos opciones: redundancia directa o redundancia indirecta. La diferencia entre estas dos opciones radica en la forma en que las computadoras intercambian su estado. Con la redundancia directa, no hay interfaz de comunicación que funcione como intermediario. Pero la redundancia indirecta necesita un componente extra que funcione como intermediario entre las computadoras. El enfoque directo tiene la ventaja de que requiere menos espacio, pero requiere más potencia para su funcionamiento, mientras que la indirecta da la ventaja de requerir menos potencia, pero incrementa la complejidad, se requiere de una serie de test para prevenir errores en el componente extra y requiere más espacio. Debido a las ventajas del enfoque directo y las dificultades que representa el enfoque indirecto, se definió que la mejor opción era utilizar un enfoque directo.

a. **Diseño.** El diseño de la redundancia similar directa para el módulo de manejo de comandos y datos se compone de dos OBCs iguales, las cuales compartirán el bus de datos I2C que controla la comunicación con los demás módulos, un bus de datos por medio del estándar UART mantendrá la comunicación entre las dos computadoras.

Figura 61. Conexión UART.

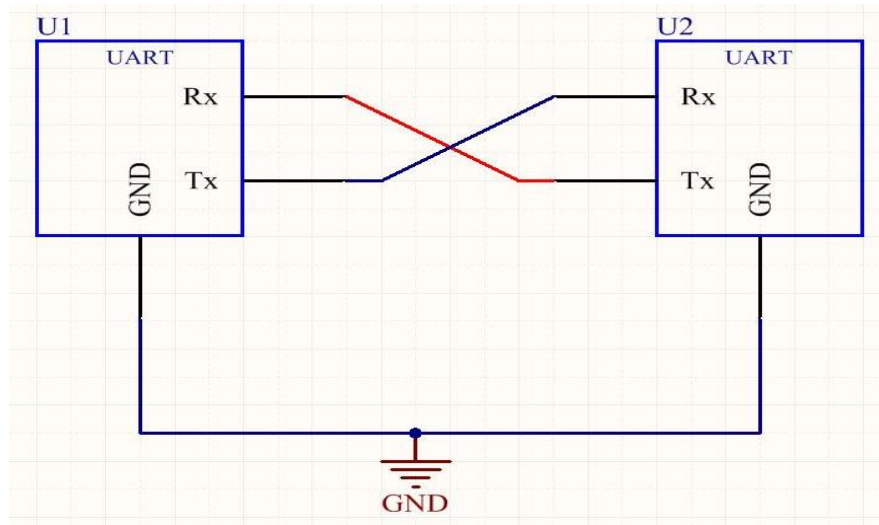
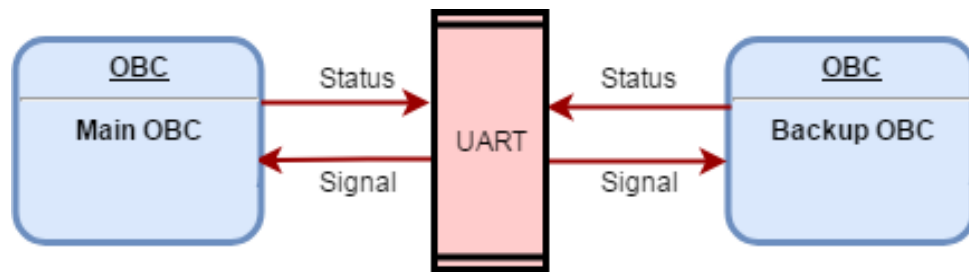


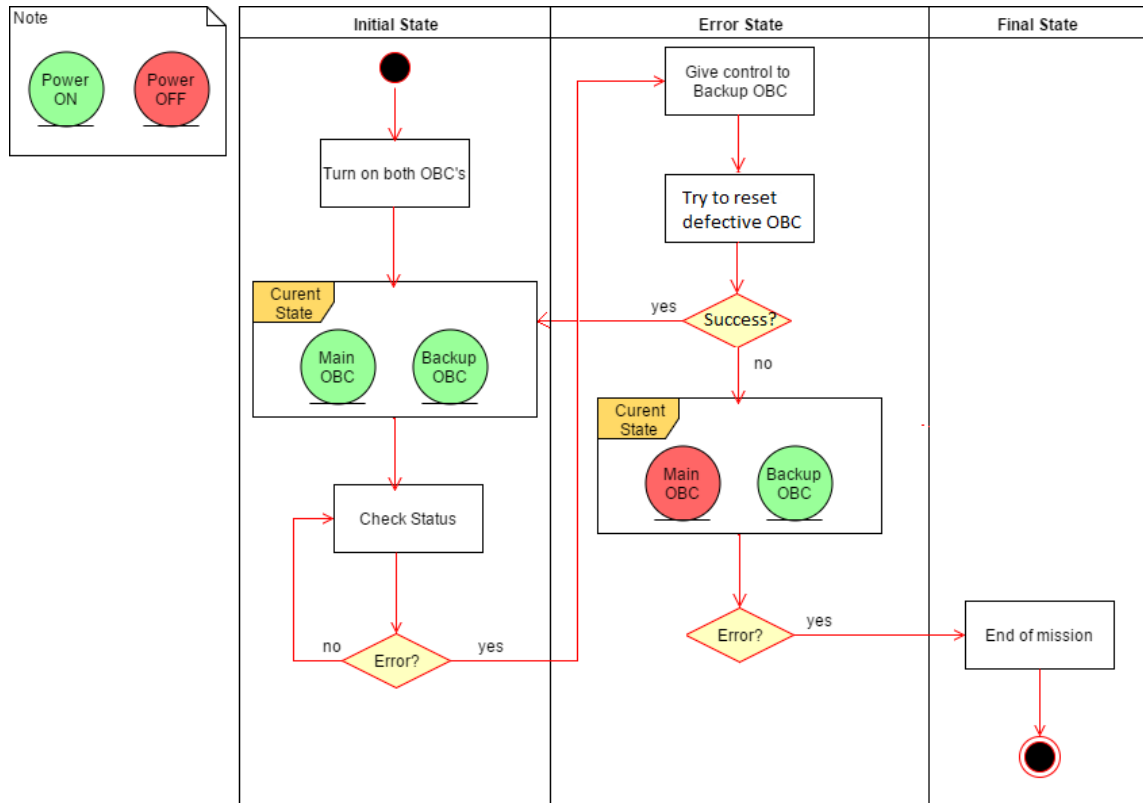
Figura 62. Interacción de redundancia.



Esto implica que ambas computadoras estarán encendidas en un estado inicial de CubeSat, debido a que ambas deben tener la capacidad de tomar el control. El cambio de control entre las computadoras está determinado por una señal constante transmitida por medio del bus de datos UART, esta señal únicamente ayuda a identificar si ha existido un problema con alguna OBC; en caso de que la señal de la OBC principal no llegue a la computadora de apoyo, la computadora de apoyo tomará el control y tratará de reestablecer a la otra computadora, funciona de igual manera en el caso contrario. Es importante mencionar que solo una computadora estará controlando los demás módulos, a pesar de que ambas están encendidas.

La siguiente figura explica de mejor manera la interacción entre las dos computadoras, la revisión de estado se realizará por medio de la comunicación UART y el error dependerá si el tiempo de respuesta ha sido sobrepasado.

Figura 63. Estados de la redundancia.



b. Implicaciones. Las implicaciones de aplicar el diseño de redundancia se listan por modulo a continuación:

1. Estructura:

- Como se necesitan dos computadores esto implica la utilización de más espacio y masa para el sistema. También se debe tomar en cuenta el espacio para las conexiones adicionales.

2. Potencia:

- Implica un mayor consumo de energía eléctrica. La computadora de respaldo debe configurarse en modo de bajo consumo de energía en el estado inicial y cambiar al modo de consumo estándar si el CubeSat pasa al estado de error en la computadora principal.
- Incorporar las dos computadoras en el diseño de los modelos electrónicos para suministrar energéticamente a ambas, además, de las conexiones adicionales requeridas para UART.

3. Carga útil:

- El modelo actual de redundancia permite a todos los dispositivos incorporados en el bus de datos ser controlados por ambas computadoras, para cualquier dispositivo que quede fuera de él, se necesitará tener un componente adicional.

c. Desarrollo. La comunicación entre las computadoras requiere la programación de las funciones de conexión, envío de señales y cambio de estado. A demás se requiere la programación para detectar el estado de la computadora (principal o respaldo) y funcionar de acuerdo con el mismo. Se utilizó C como lenguaje de programación para ambas computadoras.

d. Implementación. Desde el inicio de la computadora un hilo de ejecución es asignado para activar el funcionamiento del método de redundancia, para esto se agregó el comando de ejecución en el archivo `/etc/rc.local`. Se agregaron dos leds externos a las placas de las computadoras para indicar el estado de estas, cuando el led está encendido indica que la computadora correspondiente actualmente está funcionando como principal y tiene el control del bus de datos.

e. Pruebas. Se realizaron pruebas de cuatro tipos: unitarias, integradoras, respuesta a errores introducidos y de tiempo prolongado.

Las pruebas unitarias se realizaron sobre cada función, para esto se tenía una salida o funcionamiento esperado por cada posible estado de cada función, las funciones que se probaron son: conexión por medio de UART, enviar señal, leer señal, encender led, apagar led, activar bus de datos I2C, determinar estado y cambiar de estado.

Para las pruebas integradoras se probó el funcionamiento del sistema de redundancia al utilizar todas las funciones en los dos estados de la computadora. Las pruebas de errores introducidos utilizaron el funcionamiento de las pruebas integradoras con la distinción de tener errores a propósito que prueban los siguientes escenarios inusuales al correcto funcionamiento: una computadora deja de funcionar totalmente, los buses de datos se desconectan, los buses de datos no responden, ambas computadoras funcionan como principal y ambas computadoras funcionan como respaldo.

Las pruebas de tiempo prolongado se realizaron en periodos de 24 horas, para estas se desarrolló un programa que reiniciaba las computadoras en un tiempo aleatorio entre 5 y 30 minutos. Al reiniciarse aleatoriamente las computadoras cambiaban de estado y la principal tomaba el control del bus de datos para continuar con la captura de datos.

S. DESARROLLO E IMPLEMENTACIÓN DEL SISTEMA DE MANEJO DE COMANDOS Y DATOS

Luego de realizar el análisis y con la arquitectura de software establecida se prosiguió con el desarrollo del código, se escogió un lenguaje de programación que se acoplará a las necesidades presentes, definidas por los requerimientos, se decidió que se utilizaría C como lenguaje de programación.

La decisión de utilizar C se genera a partir de las ventajas que nos ofrece y de las necesidades que el proyecto requiere. C posee un núcleo de lenguaje muy simple, con la ventaja de poder incorporar funcionalidades complementarias como una librería para realizar la comunicación I2C, de igual manera ofrece acceso a memoria de bajo nivel permitiendo la generación espacios de memoria para el almacenamiento de datos, de igual manera C es compatible con un gran número de microcontroladores por lo cual si existe algún cambio en la computadora a bordo el código generado será igual utilizable.

Se utilizó la librería Wiring Pi la cual es una biblioteca de acceso GPIO escrita para C, esta librería es extensible y compatible con varios protocolos de comunicación; I2C está contemplado dentro de la librería Wiring Pi por lo cual se decidió utilizar para la comunicación con los componentes que lo necesiten. A partir de esto se generaron controladores para cada componente, con el fin de poder ser manejados fácilmente por medio de un programa central que administra la realización de acciones, como la inicialización de los componentes o la toma de datos.

T. PRUEBAS DE MANEJO DE COMANDOS Y DATOS

Para realizar el testeo de este módulo se diseñaron distintas pruebas definidas a continuación:

1. Componentes

a. Pruebas de consumo energético. Debido a que la computadora seleccionada operará bajo condiciones limitadas de suministro energético, es esencial reducir al máximo posible el consumo energético. La computadora no tiene un mecanismo predefinido para trabajar en modo “ahorrador” de energía o suspender su actividad temporalmente, por lo tanto, se requiere realizar configuraciones en el sistema operativo para deshabilitar funciones que reduzcan el consumo energético sin afectar las capacidades necesarias de operación.

Se investigó que con las siguientes opciones se reduce el consumo energético de la computadora:

- Instalar un sistema operativo ligero que contenga solo paquetes necesarios para el funcionamiento, sin interfaz gráfica.

- Deshabilitar interfaces periféricas que no serán utilizadas (HDMI, USB).
- Deshabilitar leds de indicaciones.

Los comandos, configuraciones y modificaciones al sistema operativo que se realizaron para reducir el consumo de energía son las siguientes:

- El sistema operativo: raspbian-jessie-lite
- Para deshabilitar el puerto HDMI: `/usr/bin/tvservice -o`. Este comando se agregó al archivo `/etc/rc.local` para realizar esta configuración cada vez que la computadora inicia.
- Para deshabilitar los leds de indicaciones:
 - `echo none | sudo tee /sys/class/leds/led0/trigger`
 - `echo 1 | sudo tee /sys/class/leds/led0/brightness`
- El led deshabilitado está titulado en la placa como 'ACT'. Para que esta configuración sea permanente se agregó también los siguientes comandos al archivo `/boot/config.txt`
 - `dtparam=act_led_trigger=none`
 - `dtparam=act_led_activelow=on`

Para calcular el consumo de energía se midió la corriente que consume la computadora y el voltaje de alimentación. Para medir la corriente se utilizó un cable USB, el cual fue cortado para intervenir el cable que transporta el voltaje y realizar la medición con un multímetro, como se observa en la siguiente figura. La medición de voltaje se realizó con otro multímetro.

Figura 64. Método para medir corriente eléctrica.



b. Pruebas de interferencias electromagnéticas. La computadora seleccionada no tiene antecedentes de operar en el espacio, por lo tanto, se investigó las interferencias propias del ambiente del espacio que afectan el funcionamiento de equipos electrónicos. Los pulsos electromagnéticos crean interferencias en la operación de los componentes y afectan de distintas maneras el funcionamiento. Se investigó la forma de generar interferencia electromagnética para afectar la Raspberry Pi y comprobar lo que esta provoca en su funcionamiento, además de los métodos para medir y observar esta interferencia.

El circuito que genera los pulsos eléctricos básicamente está compuesto de un oscilador electrónico, un transformador y un multiplicador de voltaje; es utilizado en aplicaciones comunes como: máquinas de electro shock y raquetas electrónicas para matar insectos. El campo magnético por el cual los pulsos se transfieren se compone de una bobina de 1.64 metros de cobre. Las siguientes figuras muestran el circuito que genera los pulsos eléctricos y el mecanismo completo.

Figura 65. Circuito que genera pulsos eléctricos.

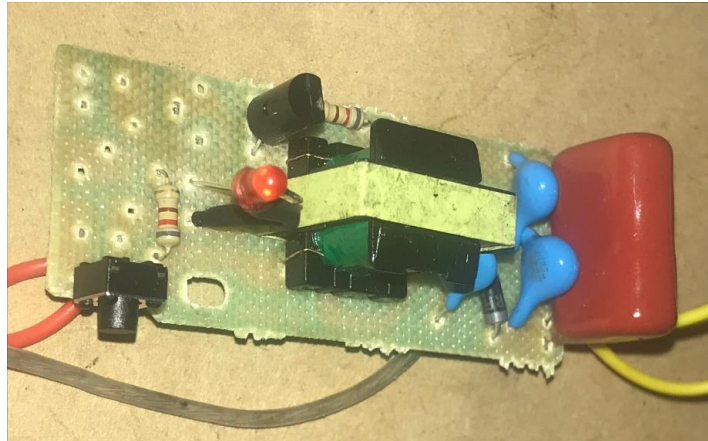
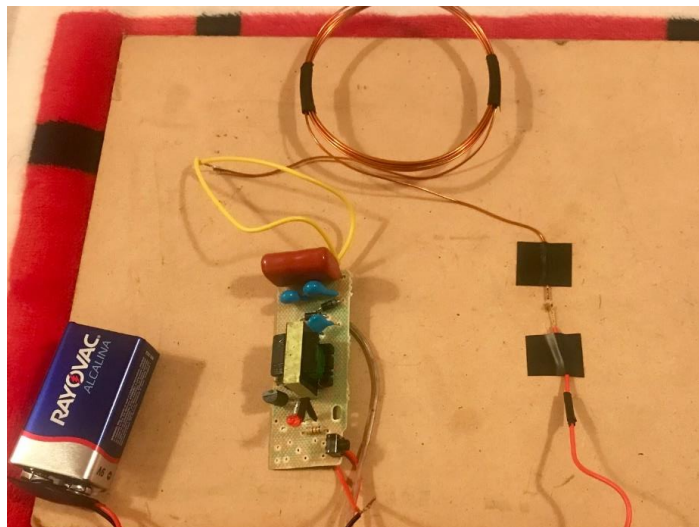


Figura 66. Mecanismo para generar interferencia electromagnética.



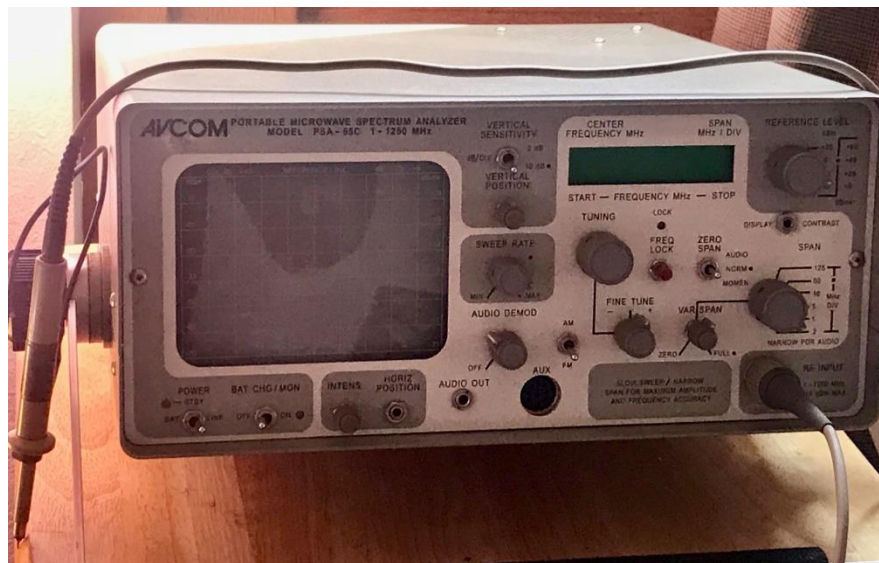
Para medir la frecuencia en la que se pueden encontrar los pulsos se utilizó un frecuencímetro, este se muestra en la siguiente figura.

Figura 67. Frecuencímetro.



La siguiente figura muestra el analizador de espectro que se utilizó para analizar los pulsos producidos por el mecanismo desarrollado.

Figura 68. Analizador de espectro.



2. Funcionales. Se realizaron pruebas unitarias para cada componente con el fin de verificar su correcto funcionamiento, para las pruebas de ADCS, Energía y Comunicaciones se utilizó un Arduino Uno para cada módulo; el Arduino Uno es una placa de circuito impreso con un microcontrolador basado en el

chip ATmega323P, posee 14 pines digitales para entrada/salida, 6 entradas analógicas, etc. Por su costo y facilidad es utilizada en proyectos que involucran sensores y actuadores.

- Cámara: se hicieron pruebas con el módulo de cámara de la Raspberry debido a que aún no hay cámara seleccionada, se verificó que se fuera capaz de tomar una fotografía y de almacenarla.
- Motor de Carrusel: para el carrusel se hicieron pruebas con un servomotor MG90S el cual se mueva por medio de señales PWM igual que el motor piezoeléctrico que se está evaluando, se verificó que el motor pudiera moverse hacia las 3 direcciones distintas donde estarán los filtros.
- HMC5883L Driver: se utilizó un Arduino Uno para simular el funcionamiento del magnetómetro, por medio de él se verificó que la secuencia de pasos de configuración y para adquirir datos estuvieran correctas.
- ADC128D818 Driver: se utilizó un Arduino Uno para simular el funcionamiento de las señales de los fotodiodos que llegan al ADC, por medio de él se verificó que la secuencia de pasos de configuración y para adquirir datos estuvieran correctas.
- Energía: se utilizó un Arduino Uno para simular el funcionamiento de los componentes del módulo de energía, estos responden a una instrucción específica (simulación de lectura de sus registros) y generan valores aleatorios entre el rango de los permitidos por sus hojas de especificaciones. (Texas Instruments, 2016)
- Comunicaciones: se utilizó un Arduino Uno para simular el funcionamiento del componente del módulo de comunicaciones, este responde a una instrucción específica (simulación de activación de escritura) y recibe los datos a transmitir, no se tuvo acceso la hoja de datos técnicos del componente seleccionado.

3. Tiempo de ejecución de acciones. Otro aspecto importante resaltado por los requerimientos son los tiempos de ejecución de cada una de las operaciones. Por esta razón se evaluó para cada una de las actividades críticas: mover carrusel, tomar fotografía, tomar datos de magnetómetro y de fotodiodos los tiempos de repuesta es decir en cuanto tiempo eran realizadas las operaciones con el fin de determinar si se cumplían los requerimientos o no.

4. Estrés. Un factor a tomar en cuenta es que el bus de datos debe estar transmitiendo constantemente y debido a la criticidad del proyecto se debe asegurar que está funcione correctamente, para asegurar su funcionamiento se simuló la recepción de datos para al menos un periodo orbital la cual se realizará aproximadamente cada 90 minutos, tomando datos de los componentes cada 2 minutos, aunque la frecuencia de datos no se ha definido en el proyecto es importante verificar que el bus de datos tenga la capacidad de transmitir constantemente. Además, estas pruebas incluían la distribución de comandos de acción de manera continua (25 veces), para la captura de fotos y para el movimiento del carrusel.

Para la prueba del módulo de energía se realizó una simulación del funcionamiento durante periodos de 24 horas, la temperatura de las baterías es un valor que debe monitorearse con mayor regularidad, por lo tanto, para este componente se realizaron mediciones cada minuto, para los otros dos componentes cada dos minutos. Para la prueba del módulo de comunicaciones se realizaron simulaciones de envío de distintos tipos de información.

U. PARÁMETROS SIMULACIÓN DEL MÓDULO DE CONTROL ACTIVO

1. **Momentos de inercia.** El primer paso de la elaboración del proyecto fue investigar y calcular los parámetros necesarios para llevar a cabo las diferentes simulaciones. Debido a que se está llevando a cabo la elaboración de un CubeSat 1U en la universidad, se utilizaron los parámetros de inercia de cada uno de sus ejes para basarse en los valores reales que tiene uno de estos satélites.

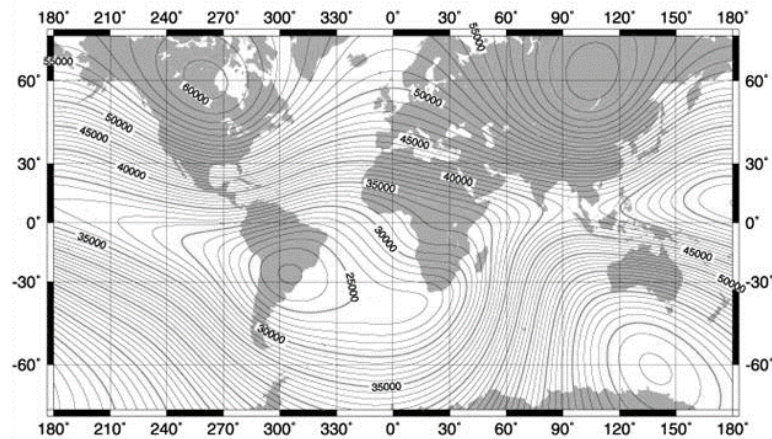
Cuadro 79. Inercias de los ejes principales del CubeSat.

Eje principal	Inercia (Kg*mm ²)
x	0.002136
y	0.000935
z	0.001996

Estos valores se obtuvieron a través de cálculos elaborados en software, son datos con los componentes que se tienen hasta el momento. Por lo tanto, en un futuro cercano pueden tener modificaciones y ser distintos a los finales. Además de los valores de inercia, también se calcularon las magnitudes de las diferentes perturbaciones en el espacio.

2. **Torque magnético.** El torque magnético se expresó como una ecuación vectorial (43), pero que tenía un modelo simplificado (44). Se hicieron algunas asunciones para facilitar los cálculos, especialmente en los grados de inclinación de la órbita. El valor que se eligió fue de 55° de latitud, es decir que está muy cercano al polo norte del planeta, esto se ve ilustrado en la Figura 69. El hecho que el satélite se encuentre en ese momento orbitando esa latitud significa que su constante de latitud magnética es fuerte por ser cercano a uno de los polos. (Zagorski, s.f.)

Figura 69. Mapa de latitudes y longitudes de la Tierra con separación de grados.



(Appinsys, 2010)

A partir de esta asunción se recopilaron los datos obteniendo el siguiente cuadro.

Cuadro 80. Parámetros para calcular la magnitud de la perturbación de torque magnético.

Parámetro	Símbolo	Valor	Unidad
Campo magnético de la Tierra	M	7.80E+15	T * m ³
Distancia satélite y centro de la Tierra	R	7,078,000.00	m
Dipolo residual	D	0.01	A*m ²
Inclinación orbital	--	90	°
Constante de latitud magnética	?	2	--

3. Torque gravitacional. El torque gravitacional se expresó como una ecuación vectorial (37), pero que tenía un modelo simplificado (38). Debido a que la órbita LEO, en la cual se lanzan los CubeSats 1U es una órbita baja, alrededor de 200-2000km, se hizo la asunción que el satélite estaría alrededor de 700km sobre la superficie de la tierra. Obteniendo así el valor presentado en el Cuadro 81. Es importante resaltar que esta perturbación puede variar dependiendo de la altura del satélite, no obstante, mientras permanezca en la misma altura no varía. (Zagorski, s.f.)

Cuadro 81. Parámetros para calcular la magnitud de la perturbación de torque gravitacional.

Parámetro	Símbolo	Valor	Unidad
Constante gravitacional de la tierra	μ	3.99E+14	m ³ /s ²
Distancia satélite y centro de la tierra	R	7,078,000.00	m
Inercia eje Y	I _y	0.000935	kg*mm ²
Inercia eje Z	I _z	0.001996	kg*mm ²

4. Torque aerodinámico. El torque aerodinámico se expresó como una ecuación vectorial (41), pero que tenía un modelo simplificado (42). El supuesto de este torque fue el área de contacto, se tomó como si toda una cara del satélite se viera afectada por los efectos aerodinámicos de restos de atmósfera. Los valores de radio del torque se eligieron de esta manera debido a que, al haber un buen diseño de inercias, el centro de masa del satélite debería de estar en el centro del satélite. Lo que sería 5 cm del centro de masa hacia la parte exterior del CubeSat. En el caso de los valores de densidad atmosférica, coeficiente de arrastre y velocidad de órbita, son valores estándar para la órbita y satélite. (Starin, 2010) (Westfall, 2015)

Cuadro 82. Parámetros para calcular la magnitud de la perturbación de torque aerodinámica.

Parámetro	Símbolo	Valor	Unidad
Densidad atmosférica	ρ	1.00E-13	kg/ m ³
Coeficiente de arrastre	C _d	2.00	--
Área de contacto	A _r	0.01	m ²
Velocidad de órbita	V	7504	m/s
Radio del torque	c _{pa} - cm	0.05	m

5. Torque de presión por radiación solar. El torque aerodinámico se expresó como una ecuación vectorial (39), pero que tenía un modelo simplificado (40). El supuesto de este torque fue el área de contacto, se tomó como si toda una cara del satélite se viera afectada por los efectos de la radiación solar. También se hizo el supuesto de que por los diferentes materiales que está hecho el CubeSat, habría una cantidad de radiación que se absorbería.

Cuadro 83. Parámetros para calcular la magnitud de la perturbación de torque de presión por radiación solar.

Parámetro	Símbolo	Valor	Unidad
Constante solar ajustada	Φ	1.37E+03	W/ m ²
Velocidad de la luz	c	300,000,000.00	m/s
Área iluminada	A _s	0.01	m ²
Factor de reflexión	q	1	--
Radio del torque	cpa - cm	0.05	m

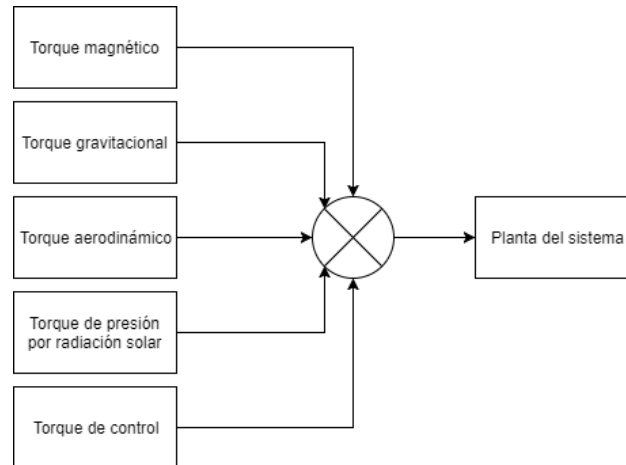
(Starin, 2010)

6. Resumen de magnitudes. Al obtener los valores adecuados, se desarrollaron las fórmulas de torques y se obtuvieron las siguientes magnitudes para las perturbaciones. Se logra observar que la mayoría de estas perturbaciones dieron resultados de magnitudes bajas. Esto es principalmente porque de hecho son valores que se ven afectados por la atmósfera de la tierra y la distancia entre objetos. El torque magnético presenta ser la perturbación con mayor magnitud por el hecho de que la tierra posee un campo magnético fuerte que la protege de perturbaciones espaciales y radiación solar. Un aspecto importante de señalar es el hecho que solo el torque magnético es cíclico, es decir no afecta de la misma manera en distintos puntos de la órbita, su magnitud reduce e incrementar. Con respecto a las demás perturbaciones, estas son más constantes por el tipo de trayectoria que tiene el satélite.

Cuadro 84. Resumen de magnitudes de torques de las perturbaciones estudiadas.

Torque	Símbolo	Magnitud calculada (N*m)	Tipo de perturbación
Presión por radiación solar	T _s	4.55667E-09	Secular
Magnético	T _m	4.39939E-07	Cíclico
Aerodinámico	T _a	2.8155E-09	Secular
Gravitacional	T _g	2.03632E-09	Secular

Figura 70. Diagrama de ingreso de perturbaciones a la planta del sistema



V. CONTROLADORES

Se seleccionaron dos tipos de controladores para controlar el satélite. Esta selección se hizo a través de la facilidad de implementar ambos controladores sin la necesidad de implementar una gran cantidad de cálculos. El primer controlador es el PID, este se escogió debido a la capacidad de ajustar los parámetros a las necesidades del sistema sin tener que cambiar directamente la localidad de los polos.

Posteriormente también se decidió por incluir el regulador lineal cuadrático como otro posible controlador para el satélite. No obstante, se tuvo que hacer la linealización del modelo no lineal para obtener las matrices A y B del sistema. Para esto se utilizó el siguiente punto de equilibrio.

$$x_o = [0 \ 0 \ 0 \ 1 \ 0 \ 0]^T$$

Se escogió este punto de equilibrio para corroborar que el sistema dejó de dar vueltas con las velocidades angulares siendo 0, y también que se logre obtener la orientación deseada debido a que los cuaterniones lleguen a 0 excepto por η que debe tener una magnitud de 1. El siguiente paso fue obtener la linealización a través de derivar parcialmente el modelo e implementar el punto de equilibrio dentro de la función. A través de esto se obtuvo el siguiente sistema linealizado.

$$A = \begin{bmatrix} \mathbf{0}_3 & \frac{1}{2}I_3 \\ \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix}, \quad X = \begin{bmatrix} q \\ \omega \end{bmatrix}, \quad B = \begin{bmatrix} \mathbf{0}_3 \\ J_{sat}^{-1} \end{bmatrix}$$

(70)

Al obtener este sistema se debía asegurar que podía ser controlable, por lo que se llevó a cabo la prueba de controlabilidad del sistema que consistía en formar la matriz Γ y obtener su Rank. Este sistema se convirtió en uno 6x6 debido a que el coeficiente η de los cuaterniones es dependiente de los otros

coeficientes lo cual lo hace que no sea controlable y no permita hacerle un control lineal cuadrático al sistema. Por esta razón se obvió en la linealización ese parámetro y se obtuvo una función de grado 6. La matriz Γ se expresa de la siguiente manera:

$$\Gamma = [B \ AB \ A^2B \ A^3B \ A^4B \ A^5B] \quad (71)$$

Al desarrollar la función se encontró que su Rank era de 6 por lo que el sistema es completamente controlable y se desarrollaría con los siguientes parámetros para las matrices Q y R.

$$Q = I_6$$

$$R = I_3$$

En caso fuera necesario penalizar el sistema para obtener reacciones de torques que se puedan alcanzar con ruedas de inercia convencionales, se ajustarán los parámetros de las velocidades angulares en la matriz Q.

W. SIMULACIÓN EN MATLAB DEL MÓDULO DE CONTROL ACTIVO

Para la simulación del sistema se hizo a través de MATLAB, para esto se utilizó el modelo no lineal que describe el sistema.

$$\begin{bmatrix} \dot{q} \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} \underline{\Omega}(\omega) \mathbf{q} \\ \underline{I}_{sat}^{-1} [\underline{S}(\omega) (\underline{I}_{sat} \omega + \mathbf{T}_{ctrl} + \mathbf{T}_{dist})] \end{bmatrix} \quad (72)$$

Además, se hicieron distintos documentos para manejar las diferentes funciones que necesitaba la simulación. Los documentos se dividieron de la siguiente manera:

- CubeSat.m – Es el programa principal que se encarga de unir todos los módulos a tomar en cuenta, estos incluyen las perturbaciones, el sistema modelado, cuaternión de error, control LQR, ingreso de datos del satélite, ingreso de datos de la órbita y presenta los resultados de forma gráfica.
- CubeSatPID.m – Es el mismo programa que CubeSat, no obstante, este introduce el control PID.
- Threeaxis_pid.m – Esta función permite el ingreso de los coeficientes del PID, también permite llevar a cabo los cálculos necesarios para incluir este tipo de control.
- Modelado_del_satelite.m – Es la función que toma en cuenta el modelado del sistema no lineal y también toma en cuenta el método de Runge Kutta grado 4 para resolverlo.
- Perturbacion_aerodinamica.m – Es la función que introduce la perturbación aerodinámica del sistema y obtiene parámetros desde el programa principal.

- `Perturbacion_gravitacional.m` – Es la función que introduce la perturbación gravitacional del sistema y obtiene parámetros desde el programa principal.
- `Perturbacion_magnetica.m` – Es la función que introduce la perturbación magnética del sistema y obtiene parámetros desde el programa principal.
- `Perturbacion_solar.m` – Es la función que introduce la perturbación por radiación solar del sistema y obtiene parámetros desde el programa principal.
- `Skew.m` – introduce la matriz skew utilizada que es una matriz 3x3.
- `Skew2.m` – introduce la matriz skew, la cual es la matriz 4x4.
- `Draw_cube.m` – es una función utilizada para poder graficar el cubo que representa al satélite en la simulación elaborada.

X. INTERFAZ DE PRUEBAS FÍSICAS

El funcionamiento del control en el mundo real es de vital importancia para el satélite. A través de una plataforma bien desarrollada es posible comprobar cómo se comportaría el sistema en el espacio. Para esto se diseñó una plataforma de una sola cara que estuviera sujeta por hilos para evitar que el sistema tenga que combatir contra la fuerza de la gravedad. A través de esta se necesita observar como rechaza las perturbaciones el sistema y como converge su velocidad angular a una igual a 0 rad/s. Se necesitaban cinco componentes básicos para esta aplicación:

- Driver puente H
- IMU, unidad de mediciones inerciales
- Microcontrolador Arduino Uno
- Motor 12V DC
- Estructura

Al tener claras las necesidades del sistema se escogieron los siguientes componentes para suplir estas necesidades.

1. **Puente H.** La necesidad del puente H es para tener una forma de alimentar el motor de 12V y controlarlo con un microcontrolador que trabaje con 5V. Una opción que es fácil de adquirir en el mercado es un driver L298N. Esta es una opción que facilita la conexión del motor debido a sus terminales de tornillo y ofrece una conexión sencilla a la fuente de alimentación y de control. Esto presentaba una facilidad a pesar de su gran tamaño, no obstante, este no iría dentro de la estructura sino en una caja aparte junto con la alimentación y microcontrolador.

Figura 71. Módulo puente H L298N.



Cuadro 85. Resumen de conexiones en el módulo L298N.

Pin L298N	Arduino/Fuente de alimentación/Motor
+12V poder	12V fuente
GND	GND fuente
IN1	Pin 2 Arduino
IN2	Pin 3 Arduino
GND	GND Arduino
Output A1	Terminal 1 motor
Output A2	Terminal 2 motor

2. Unidad de mediciones inerciales. La unidad de mediciones inerciales o IMU es un sensor indispensable en este módulo debido a que sin este sensor no hay forma fácil de encontrar diferentes parámetros como los cuaterniones, ángulos, velocidades angulares y aceleraciones del sistema. Debido a que estos valores deben mínimas perturbaciones es necesario obtener una que sea lo suficientemente robusta para filtrar la señal y dar valores reales de lectura. Por esto se hizo la elección de una IMU con referencia absoluta por lo que se obtendrán valores con respecto a un punto específico lo cual ayuda a mejorar el rendimiento del sistema. Debido a esto se escogió la IMU BNO055 de Adafruit.

Figura 72. Módulo de mediciones inerciales BNO055.



Cuadro 86. Resumen de conexiones en el módulo BNO055.

Pin BNO055	Arduino
Vin	5V
GND	GND
SCL	A5
SDA	A4

3. **Microcontrolador.** El microcontrolador se escogió por la facilidad de obtener un Arduino UNO hoy en día. Además, al funcionar este sistema como un microcontrolador como este significa que la carga computacional y de muestreo no debe ser tan rápido por lo significa que no se tiene que sobrecargar el microcontrolador para poder obtener la respuesta deseada.

Figura 73. Microcontrolador Arduino UNO.



4. **Motor DC.** Este motor se escogió debido a una serie de requerimientos. El primero era que fuera capaz de dar más de 5000 RPM, también que tuviera un eje lo suficientemente extenso para poder colocar la rueda de inercia y cualquier otro adaptador que se necesitara para asegurar la posición de la rueda. El tercer requerimiento era que en la base del eje hubiera un adaptador que permitiera ajustarlo a una estructura sin que el eje pudiera moverse y cambiar el lugar donde estuviera dando el giro la rueda. Debido a esto se encontró un motor Jameco DC de 12V de 740mA y 9820RPM.

Figura 74. Motor DC 12V.



5. Estructura. Para desarrollar la estructura se hizo un diseño básico que consistiera en la menor cantidad de piezas y que se pudiera cortar en material MDF de 3.2mm. Para esto se tomaron medidas de la IMU y del motor para poder encajarlos de la manera correcta y también unir estas piezas con tuercas y tornillos M4.

Figura 75. Sujetador del motor.

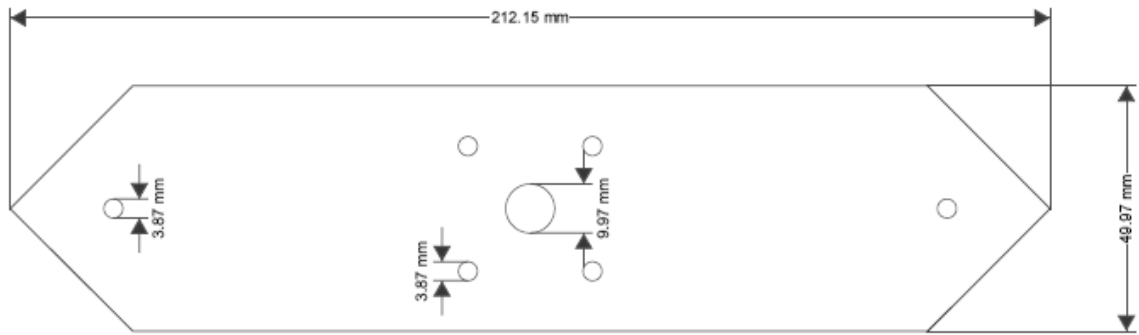


Figura 76. Estructura principal sujetador de pieza para sujeción de motor y de la IMU.

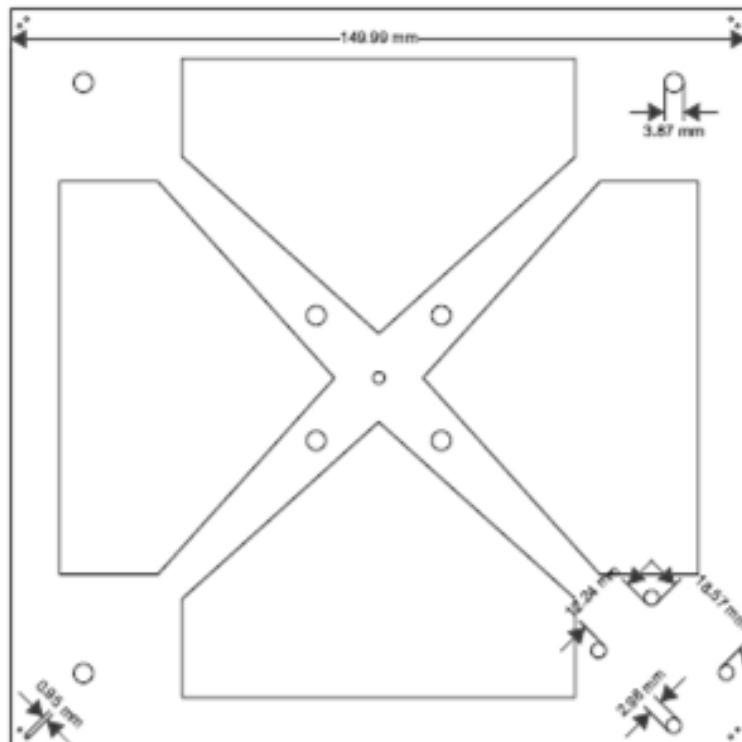


Figura 77. Estructura para sujeción del cuerpo del motor.

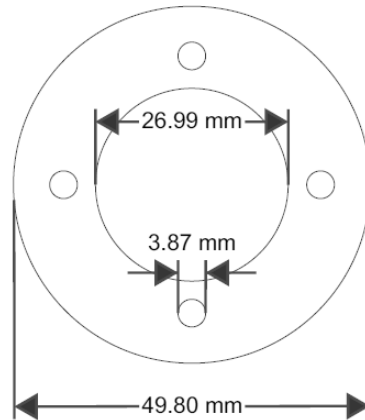
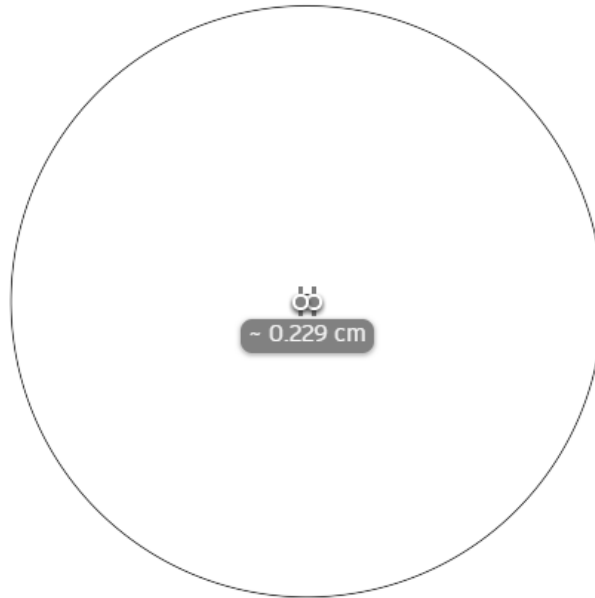


Figura 78. Rueda de inercia de 10cm de diámetro y de grosor 6.2mm.



VI. ANÁLISIS DE RESULTADOS

A. RIESGOS DEL PROYECTO

A continuación, se detallan los distintos riesgos identificados durante el diseño de este proyecto. En cuanto a la mitigación de cada riesgo, se utilizará una letra para describir el método utilizado. Se utilizará la “M” de Mitigación para indicar que se tratará de eliminar/reducir las causas y consecuencias de un riesgo; la “C” de Control para indicar que las causas y consecuencias tratarán de controlarse; la “A” para señalar que el riesgo simplemente será aceptado; y la “E” de Estudiar para indicar que se necesita más información sobre el riesgo antes de poder seleccionar una técnica de mitigación adecuada. En algunos casos se escogió más de una técnica de mitigación por riesgo. Los riesgos del módulo de Potencia serán presentados con más detalles más adelante.

A cada riesgo se le dio una identificación, la cual está formada por las primeras letras del nombre en inglés del módulo al que pertenecen y un número. A los riesgos del sistema en general se les dio las letras SYS; a los de estructura, ST; a los de control térmico; TH, a los de manejo de comandos y datos, CDH; a los de carga útil, PLD; a los de control y determinación de orientación, ACS; a los de comunicación, COMM; y a los de gestión del proyecto, PRG.

Cuadro 87. Riesgos del proyecto.

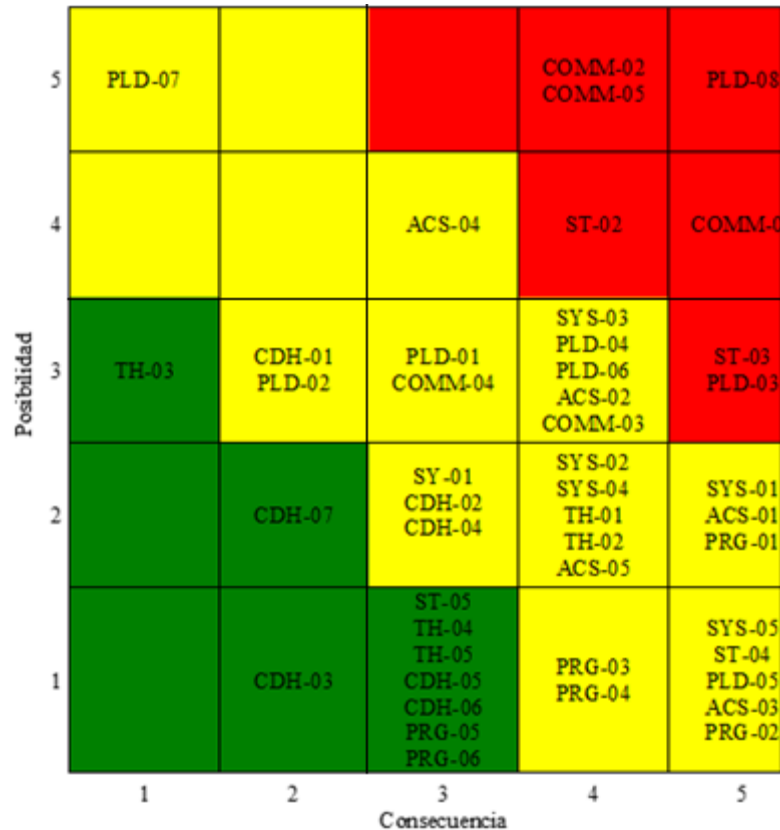
Código	Riesgo	Probabilidad	Consecuencia	Mitigación
SYS-01	El satélite no cumple con los requerimientos del vehículo de lanzamiento.	2	5	M, E
SYS-02	Los componentes no están propiamente acomodados dentro del satélite.	2	4	M, E
SYS-03	Los componentes no pueden ser integrados entre sí.	3	4	M
SYS-04	La estructura o alguno de los componentes resulta con daños.	2	4	M, A
SYS-05	Falla del vehículo de lanzamiento.	1	5	A
ST-01	Daños en la estructura antes del lanzamiento.	2	3	M
ST-02	Los componentes no permanecen fijados en su posición.	4	4	M
ST-03	La estructura no soporta las cargas durante el lanzamiento.	3	5	E, M
ST-04	La estructura no cumple con los estándares necesarios.	1	5	E, M
ST-05	Colisión con partículas en órbita.	1	3	A
TH-01	El método de control térmico no provee una protección adecuada durante el lanzamiento.	2	4	M
TH-02	El método de control térmico no provee una protección adecuada para los componentes en el espacio.	2	4	M

Continuación Cuadro 87

Código	Riesgo	Probabilidad	Consecuencia	Mitigación
TH-03	El método de control térmico es dañado antes del lanzamiento.	3	1	M
TH-04	El método de control térmico es colocado incorrectamente o dañado durante la colocación.	1	3	M
TH-05	El método de control térmico es dañado por colisión de partículas en el espacio.	1	3	A
CDH-01	Un error de programación lleva a ciclo infinito o a un error fatal.	3	2	M
CDH-02	El programa se detiene esperando la respuesta de una sub-rutina.	2	3	M
CDH-03	Error del sistema operativo o falta de respuesta.	1	2	M, A
CDH-04	Error de los protocolos de comunicación.	2	3	M, E
CDH-05	Fallas en la memoria.	1	3	M
CDH-06	Daño físico.	1	3	M
CDH-07	Pérdida de datos.	2	2	E, M
PLD-01	El carrusel se detiene en una posición incorrecta.	3	3	M, E
PLD-02	Las fotografías tomadas no están enfocadas o muestran parte del satélite.	3	2	M, E
PLD-03	Daño al sensor de la cámara debido a exposición directa de luz solar.	3	5	M, A
PLD-04	Daño al sensor de la cámara debido a radiación.	3	4	M, A
PLD-05	La conexión física entre los componentes se pierde.	1	5	M
PLD-06	El lente seleccionado es el incorrecto para el enfoque que se busca.	3	4	M
PLD-07	Las imágenes tomadas necesitarán post-procesamiento.	5	1	A
PLD-08	Las PCB utilizadas generarán una impedancia.	5	5	M, E
ACS-01	Los imanes no permanecen fijos en su posición.	2	5	M
ACS-02	El satélite no puede obtener la posición requerida.	3	4	E, M
ACS-03	Los imanes son colocados incorrectamente.	1	5	M
ACS-04	El movimiento del carrusel hará que el satélite gire sobre uno de sus ejes.	4	3	E, M, A
ACS-05	El satélite no puede obtener imágenes de la Tierra.	2	4	M, E
COMM-01	Los datos no son enviados a la estación en tierra.	4	5	E, M, A
COMM-02	El mecanismo de despliegue de la antena falla.	5	4	E, M
COMM-03	Fallas en la radio y en la computadora a bordo.	3	4	E, M, A
COMM-04	Falla en la electrónica de la radio.	3	3	E, M, A
COMM-05	Las pruebas no reflejan adecuadamente las condiciones de operación.	5	4	M, A
PRG-01	Falta de experiencia de los estudiantes.	2	5	M
PRG-02	Falta de fondos.	1	5	M
PRG-03	Documentación deficiente.	1	4	M
PRG-04	Problemas con las fechas.	1	4	M

A continuación, se presentan los riesgos posicionados en el diagrama de criticidad según el valor de su probabilidad y su consecuencia.

Figura 79. Diagrama de criticidad de todos los riesgos del proyecto.



Como se pudo observar, se logró definir cada uno de los principales riesgos que podrían afectar el desempeño del sistema durante la misión. Debido a que se le asignó un valor a la probabilidad de ocurrencia y a la consecuencia de cada riesgo, y a que se definieron los métodos de mitigación más apropiados, pueden asignarse recursos para mitigar de la mejor forma posible los riesgos más críticos para la misión, todos aquellos en el área roja. Los responsables del módulo de donde proviene el riesgo deben hacer todo lo que posible para mitigar estos, manteniendo siempre una comunicación apropiada con la persona encargada del módulo de Ingeniería en Sistemas. Los riesgos menos críticos, aquellos en el área verde, no deben ser ignorados; sin embargo, su mitigación tampoco debe ser una prioridad.

B. CARACTERÍSTICAS OPERACIONALES DE LOS COMPONENTES

Para llevar un control apropiado de cada componente se le ha asignado a cada uno un código conformado por un grupo de letras y números. Cada código comienza con las letras "PT", las cuales establecen que se trata de un componente; el siguiente grupo de letras designa el módulo al cual pertenecen los componentes, se utilizarán las mismas siglas que se han utilizado anteriormente; a continuación se

colocan tres o cuatro letras las cuales designan la función o naturaleza del componente, y las cuales sirven también para agrupar componentes similares; finalmente se coloca un número en caso exista más de un componente dentro de los grupos designados con el grupo de letras recién explicado. Los espacios sin información indican que esta no se ha obtenido, o que este tipo de datos no es válido para el componente observado.

Cuadro 88. Descripción de componentes y su masa.

Código	Descripción	Estado	Cantidad	Masa por unidad (g)
PT-PLD-CAM-001	Cámara monocromática	No definido	1	-
PT-PLD-MOT-001	Motor	No definido	1	-
PT-PLD-CAR-001	Parte superior del carrusel	Definido	1	7
PT-PLD-CAR-002	Parte inferior del carrusel	Definido	1	7
PT-PLD-CAR-003	Eje del carrusel	Definido	1	4
PT-PLD-OPT-001	Material óptico para protección del lente	Definido	1	7
PT-PLD-LENS-001	Lente óptico	No definido	1	-
PT-PLD-FILT-001 ¹	Filtro óptico de 1 442 nm	Definido	1	25
PT-PLD-FILT-002 ¹	Filtro óptico de 2 550 nm	Definido	1	25
PT-PLD-FILT-003 ¹	Filtro óptico de 3 671 nm	Definido	1	25
PT-COMM-TRA-001 ²	Transmisor VHF/UHF	Definido	1	25
PT-CDH-OBC-001	Computadora a bordo	No definido	-	-
PT-ST-BAS-001	Parte superior de la estructura	Definido	2	29
PT-ST-BAS-002	Parte lateral de la estructura	Definido	2	51
PT-ST-BAS-003	Paneles PCB	No definido	9	-
PT-ST-SCR-001	Tornillos M 2x0.4 6 mm	Definido	24	1
PT-ST-WSH-001	Arandelas #2	Definido	24	1
PT-ST-INS-001	Aislante	No definido	1	-
PT-PWR-EPS-001	Cableado	No definido	-	-
PT-PWR-EPS-002	Tabla de regulación y distribución de potencia eléctrica	No definido	1	-
PT-PWR-EPS-003 ³	Pin RBF y cordón	Definido	1	5
PT-PWR-EPS-004 ⁴	Módulo EV2400	Definido	-	-
PT-PWR-EPS-005	Compuesto Arathane 5753 A/B	Definido	1	-
PT-PWR-EPS-006 ⁵	Seguidor de Punto de Máxima Potencia y cargador de batería	Definido	1	-
PT-PWR-EPS-007 ⁶	Regulador Buck de 3.3 V	Definido	1	-
PT-PWR-EPS-008 ⁷	Sensor de temperatura	Definido	2	-
PT-PWR-EPS-009 ⁸	Capacitor cerámico de 6.8 μF	Definido	1	-
PT-PWR-EPS-010 ⁹	Monitor de voltaje/corriente	Definido	1	-
PT-PWR-EPS-011 ¹⁰	Medidor de nivel de la batería	Definido	1	-
PT-PWR-EPS-012	Cinta de <i>kapton</i>	Definido	1	-
PT-PWR-EPS-013 ⁶	Regulado <i>Buck</i> de 5 V	Definido	1	-
PT-PWR-EPS-014 ¹¹	Transistor MOSFET canal P	Definido	2	-
PT-PWR-EPS-015 ¹²	Transistor MOSFET canal dual N	Definido	2	-

Continuación Cuadro 88

Código	Descripción	Estado	Cantidad	Masa por unidad (g)
PT-PWR-EPS-016 ¹³	Transistor MOSFET canal N de 25 V	Definido	2	-
PT-PWR-EPS-017 ¹⁴	Inductor de 2.2 μ H	Definido	2	-
PT-PWR-EPS-018 ¹⁵	Capacitor cerámico de 22 μ F	Definido	2	-
PT-PWR-EPS-019 ¹⁶	Resistor de sentido de 0.03 Ohm	Definido	2	-
PT-PWR-EPS-020 ¹⁷	Resistor de 0.02 Ohm	Definido	2	-
PT-PWR-EPS-021 ¹⁵	Capacitor cerámico de 10 μ F	Definido	2	-
PT-PWR-EPS-022 ¹⁸	Resistor de sentido de 0.001 Ohm	Definido	2	-
PT-PWR-EPS-023 ¹⁹	Inductor de bobina blindado de 15 μ H	Definido	2	-
PT-PWR-EPS-024 ²⁰	Arreglo de diodos Schottky	Definido	2	-
PT-PWR-EPS-025 ²¹	Transistor MOSFET canal N de 25 V	Definido		
PT-PWR-EPS-026 ²²	Diodo Zener de 5.6 V	Definido	2	-
PT-PWR-EPS-027 ²³	Capacitor cerámico de 1 μ F	Definido	2	-
PT-PWR-EPS-028 ²⁴	Capacitor cerámico de 0.1 μ F	Definido	2	-
PT-PWR-EPS-029 ²⁵	Resistor de 499 kOhm	Definido	2	-
PT-PWR-EPS-030 ²⁵	Resistor de 165 kOhm	Definido	2	-
PT-PWR-EPS-031 ²⁵	Resistor de 93.1 kOhm	Definido	2	-
PT-PWR-EPS-032 ²⁵	Resistor 6.8 kOhm	Definido	2	-
PT-PWR-EPS-033 ²⁶	Resistor 100 Ohm	Definido	2	-
PT-PWR-EPS-034 ²⁶	Resistor de 10 MOhm	Definido	2	-
PT-PWR-EPS-035 ²⁷	Resistor de 5.1 kOhm	Definido	2	-
PT-PWR-EPS-036 ²⁸	Resistor de 10 kOhm	Definido	2	-
PT-PWR-BAT-001 ²⁹	Batería	Definido	2	40
PT-PWR-SCL-001 ³⁰	Celda solar	Definido	11	4
PT-PWR-SCL-002	Cinta kapton a doble lado	Definido	-	-
PT-PWR-SCL-003	Alambre para soldadura	Definido	-	-
PT-PWR-CON-001	Conector de 4 pines	No definido	-	-
PT-ACS-MAG-001 ³¹	Barra magnética	Definido	1	6.03
PT-ACS-MAG-002	Barra de histéresis	Definido	2	0.65
PT-ACS-SNS-001 ³²	Magnetómetro	Definido	1	5.67
PT-ACS-SNS-002 ³³	Fotodiodo	Definido	6	-
PT-ACS-CRC-001	Convertidor analógico-digital	No definido	1	-

1 - Página web de Edmund Optics, Inc.

2 - GOMspace NanoCom AX100

3 - Página web de Pumpkin Space Systems

4 - Texas Instruments EV2400 EVM Interface Board

5 - Texas Instruments bq24650 Synchronous Switch-Mode Battery Charge Controller for Solar Power with Maximum Power Point Tracking

6 - Texas Instruments TPS6213x 3-V to 17-V, 3-A Step-Down Converter In 3x3 QFN Package

7 - Texas Instruments TMP10x Temperature Sensor with I²C and SMBus Interface with Alert Function in SOT-23 Package

8 - TDK Corporation Multilayer Ceramic Chip Capacitors

9 - Texas Instruments INA3221 Triple-Channel, High-Side Measurement, Shunt and Bus Voltage Monitor with I²C-and SMBUS-Compatible Interface

- 10 - Texas Instruments bq28z610 Impedance Track Gas Gauge and Protection Solution for 1-Series to 2-Series Cell Li-Ion Battery Packs
- 11 - Infineon Technologies IRF9321PbF
- 12 - Vishay Siliconix Dual N-Channel 40-V (D-S) MOSFET
- 13 - Texas Instruments N-Channel NexFET Power MOSFETs
- 14 - Vishay Dale Low Profile, High Current Inductors
- 15 - Murata Electronics North America Chip Monolithic Ceramic Capacitors
- 16 - Yageo Current Sensor – Low TCR
- 17 - Samsung Electro-Mechanics Ultra Low Ohms (RU Series)
- 18 - Stackpole Electronics Inc. CSNL Series
- 19 - Taiyo Yuden SMD Power Inductors (NR series)
- 20 – Fairchild/ON Semiconductor BAT54C Schottky Diodes
- 21 – Vishay Siliconix N-Channel 60 V (D-S) MOSFET
- 22 – Fairchild/ON Semiconductor MM3Z2V4C-MM3Z75VC Zener Diodes
- 23 – Samsung Electro-Mechanics CL31B105KBHNNNE
- 24 – Samsung Electro-Mechanics CL21B104KBCNNNC
- 25 – Panasonic Electronic Components Precision Thick Film Chip Resistors
- 26 – Samsung Electro-Mechanics Thick-Film Chip Resistor
- 27 – Yageo Thick Film Chip Resistors
- 28 – Rohm Semiconductor Thick Film Chip Resistors
- 29 – Sparkfun 585460 Li-Polymer Battery Packs
- 30 – Azur Space 30% Triple Junction GaAs Solar Cell Assembly
- 31 – Página web de K&J Magnetics, Inc.
- 32 – Honeywell 3-Axis Digital Compass IC HMC5883L
- 33 – Vishay Semiconductor Opto Division Ambient Light Sensor

Si se suma la masa de todos los componentes presentados en el cuadro anterior, tomando en cuenta la cantidad de cada componente, se puede llegar a saber que por ahora la masa total del satélite es de 475 gramos, lo cual es menos de la mitad del límite. Sin embargo, aún falta que se escojan ciertos componentes, y por lo tanto que se agregue su masa. Además, la masa de los componentes electrónicos más pequeños no se incluye en la tabla debido a falta de información o a que se consideraba despreciable; pero es importante que una vez ensamblados estos componentes se mida la masa del sistema, ya que la masa combinada de todos estos componentes electrónicos puede volverse significativa para el sistema.

En el siguiente cuadro se presenta el rango de temperaturas operacionales de los componentes. Este es el rango de temperatura bajo el cual los componentes pueden operar apropiadamente. Es importante contar con esta información para determinar si los componentes seleccionados podrán soportar los cambios de

temperatura dentro del satélite. Es importante conocer también las dimensiones principales de cada componente; lo cual ayuda a asegurar que ninguno sobrepasa los límites del satélite en ninguna de las tres dimensiones.

Cuadro 89. Rangos de temperatura y dimensiones de componentes.

Código	Rango de temperaturas operacionales (°C)		Dimensiones (mm)		
	Mín	Máx	X	Y	Z
PT-PLD-CAM-001	-	-	-	-	-
PT-PLD-MOT-001	-	-	-	-	-
PT-PLD-CAR-001	-	-	83	83	3.55
PT-PLD-CAR-002	-	-	83	83	3.55
PT-PLD-CAR-003	-	-	8	8	30
PT-PLD-OPT-001	-	225	-	-	-
PT-PLD-LENS-001	-	-	-	-	-
PT-PLD-FILT-001	-	-	25	25	5
PT-PLD-FILT-002	-	-	25	25	5
PT-PLD-FILT-003	-	-	25	25	5
PT-COMM-TRA-001	-30	85	65	40	6.5
PT-CDH-OBC-001	-	-	-	-	-
PT-ST-BAS-001	-	-	94	100	5
PT-ST-BAS-002	-	-	100	8.5	113.5
PT-ST-BAS-003	-	-	-	-	-
PT-ST-SCR-001	-55	120	-	-	-
PT-ST-WSH-001	-	-	-	-	-
PT-ST-INS-001	-250	120	-	-	-
PT-PWR-EPS-001	-	-	-	-	-
PT-PWR-EPS-002	-	-	-	-	-
PT-PWR-EPS-003	-40	115	-	-	-
PT-PWR-EPS-004	-40	115	-	-	-
PT-PWR-EPS-005	-40	115	-	-	-
PT-PWR-EPS-006	-40	115	-	-	-
PT-PWR-EPS-007	-40	115	-	-	-
PT-PWR-EPS-008	-40	115	-	-	-
PT-PWR-EPS-009	-40	115	-	-	-
PT-PWR-EPS-010	-40	115	-	-	-
PT-PWR-EPS-011	-40	115	-	-	-
PT-PWR-EPS-012	-40	115	-	-	-
PT-PWR-EPS-013	-40	115	-	-	-
PT-PWR-EPS-014	-40	115	-	-	-
PT-PWR-EPS-015	-40	115	-	-	-
PT-PWR-EPS-016	-40	115	-	-	-
PT-PWR-EPS-017	-40	115	-	-	-
PT-PWR-EPS-018	-40	115	-	-	-
PT-PWR-EPS-019	-40	115	-	-	-
PT-PWR-EPS-020	-40	115	-	-	-
PT-PWR-EPS-021	-40	115	-	-	-
PT-PWR-EPS-022	-40	115	-	-	-
PT-PWR-EPS-023	-40	115	-	-	-
PT-PWR-EPS-024	-40	115	-	-	-
PT-PWR-EPS-025	-40	115	-	-	-
PT-PWR-EPS-026	-40	115	-	-	-

Continuación Cuadro 89

Código	Rango de temperaturas operacionales (°C)		Dimensiones (mm)		
PT-PWR-EPS-027	-40	115	-	-	-
PT-PWR-EPS-028	-40	115	-	-	-
PT-PWR-EPS-029	-40	115	-	-	-
PT-PWR-EPS-030	-40	115	-	-	-
PT-PWR-EPS-031	-40	115	-	-	-
PT-PWR-EPS-032	-40	115	-	-	-
PT-PWR-EPS-033	-40	115	-	-	-
PT-PWR-EPS-034	-40	115	-	-	-
PT-PWR-EPS-035	-40	115	-	-	-
PT-PWR-EPS-036	-40	115	-	-	-
PT-PWR-BAT-001	5	60	54	60	5.8
PT-PWR-SCL-001	-170	120	80	0.1	40
PT-PWR-SCL-002	-	-	-	-	-
PT-PWR-SCL-003	-	-	-	-	-
PT-PWR-CON-001	-	-	-	-	-
PT-ACS-MAG-001	-	80	2.7	2.7	6.35
PT-ACS-MAG-002	-	-	1	1	95
PT-ACS-SNS-001	-30	85	25.4	25.4	2
PT-ACS-SNS-002	-40	100	4	2	1.05
PT-ACS-CRC-001	-40	85	6.6	6.4	1.2

A través del cuadro anterior puede observarse que todos los componentes de los cuales se cuenta con información sobre su rango de temperaturas operacionales se encuentran dentro del límite especificado por el requerimiento MI-400. Por otro lado, a través de la observación de las dimensiones de los componentes puede notarse que ninguno sobrepasa los límites establecidos por los requerimientos DE-030 y DE-040.

Las cualidades de desgasificación también son importantes a la hora de elegir un componente. Como se mencionó anteriormente, para que un componente pueda ser incluido en el satélite solamente debe poseer materiales aceptados por JAXA. Para que puedan ser aceptados, estos materiales deben cumplir con poseer un TML menor o igual a 1% y un CVCM menor o igual a 0.1%. A continuación, se muestran los materiales principales de cada componente junto con sus respectivos niveles de TML y CVCM. Debe mencionarse que no se incluye información de desgasificación de la mayoría de los metales presentados debido a que no se encontró información al respecto en las bases de datos consultadas. Esta información fue obtenida de la página web de la NASA sobre Información de Desgasificación para Investigaciones y Reportes (*Outgassing Data for Search and Report*) y su base de datos MAPTIS.

Ya que algunos componentes aún no han sido definidos, no se cuenta con información de los materiales de tales componentes.

Cuadro 90. Materiales de los componentes junto con sus cualidades de gasificación.

Código	Materiales	TML (%)	CVCM (%)	Grado
PT-PLD-CAM-001	-	-	-	-
PT-PLD-MOT-001	-	-	-	-
PT-PLD-CAR-001	Policarbonato	0.1	0	A
PT-PLD-CAR-002	Policarbonato	0.1	0	A
PT-PLD-CAR-003	Aluminio 7075	-	-	-
PT-PLD-OPT-001	Policarbonato	0.1	0	A
PT-PLD-LENS-001	Sustrato de vidrio	0.16	0.06	A
PT-PLD-FILT-001	Sustrato de vidrio	0.16	0.06	A
PT-PLD-FILT-002	Sustrato de vidrio	0.16	0.06	A
PT-PLD-FILT-003	Sustrato de vidrio	0.16	0.06	A
PT-COMM-TRA-001	Poliamida	-	-	-
PT-CDH-OBC-001	-	-	-	-
PT-ST-BAS-001	Aluminio 7075	-	-	-
PT-ST-BAS-002	Aluminio 7075	-	-	-
PT-ST-BAS-003	-	-	-	-
PT-ST-SCR-001	Acero inoxidable	0	0	A
PT-ST-WSH-001	Acero inoxidable	0	0	A
PT-ST-INS-001	-	-	-	-
PT-PWR-EPS-001	-	-	-	-
PT-PWR-EPS-002	-	-	-	-
PT-PWR-EPS-003	-	-	-	-
PT-PWR-EPS-004	Zirconato de calcio	-	-	-
	Titanato de bario	-	-	-
	Estaño	-	-	-
	Trióxido de bismuto	-	-	-
	Sílice	0.16	0.06	A
	Zinc	-	-	-
	Cromo	-	-	-
	Cobalto	-	-	-
	Manganeso	-	-	-
	Fósforo	-	-	-
	Silicón	0.36	0.02	A
	Sulfuro	-	-	-
	Sulfito de sodio	-	-	-
	1,4-diclorobenceno	-	-	-
	Nafta solvente	-	-	-
	Silicato de plomo	-	-	-
	Óxido de rutenio	-	-	-
	Óxido de magnesio	0.03	0.01	A
	Paladio	-	-	-
	Trióxido de dicromo	-	-	-
	Flúor	-	-	-
PT-PWR-EPS-005	-	-	-	A
PT-PWR-EPS-006	Epoxy	0.37	0.03	A
	Fósforo	-	-	-
	Zinc	-	-	-
	Paladio	-	-	-
	Carbón negro	-	-	-
	Silicón dopado	-	-	-
	Sílice fundido	0.16	0.06	A

Continuación Cuadro 90

Código	Materiales	TML (%)	CVCM (%)	Grado
PT-PWR-EPS-007	Sulfuro	-	-	-
	Manganeso	-	-	-
	Epoxy	0.37	0.03	A
	Fósforo	-	-	-
	Zinc	-	-	-
	Paladio	-	-	-
	Sílice fundido	0.16	0.06	A
	Carbón negro	-	-	-
PT-PWR-EPS-008	Silicón dopado	-	-	-
	Epoxy	0.37	0.03	A
	Fósforo	-	-	-
	Zinc	-	-	-
	Paladio	-	-	-
	Sílice fundido	0.16	0.06	A
	Carbón negro	-	-	-
	Silicón dopado	-	-	-
PT-PWR-EPS-009	Silicón	0.36	0.02	A
	Titanato de bario	-	-	-
	Zirconato de bario	-	-	-
PT-PWR-EPS-010	Estaño	-	-	-
	Epoxy	0.37	0.03	A
	Cromo	-	-	-
	Estaño	-	-	-
	Zinc	-	-	-
	Paladio	-	-	-
	Sílice fundido	0.16	0.06	A
	Carbón negro	-	-	-
PT-PWR-EPS-011	Silicón dopado	-	-	-
	Paladio	-	-	-
	Sílice	0.16	0.06	A
	Epoxy	0.37	0.03	A
	Cromo	-	-	-
	Estaño	-	-	-
	Zinc	-	-	-
	Sílice fundido	0.16	0.06	A
PT-PWR-EPS-012	Carbón negro	-	-	-
	Silicón dopado	-	-	-
PT-PWR-EPS-013	-	-	-	-
PT-PWR-EPS-013	Sulfuro	-	-	-
	Manganeso	-	-	-
	Epoxy	0.37	0.03	A
	Fósforo	-	-	-
	Zinc	-	-	-
	Paladio	-	-	-
	Sílice fundido	0.16	0.06	A
	Carbón negro	-	-	-
PT-PWR-EPS-014	Silicón dopado	-	-	-
	Silicón	0.36	0.02	A
	Epoxy	0.37	0.03	A
	Sílice	0.16	0.06	A
PT-PWR-EPS-015	Estaño	-	-	-
	Epoxy	0.37	0.03	A
	Sílice	0.16	0.06	A
PT-PWR-EPS-016	Estaño	-	-	-
	Magnesio	-	-	-
PT-PWR-EPS-016	Silicón	0.36	0.02	A

Continuación Cuadro 90

Código	Materiales	TML (%)	CVCM (%)	Grado
PT-PWR-EPS-017	-	0.04	0.01	A
PT-PWR-EPS-018	Zirconato de calcio	-	-	-
	Titanato de bario	-	-	-
	Estaño	-	-	-
PT-PWR-EPS-019	Estaño	-	-	-
PT-PWR-EPS-020	Estaño	-	-	-
	Óxido de rutenio	-	-	-
	Cuproníquel	-	-	-
PT-PWR-EPS-021	Zirconato de calcio	-	-	-
	Titanato de bario	-	-	-
	Estaño	-	-	-
PT-PWR-EPS-022	Estaño	-	-	-
PT-PWR-EPS-023	Ferrita	0.49	0.03	A
PT-PWR-EPS-024	Estaño	-	-	-
	Epoxy	0.37	0.03	A
	Ablebond 84-1 LM1SR4	0.26	0.01	A
	Zinc	-	-	-
	Paladio	-	-	-
	Sílice fundido	0.16	0.06	A
	Carbón negro	-	-	-
	Silicón dopado	-	-	-
PT-PWR-EPS-025	-	-	-	-
PT-PWR-EPS-026	Sílice	0.16	0.06	A
PT-PWR-EPS-027	Titanato de calcio	-	-	-
	Estaño	-	-	-
PT-PWR-EPS-028	Zirconato de calcio	-	-	-
	Titanato de bario	-	-	-
	Estaño	-	-	-
PT-PWR-EPS-029	Zirconato de calcio	-	-	-
	Titanato de bario	-	-	-
	Estaño	-	-	-
PT-PWR-EPS-030	Estaño	-	-	-
PT-PWR-EPS-031	Estaño	-	-	-
PT-PWR-EPS-032	Estaño	-	-	-
PT-PWR-EPS-033	Estaño	-	-	-
PT-PWR-EPS-034	Óxido de rutenio	-	-	-
	Óxido de bismuto	-	-	-
	Sílice	0.16	0.06	A
	Estaño	-	-	-
PT-PWR-EPS-035	Óxido de rutenio	-	-	-
	Óxido de bismuto	-	-	-
	Sílice	0.16	0.06	A
	Estaño	-	-	-
PT-PWR-EPS-036	Óxido de rutenio	-	-	-
	Óxido de bismuto	-	-	-
	Sílice	0.16	0.06	A
	Estaño	-	-	-
PT-PWR-BAT-001	Óxido de litio cobalto	-	-	-
	Níquel de litio	-	-	-
	Manganeso de litio	-	-	-
	Grafito	0.53	0.01	A
	Carbón graficable	-	-	-
	Polietilenglicol	-	-	-
	Poliacrilonitrilo	0.59	0	A

Continuación Cuadro 90

Código	Materiales	TML (%)	CVCM (%)	Grado
PT-PWR-SCL-001	Indio galio fosfuro	-	-	-
	Arseniuro de galio	-	-	-
	Germanio	-	-	-
	Dióxido de titanio	-	-	-
PT-PWR-SCL-002	-	-	-	-
PT-PWR-SCL-003	Estaño	-	-	-
PT-PWR-CON-001	-	-	-	-
PT-ACS-MAG-001	Boro de hierro de neodimio	-	-	-
PT-ACS-MAG-002	Níquel	-	-	-
	Molibdeno	-	-	-
PT-ACS-SNS-001	-	-	-	-
PT-ACS-SNS-002	Silicón	0.36	0.02	A
PT-ACS-CRC-001	-	-	-	-

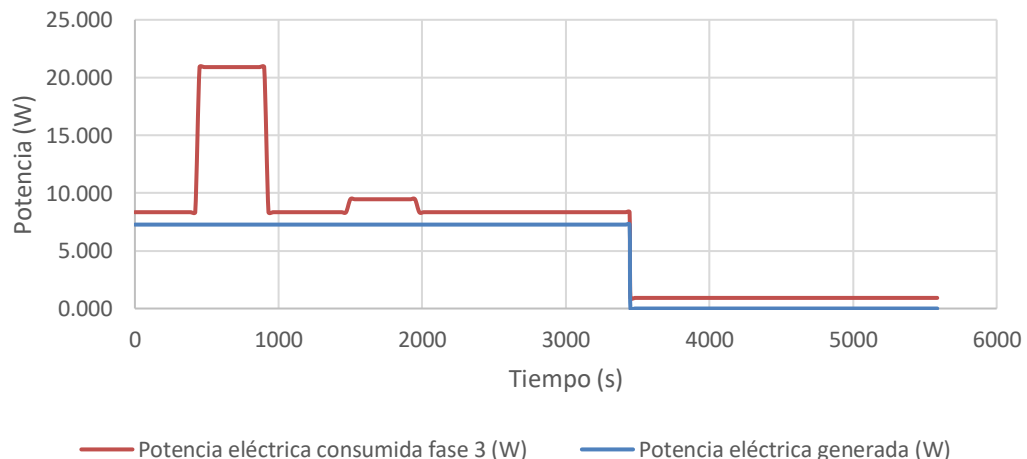
Como puede observarse en el Cuadro 90, los materiales de todos los componentes definidos hasta ahora cumplen con el requerimiento DE-500 sobre gasificación.

C. FACTIBILIDAD DE DISEÑO DE POTENCIA EN FASE 3

Se obvió la capacidad de las baterías, ya que las baterías seleccionadas por la fase anterior del Megaproyecto no habían sido utilizadas en el espacio lo que generaba muchas dudas acerca de si estas funcionarían bajo las condiciones extremas que experimentan los elementos en órbita. Además, al realizar las pruebas en Tierra a las baterías se comprobó que no eran aptas para esta aplicación. Otro supuesto fue que el sistema de distribución de energía tenía una eficiencia de un 100%, el caso más optimista posible. Esto debido a que este sistema no se había trabajado con anterioridad, incluyendo fase 3 del Megaproyecto.

Las simulaciones de potencia se realizaron computando los datos en una hoja electrónica de Microsoft Excel. Las siguientes gráficas muestran los resultados obtenidos del presupuesto de potencia de acuerdo al diseño de la fase 3 del Megaproyecto para un periodo (revolución alrededor de la Tierra).

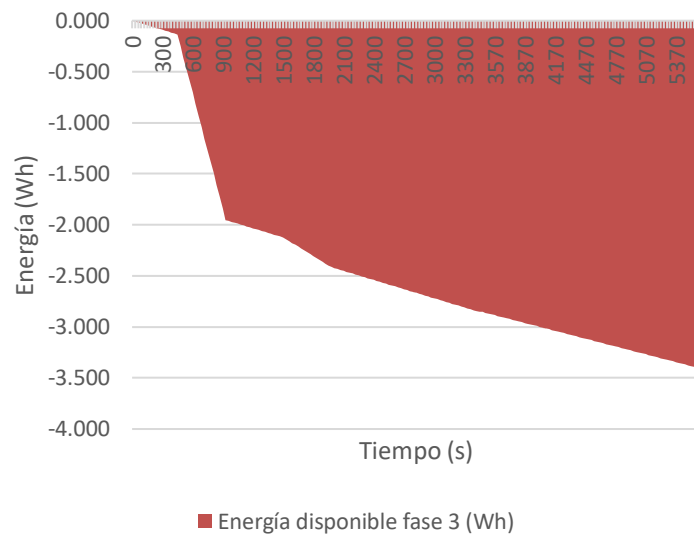
Figura 80. Presupuesto de potencia fase 3– tiempo.



En la gráfica se muestra un periodo bajo las condiciones límites. La potencia consumida posee dos picos que corresponden al modo de operación de vuelo sobre la estación y vuelo a Guatemala respectivamente. Posteriormente se muestra un declive en la potencia consumida y generada que representa la fase de eclipse o sombra a la cual se ve expuesto el satélite.

También se llevó a cabo un análisis de la energía disponible que las baterías tendrían acceso a almacenar en un periodo de revolución. Esto sin tomar en cuenta la capacidad de las baterías, ni un estado de carga inicial de las mismas. Otro supuesto fue que no se desperdiciaba energía en el circuito interno que le permite a las baterías almacenar la energía proveniente de las celdas solares.

Figura 81. Energía disponible fase 3 – tiempo.

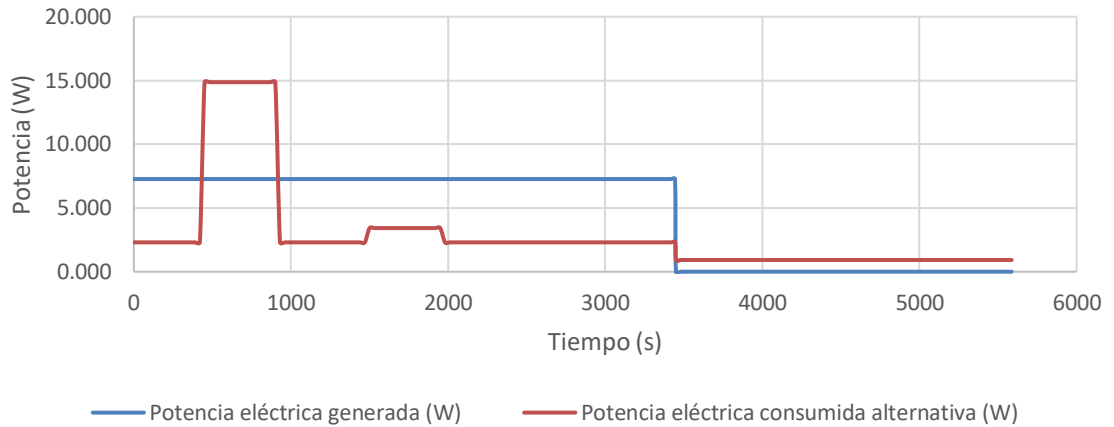


La Figura 81 muestra que la recolección de potencia eléctrica a partir del panel solar nunca es mayor que la potencia eléctrica consumida. Esto implica que nunca se tiene un excedente de energía que permita cargar las baterías. Además, la Figura 81 muestra que la energía que el satélite debe tener a disposición para operar durante un periodo es de 3.4 Wh, este es un déficit de energía neta por ciclo excesivo.

Tomando en cuenta que la misión se espera tendrá una duración aproximada de 6 meses y que un periodo típico en esta órbita toma alrededor de 90 minutos se espera que el satélite opere durante aproximadamente 2,900 revoluciones. Esto implicaría tener baterías con una capacidad de al menos 9,860 Wh cargadas. La tecnología de baterías con una mejor capacidad específica es Litio-polímero (capacidad específica de aproximadamente 240 Wh/kg. (Incell Academy, s.f) Lo que implicaría al menos 41.1 kg de baterías, lo cual no es factible tomando en cuenta que el máximo de masa para un satélite bajo el estándar de Cal Poly para CubeSats 1U es de 1.33 kg. (Cal Poly, 2009)

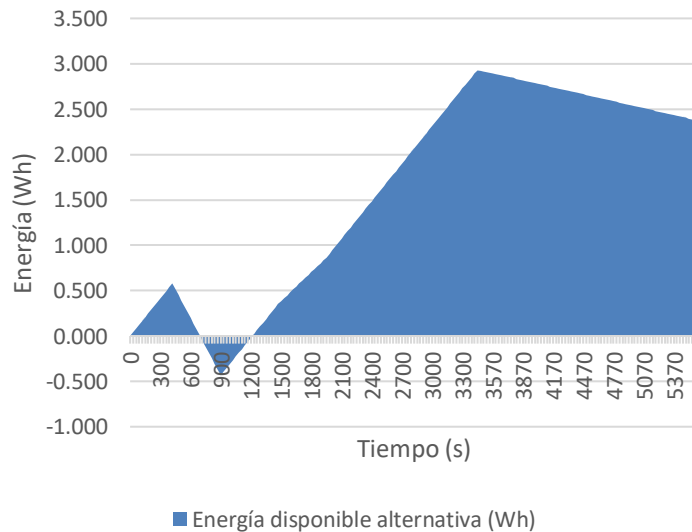
A partir de esto fue posible determinar que el modelo desarrollado durante la fase 3 del Megaproyecto no era factible desde el punto de vista de potencia. Notando que este sistema no era factible se prosiguió a realizar el análisis con la alternativa para el módulo de ADCS con un consumo más bajo de potencia.

Figura 82. Presupuesto de potencia alternativa – tiempo.



La Figura 82 muestra un sistema en el cual la potencia eléctrica generada es mayor que la potencia eléctrica consumida durante gran parte del periodo. Esto permite tener un excedente de energía que podría ser almacenado en las baterías que actúan como fuente secundaria de potencia.

Figura 83. Energía disponible alternativa – tiempo.



La Figura 83 muestra que bajo la alternativa de ADCS existía un excedente de energía neto de 2.35 W que podría ser almacenado en las baterías. Sin embargo, posteriormente se investigó el funcionamiento del

sistema ADCS “pasivo” y se encontró que el control que este tipo de sistema daba al satélite sobre su orientación no era lo suficientemente preciso como para poder implementar los paneles desplegados, por lo que esta alternativa si bien reducía el consumo de potencia también reduciría de manera significativa la recolección de potencia eléctrica y por ende tampoco era factible.

Si bien el sistema con ADCS “pasivo” tampoco era factible, la simulación ayudó a visualizar que el sistema de la fase 3 no era factible y que el reducir el consumo de potencia, en particular de los sistemas que operaban más tiempo, podría llevar a un diseño factible desde el punto de vista de potencia.

Esto llevó a un cambio radical en la forma de trabajo en el satélite, donde no se tomaban en cuenta los efectos de las decisiones de un módulo en los demás y además estableció a la potencia como uno de los factores más importantes a tomar en cuenta para la selección de componentes a través de evaluación de conceptos (*trade studies*) para los diversos módulos que componen al satélite. Además, se identificó que el consumo de ADCS y COMMS era excesivo por lo que se buscaron alternativas con menor demanda de potencia.

D. ANÁLISIS DE ALMACENAMIENTO DE ENERGÍA

Se determinó que la condición límite, con base en el análisis de la fase 3, era cuando el satélite se comunicaba con la estación terrena (0.95 Wh) y, por ende, el dimensionamiento del sistema de almacenamiento de energía se haría en base a esto.

Además de la energía que requieren las cargas es importante tomar en cuenta los límites superiores e inferiores de carga permisibles de las baterías, los cuales están dictados por la cantidad de ciclos de carga/descarga que deberían operar. Se prosiguió a definir los límites de carga tomando en cuenta una cantidad de ciclos carga/descarga de 4,500 ciclos, según el requerimiento MI-265.

Se ha demostrado que el tener una profundidad de descarga elevada puede reducir la cantidad de ciclos de carga/descargas útiles de las baterías. Por lo que es necesario tomar en cuenta cuantos ciclos se esperan durante la misión para definir apropiadamente la profundidad de descarga. Con base en el Cuadro 2 se decidió que la energía almacenada no superara el 10% de la capacidad de las baterías a seleccionar. Por lo que, las baterías deberían tener una capacidad de al menos 9.5 Wh.

E. SELECCIÓN DE COMPONENTES DEL MÓDULO DE POTENCIA

Cuadro 91. Evaluación de conceptos de celdas solares.

Factores	Normalización	Modelos de decisión							
		AZUR SPACE SCA 3G30A		Spectrolab 30.7% XTJ Solar Cells		Spectrolab 28.3% UTJ Solar Cells			
Variables	Peso (1-5)	1	10	VN	Total	VN	Total	VN	Total
Potencia	5	1.043	1.211	10	50	6	30	1	5
Masa	1	3.562	2.286	1	1	10	10	10	10
dV/dT	2	-6.70	-6.50	1	2	10	20	10	20
Precio por celda	2	329	319	9	18	1	2	10	20
Legado espacial	1	No	Sí	10	10	10	10	10	10
Total					81		72		65

Se llegó a una decisión objetiva que las celdas solares 3G30A de AZUR SPACE serían la mejor opción para la misión. Estas celdas contienen una capa superior de vidrio para protección, así como diodos para prevenir corrientes inversas. Su legado espacial incluye varias misiones de CubeSats como: OUFTI-1, ESTCube-1 y MinXSS.

Además, uno de los coordinadores de proyecto Luis Zea, mencionó que existían muchas regulaciones para la exportación de componentes espaciales fabricados en Estados Unidos de América. Debido a que Spectrolab se sitúa en Estados Unidos de América esto representaba un factor más a favor de las celdas de AZUR SPACE que no pudo ser cuantificado en la matriz de decisión. Adicionalmente, se tomó en cuenta que se buscará llevar a cabo el ensamblaje de los paneles solares con base en un método desarrollado por la Universidad de Colorado, Boulder, quienes utilizaron las celdas 3G30C de AZUR SPACE, cuyas diferencias son la falta de la capa superior de protección, conectores y diodo de protección.

Cuadro 92. Evaluación de conceptos de baterías.

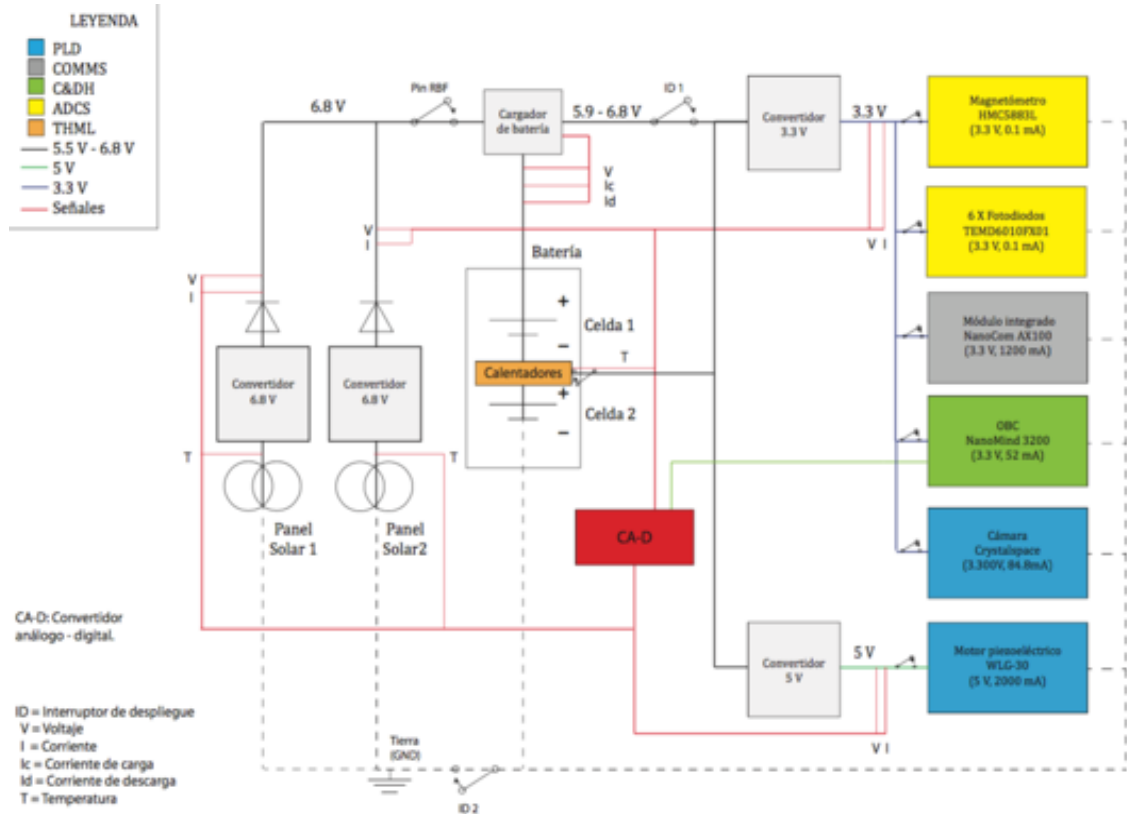
Factores	Normalización	Modelos de decisión									
		Sparkfun		Clyde Space		EXA BA01X		Gomspace			
Variables	Peso (1-5)	1	10	2Ah PRT- 08483		CubeSat Battery		Battery Array		NanoPower Cells	
				VN	Total	VN	Total	VN	Total	VN	Total
% Volumen	2	12.5%	5.1%	10	20	1	2	6	12	10	20
% Masa	3	9.5%	5.6%	10	30	1	3	3	9	6	18
Costo	5	3,200	26	10	50	5	25	1	5	7	35
Capacidad	1	10.0	26.6	5	5	1	1	10	10	6	6
Misiones exitosas	1	0	3	4	4	7	7	7	7	10	10
Temp. mínima	2	0	-30	1	2	1	2	10	20	3	6
Temp. máxima	3	40	80	1	3	1	3	10	30	2	6
Total					109		38		43		89

A partir de la evaluación de conceptos de las baterías se determinó que la mejor opción para el CubeSat sería la de Sparkfun 2Ah. Esta tiene una misión conocida que ha sido exitosa que es la del CubeSat 3U MinXSS desarrollado en la Universidad de Colorado, Boulder. Estas baterías fueron elegidas por su legado espacial y bajo costo que permitía adquirir bastantes para realizar las pruebas necesarias sin alterar las baterías que serían utilizadas en la misión.

F. SISTEMA DE DISTRIBUCIÓN DE ENERGÍA

El sistema de distribución de energía tiene una gran importancia para el funcionamiento del satélite. Este se debe comunicar con la OBC para activar y desactivar las cargas mediante interruptores, además debe distribuir la energía proveniente de los paneles solares y las baterías. La siguiente figura muestra el diseño preliminar que se contempló para el sistema de distribución de energía, el cual se basó en el sistema diseñado para el satélite tipo CubeSat 3U MinXSS de la Universidad de Colorado, Boulder.

Figura 84. Esquema de sistema de distribución de energía preliminar.



Los calentadores serán resistencias eléctricas, sin embargo, todavía no se ha determinado el valor de la resistencia a utilizar, por lo que no se incluyó en el esquema. Por claridad se decidió no incluir ciertos circuitos integrados que tienen un consumo nominal de corriente de micro amperios.

El voltaje de 5.9 a 6.8 V está dictado por el estado de carga de la batería. Todos los interruptores hacia las cargas deberán ser controlados por OBC. Además, es importante mencionar que al momento de plantear este diseño no se han seleccionado la OBC, la cámara ni el motor por lo que tampoco se sabe qué *driver* debería llevar el motor. Inclusive el módulo de COMMS se encuentra bajo análisis.

Se recomienda utilizar el método de corriente para estimar el estado de carga de la batería, ya que el método de voltaje depende de la temperatura y la corriente de descarga. El cargador de la batería es un componente autónomo por lo que no necesita que las señales pasen por OBC. (Texas Instruments, 2015)

El sensor de temperatura que se utilizará para monitorear la temperatura de las baterías tiene una precisión de ± 2 °C. (Texas Instruments, 2015) Ya que las baterías no pueden ser cargadas por debajo de los 0 °C, se recomienda que los calentadores sean activados cuando el sensor indique 10 °C y se desactiven al llegar a los 20°C. Además, los convertidores de voltaje seleccionados tienen una eficiencia que oscila entre 85 y 95% y se puede ajustar su voltaje de salida según se requiera. (Texas Instruments, 2016)

Si bien se propuso un diseño preliminar para el satélite tomando en cuenta los requerimientos funcionales, de Cal Poly y de UNOOSA/JAXA, este no será final ya que todavía se están realizando cambios, debido a la naturaleza iterativa del proyecto. No se trabajó más a profundidad en este tema, ya que se consideró que no se tenía la experiencia ni el conocimiento necesario para diseñar un circuito de potencia para aplicaciones espaciales y no se quiso arriesgar el éxito de todo el proyecto por este aspecto.

G.RIESGOS DEL MÓDULO DE POTENCIA

Cuadro 93. Riesgo 1, baterías fallan durante misión.

Causas	Probabilidad	Consecuencia	Criticidad	Prioridad	1/Prioridad	Peso	PN*	CN**
Temperatura elevada	2	3	6	1	1	0.27	0.55	0.82
Almacenaje en condiciones no apropiadas	3	2	6	1	1	0.27	0.82	0.55
No uniformidades	2	2	4	2	0.50	0.14	0.27	0.27
Baja temperatura	1	3	3	3	0.33	0.09	0.09	0.27
Fatiga mecánica	3	1	3	3	0.33	0.09	0.27	0.09
Voltaje elevado	1	2	2	4	0.25	0.07	0.07	0.14
Bajo voltaje	1	2	2	4	0.25	0.07	0.07	0.14
Total						1	2.14	2.27

*Probabilidad normalizada

**Consecuencia normalizada

Cuadro 94. Plan de mitigación. riesgo 1, baterías fallan durante misión.

Causas	TM	Acción	Módulo responsable
Temperatura elevada	T	Investigar temperatura esperada dentro del satélite.	Térmico
Almacenaje en condiciones no apropiadas	C	Evitar que las baterías sean almacenadas con alta carga, evitar altas y bajas temperaturas, 20 °C +/- 5 °C recomendable. ¹	Potencia
No uniformidades	A	No hay acción.	No hay responsable
Baja temperatura	E	Accionar calentadores cuando sensor de temperatura en batería indique temperatura de 10 °C.	C&DH, Potencia
Fatiga mecánica	C	Mantener una profundidad de descarga menor o igual al 10%. Control de expansión térmica.	C&DH, Potencia, Estructura
Voltaje elevado	C	Programación de carga máxima equivalente a un 80% de carga.	C&DH, Potencia
Bajo voltaje	C	Programación de carga mínima equivalente a un 70% de carga.	C&DH, Potencia

¹ = (Wang, et al, 2006)

Se asumirá el riesgo de no uniformidades en las baterías, ya que cambiar algo implicaría cambiar el proceso productivo del fabricante, algo que está fuera del alcance del proyecto. Sin embargo, las baterías que se utilizarán en el satélite pertenecerán al mismo lote que las baterías que serán sometidas a pruebas.

Cuadro 95. Riesgo 2, falla de sistema de distribución de energía eléctrica.

Causas	Probabilidad	Consecuencia	Criticidad	Prioridad	1/Prioridad	Peso	PN*	CN**
Falta de experiencia diseñando circuitos de potencia para aplicaciones espaciales	3	4	12	1	1	0.37	1.11	1.48
Daño por radiación espacial	2	5	10	2	0.50	0.19	0.37	0.93
Fallo en circuito de protección de batería	2	4	8	3	0.33	0.12	0.25	0.49
Corto circuito en algún componente	2	3	6	4	0.25	0.09	0.19	0.28
Gasificación de componentes	2	3	6	4	0.25	0.09	0.19	0.28
Desconexión de baterías	1	5	5	5	0.20	0.07	0.07	0.37
Algún componente no está fijado de manera correcta	1	2	2	6	0.17	0.06	0.06	0.12
Total						1	2.23	3.95

*Probabilidad normalizada

**Consecuencia normalizada

Cuadro 96. Plan de mitigación. riesgo 2, falla de sistema de distribución de energía eléctrica.

Causas	TM	Acción	Módulo responsable
Falta de experiencia diseñando circuitos de potencia para aplicaciones espaciales	A	No hay acción.	Potencia, Gestión de proyectos
Daño por radiación espacial	C	Se recomienda emplear PCBs aprobadas para aplicaciones espaciales.	Potencia
Fallo en circuito de protección de batería	C	Realizar pruebas a circuito de protección de batería.	C&DH, Potencia
Corto circuito en algún componente	E	Emplear protecciones necesarias.	Potencia
Gasificación de componentes	C	Seleccionar e investigar las propiedades de gasificación de los componentes a utilizar.	Potencia, Sistemas
Desconexión de baterías	E	Unir mediante soldadura y realizar pruebas de vibraciones.	Potencia, Estructura
Algún componente no está fijado de manera correcta	E	Realizar pruebas de vibraciones.	Estructura

Si bien no existe acción para mitigar la falta de experiencia en diseño de circuitos de potencia para aplicaciones espaciales, si se debe buscar personal con experiencia en circuitos de potencia para realizar

este diseño. En Guatemala no ha existido un proyecto de esta índole, por lo que para encontrar personal capacitado para diseñar circuitos de potencia para aplicaciones espaciales se debería buscar en el extranjero, lo que cual no parece factible.

Cuadro 97. Riesgo 3, daño a celdas solares durante instalación.

Causas	Probabilidad	Consecuencia	Criticidad	Prioridad	1/Prioridad	Peso	PN*	CN**
Malas técnicas de soldadura	3	3	9	1	1	0.41	1.24	1.24
Manejo inadecuado de celdas solares	3	2	6	2	0.50	0.21	0.62	0.41
Las celdas solares se fracturan	4	1	4	3	0.33	0.14	0.55	0.14
Almacenamiento en condiciones no apropiadas	2	2	4	3	0.33	0.14	0.28	0.28
Descarga eléctrica	2	1	2	4	0.25	0.10	0.21	0.10
Total						1	2.90	2.17

*Probabilidad normalizada

**Consecuencia normalizada

Cuadro 98. Plan de mitigación. riesgo 3, daño a celdas solares durante instalación.

Causas	TM	Acción	Módulo responsable
Malas técnicas de soldadura	E	Realizar pruebas de vibraciones.	Estructura
Manejo inadecuado de celdas solares	E	Utilizar guantes de látex, se recomienda el uso de pinzas de vacío ¹ . Evitar contacto con la mano.	Potencia
Las celdas solares se fracturan	A	No hay acción.	No hay responsable
Almacenamiento en condiciones no apropiadas	E	Almacenar en ambiente libre de humedad, fuera del alcance de terceros.	Potencia
Descarga eléctrica	E	Emplear regulación de voltaje, evitar contacto directo con polos de las celdas.	Potencia

1 = (AZUR SPACE, 2012)

No existe acción para evitar que las celdas solares se fracturen, más allá de ser cuidadoso/a al realizar la instalación, lo cual es más una recomendación que una acción a tomar. Sin embargo, se contempló este

escenario y con el fin de minimizar su impacto en el proyecto se ordenaron quince celdas solares, cuando las que se utilizarán serán once. Además, se realizó un proceso de ensamblaje para minimizar la probabilidad de daño a las celdas solares durante su instalación.

Cuadro 99. Riesgo 4, fallo de celdas solares durante misión.

Causas	Probabilidad	Consecuencia	Criticidad	Prioridad	1/Prioridad	Peso	PN*	CN**
Desconexión de celdas solares	2	3	6	1	1	0.40	0.80	1.20
Colisión con basura espacial	1	3	3	2	0.50	0.20	0.20	0.60
Daño por radiación espacial	1	2	2	3	0.33	0.13	0.13	0.27
Temperatura elevada	1	2	2	3	0.33	0.13	0.13	0.27
Baja temperatura	1	2	2	3	0.33	0.13	0.13	0.27
Total						1	1.40	2.60

*Probabilidad normalizada

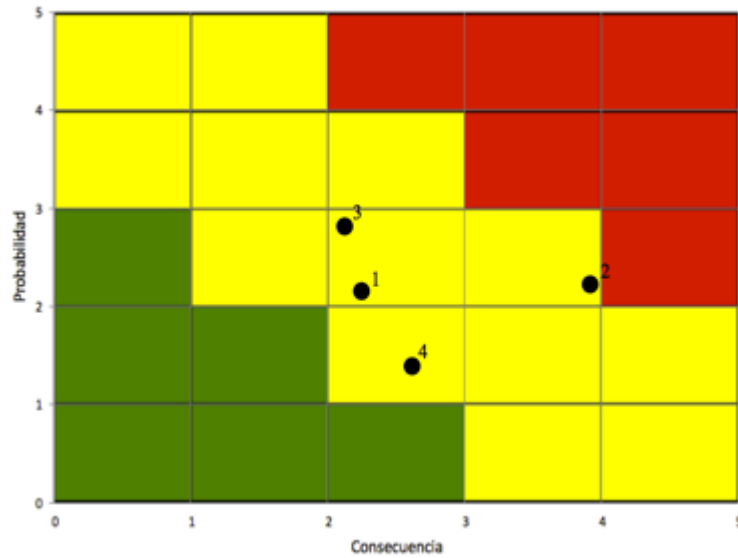
**Consecuencia normalizada

Cuadro 100. Plan de mitigación. riesgo 4, fallo de celdas solares durante misión.

Causas	TM	Acción	Módulo responsable
Desconexión de celdas solares	E	Soldar las conexiones, realizar pruebas de vibraciones.	Potencia, Estructura
Colisión con basura espacial	A	No hay acción.	No hay responsable
Daño por radiación espacial	C	Utilizar celdas solares aprobadas para el espacio con su capa protectora de vidrio (<i>coverglass</i>).	Potencia
Temperatura elevada	E	Investigar.	Térmico
Baja temperatura	E	Investigar.	Térmico

No existe una acción para mitigar la colisión con basura espacial, esta predomina en órbitas bajas como la que se planea utilizar. A pesar que no es algo muy probable, existe la posibilidad por lo que se incluyó. En cuanto a temperatura elevada y baja, actualmente el análisis térmico indica que el rango de temperatura operacional de las celdas solares será entre -44.7°C y 56.3°C, ya que la temperatura operacional de las celdas solares es desde -170 °C hasta 120 °C no habría problema en este aspecto.

Figura 85. Matriz de riesgos del módulo de potencia.



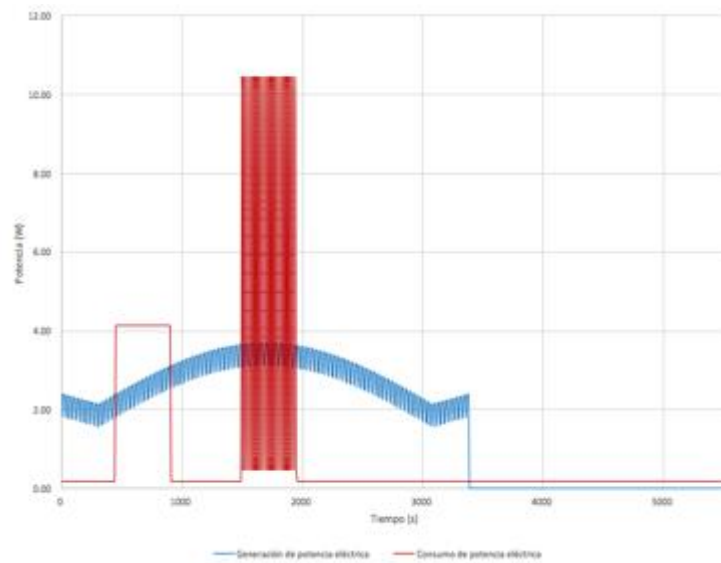
La matriz de riesgos muestra que los riesgos no se encuentran dentro del rango más crítico. Sin embargo, se debe prestar atención a ellos y seguir el plan de mitigación con el fin de disminuir su probabilidad, y/o consecuencia. Además, se deberán tomar en cuenta los riesgos del sistema integrado y desarrollar un plan de mitigación de los mismos.

H. FACTIBILIDAD DE DISEÑO DEL MÓDULO DE POTENCIA DE FASE 4

Se tomó en cuenta la carga disponible para cargar y descargar las baterías. Se asumió que el sistema de distribución de potencia tenía una eficiencia de un 75%, tomando en cuenta un caso pesimista, ya que se espera que su eficiencia sea mayor. También se tomó en cuenta el factor de temperatura como sugerido de 0.85. (Larson, *et al*, 1999)

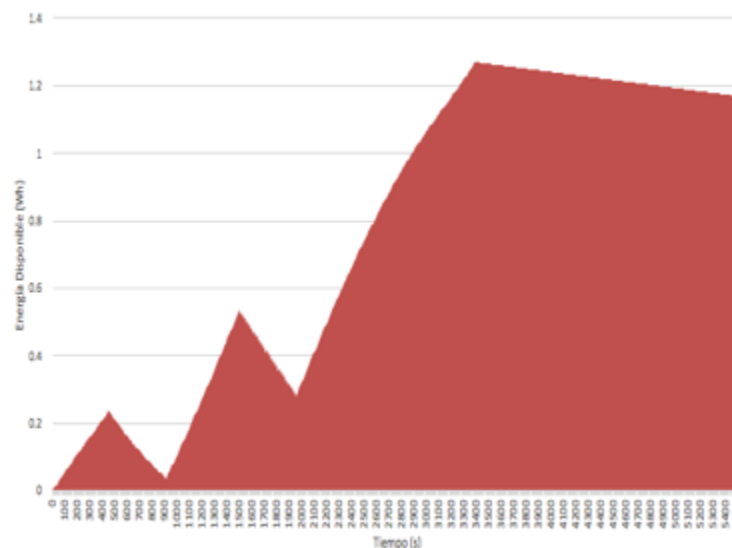
Las simulaciones de potencia se realizaron computando los datos en una hoja electrónica de Microsoft Excel. Las siguientes gráficas muestran los resultados obtenidos del presupuesto de potencia de acuerdo al diseño de la fase 4 del Megaproyecto para un periodo (revolución alrededor de la Tierra).

Figura 86. Presupuesto de potencia fase 4 – tiempo.



La oscilación en la potencia generada se debe a la rotación del satélite con respecto al eje Z. También se tomó en cuenta el ángulo límite de producción de las celdas solares que es de 85° de acuerdo al coseno de Kelly. (Oredsson, 2010) Se simuló el consumo de acuerdo a los tiempos utilizados para el estudio de factibilidad de la fase 3, para poder realizar una mejor comparación. Se tomó en cuenta que el motor piezoeléctrico funcionaba mediante pulsos, esto se refleja como una oscilación en el consumo de potencia.

Figura 87. Energía disponible fase 4 – tiempo.



En la Figura 87 se muestra que en este caso si habría energía para almacenar en las baterías, sin embargo, se recomienda que mediante programación se evalúe si es posible llevar a cabo los diversos modos de operación o si es recomendable dejar el satélite con su modo de operación de más simple “vuelo” mientras

el satélite se abastece de energía. Esto con el fin de garantizar que solo se operen los componentes no vitales cuando sea posible desde una perspectiva de potencia y energía.

Tomando en cuenta el análisis de factibilidad de la fase 4 del proyecto, se calculó la potencia eléctrica generada mínima durante la fase en la cual el satélite recibía luz solar (1.56 W) y se comparó con el consumo de potencia más elevado que se da con el vuelo sobre el objetivo (10.452 W). Con ello se estimó un consumo de potencia neto de 8.892W.

Ya que este modo opera durante 450 segundos se calculó que las baterías deberían tener una profundidad de descarga permisible de al menos 1.11 Wh. Esta profundidad de descarga equivale a un 6.61% de la capacidad de la batería, lo que indica que estarían dentro del rango que se había estimado para la misión que debería ser menor a 10%.

También, se calculó la potencia eléctrica generada y consumida promedio las cuales fueron de 1.692 W y 0.937 W, respectivamente. Si se obtiene un factor de seguridad con estos datos se encuentra que el mismo es de 1.81. Se incluyó un factor de utilización de los componentes no vitales con el fin de cuantificar el efecto que tendría la programación en la factibilidad del proyecto desde el punto de vista de este factor de seguridad.

Para tener un factor de seguridad de 2, el factor de utilización correspondiente sería de 88%, por lo que se podría accionar los modos de operación sobre el objetivo y sobre Guatemala en un 88% de las ocasiones que el satélite gire alrededor de la Tierra. Para un factor de seguridad de 3 se encontró que el factor de utilización correspondiente sería de 53%.

En la comparación entre el presupuesto de potencia de la fase 3 y fase 4, resulta evidente que es un escenario más factible el trabajado durante la fase 4. A pesar de generar menos potencia se logró contrarrestar esto mediante la asignación de un alto peso en las evaluaciones de conceptos de los componentes pertenecientes a los módulos dependientes.

La simulación realizada no será exactamente igual al caso real ya que se realizaron muchos supuestos y aproximaciones con el fin de poder comparar los resultados con los de la fase 3 y además simplificar un análisis sumamente complejo.

Es importante notar que por las características de la órbita no siempre se podrá tener comunicación con la estación terrena ubicada en Guatemala lo que altera el modo de operación del satélite reduciendo el consumo de potencia del satélite y la posibilidad de comunicar la información adquirida a la estación

terrena. También al accionar el motor piezoeléctrico se espera que se altere la rotación del satélite con respecto al eje Z por cierto periodo de tiempo, alterando la potencia eléctrica generada.

I. SISTEMA DE MANEJO DE COMANDOS Y DATOS

Para realizar la selección de la OBC se investigaron distintas opciones tomando en consideración distintos aspectos detallados con anterioridad, se recopilaron los datos requeridos con el fin de comparar y determinar la mejor opción por medio del método de *trade study*.

Cuadro 101. Ponderación de las variables esenciales para la selección de la computadora a bordo.

Variable	Ponderación	1	2	3	4	5	6	7	8	9	10
Masa (g)	5	79.0	71.2	63.4	55.7	47.9	40.1	32.3	24.6	16.8	9.0
Volumen(mm ³)	5	5187 5.7	46307.3	40738.9	35170.5	29602.1	24033.6	18465.2	12896.8	7328.4	1760.0
Potencia (W)	5	0.4	0.4	0.3	0.3	0.2	0.2	0.1	0.1	0.0	0.0
Velocidad de Reloj (GHz)	3	0.0	0.1	0.2	0.3	0.5	0.6	0.7	0.8	0.9	1.0
RAM (MB)	3	0.0	56.9	113.8	170.7	227.6	284.4	341.3	398.2	455.1	512.0
Memoria (MB)	2	0.3	111.3	222.4	333.5	444.6	555.7	666.8	777.8	888.9	1000.0
Numero de Interfaces	5	4.0	4.7	5.3	6.0	6.7	7.3	8.0	8.7	9.3	10.0
Curva de aprendizaje	3	Fácil									Difícil
Misiones	4	No									Sí
Costo (\$)	1	451.0	401.4	351.9	302.3	252.8	203.2	153.7	104.1	54.5	5.0

El cuadro anterior muestra la ponderación y la normalización de valores para las variables utilizadas en la selección de la computadora a bordo por medio del método de *trade study*, como se puede apreciar las variables que poseen una mayor ponderación son masa, volumen, potencia y el número de interfaces; el valor para las primeras dos listadas se debe a las limitaciones que presentan los estándares del CubeSat, al ser un 1U el espacio se limita a 10x10x10cm mientras que el peso debe mantenerse menor a 1.33kg; por aparte la potencia es un factor crucial para el funcionamiento del CubeSat, usualmente en los CubeSat el presupuesto de energía suele ser muy limitado por lo cual es de suma importancia que los componentes seleccionados utilicen la menor cantidad de potencia posible; por último el número de interfaces de comunicación establece la capacidad de interacción entre los distintos dispositivos y la OBC por lo cual entre mayor sea el número, mayor será la variedad de componentes que podrán ser controlados por la OBC; la variable que sigue en ponderación es la utilización de las OBC en misiones espaciales esta característica ayuda a asegurar el funcionamiento de la OBC en condiciones extremas; las ponderaciones siguientes corresponden a valores de velocidad de procesamiento y la dificultad de utilización, las cuales como se mencionó con anterioridad juegan un papel importante para la selección de OBC.

Además, en el cuadro se presentan los valores normalizados para cada una de las variables, la normalización se estableció para un intervalo de 10 valores, estos valores se calculan tomando el mejor valor y peor valor para cada característica, es decir, el que más beneficie y el que menos beneficie; estos valores son restados y divididos dentro de 10 el valor resultante representa el delta de diferencia que se necesita para llegar del peor valor situado en la posición #1 en 9 pasos hasta el mejor valor que se encuentra en la posición #10.

En el siguiente cuadro se observan los resultados finales del trade study, existen dos columnas para cada opción evaluada, la primera columna representa el valor de asignación de la normalización, este se obtiene de comparar el valor de cada característica contra los 10 valores normalizados, asignando el valor (1-10) conforme al valor normalizado que más se apegue al valor del componente para esa característica. Luego de calcular los valores normalizados para cada característica se calculan los totales multiplicando el valor asignado del proceso anterior por su respectiva ponderación; por último, se hace la sumatoria de todos los totales para cada característica y el valor resultante es la valoración que recibe cada opción.

Como se puede observar en el cuadro la mejor opción es la Raspberry Pi Zero, su total se debe a que es un dispositivo con grandes capacidades de procesamiento con 1GHz de velocidad de reloj y 512MB de RAM, acompañado de una masa (9 gramos) y volumen (65mmx30mmx5mm) muy pequeños, el único factor que pone en duda su inclusión es el consumo de potencia que ronda los 0.4 W, lo cual es un valor elevado si se toma en consideración los límites que presentan los presupuestos de energía en este tipo de satélites.

Cuadro 102. Selección de computadora a bordo utilizando el método de trade study.

Variable	Raspberry Pi Zero		STM32F407		LCP 3250		Stamp9C20		Mega 2560		MSP430F67791A		Prometheus.PC-104	
	Normalizado	Total	Normalizado	Total	Normalizado	Total	Normalizado	Total	Normalizado	Total	Normalizado	Total	Normalizado	Total
Masa (g)	10	50	5	25	8	40	5	25	6	30	1	5	1	5
Volumen(mm ³)	9	45	5	25	7	35	8	40	5	25	1	5	1	5
Potencia (W)	1	5	10	50	1	5	6	30	4	20	9	45	1	5
Clock Speed (GHz)	10	30	3	9	3	9	4	12	1	3	1	3	2	6
RAM (MB)	10	30	1	3	2	6	2	6	1	3	1	3	2	6
Memoria (MB)	10	20	1	2	2	4	2	4	1	2	1	2	1	2
Numero de Interfaces	3	15	10	50	6	30	6	30	1	5	1	5	3	15
Curva de aprendizaje	10	30	1	3	5	15	5	15	10	30	5	15	5	15
Misiones	10	40	10	40	10	40	10	40	10	40	10	40	10	40
Costo (\$)	10	10	9	9	1	1	8	8	10	10	10	10	1	1
Total	275		216		185		210		168		133		100	

Para las pruebas de consumo de energía de la Raspberry Pi se realizaron mediciones del voltaje de alimentación y de la corriente que consume en tres estados diferentes de la computadora: inicio, pasivo y funcionamiento. El estado de inicio corresponde al momento de encender la computadora mientras carga el sistema operativo y se activa. El estado pasivo corresponde al momento donde todo el sistema operativo está cargado, pero no se encuentra en funcionamiento ningún programa adicional. El estado de funcionamiento requiere que los programas para comunicación con el bus de datos estén activos y funcionando. Las mediciones se realizaron en los tres estados antes y después de haberse aplicado las mejoras de rendimiento para el consumo energético.

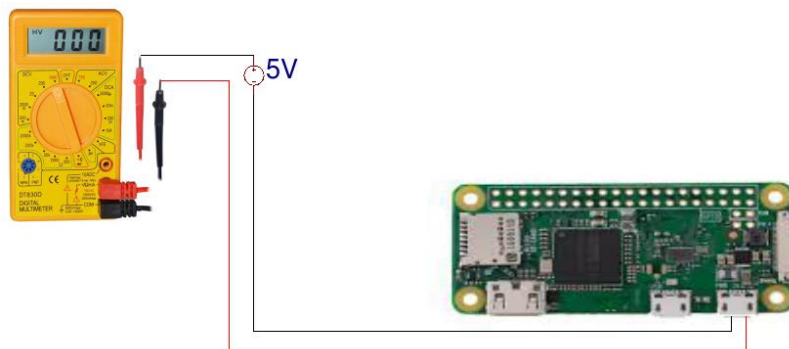
El voltaje de alimentación para la Raspberry Pi es de 4.43V como se muestra en la siguiente figura.

Figura 88. Medición de voltaje de la Raspberry Pi con multímetro.



La siguiente figura muestra el diagrama que se utilizó para realizar la medición de corriente.

Figura 89. Medición de corriente de la Raspberry Pi.



El siguiente cuadro muestra el consumo de corriente y de energía de la Raspberry Pi sin las modificaciones para reducir el consumo energético.

Cuadro 103. Mediciones de energía de la Raspberry Pi sin modificaciones

Estado	Consumo de corriente [mA]	Consumo de energía [W]
Inicio	355	1.57
Pasivo	330	1.46
Funcionamiento	335	1.48

El siguiente cuadro muestra el consumo de corriente y de energía de la Raspberry Pi con las modificaciones para reducir el consumo energético.

Cuadro 104. Mediciones de energía de la Raspberry Pi sin modificaciones

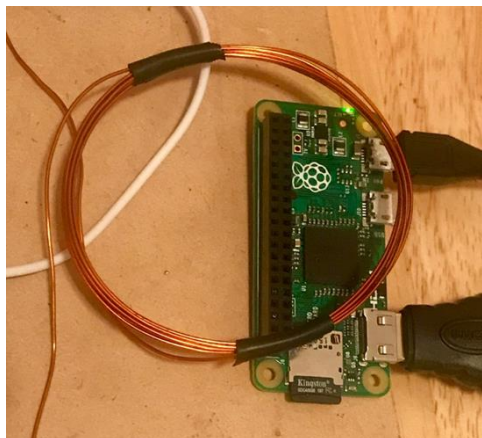
Estado	Consumo de corriente [mA]	Consumo de energía [W]
Inicio	85	0.37
Pasivo	60	0.26
Funcionamiento	65	0.28

El sistema operativo ligero instalado y las configuraciones redujeron el consumo de energía en 1.2 W para las tres configuraciones de la computadora que se probaron, esta reducción es significativa en el satélite porque da lugar a mayor distribución de los recursos energéticos en otros componentes.

Para las pruebas de interferencias electromagnéticas se expuso una Raspberry Pi (conectada únicamente a su alimentación de energía y a un monitor) al mecanismo generador de interferencias electromagnéticas. Se realizaron mediciones de distancia y se observó el efecto producido en la computadora, el frecuencímetro y el analizador de espectro se conectaron a la terminal del circuito que genera los pulsos eléctricos para analizar la interferencia.

La siguiente figura muestra la aplicación de la interferencia electromagnética sobre la Raspberry Pi.

Figura 90. Prueba de interferencia electromagnética.



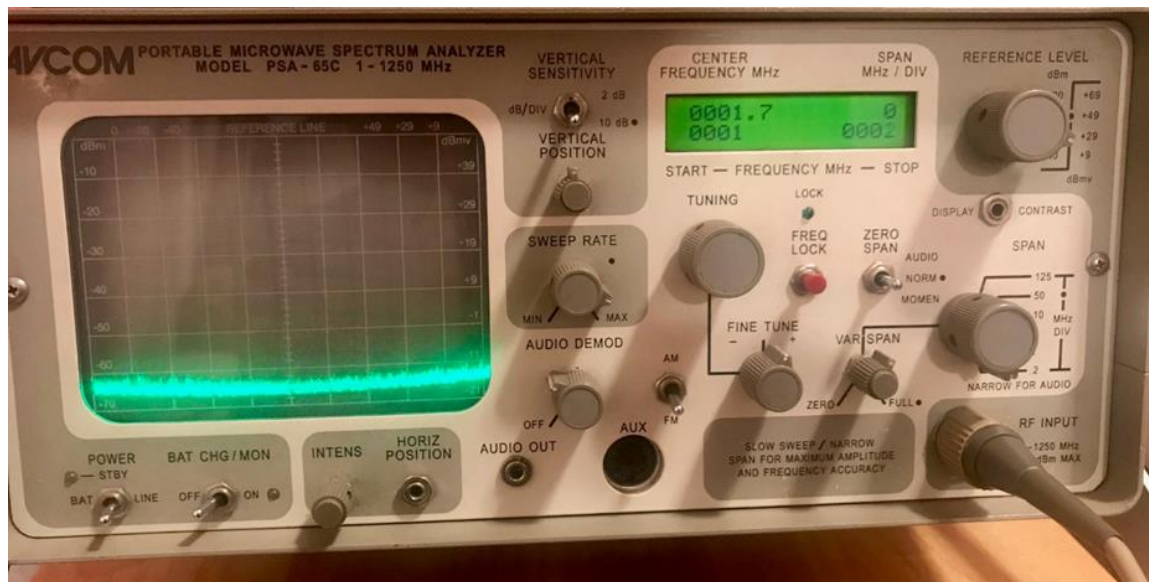
La frecuencia del pulso eléctrico aproximadamente es de 6.4920 MHz, la medición de esta frecuencia se muestra en la siguiente figura.

Figura 91. Frecuencia de los pulsos eléctricos para la prueba de interferencia electromagnética.



El analizador de espectro en la siguiente figura se encuentra inactivo.

Figura 92. Medición del analizador de espectro sin actividad electromagnética.



En las siguientes figuras se muestra la medición que realizó el analizador de espectro cuando se aplicó el pulso eléctrico y se generó la interferencia electromagnética. Para estas mediciones se configuró con 2MHz por división en el eje x y un nivel de referencia de -20dBm.

Figura 93. Medición del analizador de espectro con actividad electromagnética.

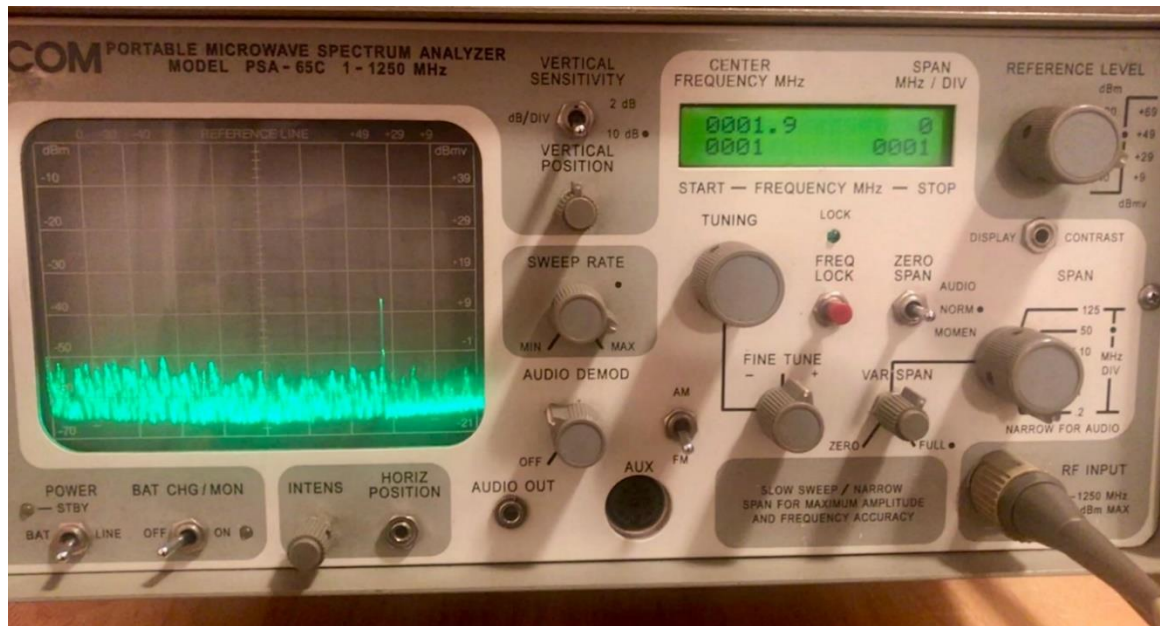
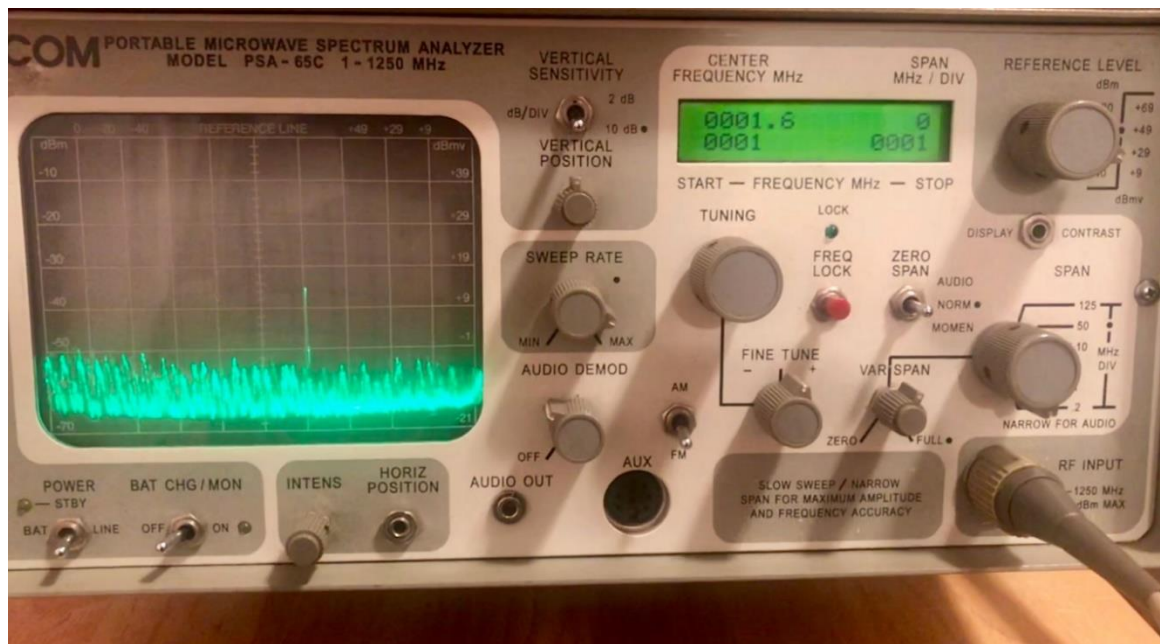


Figura 94. Medición del analizador de espectro con actividad electromagnética.



El siguiente cuadro describe la relación entre la distancia a la cual se aplicó la interferencia electromagnética y los efectos causados a la Raspberry Pi.

Cuadro 105. Efectos de la interferencia electromagnética.

Distancia [cm]	Efecto
30	Ninguno
20	Ninguno
10	Ninguno
9	Interrupción en los datos de salida de vídeo
8	Interrupción en los datos de salida de vídeo
7	Interrupción en los datos de salida de vídeo
6	Interrupción en los datos de salida de vídeo
5	Interrupción en los datos de salida de vídeo
4	Interrupción en los datos de salida de vídeo
3	Reinicio de la computadora
2	Reinicio de la computadora
1	Reinicio de la computadora
0	Reinicio de la computadora

Con los efectos que se produjeron cuando se expuso la Raspberry Pi a interferencias electromagnéticas se demostró una percepción de los riesgos y posibles métodos de protección contra el ambiente del espacio. Si bien la interferencia que se produjo no dañó completamente la computadora, es necesario que se proteja porque la radiación del espacio y los efectos naturales que provoquen singularidades eléctricas son de magnitudes mayores a las que se generó en las pruebas.

Luego de la selección de la computadora a bordo se realizaron las pruebas funcionales para carga útil: mover el carrusel, tomar fotografías y para ADCS: obtener información del magnetómetro y de los fotodiodos, esto por medio del código generado y una Raspberry Pi.

Para las pruebas para el módulo de carga útil debido a que no se han definido sus componentes (cámara y motor) se utilizaron componentes sustitutos, para el caso de la cámara se utilizó el módulo de cámara de la Raspberry y para el motor un servomotor MG90S controlado por medio de señales PWM. Estas evaluaron el envío de comandos de acción al módulo de carga útil.

Figura 95. Conexión de servomotor MG90S a la Raspberry.

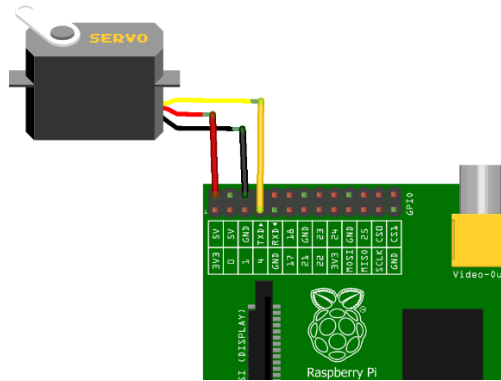


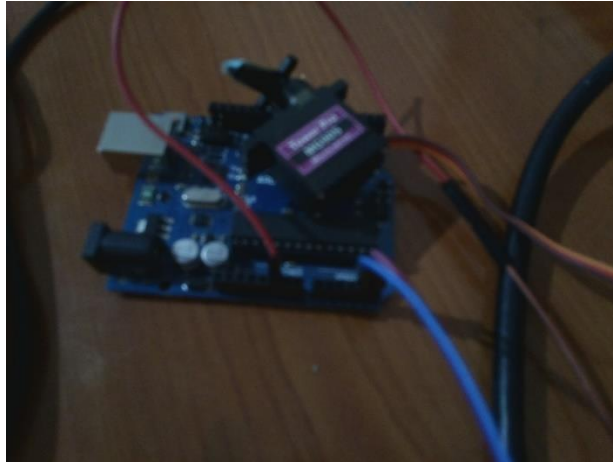
Figura 96. Conexión de módulo de cámara a la Raspberry.



Se probó la captura de imágenes y almacenamiento en una dirección específica del sistema, debido a que el módulo de cámara de la Raspberry tiene su propio driver se desarrolló un código en C para que realizará llamadas a dichas funciones con los parámetros requeridos, para el motor se evaluó el movimiento a la posición de los tres filtros estos situados 0° , 90° y 180° el movimiento es controlado por medio de señales PWM, la posición es determinada debido al tiempo de la onda.

A continuación, se presenta el resultado de la captura de imágenes por medio del módulo de la cámara de la Raspberry.

Figura 97. Captura de imagen de prueba con el módulo de cámara de la Raspberry.



El control de la cámara no representó mayor impedimento debido a que es un módulo específicamente desarrollado para la Raspberry, el sistema operativo Raspbian ya cuenta con un driver para su operación, lo que se implementó fue las llamadas a esas funciones las cuales funcionaron perfectamente,

La secuencia de imágenes a continuación son el resultado del movimiento del carrusel, como se puede apreciar se tienen tres filtros respectivamente enumerados, la posición del filtro actual (filtro con el cual se está tomando la imagen) se presenta subrayado en rojo, el movimiento para el filtro #1 es de 0° , para el #2 es de 90° y para el #3 de 180° . Para lograr su movimiento se desarrolló un driver que recibe como parámetro el número de filtro a colocar, este envía señales PWM por medio del GPIO 12 al cual está conectado el servomotor.

Figura 98. Movimiento de carrusel a 0° .

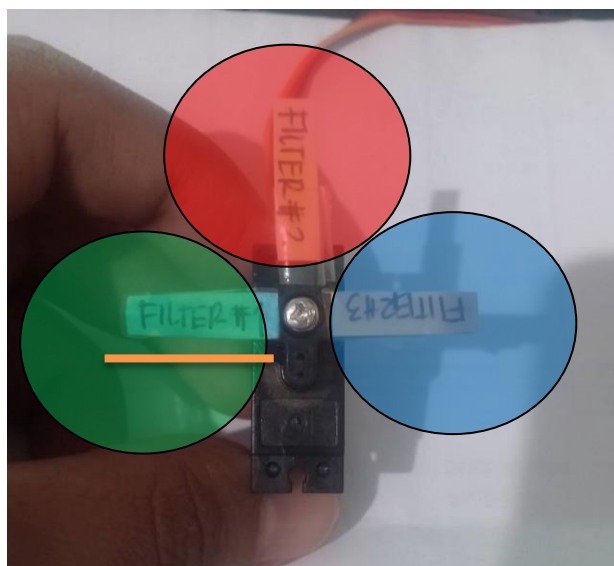


Figura 99. Movimiento de carrusel a 90°.

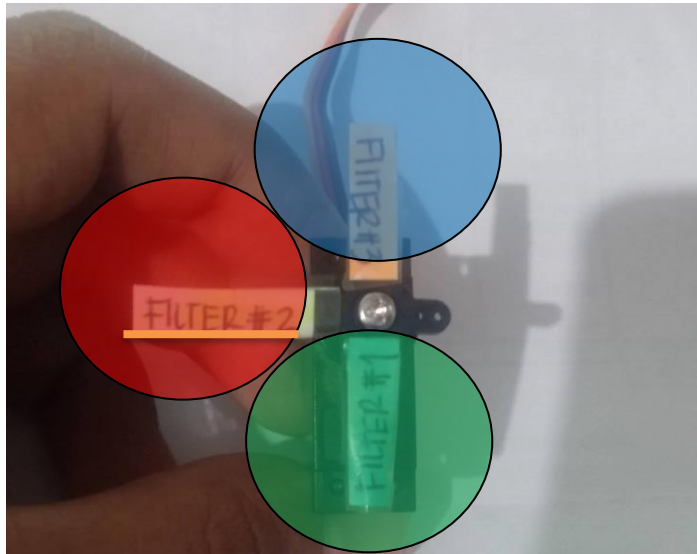
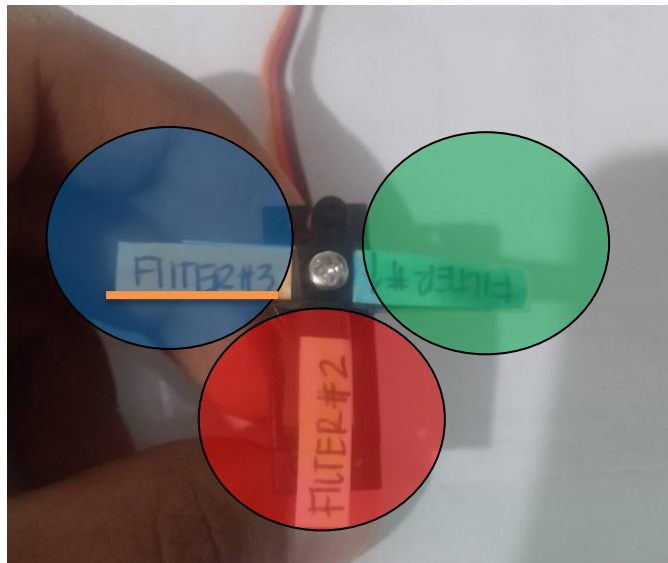


Figura 100. Movimiento de carrusel a 180°.



Aunque este no es el motor definitivo algo que se notó en las pruebas del movimiento del carrusel es que este no queda perfectamente alineado a la posición esperada, como se puede apreciar en las figuras, la posición del servomotor varía una cantidad pequeña de grados, esto es un factor importante a tomar en consideración en el diseño del carrusel para lograr que el lente de la cámara se acople bien al filtro.

En la misión el CubeSat debe ser capaz de tomar tres fotos con distinto filtro del mismo objetivo, el módulo de carga útil definió que el tiempo máximo para mover el carrusel y tomar la foto es de dos segundos. Por lo cual luego de evaluar el funcionamiento del módulo de carga útil se realizaron pruebas de

respuesta para verificar que las acciones realizadas cumplieran los requisitos funcionales 12 y 13, además el conjunto de acciones (tomar fotografía y mover carrusel) no superara los dos segundos.

Cuadro 106. Tiempos promedio de ejecución para cada una de las actividades de carga útil.

Actividad	Tomar fotografía	Mover carrusel	Tomar fotografía y mover carrusel
Tiempo promedio (s)	0.00084924	0.00084328	0.00170292

El cuadro anterior muestra el promedio del tiempo de ejecución para las actividades de carga útil, este es el promedio encontrado después de la ejecución de la actividad 25 veces, para asegurar que el resultado no se ve afectado por datos extremos, es decir, datos de tiempos muy cortos o de tiempos muy largos se calculó la desviación estándar.

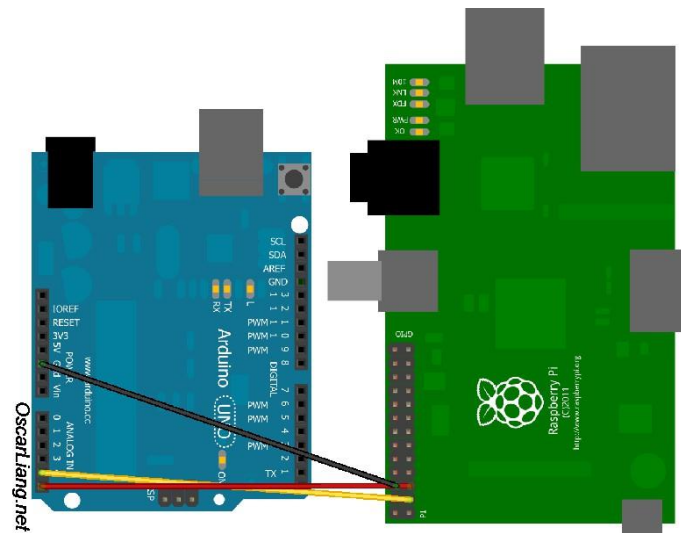
Cuadro 107. Desviación estándar para cada una de las actividades de carga útil.

Actividad	Tomar fotografía	Mover carrusel	Tomar fotografía y mover carrusel
Desviación estándar	6.28×10^{-5}	8.1×10^{-5}	1.2×10^{-4}

Los resultados de la desviación estándar presentados en el cuadro anterior nos dicen que los datos de los tiempos de ejecución se mantienen concentrados cerca de la media; estos tiempos de ejecución se encuentran entre el rango de los microsegundos, estos están muy por debajo de los requeridos que son de un segundo, incluso para los tiempos que involucran ambas actividades no se sobrepasa el segundo. Aunque aún no se hayan definido los componentes para la carga útil se puede decir que al menos se comprobó que existe la posibilidad de mover el carrusel de filtros y tomar una foto en menos de 2 segundos. Un aspecto que podría afectar el tiempo de ejecución del movimiento del carrusel es el peso que los lentes puedan tener debido a que las pruebas realizadas se realizaron con un peso sobre el servomotor.

Las pruebas para el módulo de ADCS se realizaron por medio de un Arduino Uno debido a que las pruebas con los componentes reales quedaban fuera del alcance de este trabajo, lo que se evalúa es la implementación de un protocolo útil para la interacción y la del software que garantice el intercambio de información; por lo cual se desarrolló un programa que simulara el funcionamiento de los componentes, éste funcionaba como esclavo de la Raspberry por medio del protocolo de comunicación I2C, para la obtención de datos se generaron números aleatorios dentro del rango de valores que especificaba la hoja de datos para cada componente. Con esto se probó el protocolo de comunicación establecido para la interacción (I2C) entre módulos en este caso la interacción con el módulo de ADCS, la interacción es garantizada por medio de los drivers desarrollados para cada componente los cuales permiten la configuración y la lectura de valores.

Figura 101. Comunicación de Raspberry con Arduino por medio de protocolo I2C.



Por medio del programa del Arduino se verificó la rutina de configuración y de obtención de datos para cada componente, es decir se verificó que se estuvieran enviando las instrucciones correctas a las direcciones específicas. En consola se imprimen los registros a los cuales se está accediendo, si el registro es de escritura inmediatamente después se presenta el valor que se está escribiendo por la OBC que es el maestro. Ambos dispositivos fueron primero configurados y luego se realizó la lectura de datos. A continuación, se presenta la verificación para el magnetómetro HMC5883L.

Figura 102. Verificación de comandos transmitidos por medio del protocolo de comunicación I2C para el chip HMC5883L (magnetómetro).

```

COM4 (Arduino/Genuino Uno)
[Enviar]

Data Recibida
CONFIGURATION A
value: 112
CONFIGURATION B
value: 32
REGISTER MODE
value: 1
REGISTER STATUS
REGISTER OUTPUT LSB X
REGISTER OUTPUT MSB X
REGISTER OUTPUT LSB Y
REGISTER OUTPUT MSB Y
REGISTER OUTPUT LSB Z
REGISTER OUTPUT MSB Z

Configuración
Lectura de datos

 Autoscroll
Sin ajuste de línea 9600 baudio Clear output

```

La configuración se realizó como se definió en el diseño. Como se puede apreciar en la figura anterior se escribió al registro de configuración A, este registro es el encargado de manejar el promedio de muestreo, la velocidad de salida de datos y de la configuración de medición; en él se definió un número

promedio de ocho muestras por medición, una velocidad de datos de 15 Hz y la configuración de medición normal, esta configuración está representada por la secuencia de bits 0b0111 0000 el cual representa el valor decimal de 112. Además, se configuró el registro de configuración B, este registro es el encargado de la configuración de ganancia; se estableció la ganancia por defecto de 1090 (LSb/Gauss) esta configuración está representada por la secuencia de bits 0b0010 0000 el cual representa el valor decimal de 32. Por último, se configuró el registro de modo para establecer el modo en una sola medición esta configuración está representada por la secuencia de bits 0b0000 0001 el cual representa el valor decimal de 1.

En la etapa de lectura de datos se lee a la dirección de los outputs, como se puede observar en la figura para leer cada valor (X, Y, Z) se debe leer a dos registros distintos, esto se debe a que los valores del magnetómetro son de 12 bits por lo cual se requiere de dos registros de 8 bits para su representación. A continuación, se presenta la información obtenida de correr esta secuencia de instrucciones en cinco ocasiones.

Figura 103. Datos X, Y y Z obtenidos del magnetómetro.

```

pi@raspberrypi:~/Desktop/src $ ./main
X, Y, Z: -1190.000000 | 173.000000 | -135.000000
X, Y, Z: 588.000000 | -742.000000 | -1557.000000
X, Y, Z: 1512.000000 | -1860.000000 | 882.000000
X, Y, Z: 2009.000000 | -905.000000 | 1220.000000
X, Y, Z: -735.000000 | 126.000000 | 1792.000000
pi@raspberrypi:~/Desktop/src $

```

Figura 104. Datos X, Y y Z en gauss obtenidos del magnetómetro.

```

pi@raspberrypi:~/Desktop/src $ ./main
X, Y, Z: -95.688072 | 139.174301 | 84.285713
X, Y, Z: -10.917431 | 36.605503 | -84.693878
X, Y, Z: 73.853210 | -65.963303 | 164.183670
X, Y, Z: 119.449547 | 10.000000 | 198.673477
X, Y, Z: -171.467896 | -80.917427 | 29.693878
pi@raspberrypi:~/Desktop/src $

```

A través de esta prueba se comprobó el funcionamiento del bus datos I2C de manera bidireccional debido a que se está recibiendo y mandando datos, entre el maestro OBC y el Arduino Uno simulando el magnetómetro HMC5883L que actúa como esclavo. Los datos enviados del OBC al Arduino se transmitieron sin ninguna complicación y fueron recibidos exitosamente, de la misma manera los datos enviados por el Arduino que fueron generados aleatoriamente fueron recibidos con éxito por la OBC.

Ahora, se presenta la verificación para el convertidor de señales analógico a digital ADC128D818 el cual convertirá las señales de los fotodiodos.

Figura 105. Verificación de comandos transmitidos por medio del protocolo de comunicación I2C para el chip ADC128D818.

```

COM4 (Arduino/Genuino Uno)
Ready!
REGISTER BUSY STATUS
REGISTER ADVANCED CONFIGURATION
value: 0
REGISTER CONVERSION RATE
value: 0
REGISTER CHANEL DISABLE
value: 192
REGISTER CONFIGURATION
value: 1
REGISTER BUSY STATUS
REGISTER CHANEL READING I0
REGISTER CHANEL READING I1
REGISTER CHANEL READING I2
REGISTER CHANEL READING I3
REGISTER CHANEL READING I4
REGISTER CHANEL READING I5
REGISTER CONFIGURATION
value: 8
  
```

Configuración

Lectura de datos

Autoscroll Sin ajuste de línea 9600 baudio Clear output

Para la configuración del ADC128D818, se configuró el registro de configuración avanzada en este se definió que se utilizaría el voltaje de referencia interno y que se utilizaría el modo 0 esta configuración está representada por la secuencia de bits 0b0000 0000 el cual representa el valor decimal de 0. También se configuró el registro de tasa de conversión el cual se definió en “Low Power Conversion” esta configuración está representada por la secuencia de bits 0b0000 0000 el cual representa el valor decimal de 0, se utilizó el registro de deshabilitar canales para dejar habilitados únicamente los seis canales necesarios para los fotodiodos esto por medio de la secuencia de bits 0b1100 0000 que representa el valor decimal de 192. Por último, se configuró con la secuencia de bits 0b0000 0001 el registro de configuración para habilitar el comienzo de las operaciones.

En la etapa de lectura de datos se lee a la dirección de lectura de los canales, en este caso los canales del I0 al I5 debido a que se está simulando la lectura de los seis fotodiodos. A continuación, se presenta la información obtenida de correr esta secuencia de instrucciones, en cinco ocasiones.

Figura 106. Datos I0-I5 obtenidos del convertidor de señales analógica a digital.

```

pi@raspberrypi:~/Desktop/src $ ./main
I0, I1, I2: 123.000000 | 252.000000 | 126.000000
I3, I4, I5: 84.000000 | 214.000000 | 87.000000
I0, I1, I2: 60.000000 | 18.000000 | 63.000000
I3, I4, I5: 21.000000 | 66.000000 | 24.000000
I0, I1, I2: 82.000000 | 211.000000 | 85.000000
I3, I4, I5: 43.000000 | 173.000000 | 46.000000
I0, I1, I2: 19.000000 | 233.000000 | 192.000000
I3, I4, I5: 236.000000 | 195.000000 | 68.000000
I0, I1, I2: 126.000000 | 255.000000 | 129.000000
I3, I4, I5: 87.000000 | 132.000000 | 90.000000
pi@raspberrypi:~/Desktop/src $
  
```

De la misma forma se verificó el correcto funcionamiento del bus de datos I2C para el envío y recepción de datos, en este caso entre la OBC que actúa como maestro y el Arduino Uno simulando el ADC128D818 actuando como esclavo.

Luego de evaluar su funcionamiento, al igual que con el módulo de carga útil se realizaron pruebas de respuesta para verificar que se tenga el nivel requerido por los requerimientos funcionales.

Cuadro 108. Tiempos de ejecución promedio para cada una de las actividades de ADCS.

Actividad	Toma de datos del magnetómetro	Toma de datos de los fotodiodos	Tomar de datos de ambos
Tiempo promedio (ms)	1.39136	1.97568	3.22008

El cuadro anterior muestra el promedio del tiempo de ejecución para las actividades de ADCS, este es el promedio encontrado después de la ejecución de la actividad 25 veces, para asegurar que el resultado no se ve afectado por datos extremos que puedan incumplir los requerimientos se calculó la desviación estándar.

Cuadro 109. Desviación estándar para cada una de las actividades de ADCS.

Actividad	Toma de datos del magnetómetro	Toma de datos de los fotodiodos	Tomar de datos de ambos
Desviación estándar	1.46×10^{-4}	9.68×10^{-5}	6.99×10^{-5}

Por los valores de la desviación estándar que se muestran en el cuadro anterior se puede decir que los datos están concentrados cerca de su respectiva media, lo cual nos aleja a la idea que puedan haber valores extremos que sobrepasen los límites de tiempo establecidos. El tiempo de respuesta definido por los requerimientos debía ser menor a los 300 ms para cada consulta de datos, según los tiempos promedios de ejecución estos se encuentran muy por debajo de este límite, estos no sobrepasan los 5 ms. Aunque existe un margen amplio no se puede asegurar que funcionara de igual con los componentes reales debido a que la velocidad de transmisión del I2C siempre estará limitada por la velocidad de transmisión de los otros dispositivos.

Para finalizar, debido a que el bus de datos es el que estará en constante funcionamiento cuando el satélite esté en órbita y de él depende gran parte del éxito de la misión se simuló la recepción de datos para al menos una órbita a la tierra la cual se realizará aproximadamente cada 105 minutos, tomando datos de los componentes cada 2 minutos, con el fin de verificar que los datos eran adquiridos sin complicaciones. La prueba se realizó con éxito comunicándose un total de 53 veces para cada componente. De igual manera por medio de las pruebas de estrés se determinó que el módulo de cámara de la Raspberry no es

capaz de operar de manera continua cuando se requiere tomar 25 fotografías seguidas, debido a que en las pruebas realizadas las fotografías obtenidas resultaron ser de un color totalmente negro, así mismo con el movimiento del carrusel (25 veces) se encontró que la Raspberry no es capaz de mantener el consumo de energía cuando se alimenta de 5V a 700mA provocando que está se apague, pero la misma prueba realizada con una alimentación de 5V a 1.5A fue exitosa.

Se realizaron pruebas unitarias para comprobar el funcionamiento de las funciones del bus de datos I2C correspondientes al módulo de energía y comunicaciones, la siguiente figura muestra los resultados de estas pruebas.

Figura 107. Pruebas unitarias del bus de datos I2C.

```

pi@raspberrypi:~/Desktop $ ./tests
Raspberry Pi I2C
Test unitario - Conectar esclavo: Success
Test unitario - Escribir registros a archivo de texto: Success
Test unitario - Obtener estado de carga: Success
Test unitario - Validar estado de carga: Success
Test unitario - Almacenar estado de carga: Success
Test unitario - Obtener voltaje de las baterias: Success
Test unitario - Validar voltaje de las baterias: Success
Test unitario - Almacenar voltaje de las baterias: Success
Test unitario - Obtener temperatura de las baterias: Success
Test unitario - Validar temperatura de las baterias: Success
Test unitario - Almacenar temperatura de las baterias: Success
Test unitario - Enviar imagen: Success
Test unitario - Enviar registros: Success
13 tests exitosos de 13 realizados
0 tests fallidos de 13 realizados
pi@raspberrypi:~/Desktop $

```

Para el cumplimiento de los requisitos del sistema se siguió la arquitectura planteada para realizar las pruebas con los módulos de energía y comunicaciones. Los componentes del módulo de energía se simularon con un Arduino Uno porque las pruebas con los componentes reales quedaban fuera del alcance de este trabajo, lo que realmente se quiso probar es que la computadora a bordo del satélite utilice de manera adecuada el bus de datos, interprete correctamente los datos, los valide y los almacene. Las pruebas integradoras con los componentes reales requerían la creación de una placa y la calibración de los componentes. Sin embargo, las simulaciones de los componentes se programaron de tal forma que el Arduino Uno respondiera con los valores y el formato que especifica la hoja de datos de cada componente para que las pruebas se apeguen lo más posibles a las condiciones reales del satélite. El módulo de comunicaciones tiene definido su componente, pero no se obtuvo acceso a la información del formato de las instrucciones para su comunicación con el bus de datos, por lo tanto, las pruebas con este módulo fueron más de transferencia de la información por el bus de datos y no del funcionamiento específico del componente.

Según simulaciones que se realizaron por miembros del equipo, tomando en cuenta la órbita del satélite, se estimó que el satélite se tomará una hora y treinta minutos aproximadamente para completar una órbita.

Con el fin de validar el funcionamiento de los demás módulos y en base a los requerimientos planteados, la prueba de captura de información del módulo de energía durante este tiempo se realizó exitosamente, como se observa en las siguientes figuras donde se graficaron los valores almacenados. El componente encargado de medir la temperatura de la batería tiene un muestreo más seguido porque se requiere monitorear constantemente este valor con el fin de implementar un mecanismo de calentamiento de las baterías si su temperatura se acerca a los 5 °C. En el presente trabajo no se cubrió la validación de la temperatura al llegar a este margen crítico porque todavía no está definido el mecanismo de calentamiento de las baterías ni los valores estimados en los cuales se debe accionar.

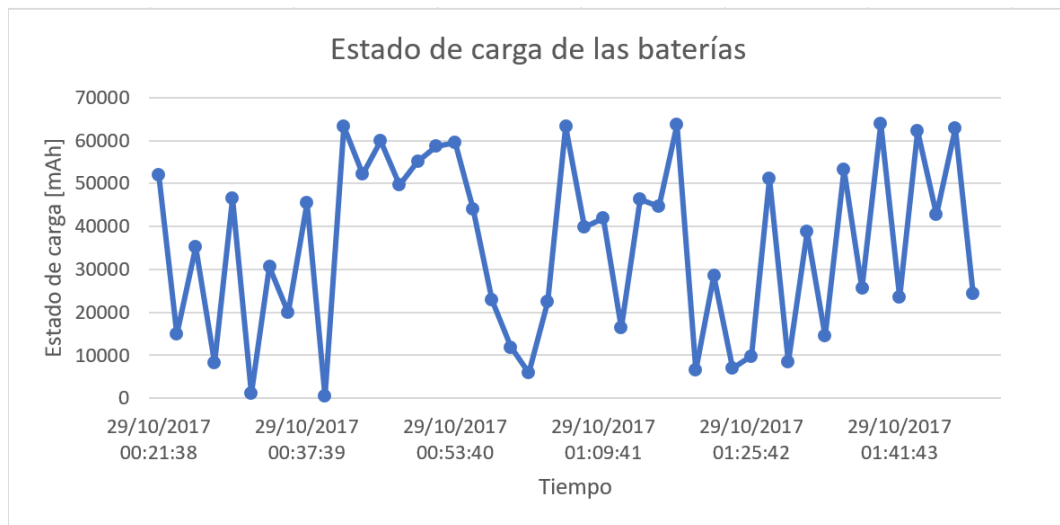
Para almacenar los valores mencionados se utilizaron las siguientes estructuras de datos.

Figura 108. Estructuras de datos.

temperatura_bateria	estado_de_carga_bateria	voltaje_bateria
+ id[3]: char	+ id[3]: char	+ id[3]: char
+ time_stamp[26]: char	+ time_stamp[26]: char	+ time_stamp[26]: char
+ valor: float	+ valor: float	+ valor: float

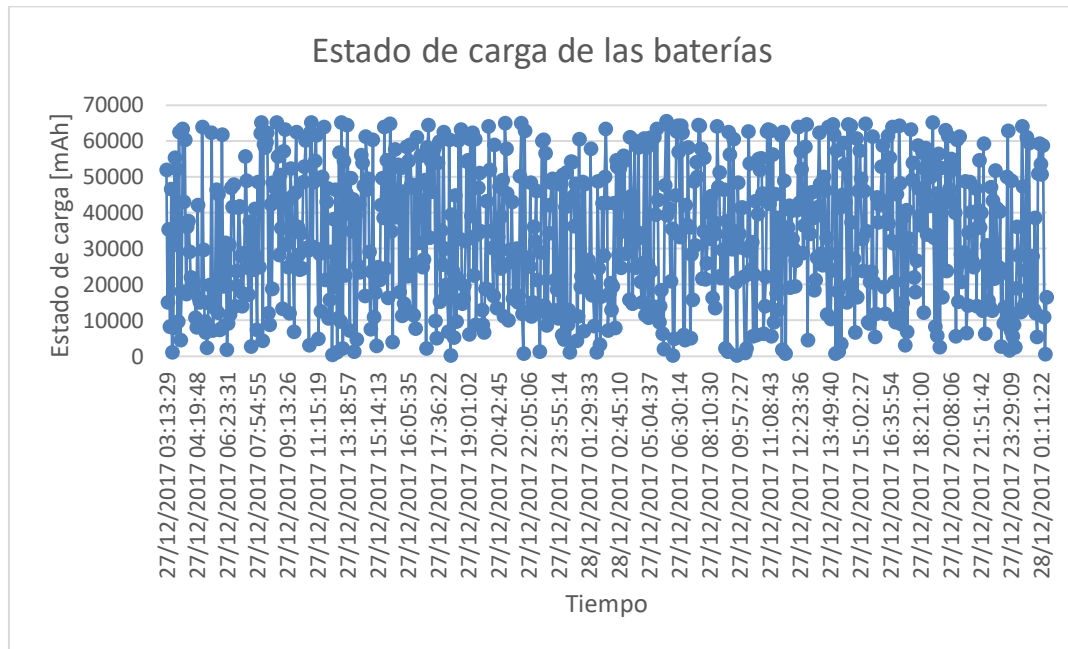
Las mediciones cada 2 minutos del estado de carga de las baterías durante 1 hora y 30 minutos se muestran en la siguiente figura.

Figura 109. Mediciones durante 1 hora y 30 minutos de estado de carga de las baterías.



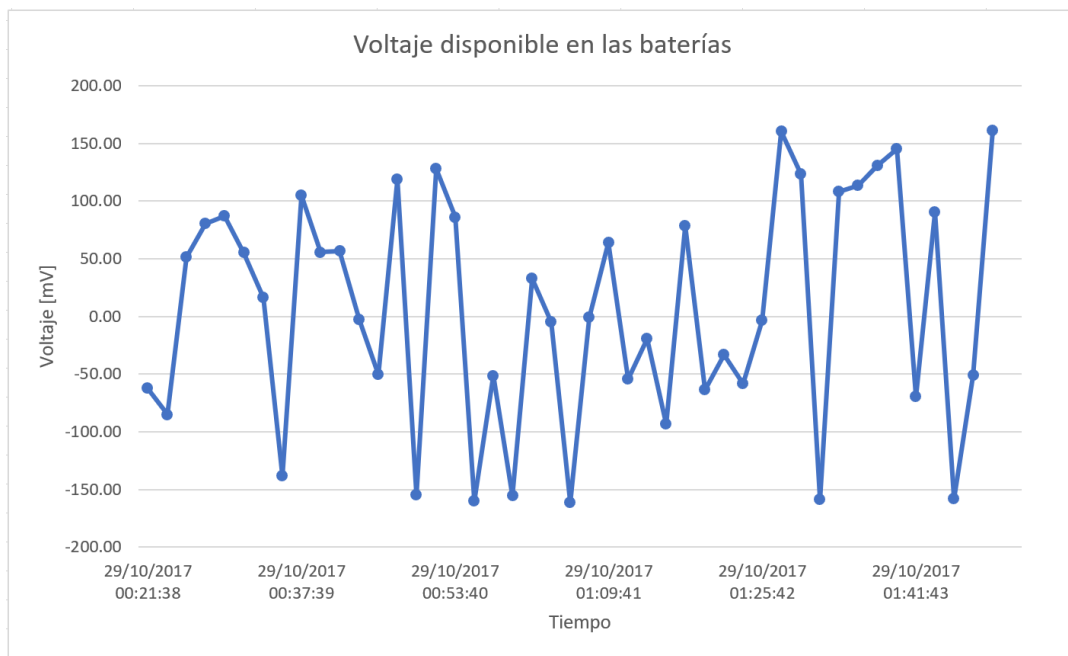
Las mediciones cada 2 minutos del estado de carga de las baterías durante 24 horas se muestran en la siguiente figura.

Figura 110. Mediciones durante 24 horas de estado de carga de las baterías.



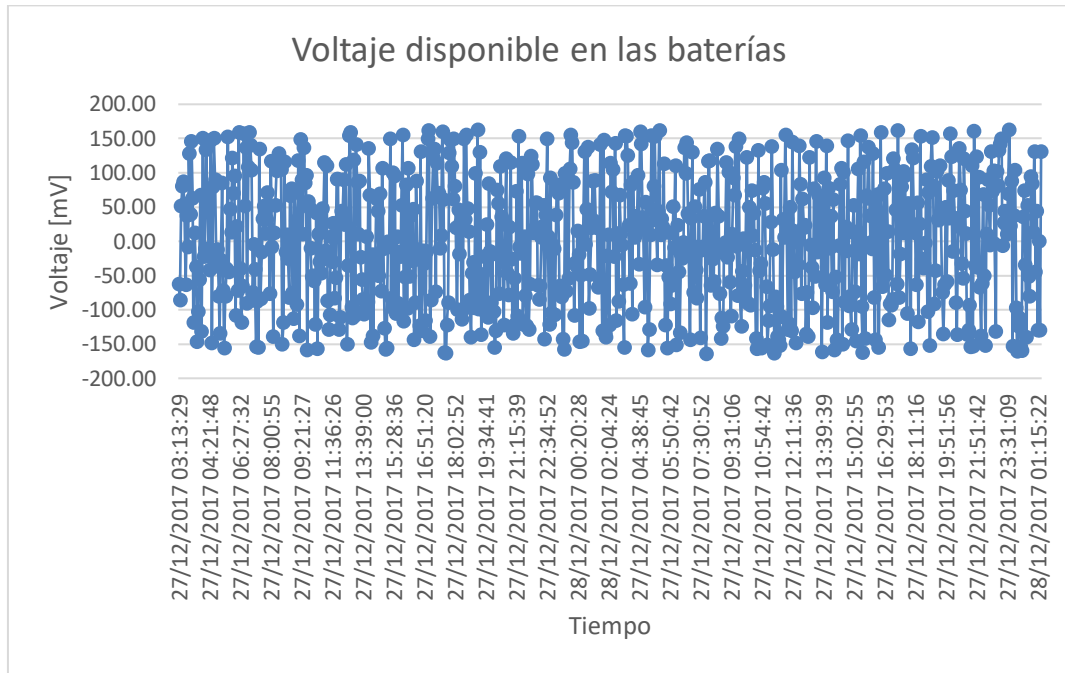
Las mediciones cada 2 minutos del voltaje disponible en las baterías durante 1 hora y 30 minutos se muestran a continuación.

Figura 111. Mediciones durante 1 hora y 30 minutos del voltaje disponible en las baterías.



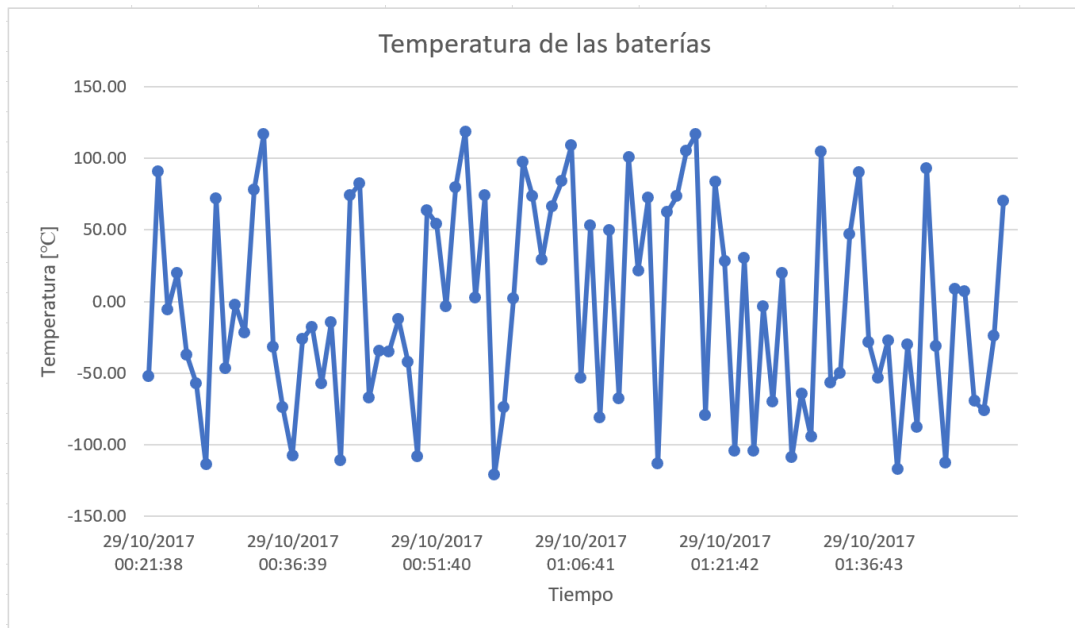
Las mediciones cada 2 minutos del voltaje disponible en las baterías durante 24 horas se muestran a continuación.

Figura 112. Mediciones durante 24 horas del voltaje disponible en las baterías.



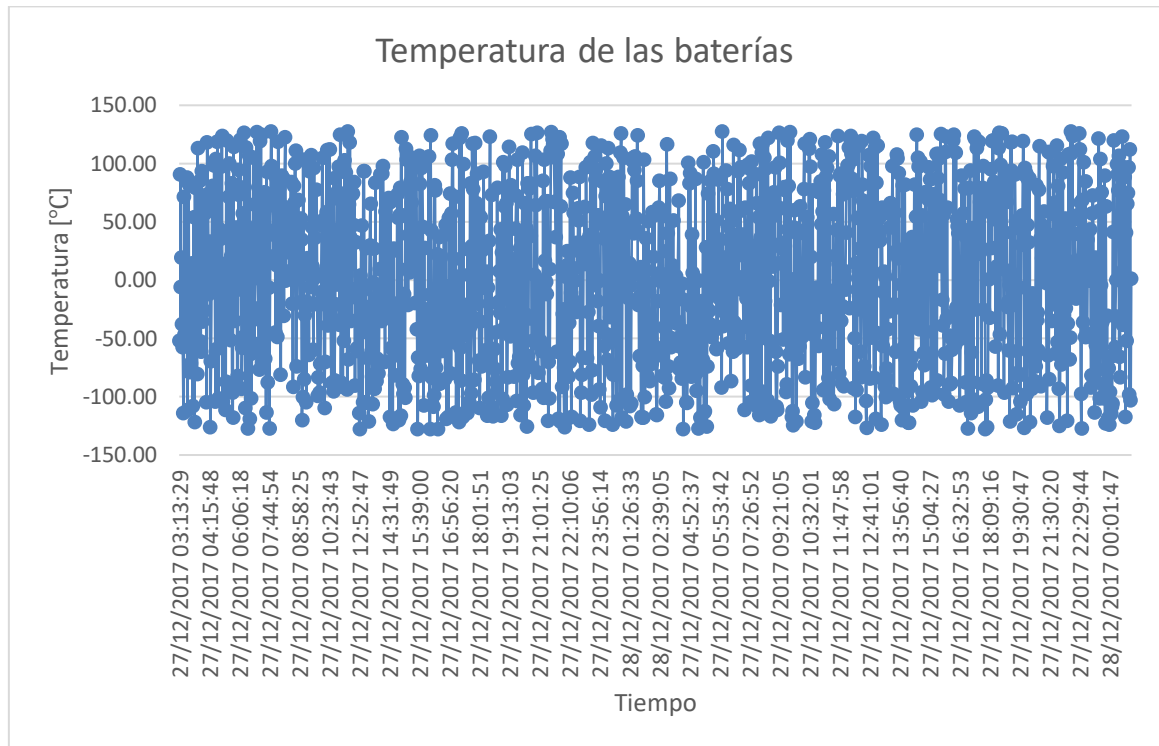
Las mediciones cada minuto de la temperatura de las baterías durante 1 hora y 30 minutos se muestran en la siguiente figura.

Figura 113. Mediciones durante 1 hora y 30 minutos de la temperatura de las baterías.



Las mediciones cada minuto de la temperatura de las baterías durante 24 horas se muestran en la siguiente figura.

Figura 114. Mediciones durante 24 horas de la temperatura de las baterías.



El tiempo de envío de la información almacenada de los tres componentes del módulo de energía durante el tiempo aproximado de una órbita del satélite y el tiempo de envío de dos imágenes se muestran en el siguiente cuadro.

Cuadro 110. Tiempos de envío de información al módulo de comunicaciones.

Datos	Tamaño [KiloBytes]	Tiempo [s]
Estado de carga de las baterías, voltaje almacenado en las baterías y temperatura de las baterías.	6.61	17
Imagen con resolución de 720x480	337.5	158
Imagen con resolución de 1280x720	900	562

Figura 115. Tiempo de envío de datos del módulo de energía.

```

pi@raspberrypi:~/Desktop $ ./master
Raspberry Pi I2C
Esclavo 8 conectado
Conectando con modulo de comunicaciones
Resuesta del modulo de comunicaciones: 00000001
Enviando 6.61 KBytes de logs
At time: 29/10/2017 19:25:03
Transmision finalizada
At time: 29/10/2017 19:25:20

```

Figura 116. Tiempo de envío de imagen de 337.5 KBytes.

```

pi@raspberrypi:~/Desktop $ gcc -o master rpi_i2c_master.c -std=c99 -lm -w
pi@raspberrypi:~/Desktop $ ./master
Raspberry Pi I2C
Esclavo 8 conectado
Conectando con modulo de comunicaciones
Resuesta del modulo de comunicaciones: 00000001
Enviando imagen de 337.5 KBytes
At time: 29/10/2017 18:41:40
Transmision finalizada
At time: 29/10/2017 18:44:18

```

Figura 117. Tiempo de envío de imagen de 900 KBytes.

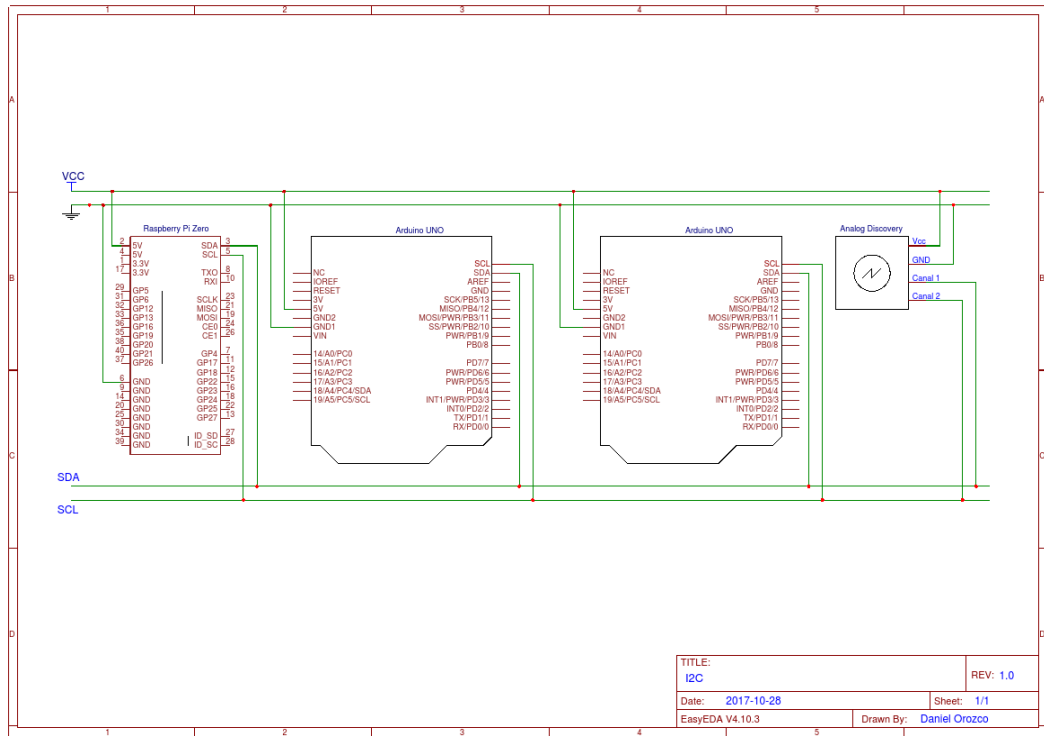
```

pi@raspberrypi:~/Desktop $ gcc -o master rpi_i2c_master.c -std=c99 -lm -w
pi@raspberrypi:~/Desktop $ ./master
Raspberry Pi I2C
Esclavo 8 conectado
Conectando con modulo de comunicaciones
Resuesta del modulo de comunicaciones: 00000001
Enviando imagen de 900 KBytes
At time: 29/10/2017 19:14:20
Transmision finalizada
At time: 29/10/2017 19:23:42

```

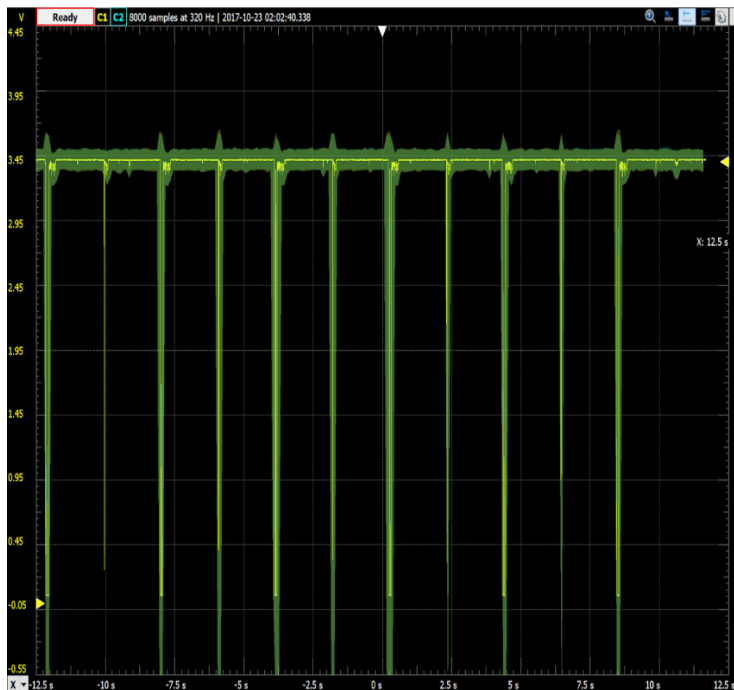
Para comprobar el funcionamiento del bus de datos se utilizó un osciloscopio y analizador lógico de señales. Se utilizaron los métodos definidos en la arquitectura de software, específicamente las que obtienen información de los componentes del módulo de energía. Con estos métodos se realizaron mediciones con los instrumentos y se obtuvieron gráficas del comportamiento de las señales del bus de datos. El ambiente con los instrumentos utilizados para estas mediciones se muestra en la siguiente figura.

Figura 118. Mediciones con osciloscopio del bus de datos.



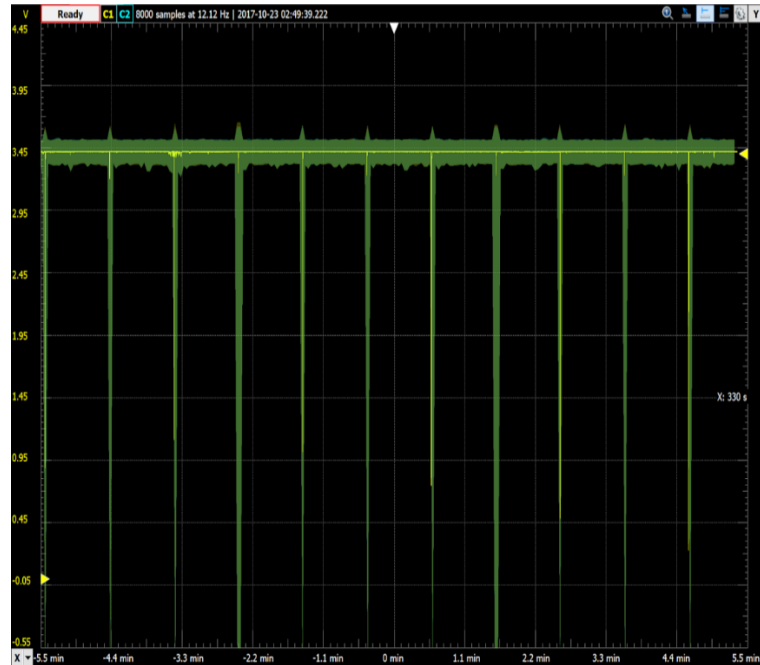
La siguiente figura contiene las mediciones realizadas con el osciloscopio para una prueba de comunicación que duró 20 segundos.

Figura 119. Mediciones del bus de datos por 20 segundos.



La siguiente figura contiene las mediciones realizadas con el osciloscopio para una prueba de comunicación que duró 10 minutos.

Figura 120. Mediciones del bus de datos por 10 minutos.



Las siguientes figuras contienen una medición realizada con el osciloscopio para observar el cambio de las señales en la transmisión de datos. En las primeras dos figuras se realizó una lectura, en la última se realizaron tres lecturas.

Figura 121. Medición de una lectura del bus con escala de tiempo de 5 ms.



Figura 122. Medición de una lectura del bus de datos con escala de tiempo de 20 ms.

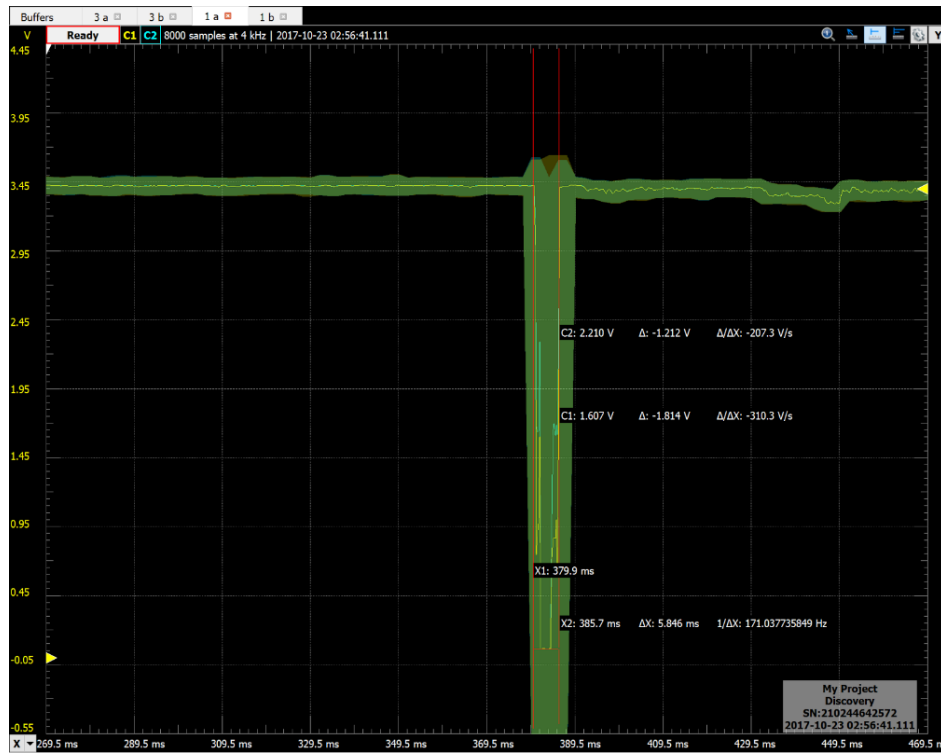
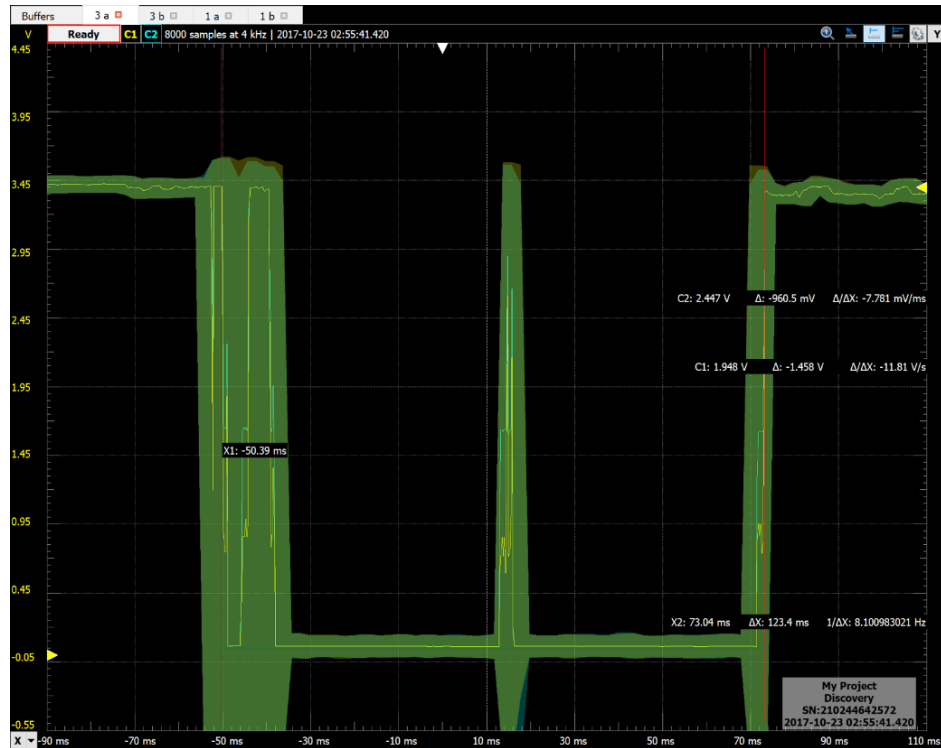


Figura 123. Medición de tres lecturas del bus de datos.



El siguiente cuadro contiene la cantidad de registros que se almacenaron durante 24 horas y la cantidad teórica de registros que debieron ser almacenados.

Cuadro 111. Cantidad de mediciones del bus de datos durante 24 horas

Componente	Cantidad de registros teórica	Cantidad de registros práctica	Porcentaje de registros almacenados
BQ28Z610	720	705	97.9%
INA3221	720	705	97.9%
TMP101	1440	1410	97.9%

Durante el periodo de tiempo de las pruebas no surgió ningún error que provocara la modificación en los datos por parte del bus de datos y se transmitieron un 97% de los valores satisfactoriamente. El comportamiento del bus de datos en la comunicación con los componentes se midió utilizando un osciloscopio y analizador lógico de señales. Las mediciones no se realizaron sobre la prueba de una hora y treinta minutos porque el instrumento de medición tenía limitaciones en cuanto a la frecuencia de mediciones. Como se observa en las figuras 121 y 122 el tiempo en el cual se reporta el valor más bajo en el voltaje de la señal del bus de datos es entre 5 y 6 milisegundos. Al hacer las mediciones sobre una hora y media la frecuencia que establecía el osciloscopio brindaba mediciones cada segundo, por lo tanto no se apreciaba una caída significativa de voltaje cuando se realizaba la transmisión de datos. Este fenómeno se puede apreciar en las figuras 119 y 120 porque para las pruebas de mediciones durante 10 minutos el osciloscopio brindaba mediciones cada 80 milisegundos, y para la prueba de 20 segundos cada 3.12 milisegundos. En ambas figuras se puede observar la diferencia que existe en los valores que se reportan en la caída de la señal para el voltaje, para la prueba de 10 minutos la señal no reporta valores tan bajos. Esto no significa que el bus de datos no está funcionando, al contrario, las pruebas reportan que los datos fueron transmitidos exitosamente con el bus de datos sin ninguna pérdida o alteración.

Las siguientes figuras muestran el funcionamiento de las dos Raspberry Pi que componen el método de redundancia.

Figura 124. Método de redundancia para la primera Raspberry Pi.

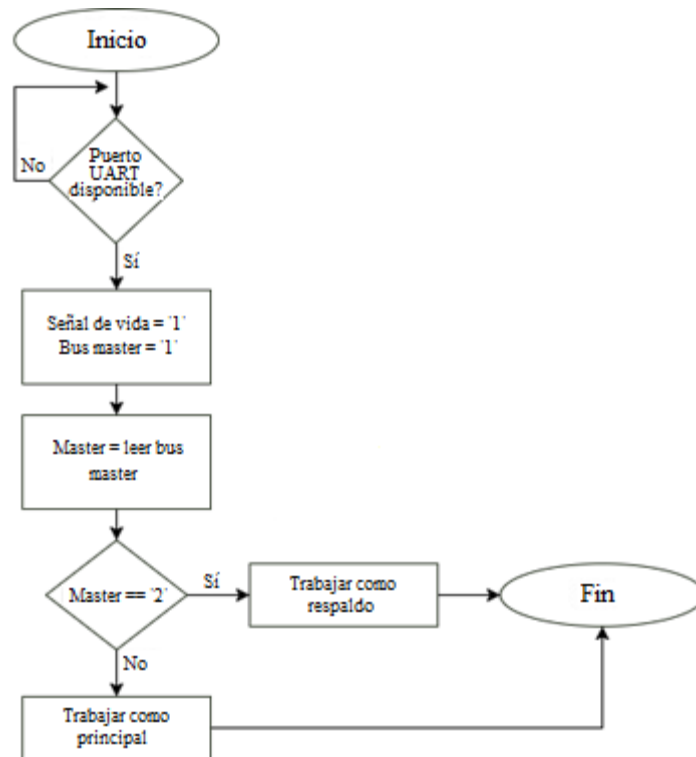
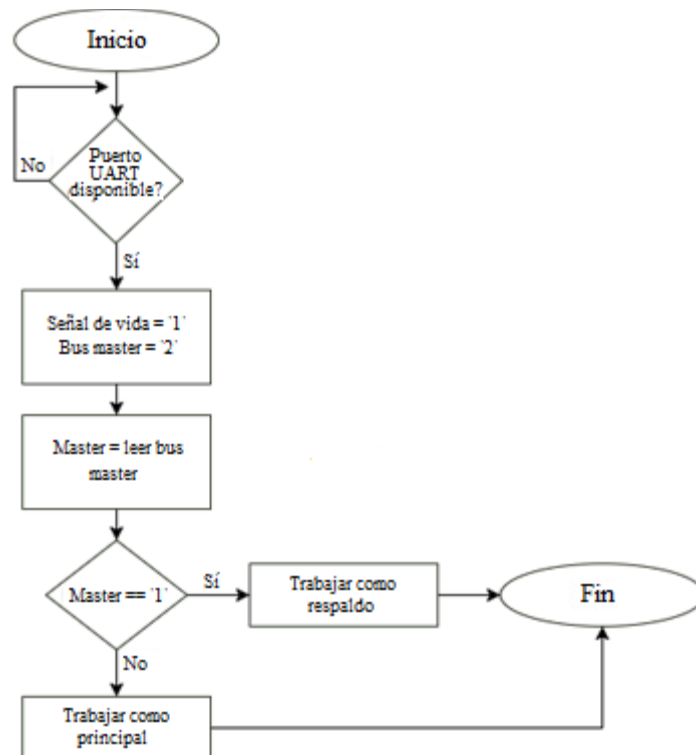
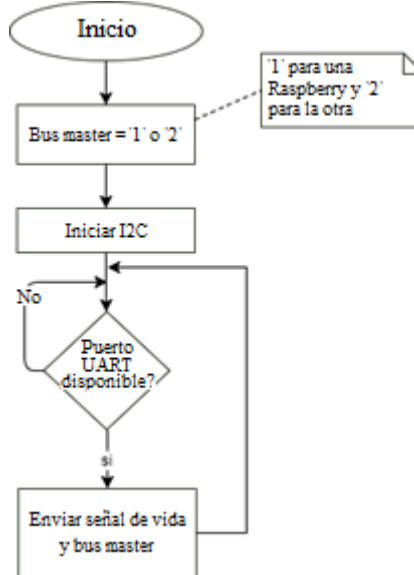


Figura 125. Método de redundancia para la segunda Raspberry Pi.



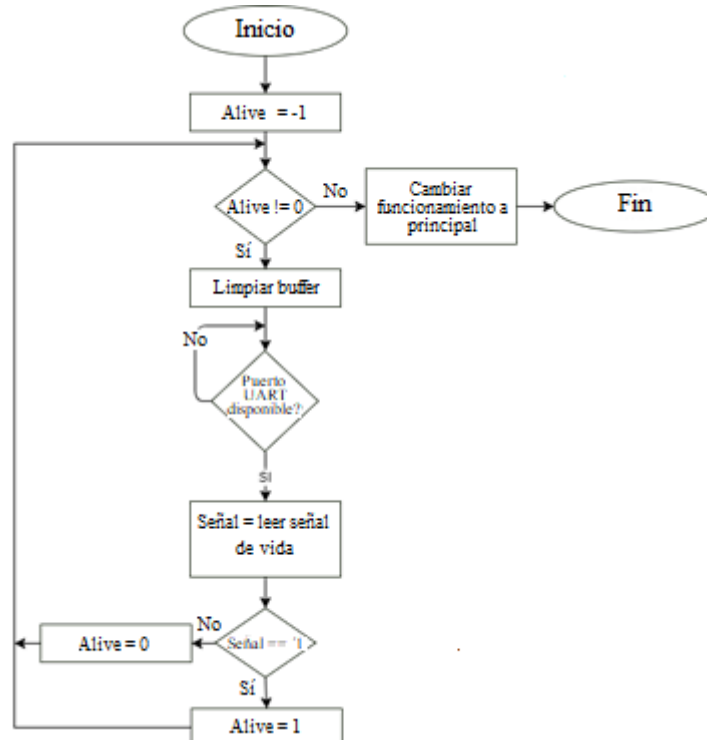
La siguiente figura muestra el funcionamiento del método de redundancia para la computadora principal.

Figura 126. Diagrama de flujo para computadora principal del método de redundancia.



La siguiente figura muestra el funcionamiento del método de redundancia para la computadora de respaldo.

Figura 127. Diagrama de flujo para computadora de respaldo del método de redundancia.



Se realizaron pruebas unitarias para comprobar el funcionamiento del método de redundancia, la siguiente figura muestra el resultado de estas.

Figura 128. Pruebas unitarias del mecanismo de redundancia.

```

pi@raspberrypi:~/Desktop $ gcc -o tests_redundancia tests_r
pi@raspberrypi:~/Desktop $ ./tests_redundancia
Test unitario - Conectar UART: Success
Test unitario - Enviar señal: Success
Test unitario - Leer señal: Success
Test unitario - Encender LED: Success
Test unitario - Apagar LED: Success
Test unitario - Activar I2C: Success
Test unitario - Determinar estado: Success
Test unitario - Cambiar estado: Success
8 tests exitosos de 8 realizados
0 tests fallidos de 8 realizados
pi@raspberrypi:~/Desktop $

```

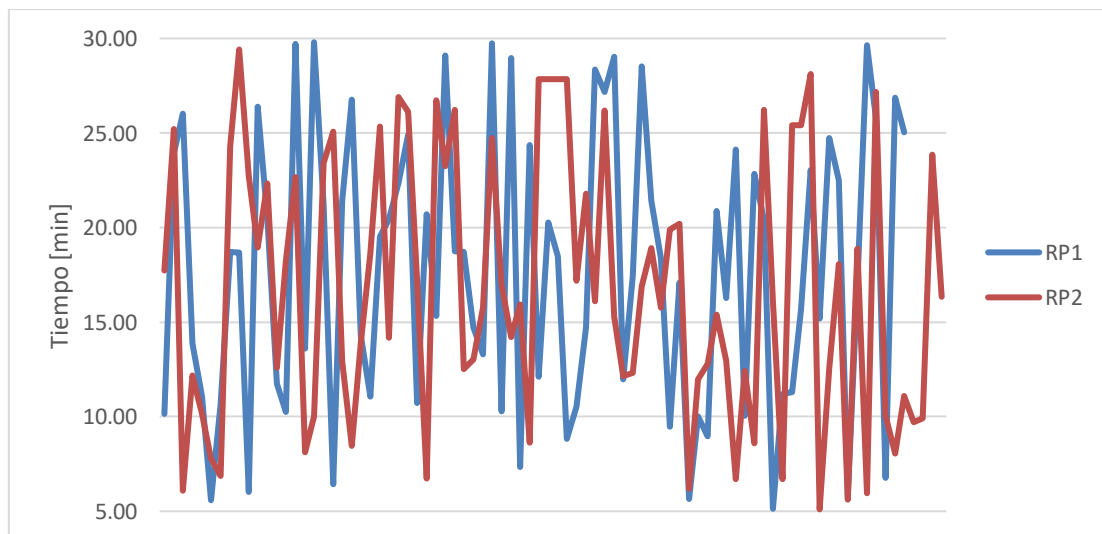
El siguiente cuadro contiene la cantidad de veces que se reiniciaron las computadoras en las pruebas de tiempo prolongado y el tiempo promedio que operaron hasta reiniciarse.

Cuadro 112. Cantidad de reinicios de computadoras y tiempos de operación.

Computadora	Cantidad de reinicios	Tiempo de operación [m]
Raspberry Pi 1	80	17.70
Raspberry Pi 2	84	16.83

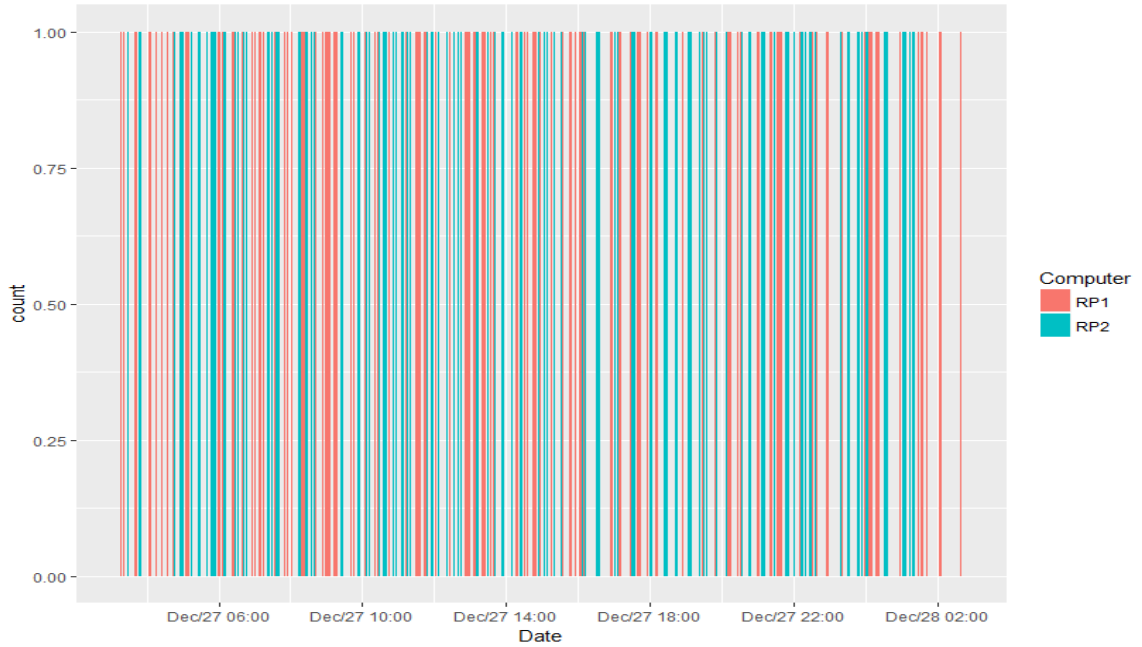
La siguiente figura muestra la comparación entre los tiempos de operación antes de reiniciarse de las dos computadoras.

Figura 129. Tiempos de operación antes de reinicio.



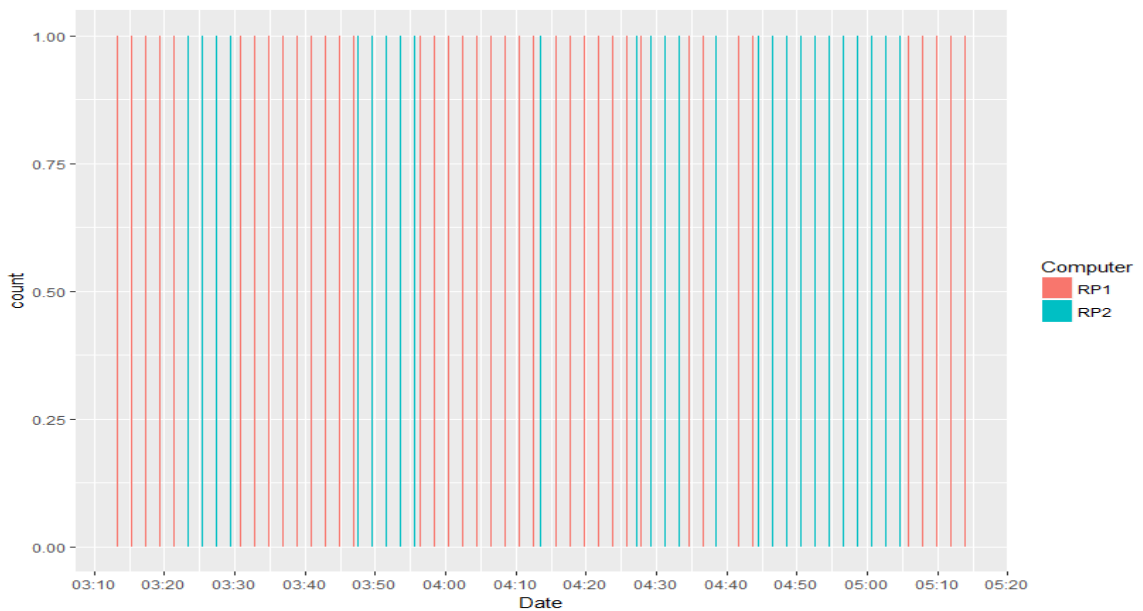
Cuando se analizaron los archivos que almacenan los registros obtenidos de las lecturas del bus de datos, se observó el comportamiento de las computadoras durante las pruebas de tiempo prolongado, como lo muestra la siguiente figura.

Figura 130. Funcionamiento de computadoras en pruebas de tiempo prolongado.



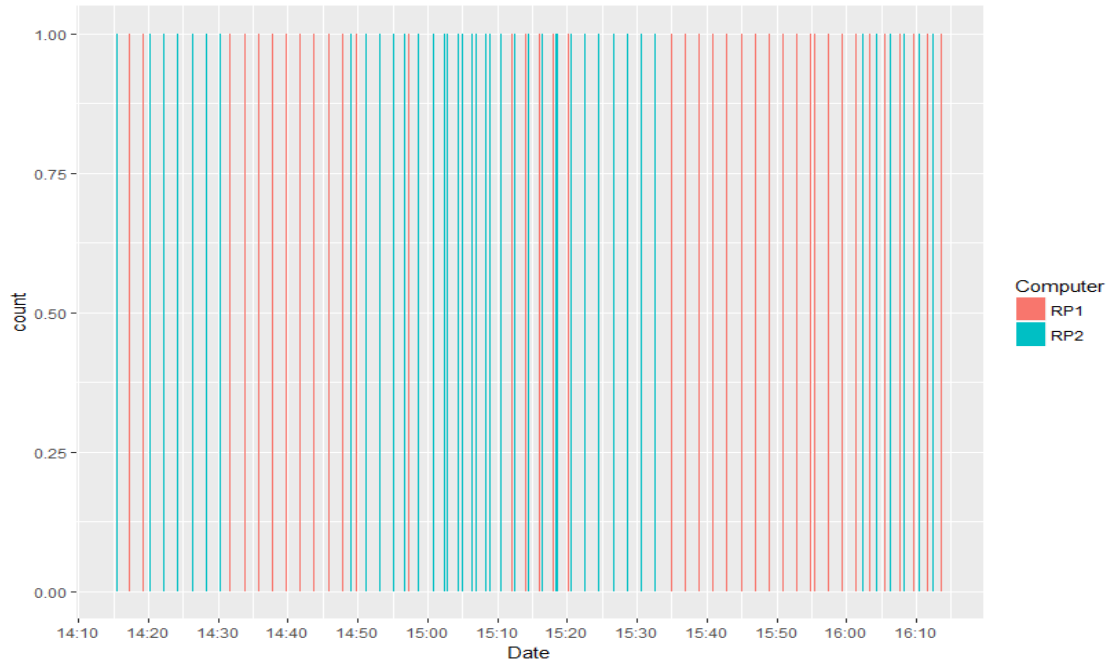
La siguiente figura muestra el funcionamiento de las computadoras en las primeras dos horas de las pruebas de tiempo prolongado.

Figura 131. Funcionamiento de redundancia en las primeras dos horas.



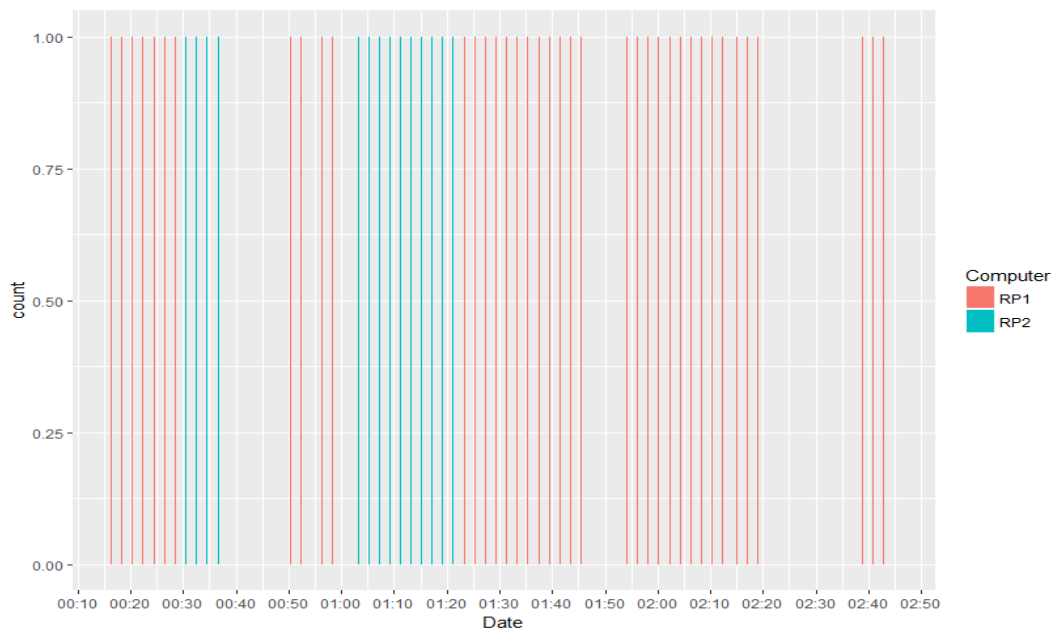
La siguiente figura muestra el funcionamiento de las computadoras por dos horas en la mitad de las pruebas de tiempo prolongado.

Figura 132. Funcionamiento de redundancia a mitad de la prueba de tiempo prolongado.



La siguiente figura muestra el funcionamiento de las computadoras en las últimas tres horas de las pruebas de tiempo prolongado.

Figura 133. Funcionamiento de redundancia en las últimas tres horas.



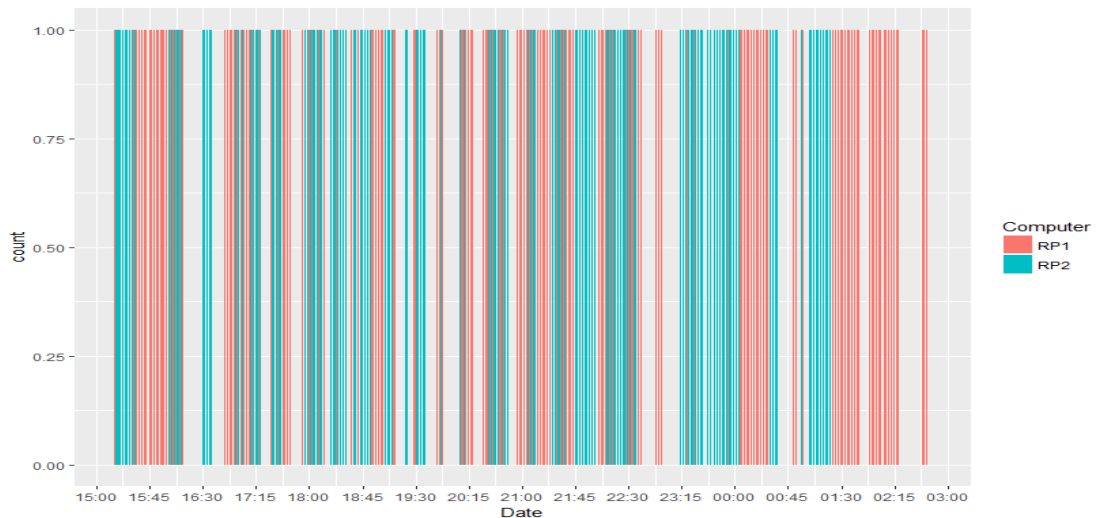
La siguiente figura muestra el funcionamiento de las computadoras en la primera mitad de las pruebas de tiempo prolongado.

Figura 134. Funcionamiento de redundancia en las primeras doce horas.



La siguiente figura muestra el funcionamiento de las computadoras en la segunda mitad de las pruebas de tiempo prolongado.

Figura 135. Funcionamiento de redundancia en las últimas doce horas.



Se comprobó que el método de redundancia funciona correctamente ya que en las pruebas de tiempo prolongado el bus de datos no se saturó o dejó de funcionar la comunicación para obtener los valores de los componentes. El mecanismo permitió que las computadoras interpretaran correctamente el estado del sistema y que operaran de acuerdo con esto. Al analizar los registros de operaciones de las computadoras se observó que en las últimas horas de ejecución los valores reportados presentan mucha diferencia en las

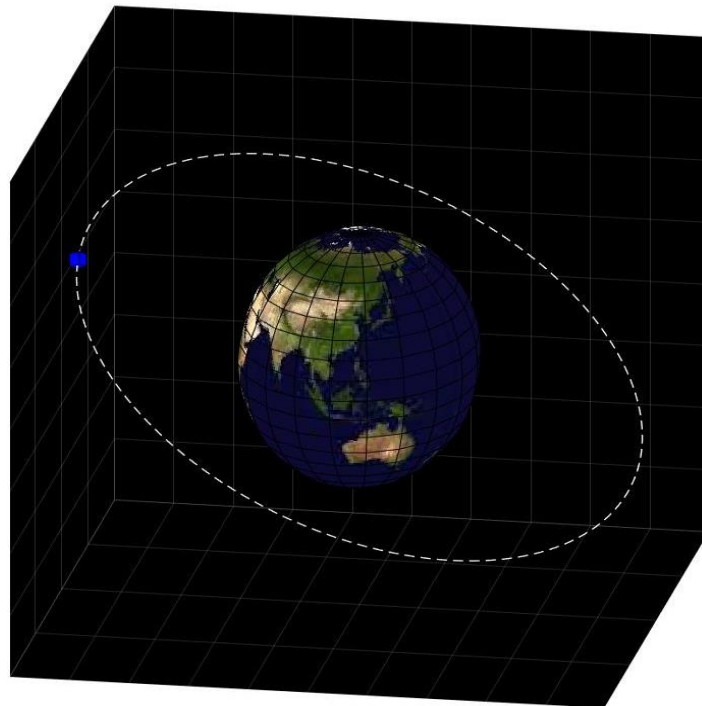
horas de operación y existen varias medidas realizadas por ambas computadoras en un mismo periodo de tiempo, esto no es indicador que el mecanismo estaba funcionando incorrectamente, ya que de haber sido así el bus de datos se hubiera detenido y las mediciones no hubieran continuado (como se comprobó en las pruebas de errores introducidos cuando actúan dos computadoras como principales). La razón de esto es que la Raspberry Pi no cuenta con un reloj integrado, este componente es el encargado, entre otras cosas, de mantener la hora de la computadora actualizada aun cuando este apagada. Por lo tanto, al no tener un reloj integrado cada vez que se reinicia la computadora el tiempo que maneja se ve afectado con retrasos.

A pesar de que el mecanismo de redundancia funcionó correctamente, este no cumple con el requisito de consumo energético del satélite. Al medir este consumo en las operaciones resultó que el sistema completo con ambas computadoras necesita 0.56 W para su funcionamiento. Esto sobrepasa en 0.16 W el máximo asignado para el módulo de manejo de comandos y datos. Para que este mecanismo se pueda implementar sin correr riesgos de no tener energía suficiente, es necesario disponer de al menos 0.56 W para el módulo o de implementar computadoras con un menor consumo energético.

J. SIMULACIÓN DE ORIENTACIÓN DEL SATÉLITE

Como resultado de la simulación se obtuvo una animación para observar cómo se logra estabilizar el sistema después de cierto tiempo en el espacio.

Figura 136. Resultado final de la simulación, animación del planeta con un satélite con cambios de orientación.



K. ESTABILIDAD DEL SISTEMA

Para encontrar la estabilidad del sistema utilizando el método directo de Lyapunov, se debe tener un candidato que ayude a comprobar cómo cambia la energía del sistema. Debido a que la parte que interesa es la rotacional y el cambio de energía, se puede utilizar como candidato la energía cinética rotacional.

$$V(x) = \frac{1}{2} \omega^T I_{sat} \omega \quad (73)$$

Por lo tanto, si se reemplazan los vectores de inercia y velocidades angulares en la ecuación (73) se obtiene:

$$V(\omega) = \frac{1}{2} [\omega_x \quad \omega_y \quad \omega_z] \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}$$

$$V(\omega) = \frac{1}{2} (\omega_x^2 I_x + \omega_y^2 I_y + \omega_z^2 I_z) > 0 \quad \forall \omega \neq 0 \quad (74)$$

La ecuación (74) demuestra que el candidato demuestra el primer requisito de Lyapunov para comprobar la estabilidad del sistema por lo que el siguiente paso es realizar el cálculo de la derivada.

$$\dot{V}(\omega) = \frac{\delta V}{\delta \omega} f(\omega) \quad (75)$$

Donde $f(\omega)$ son las ecuaciones de Euler (31), despejadas para $\dot{\omega}$. Sin tomar en cuenta cualquier acción externa a la propia del satélite, es decir, torques de control ni perturbaciones.

$$\dot{V}(\omega) = [\omega_x \quad \omega_y \quad \omega_z] \begin{bmatrix} \frac{1}{I_x} (I_y - I_z) \omega_y \omega_z \\ \frac{1}{I_y} (I_z - I_x) \omega_z \omega_x \\ \frac{1}{I_z} (I_x - I_y) \omega_x \omega_y \end{bmatrix}$$

$$\dot{V}(\omega) = (I_x - I_z) \omega_x \omega_y \omega_z + (I_y - I_z) \omega_x \omega_y \omega_z + (I_z - I_x) \omega_x \omega_y \omega_z = 0 \quad (76)$$

El resultado de la ecuación (76) no demuestra nada sobre el sistema más que es estable. No obstante, existe la posibilidad de integrar la acción de control en el sistema y así comprobar si el control ayudaría a encontrar si el sistema es globalmente asintóticamente estable.

$$\dot{V}(\omega) = -k_x \omega_x^2 - k_y \omega_y^2 - k_z \omega_z^2 < 0 \quad (77)$$

Al obtener este resultado se demuestra que el sistema es asintóticamente estable, por lo que se debe comprobar el límite al infinito.

$$\lim_{\omega \rightarrow \infty} V(\omega) = \infty$$

(78)

A través de las ecuaciones (74), (77) y (78) se demuestra que el sistema es globalmente asintóticamente estable (G.A.S.).

L. CONTROLADOR PID

El controlador PID fue utilizado solamente para controlar la velocidad angular del satélite. Esto presentó una desventaja debido a que se debe trabajar de manera desacoplada las variables de estado del sistema. Esto es un problema debido a que se tendrá siempre varios sistemas de una entrada y una salida desacoplados que no permite visualizar de manera directa como le afecta un eje al otro. A continuación, se presentan los resultados de velocidades angulares, torques de control y de evolución de cuaterniones.

Figura 137. Estados de los cuaterniones del sistema con un controlador PID.

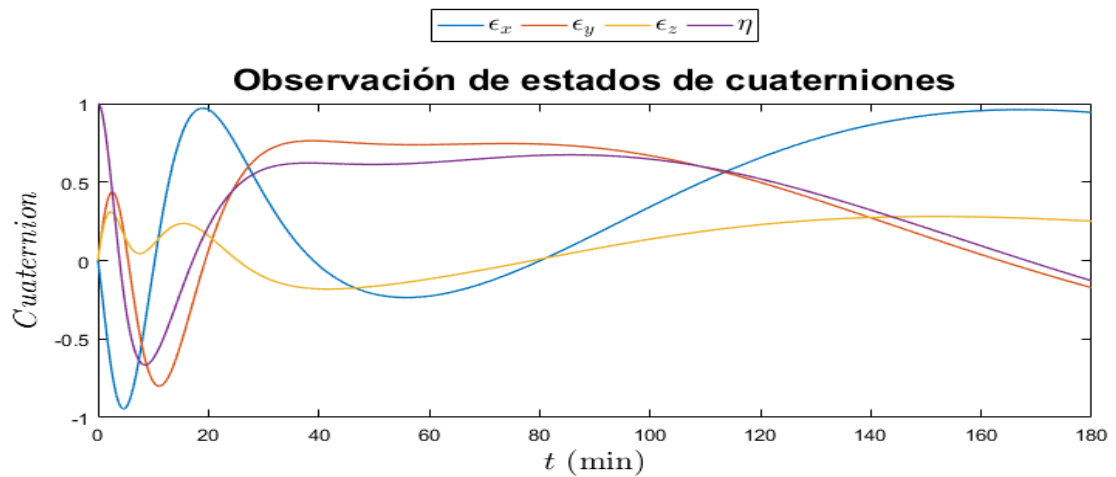
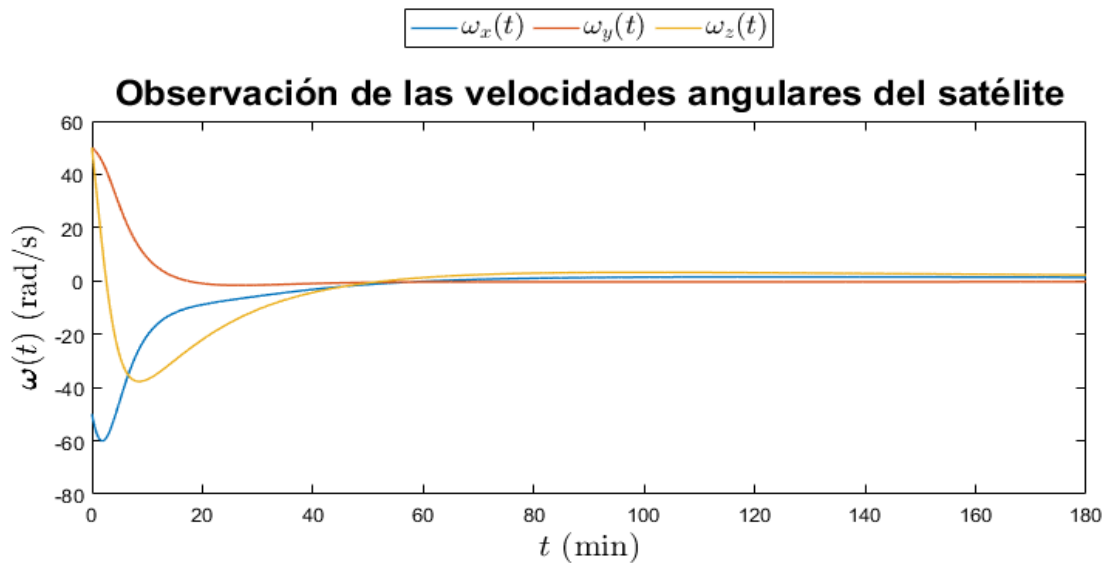
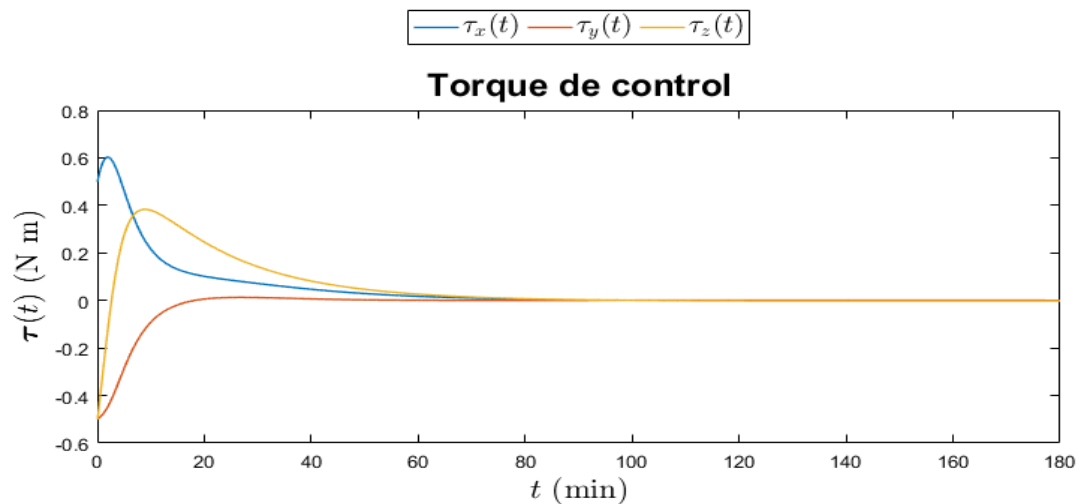


Figura 138. Velocidades angulares del sistema con un controlador PID.



Se puede observar que se obtiene un resultado bastante bueno con respecto a las velocidades angulares del sistema debido a que convergen a 0 o al menos a un valor bastante aproximado a 0. Esto quiere decir que el control está actuando de la manera correcta y está extrayendo la energía del sistema. El tiempo en estas simulaciones realmente es menor debido a que por cuestiones de simulación se colocó en minutos la escala cuando realmente cada un minuto representa 500 microsegundos. En el caso de los cuaterniones, se puede observar que realmente no se obtiene el comportamiento deseado, por lo que el satélite no se queda en la posición de referencia. Esto es porque como en el control PID están todas las entradas y salidas desacopladas no se colocó un control que tomara en cuenta tanto la velocidad como orientación por lo que ese recurso no se está tomando en cuenta porque no funcionaría adecuadamente.

Figura 139. Torques de control obtenidos para cada eje con un controlador PID.



Por último, se puede observar que los torques de control no tienen una magnitud muy grande, lo que permite que sí se pueda implementar ese control. Esto es importante porque si se necesitaran torques muy grandes las ruedas de inercia o electroimanes deberían ser más grandes y luego de cierta magnitud ya no caben dentro del satélite o tampoco son fáciles de encontrar en el mercado. No obstante, un aspecto importante que se encontró es que el control PID realmente terminó siendo un PI debido a que la parte derivativa vuelve inestable al sistema.

M. CONTROLADOR REGULADOR LINEAL CUADRÁTICO (LQR)

El control LQR resultó ser un experimento interesante para el satélite. El primer paso fue obtener la matriz de constantes para controlar el sistema y acoplar una variable de estado con otras e ingresar ese control al modelo no lineal. La matriz K fue la siguiente.

$$K = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Esta matriz representa la forma que se deben acoplar las variables de estado de la manera más eficiente para poder llegar a controlar el satélite. Esto quiere decir lo siguiente:

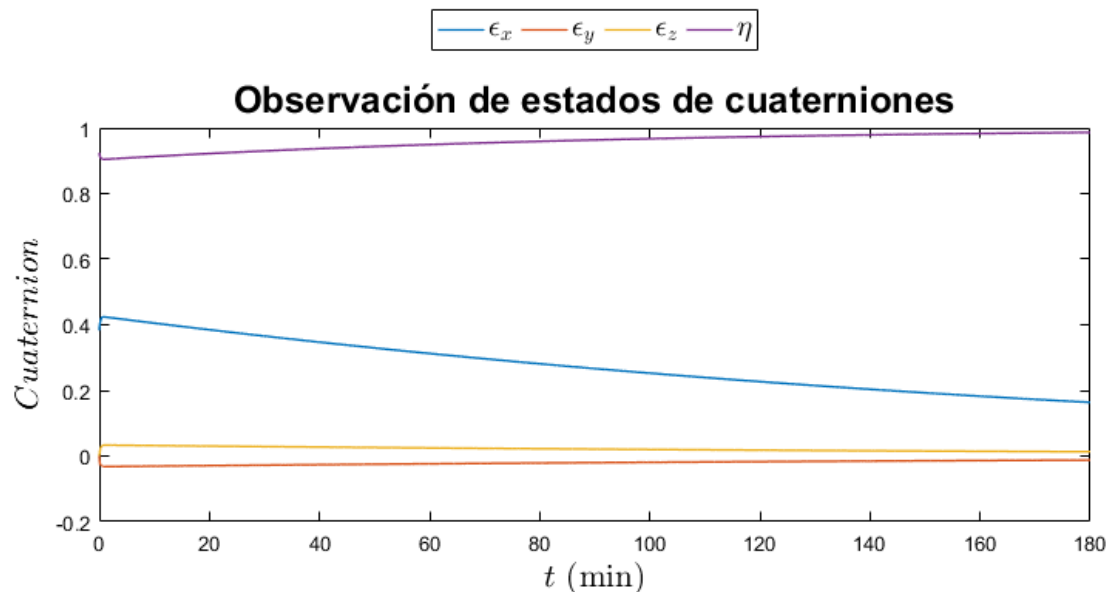
$$u = -kx \quad (79)$$

Donde x son las variables de estado, es decir los cuatro cuaterniones y las velocidades angulares. Esto daría como resultado que u se convierta en una combinación del cuaternión que describe el ángulo en un eje y la velocidad angular que describe como cambia ese ángulo, por consiguiente, esto se convierte en una solución de la siguiente forma:

$$u = -K_p \epsilon - K_d \omega \quad (80)$$

Esta expresión es importante porque demuestra que el control ideal para controlar el satélite es un PD, porque la parte proporcional es con el ángulo, es decir el cuaternión, y la parte derivativa es como cambia ese, es decir, la velocidad angular. Esto simplifica la solución al problema de control. A continuación, se presentan los resultados del control LQR.

Figura 140. Estado de cuaterniones con control LQR sin penalización.



Se puede observar que los cuaterniones en esta ocasión si llegan a cumplir los valores esperados. Esto quiere decir que el control no solo está haciendo que se reduzca la velocidad angular de cada eje sino

también está llegando a la referencia que se había establecido con el cuaternión de error. Es por esto que el término η llega a 1 y los otros tres valores convergen a 0.

Figura 141. Velocidades angulares del sistema con control LQR sin penalización.

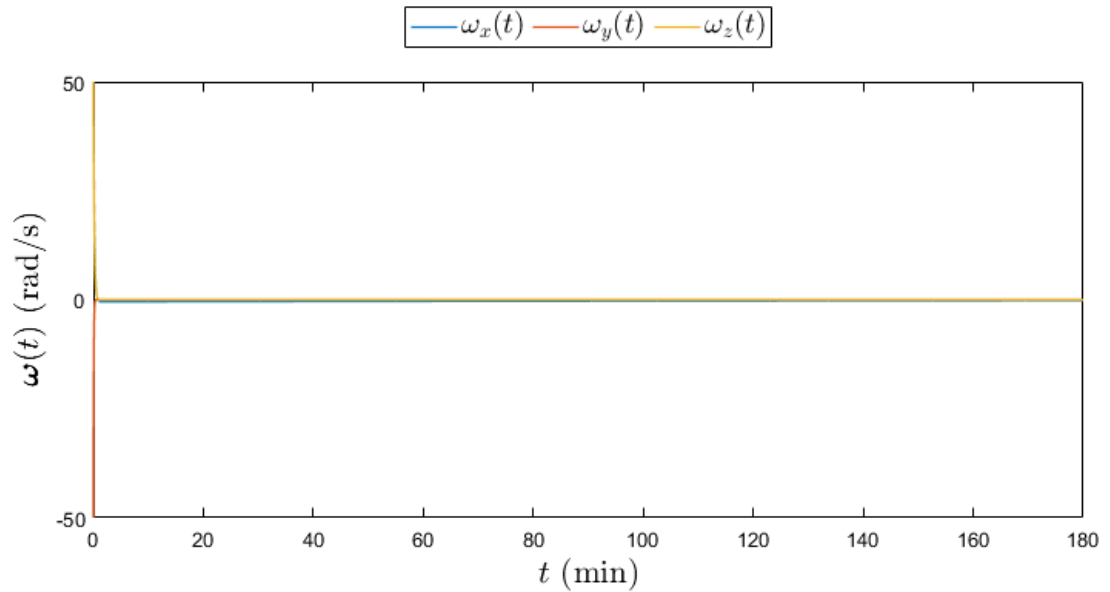
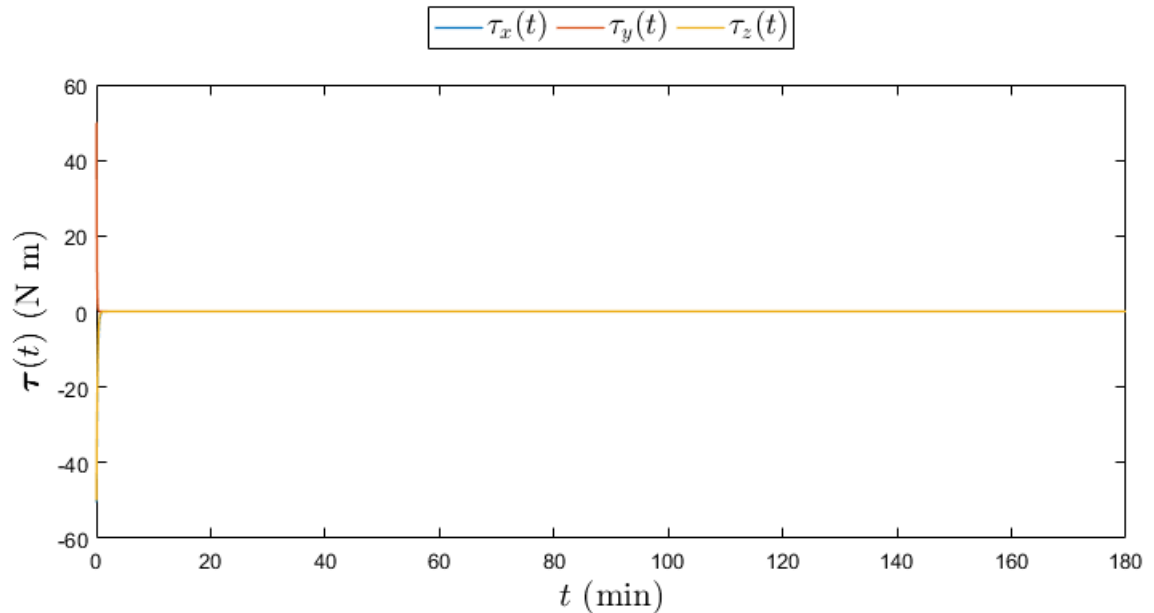


Figura 142. Torques de control del sistema obtenidos con control LQR sin penalización.



De primer plano se puede observar que este control presentó un problema discutido anteriormente, el hecho que los torques deben ser de magnitud entre 0 a 6 Nm para que sea factible implementar el control. Esto es principalmente porque en el LQR no se penalizó ninguna variable de estado. Es por esta misma

razón que el control actúa de manera tan abrupta y trata de reducir en una misma acción de perturbación. A pesar de no ser un resultado esperado, se hizo un cambio y se penalizaron las velocidades angulares por lo que se obtuvo la siguiente matriz K.

$$K = \begin{bmatrix} 1 & 0 & 0 & 0 & 0.025 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0.025 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0.025 \end{bmatrix}$$

Esta nueva matriz obtuvo mejores resultados, los cuales se presentan a continuación:

Figura 143. Estado de cuaterniones con control LQR con penalización.

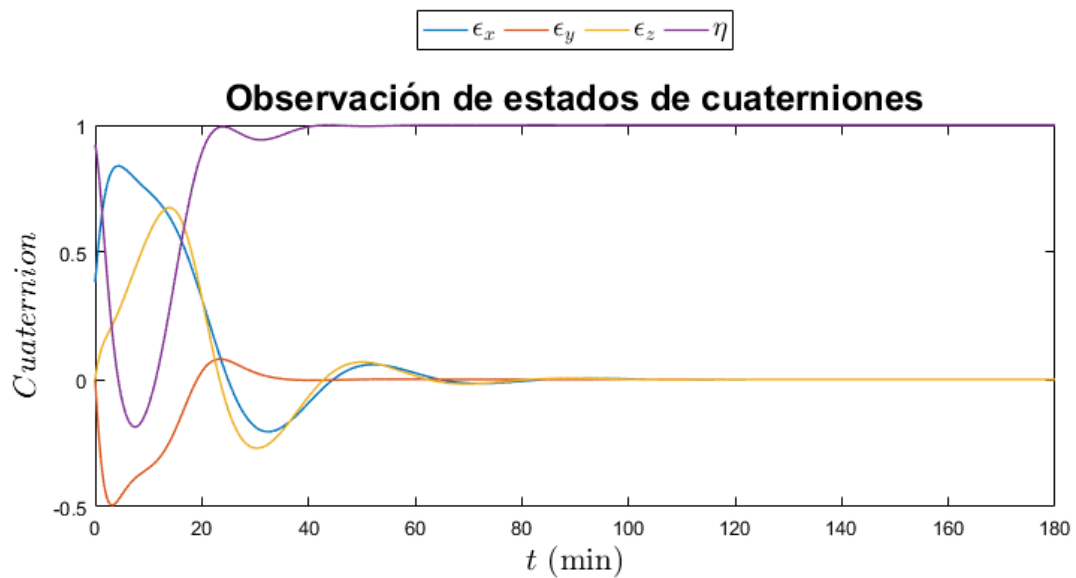
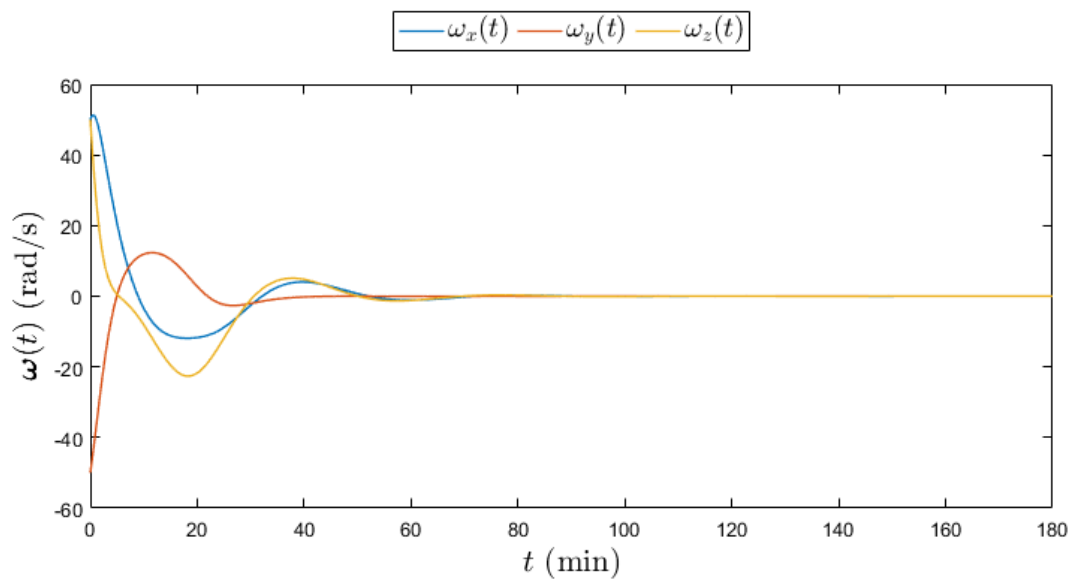
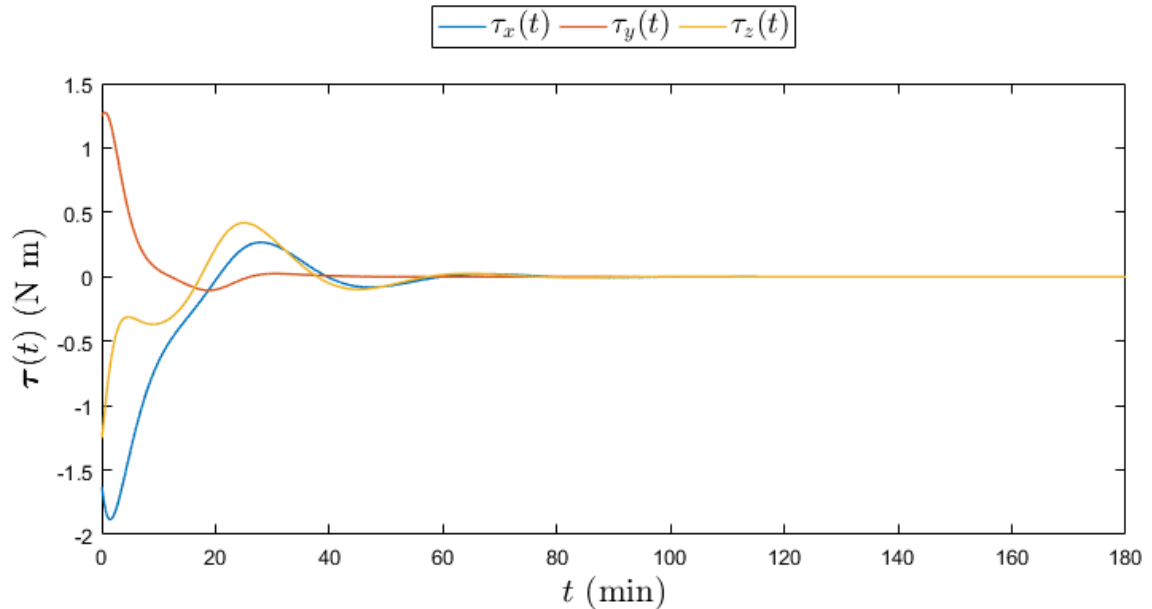


Figura 144. Velocidades angulares del sistema con control LQR con penalización.



En los resultados del LQR con penalización se puede observar que ahora se obtiene de una manera más suavizada y más lenta. Esto podría parecer una disminución de rendimiento del control, no obstante, esto no es así. El porqué esto no es paso hacia atrás en el control LQR es por los torques de control.

Figura 145. Torques de control del sistema obtenidos con control LQR con penalización.

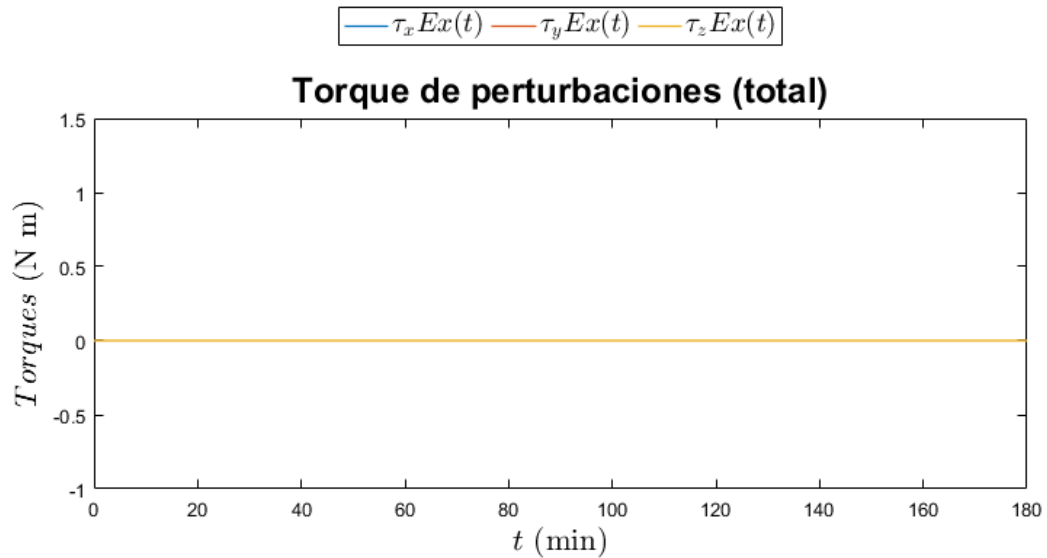


Al observar la gráfica de torques de control se puede observar que se pasó de un torque de 60Nm a uno de 1.2Nm lo cual presenta una gran mejora en el control y lo hace implementable. Al comparar este control con el PID hay algunas diferencias que vale la pena resaltar. El control PID es más estable para reducir la velocidad angular y usar magnitudes más bajas que las utilizadas por el LQR. No obstante, en el LQR se logra estabilizar también la orientación del satélite, lo cual no se puede hacer con el PID de manera convencional. También es un poco más rápido en cuestión de tiempo que el PID.

N. PERTURBACIONES

Todos los controladores pasaron la prueba al tener velocidades angulares iniciales, de la misma manera, se hizo control con las perturbaciones presentes. Como se pudo observar en la metodología, al momento de calcular los valores máximos de las cuatro perturbaciones, sus magnitudes eran de grados de magnitud unidades micro y nano. Esto es una ventaja por como se observa en la siguiente figura no afecta realmente al satélite.

Figura 146. Torques de perturbaciones totales del sistema.



Se puede observar que realmente las perturbaciones no son significativas para el satélite, y este es capaz de rechazarlas solo por el hecho de tener una masa que necesita más fuerza para rotar y también por el control. Algo más que hace que estas perturbaciones sean de magnitud baja es su dependencia de las dimensiones del satélite. Con esto en cuenta, se hizo un experimento para encontrar la magnitud que debería tener las perturbaciones para poder descontrolar el satélite. Esto resultó ser que la suma de todas las perturbaciones multiplicadas por un factor de 10^5 es necesario. A continuación, se observan el resultado de una perturbación de esta índole.

Figura 147. Estados de cuaterniones con perturbaciones alteradas y control LQR penalizado.

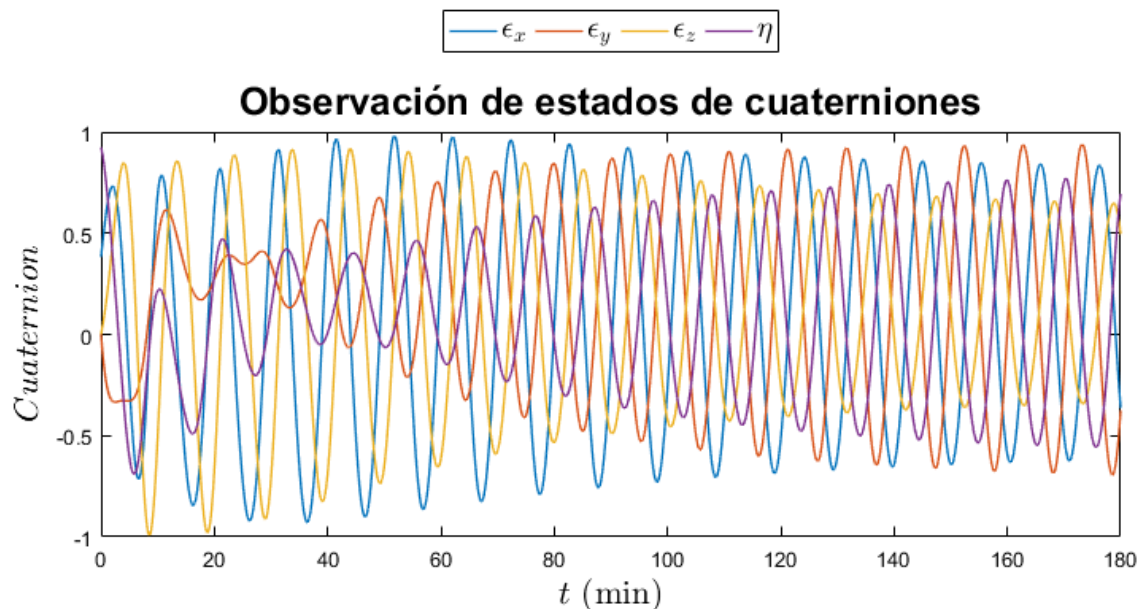


Figura 148. Velocidades angulares con perturbaciones alteradas y control LQR penalizado.

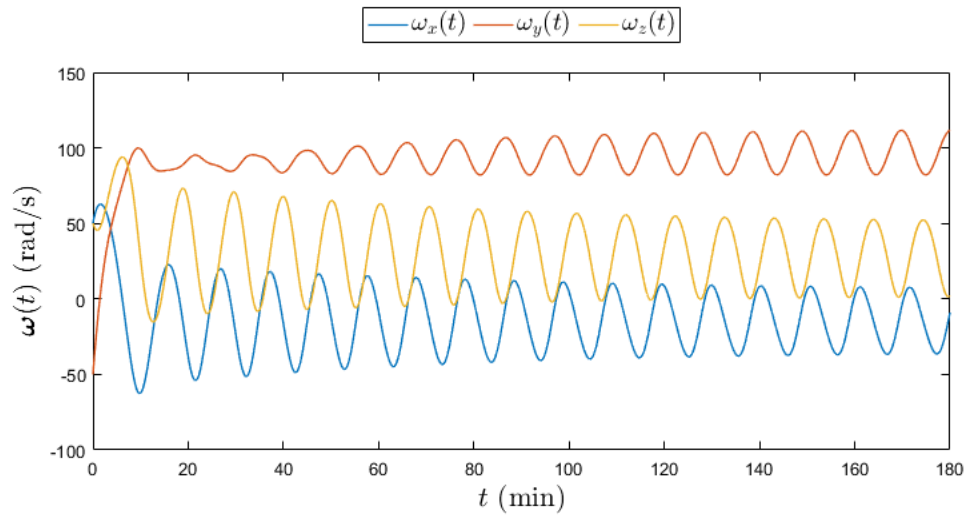


Figura 149. Torques de control con perturbaciones alteradas y control LQR penalizado.

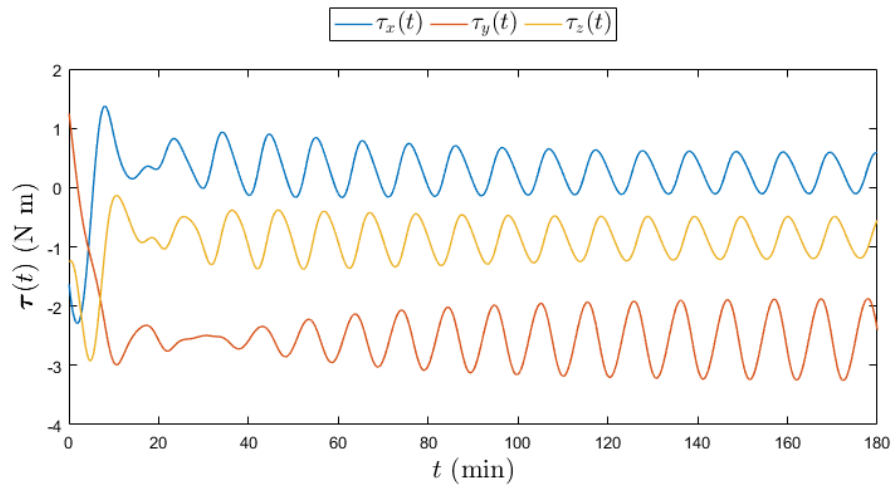
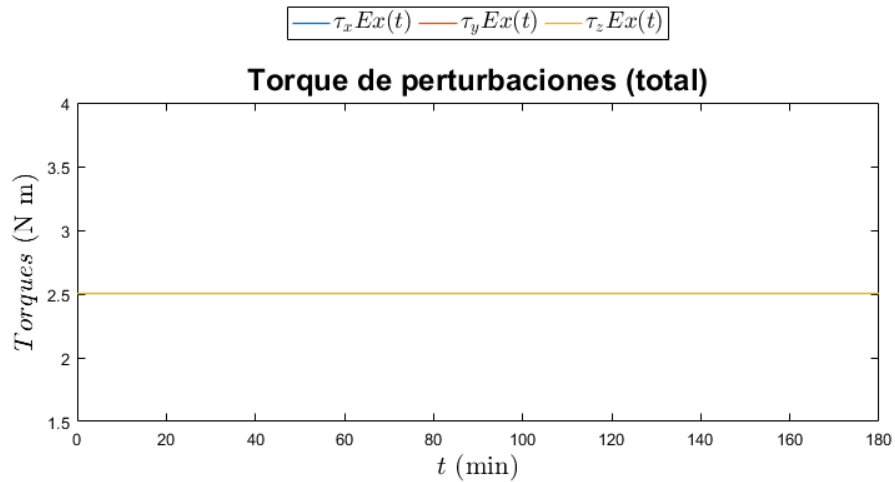


Figura 150. Torques de perturbaciones alteradas totales del sistema.



Con estas imágenes se comprueba que el control no tendría la capacidad de controlar perturbaciones de esa magnitud. No obstante, esto no es un problema porque de manera normal el CubeSat no recibiría magnitudes similares. Por lo que se puede decir que las perturbaciones en el espacio no son un factor tan relevante a la hora de hacer el control de un satélite, hay otros aspectos como la orientación que se vuelven más difíciles de controlar.

O.PLATAFORMA DE PRUEBAS FÍSICAS.

Luego de diseñar la plataforma, esta se construyó y se conectaron todos los sensores y actuadores necesarios para su funcionamiento.

Figura 151. Plataforma de pruebas vista lateral.

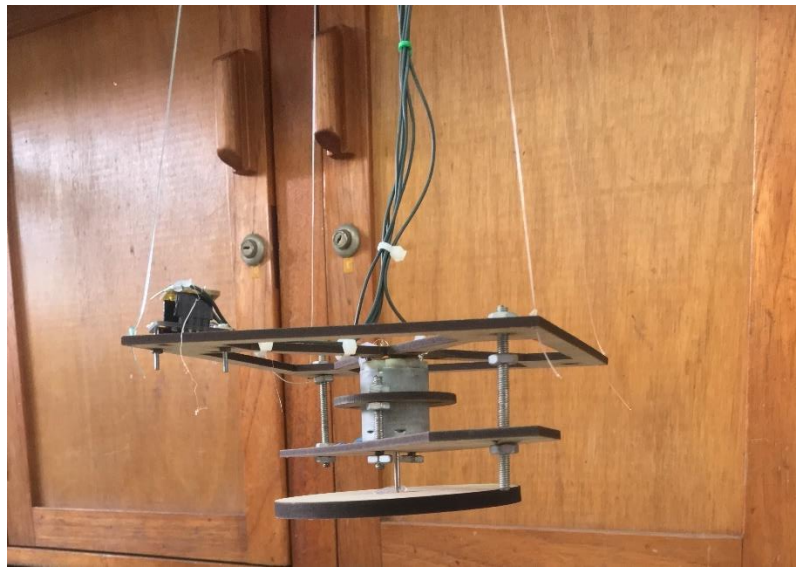


Figura 152. Plataforma de pruebas vista aérea.



Figura 153. Plataforma integrada con caja de comando.

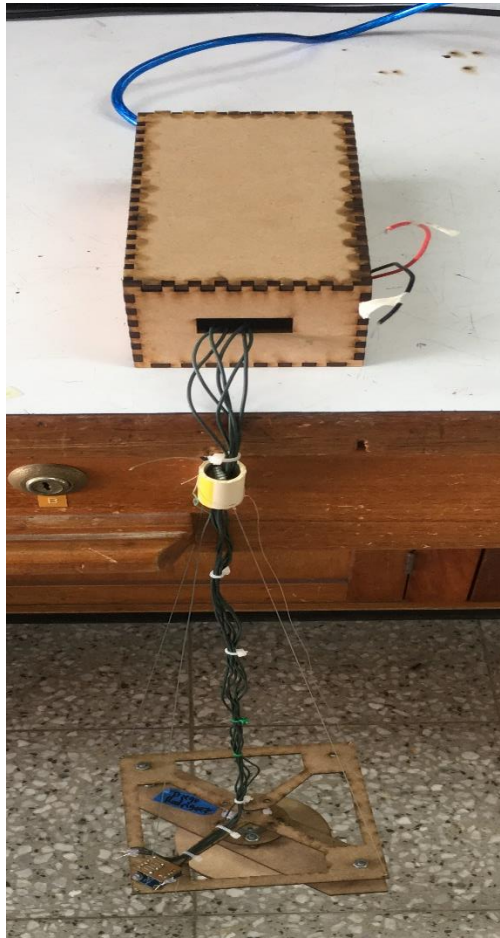
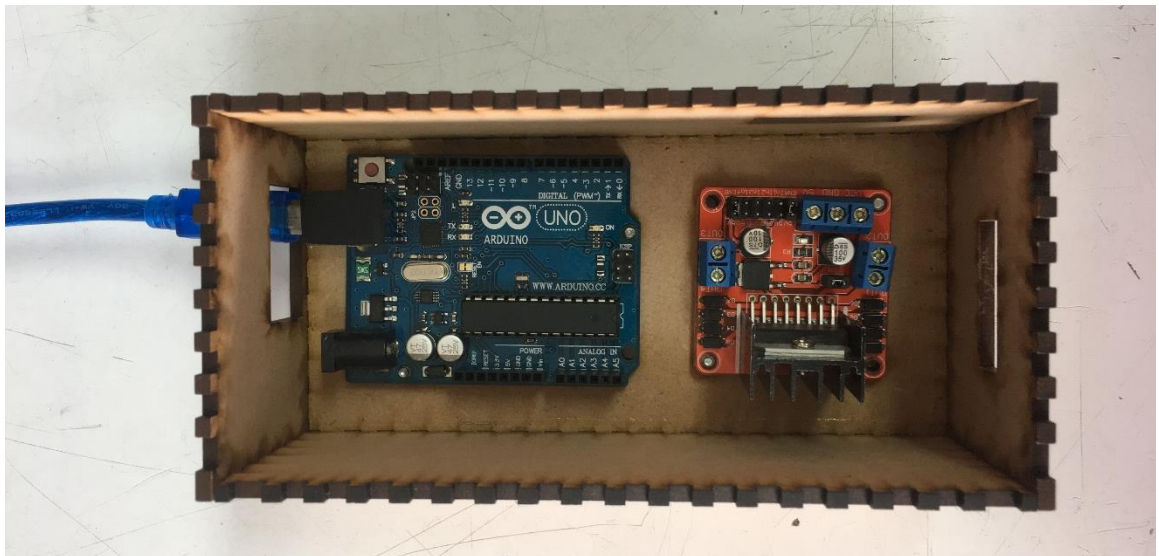


Figura 154. Componentes dentro de la caja de comando.



Con todos los componentes establecidos en su lugar se utilizó un lector serial para leer los valores que la IMU enviaba al Arduino. Esto fue de vital importancia para observar que el control realmente hiciera que las velocidades angulares convergerán a 0 rad/s. Esto también se podía visualizar a través de la vista, pero se necesitaba cuantificar el proceso. Por lo que se llevaron a cabo dos pruebas con la plataforma y así mostrar las diferencias. La primera prueba fue enrollar la plataforma en su eje z y así hacer que los hilos se desenredaran. Esto pretendía demostrar el impulso que se le da al satélite al soltarlo en el espacio.

Figura 155. Prueba de varias vueltas de la plataforma sin acción de control.

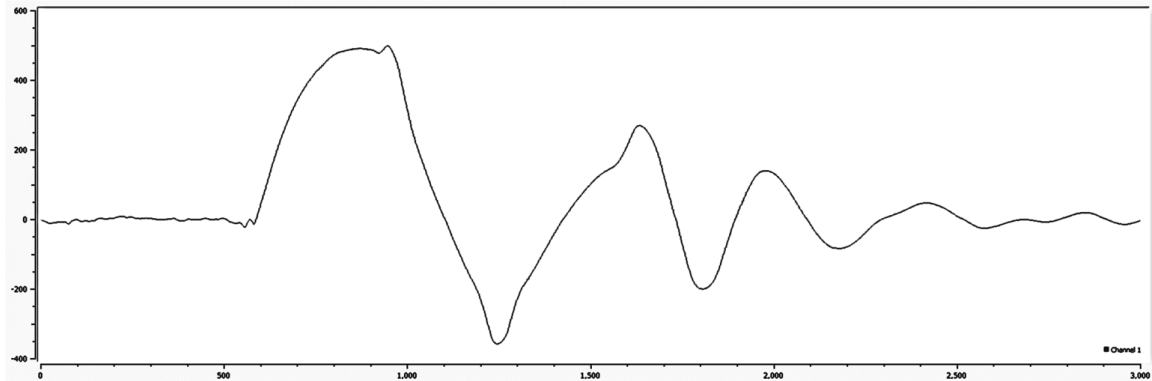
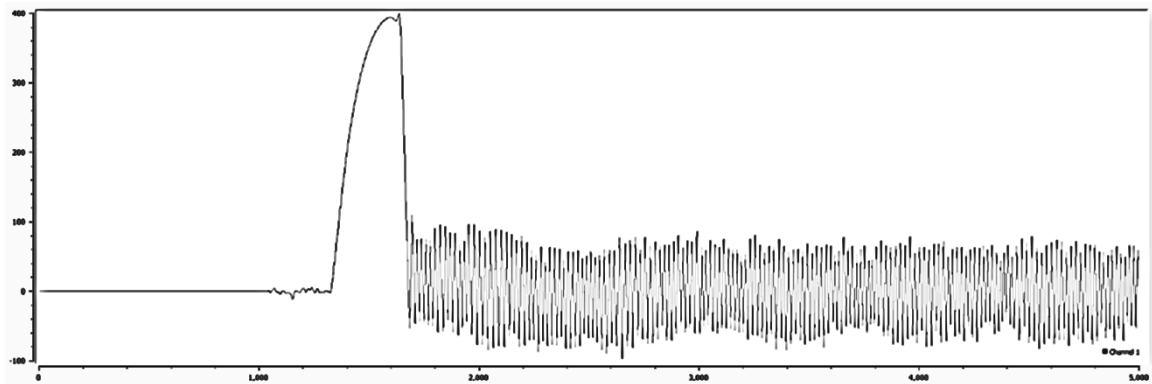


Figura 156. Prueba de varias vueltas de la plataforma con acción de control.



En este caso se puede observar que, sin el control, se cumple un comportamiento sinusoidal con una exponencial negativa que va disminuyendo la perturbación del sistema hasta que llega al equilibrio. Esto sucede de esta manera debido a que existe una presencia de fricción en el sistema, lo cual no ocurre en el espacio. No obstante, se observa que tarde una cierta cantidad de muestras para estabilizarse. Mientras que en el caso de la corrida con control, se observa que hay una reacción inmediata al prender la fuente de voltaje que permite que el motor se accione. Se puede observar también que la señal no queda perfectamente en 0 grados/s, esto es porque la rueda de inercia no tiene una forma de parar de manera adelantada, por lo que permanece tratando de compensar esa acción extra que permanece en el sistema. La

segunda prueba consistía de tener la plataforma en 0 grados/s y aplicarle dos golpes con la mano y observar la reacción del sistema.

Figura 157. Prueba de dos golpes en lados opuestos a la plataforma sin acción de control.

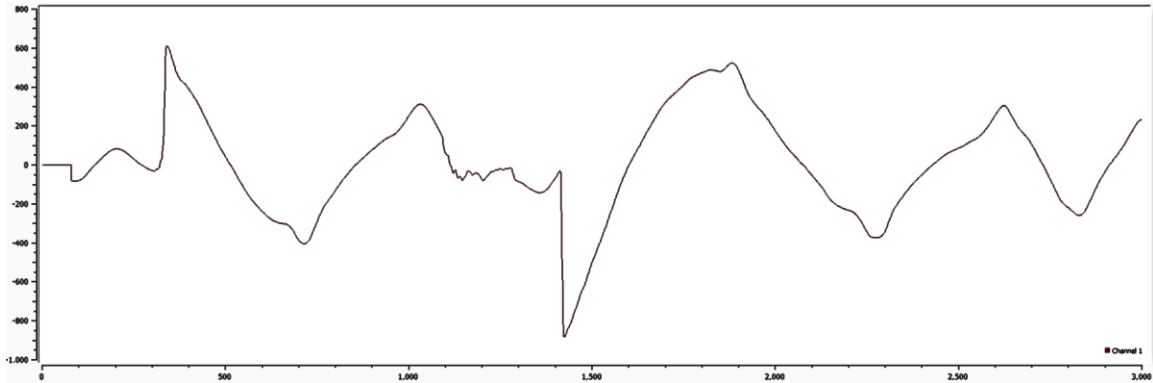
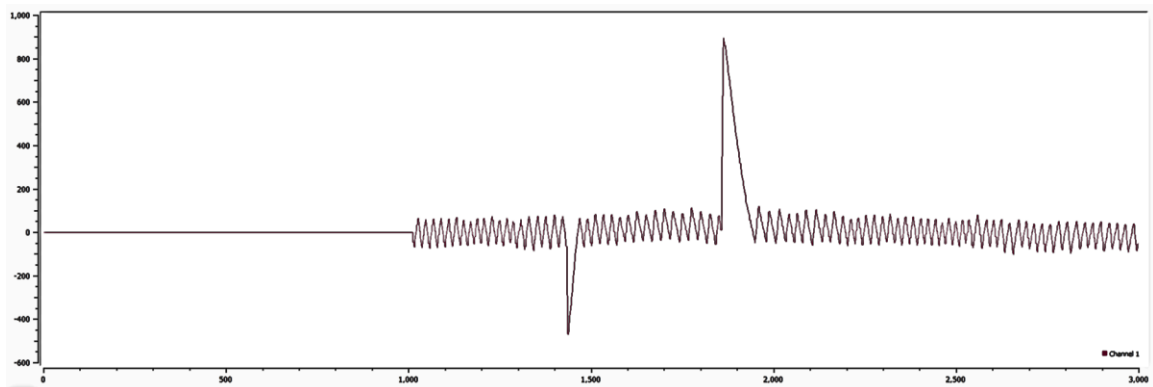


Figura 158. Prueba de dos golpes en lados opuestos a la plataforma con acción de control.



En esta prueba se observa que hubo presencia de dos golpes en distintas direcciones y distintos tiempos. En el caso de la plataforma sin control se observa que empieza a oscilar y realmente no se estabiliza de manera adecuada. Mientras que cuando hay una acción de control se observa que se repele de manera rápida la perturbación. Con estas dos pruebas se observa que el control realmente si hace lo que debe hacer, no obstante, no lo hace de la manera más óptima.

A grandes rasgos es importante tomar en cuenta dos cosas que se comprobaron en la investigación, la primera es que el sistema es globalmente asintóticamente estable, esto es importante porque significa que el satélite siempre llegará al equilibrio. La segunda es que los controladores sean simples y no necesiten leyes complejas. Estas dos comprobaciones ayudan a visualizar de mejor manera porque es posible que el control pasivo logre controlar el satélite. La realidad es que solo se necesita un actuador o componente que se oponga al movimiento que lleva el satélite, esto sacará energía del sistema hacia el entorno y podrá reducir las velocidades angulares.

VII. CONCLUSIONES

A. INGENIERÍA DE SISTEMAS Y GESTIÓN DE PROYECTOS

A través de la revisión de los objetivos de la misión se logró definir un grupo preliminar de los requerimientos que el sistema desarrollado debe cumplir para poder alcanzar tales objetivos. En el Cuadro 5 se muestran estos requerimientos. Sin embargo, para que la misión sea exitosa se debe cumplir también con una serie de requerimientos impuestos por JAXA y las recomendaciones de seguridad de la ISS. El cumplimiento de estos requerimientos asegurará el correcto despliegue y operación del sistema en las condiciones previstas. Tales requerimientos se definen en los cuadros 6 y 7. A medida que el proyecto madure los requerimientos de este deben ser revisados. Todos estos requerimientos deberán ser verificados y validados a través de los métodos mencionados.

A través del análisis de riesgos se llegó a la conclusión de que aquellos con la criticidad más alta son la falla de la estructura del satélite, la falta de fijación de los componentes, el daño de las celdas solares durante su instalación, el daño del sensor monocromático debido a exposición solar, la generación de impedancia en las PCBs, la falta de comunicación entre el satélite y la estación en Tierra, el fallo en el mecanismo de despliegue de la antena, y la incapacidad para reflejar las condiciones de operación en las pruebas. Estos riesgos pueden ser mitigados a través de la correcta selección de los componentes, la implementación de planes de instalación para los componentes más críticos del satélite, y la realización extensiva de pruebas tratando de replicar de la mejor forma posible las condiciones de operación. Una de las pruebas a realizar es la Prueba de Integración explicada durante el desarrollo del trabajo. En esta se comprobará que todos los módulos cumplen con sus objetivos cuando trabajan como un sistema.

Se comprobó que ninguno de los componentes seleccionados hasta ahora superan las dimensiones máximas del satélite en ninguno de sus tres ejes. De igual forma, se comprobó también que actualmente la suma de la masa de todos los componentes del satélite no supera la masa máxima del satélite; sin embargo, aún no se han seleccionado todos los componentes. Así mismo, se determinó que todos los materiales del satélite cuyas propiedades de gasificación pudieron ser obtenidas cumplen con los requerimientos relacionados con este problema. Finalmente, se comprobó también que todos los componentes de los cuales se cuenta con información sobre su rango de temperaturas operacionales se encuentran dentro del límite especificado por el requerimiento MI-400.

B. MÓDULO DE POTENCIA

Se diseñó un sistema preliminar de potencia para el satélite tipo CubeSat. Para ello se analizó: factibilidad del sistema, requerimientos funcionales, selección de componentes, sistema de distribución de energía, interfaces, riesgos, y procedimientos de ensamblaje del módulo de potencia.

Se determinó que el arreglo físico de las celdas solares en base a las características de la misión y el comportamiento esperado del satélite estará conformado de dos celdas solares en cada una de las caras externas del satélite, con la excepción de la cara con la apertura para la cámara, en la cual solo se pondrá una celda solar.

Se establecieron requerimientos funcionales que sirvieron como base para el diseño del sistema de potencia del CubeSat. Estos se dividen para los sistemas de recolección de energía, almacenamiento de energía y distribución de energía, estos se encuentran en los cuadros 38 – 40.

Se seleccionó mediante una evaluación de conceptos, las celdas solares 3G30A de AZUR SPACE, previamente utilizadas en CubeSats OUFTI-1, ESTCube-1 y MinXSS que se utilizarán para recolectar energía solar.

Se estableció la capacidad de al menos 9.5Wh de una batería de Li-ion con un límite inferior y superior de carga de 70% y 80%, respectivamente, con base a las condiciones de la órbita y el consumo de potencia estimado.

Se seleccionó mediante una evaluación de conceptos, la batería de Li-ion de Sparkfun 2Ah, previamente utilizada en CubeSat 3U MinXSS, que se utilizará para almacenar energía eléctrica.

Se planteó un diseño preliminar del circuito interno para la distribución de energía del satélite tomando en cuenta las características de las cargas que requieren los módulos dependientes, ver Figura 68.

El sistema propuesto por la fase 4 es más factible que el propuesto durante la fase 3 del Megaproyecto. Si bien el sistema de la fase 4 es más factible que el de la fase 3, aún es necesario realizar ajustes para incrementar su factibilidad aún más.

Se identificaron las interfaces mecánicas, eléctricas, de programación y de servicios suministrados.

Se determinaron los riesgos del módulo de potencia y los planes de mitigación para los mismos, ver cuadros 92 – 99.

Se ejemplificó la criticidad de los riesgos, ver Figura 69.

Se establecieron los procedimientos de ensamblaje de los paneles solares y las baterías con base en los procedimientos de ensamblaje que se utilizaron en el CubeSat MinXSS.

Se desarrolló el protocolo de pruebas a realizar para comprobar el correcto funcionamiento de los componentes y el sistema ensamblado, estos se detallan en la sección “PRUEBAS DEL MÓDULO DE POTENCIA”.

C. MÓDULO DE MANEJO DE COMANDOS Y DATOS

Se determinó por medio del motor sustituto que la rotación a una posición establecida no es precisa, por lo tanto, es un aspecto que debe contemplarse en el diseño del carrusel.

Se determinó que la arquitectura de hardware centralizada es la mejor opción para este proyecto, debido a que utiliza una menor cantidad de espacio, peso y consumo de potencia.

Se determinó que la Raspberry Pi Zero era la mejor opción para OBC de las opciones evaluadas, debido a que es un dispositivo con grandes capacidades de procesamiento, acompañado de una masa y volumen muy pequeños.

Se comprobó que la Raspberry Pi Zero no está diseñada con compatibilidad electromagnética.

Se comprobó que el mecanismo de redundancia no cumple con los requisitos de consumo energético de la misión.

Se comprobó con la ayuda del Arduino Uno, que el software implementado permite la configuración y recolección de datos de los chips HML5883L y ADC128D818.

Se comprobó que el protocolo de comunicación I2C permite la interacción entre la OBC y los demás módulos, por lo cual es útil para este proyecto.

Se determinó que la cámara de la Raspberry Pi no es capaz de capturar 25 fotografías de manera continua.

El movimiento del motor provoca un consumo de energía mayor que el proporcionado por la Raspberry alimentada por medio de 5V a 700mA al realizarlo constantemente (25 veces), provocando que está se apague.

La forma de almacenamiento y empaquetamiento de datos correspondientes al estado de los módulos es la correcta porque permite hacer análisis y comprobar su funcionamiento.

D. MÓDULO DE CONTROL ACTIVO

Se logró controlar de manera adecuada la orientación y velocidad angular del satélite utilizando el modelo no lineal con controladores PID y LQR, esto indica que el control de un satélite puede llevarse a cabo con técnicas de control clásico y también de control moderno.

El control LQR tuvo un mejor resultado al momento de acoplar las variables del sistema y llevar al sistema a convergir a los valores deseados que se tenían como punto de equilibrio.

Los valores obtenidos de torque de control son aceptables y concuerdan con los torques que brindan las ruedas de inercia y electroimanes del mercado actual para satélites tipo CubeSat.

Ambos controladores desarrollados demostraron tener una acción robusta contra perturbaciones en el espacio, lo suficiente como para que estas no sean un problema para el control. Se concluyó también que el sistema tendría que tener perturbaciones 100,000 veces más grandes de las actuales para que realmente vuelva inestable al satélite. No obstante, las perturbaciones se acumulan y pueden llegar a saturar los actuadores.

La plataforma desarrollada permitió comprobar la robustez y eficiencia de los controles desarrollados. Esto es importante debido a que a partir de estos datos se encontraron nuevas necesidades para tomar en cuenta en futuros estudios de control de satélites.

VIII. RECOMENDACIONES

A. INGENIERÍA DE SISTEMAS Y GESTIÓN DE PROYECTOS

En cuanto a los requerimientos y los riesgos, es importante que todo el equipo del Megaproyecto, no solamente la persona encargada del módulo de Ingeniería en Sistemas, se dediquen a revisarlos y actualizarlos a medida que el proyecto avanza. Es probable que a medida que el proyecto madure aparezcan nuevos requerimientos y riesgos que deban tomarse en cuenta, o que otros deban ser eliminados o re-escritos si las necesidades de la misión lo exigen. De igual forma, una parte importante del análisis de riesgos es que se observe cómo varían estos a lo largo de la fase de diseño. Para evidenciar estas variaciones, los valores de probabilidad y consecuencia cambian según se requiera. La decisión de qué cambios se realizan debe ser un pacto entre la persona encargada del módulo de Ingeniería en Sistemas y la persona o personas encargadas del módulo en el cual ocurre el cambio.

Ya que aún no se ha finalizado con la fase de definición de requerimientos es importante que el grupo que trabaja en el proyecto dedique cierta cantidad de tiempo al mes, o incluso a la semana, para la revisión de estos. Es común que se tengan que reevaluar los requerimientos en este tipo de proyectos. Es importante también que se mantenga bien informado al equipo de cualquier información proveniente de JAXA, ya que gran parte de los requerimientos dependerán de las decisiones de esta agencia. Igualmente, para la revisión de los riesgos es recomendable que el equipo se reúna constantemente para discutirlos.

Una vez se comience con la verificación y validación de los requerimientos es importante que se documente todo el proceso de manera que no existan dudas sobre si cierto requerimiento fue o no verificado y validado. Para esto es recomendable que se generen documentos en los cuales se explique cómo se verificó y validó cada requerimiento. Tales documentos deberían estar referenciados en la matriz de verificación y validación, de manera que puedan identificarse fácilmente y que permitan determinar si aún no existen planes de verificación y validación para algún requerimiento.

Debe resaltarse que el simple hecho de definir los riesgos no hará que estos sean mitigados. Para estos se necesita que se lleven a cabo planes de mitigación para cada riesgo. Es por esto que se recomienda que todos los encargados de los distintos módulos revisen constantemente el estado de la matriz de riesgos del proyecto. Esto les ayudará en la toma de decisiones sobre cómo puede mitigarse cada riesgo. Adicionalmente, sería recomendable que a la matriz de riesgos se le añadiera información más detallada sobre cómo se planea mitigar cada riesgo.

Es importante que a medida que más componentes vayan siendo elegidos se continúe documentando sus propiedades físicas y operacionales. Toda esta información será útil para finalizar el diseño de algunos módulos, al igual que para asegurar que todos los componentes cumplen con los requerimientos ya

establecidos. Es recomendable también que se continúe buscando información sobre las propiedades de gasificación de los materiales en los componentes del satélite. Toda esta información debería ser revisada con cierta frecuencia para asegurar que aún se está tomando en cuenta.

Debido a que no existe información sobre las propiedades de gasificación de todos los materiales listados, es importante que se lleve a cabo apropiadamente la prueba de termo-vacío del satélite. Esta prueba podría demostrar si la gasificación excesiva de alguno de los componentes podría generar problemas. Así, es recomendable que se ponga atención a este detalle durante la prueba de termo-vacío.

Sobre la Prueba de Integración, a pesar de que no es necesario seguir estrictamente el cronograma, es importante que se utilice como una guía a través de toda la prueba debido a que cada fase y función del satélite debe ser probada en 24 horas. Así, el cronograma provee la flexibilidad suficiente como para que las personas que realicen la prueba puedan decidir cuándo es el momento indicado para inicializar cierta función del sistema. Esta flexibilidad es importante debido a que es muy probable que el sistema no se comporte como se espera durante la prueba, lo cual significa que es probable que el sistema tenga que ser ajustado durante la prueba.

Ya que las celdas solares serán utilizadas durante la prueba, es importante colocarlas en un área donde reciban la mayor cantidad de luz solar posible. Es importante también señalar que la prueba debe empezar tan puntualmente como sea posible para aprovechar de mejor manera la luz solar.

Otra cosa que debe tomarse en cuenta sobre esta prueba es que antes de realizarse deben haberse realizado pruebas para los módulos individuales. Debe comprobarse que los módulos operan adecuadamente antes de poder probarse en el sistema integrado. Si uno de los módulos no opera correctamente por sí solo, se estará perdiendo tiempo al integrar y probar en el sistema un módulo defectuoso. Además, esta no será la única prueba de integración que se deberá realizar ya que se necesitará planear otras para verificar y validar ciertos requerimientos del sistema que no pueden ser comprobados con una prueba en un laboratorio.

Ya que algunos de los componentes aún están siendo seleccionados y que probablemente el diseño de algunos módulos varíe a medida que el proyecto madura, es recomendable que tanto la prueba de integración como el resto de las pruebas sean revisadas una vez se tenga un diseño definitivo del satélite para asegurar que estas aún cumplen con su función de verificar y validar ciertos requerimientos. Cualquier cambio que deba ser realizado en la planificación de las pruebas debe ser correctamente documentado de manera que no se pase por alto al momento de llevarla a cabo.

B. MÓDULO DE POTENCIA

Para evitar los problemas relacionados con temperatura en las baterías, se recomienda que las mismas se mantengan entre 10 °C y 60 °C, por lo que los calentadores deberían activarse cuando las baterías alcancen los 10 °C, y desactivarse al llegar a los 20 °C. Además, se recomienda evitar que las baterías sean almacenadas con alta carga, y evitar altas y bajas temperaturas, 20 °C +/- 5 °C.

Para estimar el estado de carga de las baterías se recomienda utilizar el método de corriente, ya que el método de voltaje depende de la temperatura y la corriente de descarga.

El diseño de las PCBs tanto externas, como internas deberá ser realizado, para lo que se recomienda conseguir a personal capacitado y experimentado en estos aspectos. Además, se deberá tomar en cuenta el procedimiento de ensamblaje para los paneles solares.

Se recomienda analizar los riesgos y darles seguimiento, además de llevar a cabo el plan de mitigación planteado en la sección “RIESGOS DEL MÓDULO DE POTENCIA” para cada uno de los riesgos identificados.

Las simulaciones de potencia y energía tienen muchos factores que pueden afectar los resultados, y que francamente se desconocen a totalidad. Por lo que no se recomienda tomar como una exacta del funcionamiento del satélite, sino más con fines de dar una idea de la factibilidad del proyecto y poder evaluar el progreso del módulo con el paso del tiempo.

Para el ensamblaje de las celdas solares se recomienda tener mucho cuidado, ya que las celdas solares y las PCBs pueden ser bastante frágiles. Además, se recomienda evitar el contacto con la mano y utilizar el equipo necesario: guantes de látex, pinzas de vacío, etc.

Se recomienda seguir la secuencia de pasos detallada en la sección “PROCEDIMIENTOS DE ENSAMBLAJE DEL MÓDULO DE POTENCIA” para la instalación de las celdas solares y en caso de duda acudir a la referencia Dahir, *et al*, 2017. De ser posible, contactar directamente al Dr. Andrew Dahir de la Universidad de Colorado, Boulder para resolución de dudas.

Para el ensamblaje de las baterías se recomienda analizar la factibilidad de utilizar la estructura interna tomando en cuenta aspectos como masa y volumen. En caso de ser posible se debe diseñar y analizar desde un punto de vista estructural y térmica. Además, de ser posible la PCB que contenga esta estructura no deberá tener capas de cobre para aislar de cierta manera las baterías del resto del satélite.

Se recomienda seguir la secuencia de pasos listada en la sección “PROCEDIMIENTOS DE ENSAMBLAJE DEL MÓDULO DE POTENCIA” para ensamblar las baterías, además de utilizar el equipo necesario.

Para verificar el correcto funcionamiento de los paneles solares se recomienda realizar las pruebas de electroluminiscencia, termografía e iluminación, la cual a su vez se recomienda sea artificial.

Se recomienda seguir el protocolo de pruebas establecido para las baterías son de inspección visual, voltaje entre terminales, mediciones de voltaje, corriente y potencia para varios ciclos de carga/descarga. Además, se recomienda probar las protecciones de la batería, y medir la pérdida total de masa.

Las pruebas recomendadas para el sistema integrado son de funcionamiento en termo vacío, para simular las condiciones de operación en el espacio y pruebas de vibraciones. Estas pruebas se deberían realizar para todos los módulos.

C. MÓDULO DE MANEJO DE COMANDOS Y DATOS

Recubrir la Raspberry Pi Zero para darle compatibilidad electromagnética y que no afecte y sea afectada por interferencias electromagnéticas.

Implementar el mecanismo de redundancia con computadoras que tengan menor consumo energético a la Raspberry Pi Zero.

Modificar las instrucciones de transferencia de datos con el módulo de comunicaciones cuando se tenga la información técnica del protocolo de comunicación del componente.

Diseñar un prototipo de la placa del sistema de control y procesamiento de datos para realizar pruebas de integración con los componentes reales de los demás módulos del satélite.

Establecer prioridades en la información que se almacena, de esta forma se puede decidir cuales datos son primordiales enviar a la estación terrestre si se tiene un periodo corto de transmisión.

D. MÓDULO DE CONTROL ACTIVO

Se recomienda mejorar los modelos de perturbaciones para que dependan de la posición del satélite con respecto al sol y la tierra. Esto con el fin de mejorar la precisión con la que estos factores externos afectan al satélite.

Es recomendable desarrollar investigación alrededor de temas como ruedas de inercia y también de electroimanes. De esta manera se podría incluir lazos de control externos e internos en cascada para desarrollar estos actuadores en la misma universidad y disminuir los costos de la misión.

Implementar el filtro de Kalman para simular el ruido que existe en el espacio y también como observador del sistema para obtener una mejor aproximación de los valores reales de las variables de estado del sistema. Es decir que se implemente un control tipo LQG.

Se recomienda mejorar la plataforma con respecto a su tamaño, su actuador, y remover su necesidad de alambres para que pueda funcionar en plataformas de aire comprimido para actuar libremente.

Es recomendable desarrollar un sistema de reconocimiento de posición del satélite para cambiar la referencia de ángulo constantemente y así seguir una trayectoria en lugar de una sola posición durante toda la misión. Esto haría que la misión tuviera más éxito al momento de rastrear un lugar en específico en la tierra y no perder energía en momentos donde no sea necesario utilizar el control.

Se recomienda implementar ecuaciones más complejas de posicionamiento del sistema para tener valores más exactos y constantes. Esto con el fin de poder describir mejor la trayectoria del satélite y específicamente reconocer la variación de todos los factores que le afectan externamente.

IX. BIBLIOGRAFÍA

- [1] Adafruit Industries. *HMC5883L Datasheet*. https://cdn-shop.adafruit.com/datasheets/HMC5883L_3-Axis_Digital_Compass_IC.pdf. [1 de octubre de 2017].
- [2] Agencia Espacial Civil Ecuatoriana (EXA). 2016. *BAOX: 1U High Capacity Battery Packs*. Guayaquil. 2 págs.
- [3] Agencia de Exploración Espacial Japonesa. 2015. *JEM Payload Accommodation Handbook - Vol.8- Small Satellite Deployment Interface Control Document*. Agencia de Exploración Espacial Japonesa 40 págs.
- [4] Agolli, J. *et al.* “Design and Analysis of the Sphinx-NG CubeSat” Worcester. Pp.190, marzo 2017.
- [5] Ams AG. 2016. TSL237T, TSL237CL: High-Sensitivity Light-to-Frequency Converter. Unterpremstätten. 27 págs.
- [6] Andreotiti, Jorge. Motor Piezoeléctrico. <http://egresadoselectronicaunc.blogspot.com/2014/12/motor-piezoelctrico.html>. [10 de octubre de 2017].
- [7] Appinsys. “earth Magnetic Field and Climate Variability” 2010. [En línea]: <http://appinsys.com/globalwarming/earthmagneticfield.htm>
- [8] Arduino. *ARDUINO UNO REV3*. <https://store.arduino.cc/arduino-uno-rev3>. [10 de octubre de 2017].
- [9] Azur Space. 2012. *Handling of Semiconductor Products*. Heilbronn. 2 págs.
- [10] Azur Space. *30% Triple Junction GaAs Solar Cell Assembly*. http://www.azurspace.com/images/products/0003401-01-01_DB_3G30A.pdf [15 de agosto de 2017].
- [11] Balcells, J. 1992. *Interferencias electromagnéticas en sistemas electrónicos*. Marcombo.
- [12] Barrios, K. *et al.* 2016. *CubeSat Fase 3*. Tesis en modalidad de Megaproyecto de Universidad del Valle de Guatemala. Ciudad de Guatemala, Guatemala.: Facultad de Ingeniería. 185 págs.
- [13] Battery University. 2016. *BU-808: How to Prolong Lithium-based Batteries*. http://batteryuniversity.com/learn/article/how_to_prolong_lithium_based_batteries (22 de septiembre de 2016)
- [14] Blanchard, Benjamin; y W. Fabrycky. 2013. *Systems engineering and analysis*. 5ª ed. New Jersey: Prentice Hall. 846 págs.
- [15] Brathen, G. “Design of Attitude Control System of a Double Cubesat”. Trondheim, Noruega. Pp. 114, enero 2013.
- [16] Brumbaugh, Katharine; y E. Lightsey. 2013. <<Application of Risk Management to University

- CubeSat Missions». *Journal of Small Satellites*. 2 (1): 147-160.
- [17] Can, K. *et al.* Design, construction, and test of a reliable, redundant on-board computer (OBC) for the OUFTI-1 CubeSat of the University of Liège. <http://orbi.ulg.ac.be/bitstream/2268/107132/1/OBC.pdf> [10 de octubre de 2017].
- [18] Cannon, R. “Some Basic Response Relations for Reaction-Wheel Attitude Control” en *The American Rocket Society*. Pp.49.
- [19] CH Robotics. 2014. *UM7 Datasheet*. Utah. 79 págs.
- [20] Clawson, J, *et al.* 2002. <<Spacecraft Thermal Environments>>. *Spacecraft Thermal Control Handbook, Volume I: Fundamental Technologies*. 2ª ed. Estados Unidos de América: The AeroSpace Press. págs. 21-69.
- [21] Clyde Space. 2016. *10 Whr CubeSat Battery*. <https://www.clyde.space/products/48-10whr-cubesat-battery> (21 de enero de 2017)
- [22] Cote, K., Gabriel, J., Patel, B., Ridley, N., Taillefer, Z., & Tetreault, S. 2011. «Mechanical, Power, and Propulsion Subsystem Design for a CubeSat». *Worcester Polytechnic Institute* [Massachusetts].
- [23] CubeSatShop. 2016. *Cube ADCS*. <https://www.cubesatshop.com/product/cube-adcs/> (31 de agosto de 2016)
- [24] CubeSatShop. 2016. *S-Band Patch Antenna RHCP for HISPICO*. <https://www.cubesatshop.com/product/s-band-patch-antenna-rhcp-hispico/> (31 de agosto de 2016)
- [25] Dahir, A, *et al.* 2017. *Advancement, Testing and Validation of an Innovative SmallSat Solar Panel Fabrication Process*. Actas de la Conferencia AIAA/USU sobre Pequeños Satélites. 6 págs.
- [26] De Dijcker, S. Design of the On-Board Computer of the Belgian OUFTI-1 CubeSat. http://www.leodium.ulg.ac.be/cmsms/uploads/2011_04_22_CalPoly_De_Dijcker_OUFTI-1.pdf [10 de octubre de 2017].
- [27] Ebersole, Daryl. 2016. *Lithium Battery Failures – Expert Introduction to Fire & Explosion Investigations*. <http://www.robsonforensic.com/articles/lithium-battery-fire-expert-witness> (12 de septiembre de 2017).
- [28] Edmund Optics, Inc. *442nm CWL, 25mm Dia., Hard Coated OD 4 10nm Bandpass Filter*. <https://www.edmundoptics.com/optics/optical-filters/bandpass-filters/442nm-cwl-25mm-dia.-hard-coated-od-4-10nm-bandpass-filter/#downloads> [18 de junio de 2017]
- [29] Electronica Unicrom. Funcionamiento de fotodiodo. https://web.archive.org/web/20060430230802/http://www.unicrom.com/Tut_fotodiodo.asp [10 de octubre de 2017].

- [30] ESA. *On board computer and data handling*. http://www.esa.int/Our_Activities/Space_Engineering_Technology/Onboard_Computer_and_Data_Handling/Onboard_Computers. [10 de octubre de 2017].
- [31] E-con Systems India Pvt. Ltd. 2015. *Camera Module Datasheet*. Guindy. 12 págs.
- [32] Fairchild/ON Semiconductor. *BAT54C Schottky Diodes*. <https://www.fairchildsemi.com/datasheets/BA/BAT54.pdf> [13 de agosto de 2017]
- [33] Fairchild/ON Semiconductor. *MM3Z2V4C-MM3Z75VC Zener Diodes*. <https://www.fairchildsemi.com/datasheets/MM/MM3Z10VC.pdf> [23 de julio de 2017]
- [34] Faulhaber. 2017. *Brushless Flat DC-Micromotors*. Croglío. 2 págs.
- [35] FeeTech. *s.f. Specification of Product, 6V 1.5 kg.cm Analog Continuous Rotation Servo*. Shenzhen. 2 págs.
- [36] Forbes, J. *et al.* "Spacecraft dynamics and control, An introduction". Pp. 589, 2013.
- [37] Fortescue, Peter; G. Swinerd y J. Stark. 2011. *Spacecraft systems engineering*. 4ª ed. Reino Unido: Wiley. 691 págs.
- [38] Froelich, R. "Reaction Wheel Attitude control for space vehicles". Pp.32, octubre 1959.
- [39] Fuglsang, K. Vinther, K. "Attitude Determination and Control System for AAUSAT3" Aalborg. Pp.24, junio 2010.
- [40] Gebara, Christine; y D. Spencer. 2016. <<Verification and Validation Methods for the Prox-1 Mission>>. *Conference on Small Satellites*. 8 (3): 1-8.
- [41] GOMspace. *NanoCom AX100*. <https://gomspace.com/UserFiles/Subsystems/datasheet/gs-ds-nanocom-ax100-33.pdf> [23 de junio de 2017]
- [42] GOMspace. *NanoMind A3200, Datasheet*. Aalborg. 14 págs.
- [43] GOMspace. *NanoPower Battery, Datasheet*. Aalborg. 6 págs.
- [44] Guías Prácticas. Magnetómetros. <http://www.guiaspracticas.com/detectores-de-metales/magnetometros>. [10 de octubre de 2017].
- [45] Hao, Z. "Detumbling and Aerostable Control for Cubesats". Pp. 127, 2013.
- [46] Heavens Above. 2017. *ISS – Orbit*. <http://www.heavens-above.com/orbit.aspx?satid=25544> (15 de septiembre de 2017)
- [47] Henrik. *How does PCB Motor's technology work?* <https://pcbmotor.com/behind-our-technology/>. [10 de octubre de 2017].
- [48] Honeywell. *3-Axis Digital Compass IC HMC5883L*. https://cdn-shop.adafruit.com/datasheets/HMC5883L_3-Axis_Digital_Compass_IC.pdf [12 de agosto de 2017]
- [49] Hyperion Technologies. 2016. *iACS200 Series Attitude Control Systems*. Delft. 2 págs.
- [50] Incell Academy. *s.f. Comparison Common Lithium Technologies*. 6 págs.
- [51] Infineon Technologies. *IRF9321PbF*. <https://www.infineon.com/dgdl/irf9321pbf.pdf?fileId=5546d462533600a40153561125fc1d9b>

[28 de julio de 2017]

- [52] Jordan, Fabien. 2006. *Electrical Power System (EPS)*. Tesis Escuela Politécnica Federal de Lausana. Suiza, Lausana.: Facultad Superior de Ingeniería y Gestión del Cantón de Vaud. 43 págs.
- [53] K&J Magnetics, Inc. *Disc/Cylinder Magnets*. <https://www.kjmagnetics.com/proddetail.asp?prod=D84> [8 de agosto de 2017]
- [54] King, R, *et al.* 2012. <<Solar cell generations over 40% efficiency>> Progress in Photovoltaics. XX (6): 801-815.
- [55] Krogh, K., Schreder, E., & Bak, T. 2002. «Attitude Determination for AAU CubeSat». *Aalborg University* [Aalborg].
- [56] Kuuste, H, *et al.* 2014. <<Imaging system for nanosatellite proximity operations>> Proceedings of the Estonian Academy of Sciences. LXIII (2): 250-257.
- [57] L. Bass, P. Clements, R. Kazman, 2003 *Software Architecture in Practice*. 2ª ed., Estados Unidos: Addison Wesley. 560 paginas.
- [58] Laizans, K, *et al.* 2014. «Design of the fault tolerant command and data handling subsystem for ESTCube-1». *Proceeding of Estonian Academy of Science*[Estonia]. 63: 222-231.
- [59] Lalli, V, *et al.* 1997. *Training Manual for Elements Interface Definition and Control*. Cleveland: Lewis Research Center. 51 págs.
- [60] Larson, W; y J. Wertz. 1999. *Space mission, analysis and design*. 3ª edición. Estados Unidos de América: Space Technology Library. 976 págs.
- [61] Launchbury-Clarc, C. Fault Redundant On-Board Computer using COTS components. <http://personal.ee.surrey.ac.uk/Personal/C.Bridges/AAReST/Files/2015,%20Oliver%20Launchbury-Clark,%20BSc%20Thesis.pdf> [10 de octubre de 2017].
- [62] Li, j. *et al.* “Design of Attitude Control Systems for CubeSat-Class Nanosatellite” in *Journal of Control Science and Engineering*. Canada. Pp.16, diciembre 2012.
- [63] Loff, Sarah. *CubeSats Overview*. https://www.nasa.gov/mission_pages/cubesats/overview. [10 de octubre de 2017].
- [64] Mabrouk, Elizabeth. *What are SmallSats ad Cubesats?*. <https://www.nasa.gov/content/what-are-smallsats-and-cubesats> [10 de octubre de 2017].
- [65] Mason, J, *et al.* 2015. *Miniature X-Ray Solar Spectrometer (MinXSS) – A Science-Oriented, University 3U CubeSat*. <https://arxiv.org/pdf/1508.05354.pdf> (20 de septiembre de 2017)
- [66] Massachusetts Institute of Technology (MIT) Electric Vehicle Team. 2008. *A Guide to Understanding Battery Specifications*. http://web.mit.edu/evt/summary_battery_specifications.pdf (22 de septiembre de 2016).
- [67] MATLAB. “PID controller, Discrete PID controller”. [En línea]: <http://www.mathworks.com/help/simulink/slref/pidcontroller.html#br9ejg0-18>

- [68] Microchip Technology Inc. 2011. *64/80-Pin High Performance, 256 Kbit to 1 Mbit Enhanced Flash Microcontrollers with A/D*. Arizona. 360 págs.
- [69] Murata Electronics North America. *Chip Monolithic Ceramic Capacitors*, <http://www.murata.com/~media/webrenewal/support/library/catalog/> [22 de julio de 2017]
- [70] NASA. "Small Spacecraft Technology state of the Art". Moffett Field, California. Pp. 177, Diciembre 2015.
- [71] NASA. 2007. *NASA Systems Engineering Handbook*. Washington, D.C. NASA. 340 págs.
- [72] NASA. 2008. *Agency risk management procedural requirements*. Washington, DC. NASA. 7 págs.
- [73] NASA. 2010. *NPR 7120.5, NASA Space Flight Program and Project Management Processes and Requirements*. Washington, D.C. NASA. 144 págs.
- [74] NASA. *MAPTIS*. <https://maptis.nasa.gov/> [25 de julio de 2017]
- [75] NASA. *Outgassing Data for Search and Report*. https://outgassing.nasa.gov/cgi/uncgi/search/search_html.sh [23 de julio de 2017]
- [76] Oficina del Programa de la Estación Espacial. 1999. *Space Station Electrical Bonding Requirements*. Houston. NASA. 45 págs.
- [77] Oficina del Programa de la Estación Espacial. 2002. *Space Station Requirements for Electromagnetic Compatibility*. Houston. NASA. 55 págs.
- [78] Oficina del Programa de la Estación Espacial. 2001. *Space Station Electromagnetic Emission and Susceptibility Requirements*. Houston. NASA. 155 págs.
- [79] Onojo, J. *et al.* "Design of Linear Quadratic Regulator for the Three-Axis Attitude Control System Stabilization of Microsatellites" *International Journal of Scientific & Engineering Research*, Volume 7, junio 2016.
- [80] Oredsson, M. 2010. *Electrical Power System for the CubeSTAR Nanosatellite*. Tesis Universidad de Oslo. Noruega, Oslo.: Facultad de Matemática y Ciencias Naturales. 230 págs.
- [81] Panasonic Electronic Components. *Precision Thick Film Chip Resistors*. <https://industrial.panasonic.com/cdbs/www-data/pdf/RDA0000/AOA0000C304.pdf> [23 de julio de 2017]
- [82] Pi Supply. 2017. *Raspberry Pi Zero and Cable Kit, Description*. <https://www.pi-supply.com/product/raspberry-pi-zero-cable-kit/> (31 de agosto de 2016)
- [83] Pumpkin Space Systems. *Remove-Before-Flight (RBF) Pin and Lanyard*. [http://www.pumpkinspace.com/store/p106/Remove-Before-Flight-\(RBF\)-Pin-and-Lanyard.html](http://www.pumpkinspace.com/store/p106/Remove-Before-Flight-(RBF)-Pin-and-Lanyard.html) [19 de junio de 2017]
- [84] Programa CubeSat. 2014. *CubeSat Design Specification Rev. 13*. San Luis Obispo. Universidad Politécnica del Estado de California. 16 págs.
- [85] Qorvo. 2017. *TQP7M9105, 1W High Linearity Amplifier*. Carolina del Norte. 13 págs.

- [86] Quadrino, M. *et al.* “Testing the Attitude Determination and Control of a CubeSat with Hardware-in-the-Loop” Estados Unidos. Pp.184, junio 2014.
- [87] RED. *Color vs Monochrome Sensors*. <http://www.red.com/learn/red-101/color-monochrome-camera-sensors>. [10 de octubre de 2017.]
- [88] Richtek Technology Corporation. 2014. *Understanding the characteristics of Li-ion batteries and Richtek power management solutions*. Taiwan. 21 págs.
- [89] Robots Argentina. *Comunicación – Bus I2C, descripción y funcionamiento*. http://robots-argentina.com.ar/Comunicacion_busI2C.htm [10 de octubre de 2017].
- [90] Rohm Semiconductor. *Thick Film Chip Resistors*. http://rohms.rohm.com/en/products/databook/datasheet/passive/resistor/chip_resistor/mcr_g.pdf [18 de agosto de 2017]
- [91] Sabri, S., Yuhaniz, S., & Kamardin K. 2016. «Designing a low cost CubeSat’s Command and Data Handling subsystem kit». *ARPN Journal Engineering and Applied Sciences* [Malaysia]. 11(10):6259-6264.
- [92] Samsung Electro-Mechanics. *CL21B104KBCNNNC*. http://www.samsungsem.com/kr/support/product-search/mlcc/_icsFiles/afieldfile/2016/08/18/S_CL21B104KBCNNNC.pdf [12 de agosto de 2017]
- [93] Samsung Eletro-Mechanics. *CL31B105KBHNNNE*. http://www.samsungsem.com/kr/support/product-search/mlcc/_icsFiles/afieldfile/2016/08/18/S_CL31B105KBHNNNE.pdf [12 de agosto de 2017]
- [94] Samsung Electro-Mechanics. *Thick-Film Chip Resistor*. http://www.samsungsem.com/kr/support/library/product-catalog/_icsFiles/afieldfile/2015/09/22/CHIP_RESISTOR_2015.pdf [25 de julio de 2017]
- [95] Samsung Electro-Mechanics. *Ultra Low Ohms (RU Series)*. https://media.digikey.com/pdf/Data%20Sheets/Samsung%20PDFs/RU_Series_ds.pdf [12 de agosto de 2017]
- [96] Sandberg, Ariel. 2016. *Streamlining CubeSat Solar Panel Fabrication Processes*. Actas de la Conferencia AIAA/USU sobre Pequeños Satélites. 12 págs.
- [97] Selva, D; D. Krejci. (2012). <<A survey and assessment of the capabilities of CubeSats for Earth observation>> *Acta Astronautica*. LXXIV (5): 50-58.
- [98] Shah, A., Afaq, S., & U. Arshad. 2013. «General system design of Cubesat in LEO for IR imaging». *International Conference on Aerospace Science & Engineering* [ICASE].: 1-5.
- [99] Sparkfun. *585460 Li-Polymer Battery Packs*. <https://www.sparkfun.com/datasheets/Batteries/UnionBattery-2000mAh.pdf> [12 de agosto de 2017]

- [100] Spectrolab, Inc. 2010. 28.3% Ultra Triple Junction (UTJ) Solar Cells. California. 2 págs.
- [101] Spectrolab, Inc. 2016. 30.7% Triple Junction Space Grade Solar Cell, XTJ Prime. California. 2 págs.
- [102] Stackpole Electronics Inc. CSNL Series. <https://www.seielect.com/Catalog/SEI-csnl.pdf> [12 de agosto de 2017]
- [103] Starin, S. "Attitude Det[ermination and Control Systems". Estados Unidos. Pp.33, 2010.
- [104] Steiner, Nicolas. 2007. EPS subsystem: Electrical qualification tests. Tesis Escuela Politécnica Federal de Lausana. Suiza.: Facultad Superior de Ingeniería y Gestión del Cantón de Vaud. 208 págs.
- [105] Swift Navigation Inc. 2016. Piksi Datasheet. California. 4 págs.
- [106] Taiyo Yuden. 2017. SMD Power Inductors (NR series). 3 págs.
- [107] Teckeleo. 2017. Piezoelectric Motors with Built-in Sensors. Valbonne. 6 págs.
- [108] Telemaq. 2017. User Manual, Piezoelectric Motor WLG-30. Valbonne. 2 págs.
- [109] TDK Corporations. Corporation Multilayer Ceramic Chip Capacitors. https://product.tdk.com/info/en/catalog/datasheets/mlcc_commercial_general_en.pdf [26 de octubre de 2017]
- [110] Texas Instruments. ADC128D818 Datasheet. <http://www.ti.com/lit/ds/symlink/adc128d818.pdf>. [1 de octubre de 2017].
- [111] Texas Instruments. bq24650 Synchronous Switch-Mode Battery Charge Controller for Solar Power With Maximum Power Point Tracking. <http://www.ti.com/lit/ds/symlink/bq24650.pdf>. [15 de julio de 2017]
- [112] Texas Instruments. bq28z610 Impedance Track Gas Gauge and Protection Solution for 1-Series to 2-Series Cell Li-Ion Battery Packs. <http://www.ti.com/lit/ds/symlink/bq28z610.pdf> [18 de julio de 2017]
- [113] Texas Instruments. EV2400 EVM Interface Board. <http://www.ti.com/lit/ug/sl00446c/sl00446c.pdf> [12 de julio de 2017]
- [114] Texas Instruments. INA3221 Triple-Channel, High-Side Measurement, Shunt and Bus Voltage Monitor with I²C-and SMBUS-Compatible Interface. <http://www.ti.com/lit/ds/symlink/ina3221.pdf> [22 de julio de 2017]
- [115] Texas Instruments. 2017. Low-Cost Low-Power 2.4 GHz RF Transceiver. Texas. 88 págs.
- [116] Texas Instruments. N-Channel NexFET Power MOSFETs. <http://www.ti.com/lit/ds/symlink/csd16412q5a.pdf> [2 de agosto de 2017]
- [117] Texas Instruments. TMP10x Temperature Sensor With I²C and SMBus Interface with Alert Function in SOT-23 Package. <http://www.ti.com/lit/ds/symlink/tmp100.pdf> [25 de julio de 2017]

- [118] Texas Instruments. TPS6213x 3-V to 17-V, 3-A Step-Down Converter In 3x3 QFN Package. <http://www.ti.com/lit/ds/symlink/tps62132.pdf> [25 de julio de 2017]
- [119] The Boston Consulting Group. 2010. Batteries for Electric Cars: Challenges, Opportunities and the Outlook to 2020. <https://www.bcg.com/documents/file36615.pdf> [22 de septiembre de 2016]
- [120] Tilley, S. *et al.* “Onboard attitude control using reaction wheels”. 2001. [En línea]: <https://www.google.com/patents/US6285928>
- [121] Tomasi, W. 2003. *Sistemas de comunicaciones electrónicas*. México: Pearson Educación.
- [122] Tran, E. 1999. *Environment/EMC/EMI*. Obtenido de Carnegie Mellon University: https://users.ece.cmu.edu/~koopman/des_s99/environment/ [20 de noviembre de 2017]
- [123] UML-Diagrams. s.f.. *UML 2.5 Diagrams Overview*. Obtenido de uml-diagrams: <http://www.uml-diagrams.org/uml-25-diagrams.html> [13 de octubre de 2017]
- [124] Universidad de Adquisición de Defensa. 2006. Risk Management Guide for DoD Acquisition. Departamento de Defensa. Estados Unidos de América. 92 págs.
- [125] Vishay Dale. Low Profile, High Current Inductors. <http://www.vishay.com/docs/34295/sc15ah01.pdf> [3 de agosto de 2017]
- [126] Vishay Semiconductor Opto Division. Ambient Light Sensor. <http://www.vishay.com/docs/81308/temd6010.pdf> [14 de agosto de 2017]
- [127] Vishay Siliconix. Dual N-Channel 40-V (D-S) MOSFET. <http://www.vishay.com/docs/65366/si7288dp.pdf> [23 de julio de 2017]
- [128] Vishay Siliconix. N-Channel 60 V (D-S) MOSFET. <http://www.vishay.com/docs/71333/2n7002k.pdf> [13 de agosto de 2017]
- [129] Vries, W. “Cubesat Drag Calculations”. Estados Unidos, pp. 9, septiembre 2010.
- [130] Wang, *et al.* 2006. Li-Polymer Battery Packs, Specification. Boulder. 6 págs.
- [131] Westfall, A. “Design of an Attitude Control System for Spin-Axis Control of a 3U Cubesat”. Diciembre 2015.
- [132] Woodbank Communications Ltd. 2005. Battery Life (and Death). <http://www.mpoweruk.com/life.htm> (22 de septiembre de 2016)
- [133] Woodbank Communications Ltd. 2005. Cell Chemistries – How Batteries Work. <http://www.mpoweruk.com/chemistries.htm> [22 de septiembre de 2016]
- [134] Woodbank Communications Ltd. 2005. Lithium Battery Failures. http://www.mpoweruk.com/lithium_failures.htm (22 de septiembre de 2016)
- [135] Yageo. Current Sensor – Low TCR. <http://www.yageo.com/documents/recent/PYu-> [12 de agosto de 2017]
- [136] Yageo. Thick Film Chip Resistors. http://www.yageo.com.tw/exep/pages/download/literatures/PYu-R_INT-thick_7.pdf [15 de agosto de 2017]

- [137] Zagorski, P. "Modeling disturbances influencing an Earth-orbiting satellite" in Nauka. Pp.6, s.f.
- [138] ZARM Technik AG. 2013. Magnetic Torquers for Microsatellites. Bremen. 2 págs.

X. ANEXOS

A. REQUERIMIENTOS DE CAL POLY

Algunas de las especificaciones con las que trabajan la mayoría de los satélites CubeSat provienen de las Especificaciones de Diseño de CubeSat de la Universidad Politécnica del Estado de California, Cal Poly por sus siglas en inglés. Al inicio del proyecto se trabajó el diseño del satélite bajo estas especificaciones, por lo que se definió un grupo de requerimientos tomando en cuenta estos. Se notará que la mayoría de estos son iguales, o tienen cierta equivalencia con los requerimientos obtenidos de las especificaciones de JAXA. Cada requerimiento se identifica con dos números; el primero se le asigna según el grupo de requerimientos al que pertenece, y el segundo según el número de requerimiento que posee dentro de su grupo. A los requerimientos generales se les asignó el número 1; a los mecánicos, el 2; a los eléctricos, el 3; y a los operacionales, el 4. De igual forma se mostrarán las técnicas de verificación escogidas para cada requerimiento, siguiendo los formatos explicados en la sección de Matriz de Requerimientos, Verificación y Validación.

Cuadro 113. Requerimientos de Cal Poly.

No.	Requerimiento	Verificación
1.1	Todos los componentes deben permanecer sujetos al satélite durante el lanzamiento, la eyección y la operación.	R
1.2	No deben utilizarse pirotécnicos.	R
1.3	La energía química total almacenada no debe exceder los 100 Watt-hora.	R, A
1.4	Los materiales peligrosos del satélite deben cumplir con el documento AFSPCMAN 91-710, Volumen 3.	R
1.5	Todos los materiales utilizados deben ser aprobados por la lista de materiales de baja desgasificación de la NASA.	R
1.5.1	Los materiales deben poseer una TML menor o igual a 1.0%.	R
1.5.2	Los materiales deben poseer una CVCM menor o igual a 0.1%.	R
1.6	El campo magnético fuera del envoltorio estático del satélite debe ser menor a 0.5 Gauss sobre el campo magnético de la Tierra.	R
2.1	El sistema de coordenadas del satélite debe ser el mismo que el del sistema de despliegue.	R
2.2	Ningún componente debe exceder los 6.5 milímetros en la dirección normal a las superficies externas.	R, I
2.3	Los rieles y las paredes del P-POD no deben ser utilizados para sostener desplegables.	R
2.3.1	Los componentes exteriores no deben tocar las superficies interiores del P-POD, con excepción de los rieles.	R
2.4	El ancho mínimo de los rieles debe ser de 8.5 milímetros.	R, I
2.5	La aspereza máxima de los rieles debe ser de 1.6 micrómetros.	R
2.6	Los bordes de los rieles deben estar redondeados en un radio mínimo de 1 milímetros.	R, I
2.7	El final de los rieles en la superficie Z+ debe tener un área superficial de por lo menos 6.5 x 6.5 milímetros.	R, I
2.8	Por lo menos el 75% de la longitud de los rieles debe estar en contacto con los rieles del P-POD, esto quiere decir que por lo menos 85.1 milímetros deben estar en contacto.	R, I
2.9	La masa máxima del satélite debe ser de 1.33 kilogramos.	R, I

Continuación Cuadro 113

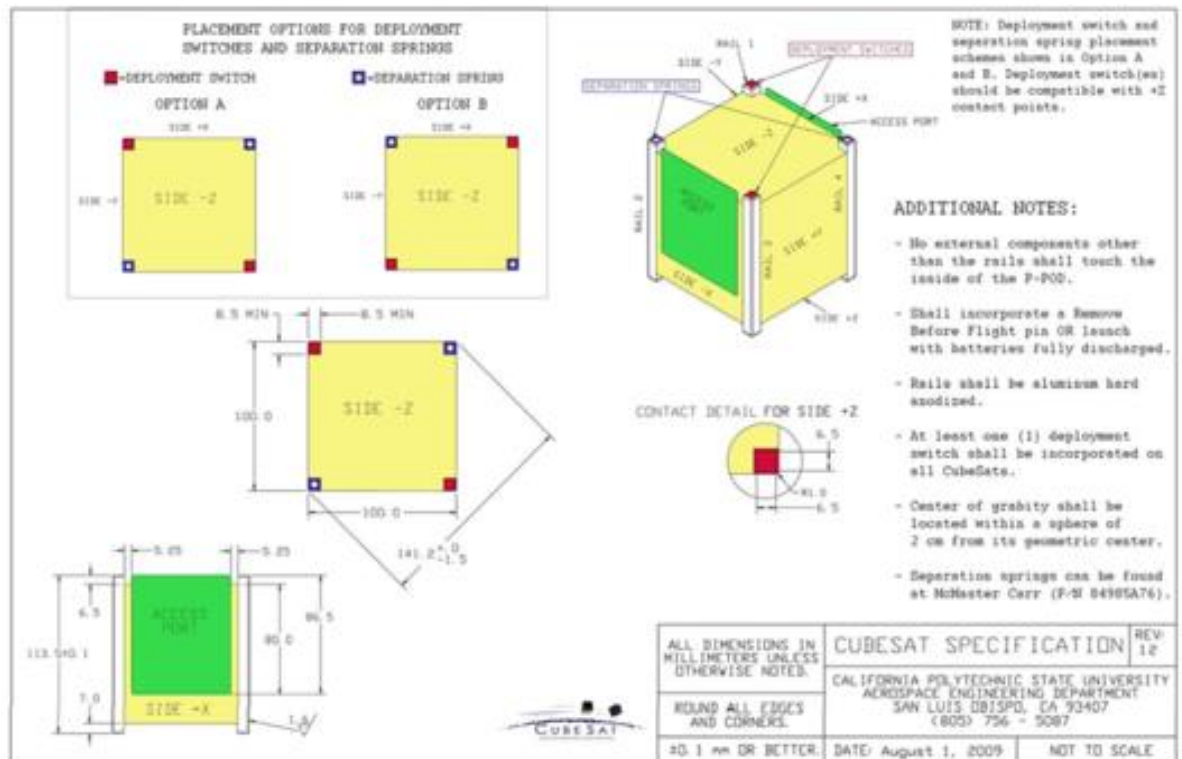
No.	Requerimiento	Verificación
2.10	El centro de gravedad del satélite debe estar localizado a 2 centímetros de su centro geométrico en las direcciones X, Y y Z.	A
2.11	Aluminio 7075, 6061, 5005 y/o 5052 debe utilizarse para la estructura y los rieles del satélite.	R
2.12	Los rieles y sus bases deben ser de aluminio anodizado duro.	R
2.13	El satélite debe utilizar resortes de separación en las bases de los rieles designados.	R
2.13.1	El material de los resortes debe ser acero inoxidable.	R
2.13.2	El largo de lanzamiento del resorte debe ser de por lo menos 0.16 pulgadas sobre la superficie de la base utilizada.	R, I
2.13.3	Las fuerzas inicial y final del resorte deben ser 0.14 y 0.9 libras fuerza, respectivamente.	A
2.13.4	Los resortes de separación comprimidos deben estar al nivel, o bajo el nivel, de la base utilizada.	R
2.13.5	Los resortes deberán poseer la especificación 8-36 UNF-2B.	R
2.14	El satélite deberá poseer un ancho de 100 ± 0.1 milímetros.	R, I
2.15	El satélite deberá poseer un alto de 113.5 ± 0.1 milímetros.	R, I
2.16	Para aceleraciones cuasi-estáticas debe aplicarse un factor de seguridad de 1.25.	A
3.1	Ningún electrónico debe estar activo durante el lanzamiento.	R
3.2	Durante el lanzamiento las baterías deben estar totalmente descargadas o desactivadas.	
3.3	El satélite debe incluir por lo menos un interruptor de despliegue en la base del riel designado para apagar completamente el satélite.	R
3.4	En el estado de actuación, el interruptor de despliegue del satélite debe desconectar eléctricamente el módulo de potencia, incluyendo cualquier reloj de tiempo real.	R
3.5	El interruptor de despliegue debe estar en su estado de actuación todo el tiempo mientras esté instalado en el P-POD.	R
3.5.1	En su estado de actuación, el interruptor de despliegue debe estar al nivel, o más abajo, de la base del riel.	R
3.6	Si el interruptor de despliegue cambia al estado de actuación para después regresar a su estado anterior los contadores de despliegue y transmisión deberán reiniciarse.	R
3.7	El satélite debe incluir un pin RBF o ser lanzado con las baterías completamente descargadas.	R
3.7.1	El pin RBF debe cortar la transmisión de potencia en todo el satélite una vez colocado.	R
3.7.2	El pin RBF debe ser accesible desde la compuerta de acceso del satélite.	R
3.7.3	El pin RBF no debe sobrepasar por más de 6.5 milímetros a la superficie de los rieles una vez insertado.	R, I
3.8	Todos los conectores umbilicales deben estar localizados en la compuerta de acceso del satélite.	R
3.9	Todos los diagnósticos y recargas de las baterías deben realizarse mientras los interruptores de despliegue no se encuentran presionados.	R
3.10	El satélite debe incluir protección en el circuito de la baterías para la carga y descarga.	R
3.11	El satélite debe poseer un inhibidor de radio frecuencias y una potencia de salida por radiofrecuencias menor a 1.5 Watt, o poseer dos inhibidores de radio frecuencias independientes.	R
3.12	El satélite debe poseer la capacidad de recibir comandos para apagarse.	R
2.13	Las transmisiones de radio frecuencias mayores a 1 mW deben esperar para transmitir por lo menos 30 minutos después de que los interruptores de despliegue hayan sido activados.	R

Continuación Cuadro 113

No.	Requerimiento	Verificación
4.1	El satélite debe cumplir con los acuerdos y restricciones para licencias de radio del país donde se está desarrollando.	R
4.2	El diseño de la misión y el hardware del satélite deben cumplir con el estándar NPR 8715.6.	R
4.2.1	Cualquier componente que re-entre en la atmósfera debe hacerlo con una energía menor a 15 Joule.	A
4.3	Todos los desplegables deben esperar por lo menos 30 minutos para desplegarse después de que el interruptor de despliegue haya sido activado.	R
4.4	El satélite no debe generar o transmitir ninguna señal desde el momento que es integrado en el P-POD hasta después de 45 minutos del despliegue desde el mismo.	R
4.5	El tiempo de decaimiento orbital del satélite debe ser menor a 25 años después del fin de la misión.	A

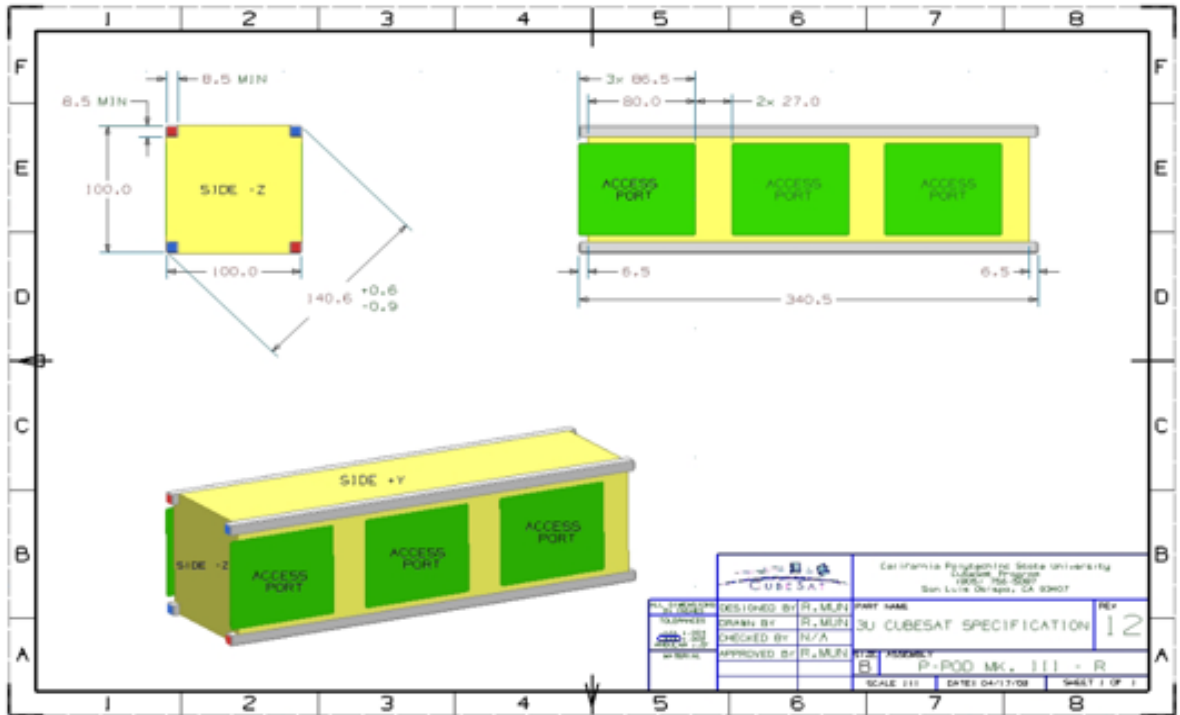
B. PLANOS E IMÁGENES POR JAXA Y CAL POLY

Figura 159. Puerto de acceso CubeSat 1U, estándar Cal Poly.



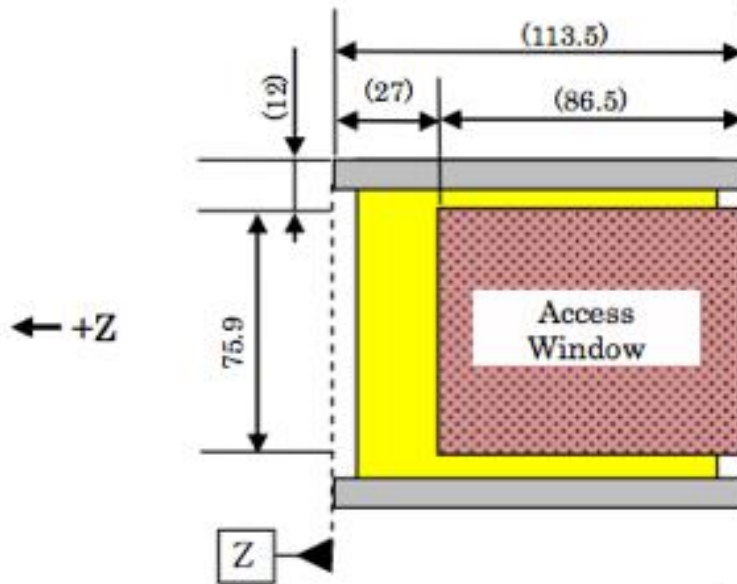
(Cal Poly, 2009).

Figura 160. Puertos de acceso CubeSat 3U, estándar Cal Poly.



(CalPoly, 2009).

Figura 161. Ventana de acceso CubeSat 1U, estándar UNOOSA/JAXA.



(JAXA, 2015).

C. CÓDIGO FUENTE DE MANEJO DE COMANDOS Y DATOS

1. Código para lectura de datos de ADCS

```

#include "hmc5883lDriver.h"
#include "adc128d818Driver.h"

static int powerStatusA=0;
typedef struct {
    int indice;
    dataHMC5883L data[45];
} logMagnetometer;
typedef struct {
    int indice;
    dataADC128D818 data[45];
} logPhotodiodes;

static logMagnetometer logMagne;
static logPhotodiodes logPhot;

void readDataADCS(){
    if(powerStatusA==0){
        dataHMC5883L dataMagnetometer;
        dataADC128D818 dataADC;
        //leer Magnetometro
        //hmcInit(&dataMagnetometer);
        //hmcRead(&dataMagnetometer);
        //printf("X,      Y,      Z:      \t%f      |      %f      |      %f      \n",
dataMagnetometer.data.x,dataMagnetometer.data.y,dataMagnetometer.data.z);
        //leer adc
        adcInit(&dataADC);
        adcRead(&dataADC);
        printf("I0, I1, I2: \t%f | %f | %f \n", dataADC.data.I0,dataADC.data.I1,dataADC.data.I2);
        printf("I3, I4, I5: \t%f | %f | %f \n", dataADC.data.I3,dataADC.data.I4,dataADC.data.I5);

        //seteamos valores
        logMagne.indice=logMagne.indice+1;
    }
}

```

```

        if(logMagne.indice>44){
            logMagne.indice=0;
        }
        logMagne.data[logMagne.indice]=dataMagnetometer;
        logPhot.indice=logPhot.indice+1;
        if(logPhot.indice>44){
            logPhot.indice=0;
        }
        logPhot.data[logPhot.indice]=dataADC;

    }
}

logMagnetometer getMagnetometerData(){
    return logMagne;
}

logPhotodiodes getPhodiodesData(){
    return logPhot;
}

void lowPowerADCS(){
    powerStatusA=1;
}

void okPowerADCS(){
    powerStatusA=0;
}

```

2. Código para lectura de datos del ADC128D818.

```

#include <stdio.h>
#include <stdlib.h>
#include <stdint.h>
#include <string.h>
#include <math.h>
#include <wiringPi.h>

```

```

#include <wiringPiI2C.h>

// Direccion del I2C
#define ADDRESSADC 0x1D
#define REGISTER_CONFIGURATION 0x00
#define REGISTER_INTERRUPT_STATUS 0x01
#define REGISTER_INTERRUPT_MASK 0x03
#define REGISTER_CONVERSION_RATE 0x07
#define REGISTER_CHANEL_DISABLE 0x08
#define REGISTER_ONE_SHOT 0x09
#define REGISTER_DEEP_SHUTDOWN 0x0A
#define REGISTER_ADVANCED_CONFIGURATION 0x0B
#define REGISTER_BUSY_STATUS 0x0C
#define REGISTER_CHANNEL_READING_I0 0x20
#define REGISTER_CHANNEL_READING_I1 0x21
#define REGISTER_CHANNEL_READING_I2 0x22
#define REGISTER_CHANNEL_READING_I3 0x23
#define REGISTER_CHANNEL_READING_I4 0x24
#define REGISTER_CHANNEL_READING_I5 0x25
#define REGISTER_CHANNEL_READING_I6 0x26
#define REGISTER_CHANNEL_READING_I7 0x27
#define REGISTER_MANUFACTURER_ID 0x3E
#define REGISTER_REVISION_ID 0x3F
#define READY 0x01
#define LOCK 0x02

typedef struct {
    int fd;
    int status;
    struct {
        float I0;
        float I1;
        float I2;
        float I3;
        float I4;
        float I5;
    } data;
} data;

```

```

} dataADC128D818;

void adcStatus(dataADC128D818 *dataADC128D818) {
    int status = 0;
    status = wiringPi2CReadReg8(dataADC128D818->fd, REGISTER_BUSY_STATUS);
    delay(100);
    if(status!=1){
        dataADC128D818->status = READY;
    }else{
        dataADC128D818->status = LOCK;
    }
}

//lectura de datos
void adcRead(dataADC128D818 *dataADC128D818) {
    int I0 = 0;
    int I1 = 0;
    int I2 = 0;
    int I3 = 0;
    int I4 = 0;
    int I5 = 0;

    //Leemos el registro de Estatus
    adcStatus(dataADC128D818);

    if( dataADC128D818->status == READY ) {
        //leemos valores de x
        I0=wiringPi2CReadReg8(dataADC128D818-
>fd,REGISTER_CHANNEL_READING_I0);
        delay(100);
        I1=wiringPi2CReadReg8(dataADC128D818-
>fd,REGISTER_CHANNEL_READING_I1);
        delay(100);
        I2=wiringPi2CReadReg8(dataADC128D818-
>fd,REGISTER_CHANNEL_READING_I2);
        delay(100);
    }
}

```

```

        I3=wiringPiI2CReadReg8(dataADC128D818-
>fd,REGISTER_CHANNEL_READING_I3);
        delay(100);
        I4=wiringPiI2CReadReg8(dataADC128D818-
>fd,REGISTER_CHANNEL_READING_I4);
        delay(100);
        I5=wiringPiI2CReadReg8(dataADC128D818-
>fd,REGISTER_CHANNEL_READING_I5);
        delay(100);
        if(I0<0 || I0>4095){
            dataADC128D818->data.I0=I0;
        }
        if(I1<0 || I1>4095){
            dataADC128D818->data.I1=I1;
        }
        if(I2<0 || I2>4095){
            dataADC128D818->data.I2=I2;
        }
        if(I3<0 || I3>4095){
            dataADC128D818->data.I3=I3;
        }
        if(I4<0 || I4>4095){
            dataADC128D818->data.I4=I4;
        }
        if(I5<0 || I5>4095){
            dataADC128D818->data.I5=I5;
        }

        //modo apagado
        wiringPiI2CWriteReg8(dataADC128D818->fd,REGISTER_CONFIGURATION,0x08);
        delay(100);
    }

}

//inicializacion
int adcInit(dataADC128D818 *dataADC128D818) {
    //configuramos para empezar a transmitir (interface descriptor)

```

```

dataADC128D818->fd = wiringPiI2CSetup(ADDRESSADC);

if( dataADC128D818->fd == -1 ){
    printf("nel perro");
    return -1;
}

dataADC128D818->data.I0 = 0;
dataADC128D818->data.I1 = 0;
dataADC128D818->data.I2 = 0;
dataADC128D818->data.I3 = 0;
dataADC128D818->data.I4 = 0;
dataADC128D818->data.I5 = 0;

adcStatus(dataADC128D818);
if( dataADC128D818->status == LOCK ) {
    return -1;
}
//Definimos la configuracion por default
//vref internal
//mode operation 0
wiringPiI2CWriteReg8(dataADC128D818->fd,REGISTER_ADVANCED_CONFIGURATION,
0x00);
delay(100);
//Definir el conversion rate
wiringPiI2CWriteReg8(dataADC128D818->fd,REGISTER_CONVERSION_RATE,0x00);
delay(100);
//enable or disable channels
wiringPiI2CWriteReg8(dataADC128D818->fd,REGISTER_CHANEL_DISABLE,0xC0);
delay(100);
//limit register no necesario
//start y clear int
wiringPiI2CWriteReg8(dataADC128D818->fd,REGISTER_CONFIGURATION,0x01);
delay(100);
return 0;
}

```

3. Código para lectura de datos del HMC5883L Driver.

```

#include <stdio.h>
#include <stdlib.h>
#include <stdint.h>
#include <string.h>
#include <math.h>
#include <wiringPi.h>
#include <wiringPiI2C.h>

// Direccion del I2C
#define ADDRESS 0x1E
//Definicion de los Registros
#define REGISTER_CONFIGURATION_A 0x00 //Velocidad de salida de datos
#define REGISTER_CONFIGURATION_B 0x01 //Ganancia del Dispositivo
#define REGISTER_MODE 0x02 //Funcionamiento del dispositivo
#define REGISTER_OUTPUT_MSB_X 0x03
#define REGISTER_OUTPUT_LSB_X 0x04
#define REGISTER_OUTPUT_MSB_Z 0x05
#define REGISTER_OUTPUT_LSB_Z 0x06
#define REGISTER_OUTPUT_MSB_Y 0x07
#define REGISTER_OUTPUT_LSB_Y 0x08
#define REGISTER_STATUS 0x09
#define REGISTER_IDENTIFICATION_A 0x0A
#define REGISTER_IDENTIFICATION_B 0x0B
#define REGISTER_IDENTIFICATION_C 0x0C
//Estatus del dataHMC5883L
#define READY 0x01
#define LOCK 0x02
//Conversion de GAUSS a MicroTesla
#define CONVERSIONTESLA 100

static float _dataHMC5883L_Gauss_LSB_XY = 1090;
static float _dataHMC5883L_Gauss_LSB_Z = 980;
static float scale= 0.92;

//Estructura para recibir los datos y enviar parametros

```

```

typedef struct {
    int fd;
    int status;
    struct {
        float x;
        float y;
        float z;
    } dataC;
    struct {
        float x, y, z;
    } data;
} dataHMC5883L;

//lectura de status del chip
void status(dataHMC5883L *dataHMC5883L) {
    int status = 0;
    status = wiringPiI2CReadReg8(dataHMC5883L->fd, REGISTER_STATUS);
    dataHMC5883L->status = status;
}

//lectura de datos
void hmcRead(dataHMC5883L *dataHMC5883L) {
    int valX = 0;
    int valX1 = 0;
    int valY = 0;
    int valY1 = 0;
    int valZ = 0;
    int valZ1 = 0;

    //Leemos el registro de Estatus
    status(dataHMC5883L);

    if( dataHMC5883L->status == READY ) {
        //leemos valores de x
        valX = wiringPiI2CReadReg8(dataHMC5883L->fd, REGISTER_OUTPUT_LSB_X);
        delay(100);
    }
}

```

```

valX1 = wiringPiI2CReadReg8(dataHMC5883L->fd, REGISTER_OUTPUT_MSB_X);
delay(100);
//leemos valores de y
valY = wiringPiI2CReadReg8(dataHMC5883L->fd, REGISTER_OUTPUT_LSB_Y);
delay(100);
valY1 = wiringPiI2CReadReg8(dataHMC5883L->fd, REGISTER_OUTPUT_MSB_Y);
delay(100);
//leemos valores de z
valZ = wiringPiI2CReadReg8(dataHMC5883L->fd, REGISTER_OUTPUT_LSB_Z);
delay(100);
valZ1 = wiringPiI2CReadReg8(dataHMC5883L->fd, REGISTER_OUTPUT_MSB_Z);

//se combinan los valores para hacer el registro de 16 bits
dataHMC5883L->data.x = (int16_t)(valX | ((int16_t)valX1 << 8));
dataHMC5883L->data.y = (int16_t)(valY | ((int16_t)valY1 << 8));
dataHMC5883L->data.z = (int16_t)(valZ | ((int16_t)valZ1 << 8));

//verificacion de valores
if(dataHMC5883L->data.x<-2048 ||dataHMC5883L->data.x>2047){
    dataHMC5883L->data.x=0;
}

if(dataHMC5883L->data.y<-2048 ||dataHMC5883L->data.y>2047){
    dataHMC5883L->data.y=0;
}

if(dataHMC5883L->data.z<-2048 ||dataHMC5883L->data.z>2047){
    dataHMC5883L->data.z=0;
}

//Calculo del campo magnetico convertido a microteslas
dataHMC5883L->dataC.x = dataHMC5883L->data.x / _dataHMC5883L_Gauss_LSB_XY *
CONVERSIONTESLA;
dataHMC5883L->dataC.y = dataHMC5883L->data.y / _dataHMC5883L_Gauss_LSB_XY *
CONVERSIONTESLA;
dataHMC5883L->dataC.z = dataHMC5883L->data.z / _dataHMC5883L_Gauss_LSB_Z *
CONVERSIONTESLA;

```

```

    }
}

//inicializacion
char hmcInit(dataHMC5883L *dataHMC5883L) {
    //configuramos para empezar a transmitir (interface descriptor)
    dataHMC5883L->fd = wiringPiI2CSetup(ADDRESS);

    if( dataHMC5883L->fd == -1 ){
        return -1;
    }
    dataHMC5883L->status      = 0;
    dataHMC5883L->data.y      = 0;
    dataHMC5883L->data.z      = 0;
    dataHMC5883L->dataC.x     = 0;
    dataHMC5883L->dataC.y     = 0;
    dataHMC5883L->dataC.z     = 0;

    //Definimos la configuracion por defecto
    wiringPiI2CWriteReg8(dataHMC5883L->fd, REGISTER_CONFIGURATION_A, 0x70);
    delay(100);
    //se define el nivel de ganancia
    wiringPiI2CWriteReg8(dataHMC5883L->fd, REGISTER_CONFIGURATION_B, 0x20);
    delay(100);
    //Se habilita el magnetometro
    wiringPiI2CWriteReg8(dataHMC5883L->fd, REGISTER_MODE, 0x01);
    delay(100);
    return 0;
}

```

4. Código para manejo de la carga útil

```

#include "camera.h"
#include "motor.h"

typedef struct {

```

```

        int step;
        int filter;
    } logPayload;
    static logPayload logPay;

    void takePicture(){
        takePictur();
    }

    void moveCarrousel(){
        moveCarrouse();
    }

    void lowPowerPayload(){
        setPowerStatusC(1);
        setPowerStatusM(1);
    }

    void okPowerPayload(){
        setPowerStatusC(0);
        setPowerStatusM(0);
    }

    void readDataPayload(){
        int step=getStep();
        int filter=getFilter();
        logPay.step=step;
        logPay.filter=filter;
    }

    logPayload getPayloadData(){
        return logPay;
    }
}

```

5. Código para manejo de la cámara

```
#include <stdio.h>
```

```
#include <string.h>

static int filter = 0;
static int powerStatusC=0;

void takePictur(){
    char command[80];
    char name[64];
    if(powerStatusC==0){
        if(filter==0){
            strcpy(command,"raspistill -t 100 -o /home/pi/Desktop/camera/");
            strcpy(name,"filter1-red.jpg -n");
        }
        if(filter==1){
            strcpy(command,"raspistill -t 100 -o /home/pi/Desktop/camera/");
            strcpy(name,"filter2-blue.jpg -n");
        }
        if(filter==2){
            strcpy(command,"raspistill -t 100 -o /home/pi/Desktop/camera/");
            strcpy(name,"filter3-green.jpg -n");
        }
        strcat(command,name);
        system(command);
        filter++;
        if(filter>2){
            filter=0;
        }
    }
}

void setPowerStatusC(int value){
    powerStatusC=value;
}

int getFilter(){
    return filter;
}
```

6. Código para manejo del motor.

```
#include <stdio.h>
#include <string.h>

static int step = 2;
static int powerStatusM=0;

void moveCarrouse(){
    char command[80];
    char parameter[10];
    if(powerStatusM==0){
        if(step==0){
            strcpy(command,"python servoDriver.py 0");
        }
        if(step==1){
            strcpy(command,"python servoDriver.py 1");
        }
        if(step==2){
            strcpy(command,"python servoDriver.py 2");
        }
        system(command);
        step++;
        if(step>2){
            step=0;
        }
    }
}

void setPowerStatusM(int value){
    powerStatusM=value;
}

int getStep(){
    return step;
}
```

7. Código para manejo del driver del motor

```

import sys
import RPi.GPIO as GPIO
import time

def move(step):
    GPIO.setmode(GPIO.BOARD)
    GPIO.setup(12, GPIO.OUT)
    p = GPIO.PWM(12, 50)
    if(step=="0"):
        p.start(2.5)
        p.ChangeDutyCycle(2.5)
        time.sleep(1)
    if(step=="1"):
        p.start(2.5)
        p.ChangeDutyCycle(7.5)
        time.sleep(1)
    if(step=="2"):
        p.start(7.5)
        p.ChangeDutyCycle(12.5)
        time.sleep(1)

    p.stop()
    GPIO.cleanup()

move(sys.argv[1]);

```

8. Simulación HMC5883L Arduino

```

#include <Wire.h>

#define SLAVE_ADDRESS 0x1E

int number = 0;
int numberX=0;
int numberY=0;
int numberZ=0;
char LSBX=0;

```

```

char MSBX=0;
char LSBZ=0;
char MSBZ=0;
char LSBY=0;
char MSBY=0;
int reader=0;
/*int commandStatus=9;
int commandLSBX=4;
int commandMSBX=3;
int commandLSBZ=6;
int commandMSBZ=5;
int commandLSBY=8;
int commandMSBY=7;*/
int commandActual=0;

void setup() {
  Serial.begin(9600); // start serial for output
  // initialize i2c as slave
  Wire.begin(SLAVE_ADDRESS);
  // define callbacks for i2c communication
  Wire.onReceive(receiveData);
  Wire.onRequest(sendData);

  Serial.println("Data Recibida");
}

void loop() {
}

// callback for received data
void receiveData(int byteCount){
  while(Wire.available()) {
    number = Wire.read();
    commandActual=number;
    if(reader==0){
      switch(commandActual){

```

```
case 0:
    Serial.println("CONFIGURATION A");
    reader=1;
    break;
case 1:
    Serial.println("CONFIGURATION B");
    reader=1;
    break;
case 2:
    Serial.println("REGISTER MODE");
    reader=1;
    break;
case 3:
    Serial.println("REGISTER OUTPUT MSB X");
    break;
case 4:
    Serial.println("REGISTER OUTPUT LSB X");
    break;
case 5:
    Serial.println("REGISTER OUTPUT MSB Z");
    break;
case 6:
    Serial.println("REGISTER OUTPUT LSB Z");
    break;
case 7:
    Serial.println("REGISTER OUTPUT MSB Y");
    break;
case 8:
    Serial.println("REGISTER OUTPUT LSB Y");
    break;
case 9:
    Serial.println("REGISTER STATUS");
    break;
}
}else{
    Serial.print("value: ");
    Serial.println(number);
}
```

```
    reader=0;
  }
}
}

// callback for sending data
void sendData(){
  switch(commandActual){
    case 9:
      Wire.write(1);
      randomSeed(millis());
      numberX=random(-2048,2047);
      numberY=random(-2048,2047);
      numberZ=random(-2048,2047);
      LSBX=numberX & 0xFF;
      MSBX=numberX >> 8;
      LSBY=numberY & 0xFF;
      MSBY=numberY >> 8;
      LSBZ=numberZ & 0xFF;
      MSBZ=numberZ >> 8;
      break;
    case 4:
      Wire.write(LSBX);
      break;
    case 3:
      Wire.write(MSBX);
      Serial.println(numberX);
      break;
    case 8:
      Wire.write(LSBY);
      break;
    case 7:
      Wire.write(MSBY);
      Serial.println(numberY);
      break;
    case 6:
      Wire.write(LSBZ);
```

```

        break;
    case 5:
        Wire.write(MSBZ);
        Serial.println(numberZ);
        break;
    }
}

```

9. Simulación ADC128D818 Arduino

```

#include <Wire.h>

#define SLAVE_ADDRESS 0x1D
int number = 0;
int I0=0;
int I1=0;
int I2=0;
int I3=0;
int I4=0;
int I5=0;
int commandActual=0;
int reader=0;

void setup() {
    Serial.begin(9600); // start serial for output
    // initialize i2c as slave
    Wire.begin(SLAVE_ADDRESS);
    // define callbacks for i2c communication
    Wire.onReceive(receiveData);
    Wire.onRequest(sendData);

    Serial.println("Ready!");
}

void loop() {
}

```

```
// callback for received data
void receiveData(int byteCount){
    while(Wire.available()) {
        number = Wire.read();
        commandActual=number;
        if(reader==0){
            switch(commandActual){
                case 0:
                    Serial.println("REGISTER CONFIGURATION");
                    reader=1;
                    break;
                case 7:
                    Serial.println("REGISTER CONVERSION RATE");
                    reader=1;
                    break;
                case 8:
                    Serial.println("REGISTER CHANEL DISABLE");
                    reader=1;
                    break;
                case 11:
                    Serial.println("REGISTER ADVANCED CONFIGURATION");
                    reader=1;
                    break;
                case 12:
                    Serial.println("REGISTER BUSY STATUS");
                    break;
                case 32:
                    Serial.println("REGISTER CHANEL READING I0");
                    break;
                case 33:
                    Serial.println("REGISTER CHANEL READING I1");
                    break;
                case 34:
                    Serial.println("REGISTER CHANEL READING I2");
                    break;
                case 35:
```

```
        Serial.println("REGISTER CHANEL READING I3");
        break;
    case 36:
        Serial.println("REGISTER CHANEL READING I4");
        break;
    case 37:
        Serial.println("REGISTER CHANEL READING I5");
        break;
    }
}
else{
    Serial.print("value: ");
    Serial.println(number);
    reader=0;
}
}
}

// callback for sending data
void sendData(){
    switch(commandActual){
        case 12:
            Wire.write(0);
            break;
        case 32:
            randomSeed(millis());
            I0=random(0,4095);
            Wire.write(I0);
            break;
        case 33:
            randomSeed(millis());
            I1=random(0,4095);
            Wire.write(I1);
            break;
        case 34:
            randomSeed(millis());
            I2=random(0,4095);
            Wire.write(I2);
```

```

        break;
    case 35:
        randomSeed(millis());
        I3=random(0,4095);
        Wire.write(I3);
        break;
    case 36:
        randomSeed(millis());
        I4=random(0,4095);
        Wire.write(I4);
        break;
    case 37:
        randomSeed(millis());
        I5=random(0,4095);
        Wire.write(I5);
        break;
    }
}

```

10. Código para lectura de componentes del módulo de energía y traspaso de información al módulo de comunicaciones

```

/*
Daniel Orozco
@ UVG
*/

#include <fcntl.h>
#include <linux/i2c-dev.h>
#include <linux/i2c.h>
#include <stdio.h>
#include <unistd.h>
#include <time.h>
#include <math.h>

#define n_temperatura 91
#define n_voltaje 46

```

```
#define n_estado_carga 46
#define slave1_address 0x13
#define slave2_address 0x08

#define img_path "/home/pi/Desktop/img.bmp"

int deviceHandle1;
int deviceHandle2;
int readBytes1;
int readBytes2;
char buffer1[7];
char data1[16];
char buffer2[7];
char data2[8];

time_t timer;
char time_buffer[26];
struct tm* tm_info;

struct temperatura_bateria
{
    char id[3];
    char time_stamp[26];
    float valor;
} log_temperatura_bateria[n_temperatura];

struct estado_de_carga_bateria
{
    char id[3];
    char time_stamp[26];
    float valor;
} log_estado_de_carga_bateria[n_estado_carga];

struct voltaje_bateria
{
    char id[3];
    char time_stamp[26];
```

```

    float valor;
} log_voltaje_bateria[n_voltaje];

int nc1 = 0;
int nc2 = 0;
int nc3 = 0;

int binTwosComplementToInt(char binary[], int bits);
int connect_slave_1(int slave_address);
int connect_slave_2(int slave_address);
void logs_to_txts_1();
void logs_to_txts_2();
void estado_de_carga();
int get_estado_de_carga();
int validar_estado_de_carga();
void voltaje_de_baterias();
float get_voltaje_de_baterias();
int validar_voltaje_de_baterias(float valor);
void temperatura_de_baterias();
float get_temperatura_de_baterias();
int validar_temperatura_de_baterias(float valor);
void enviar_img(char *fileName);
void enviar_datos_a_transmitir(int tipo);

void unitary_tests(){
    int n_success = 0;
    int total = 13;
    if(unitary_test_connect_slave(slave1_address)){
        printf("Test unitario - Conectar esclavo: Success\n");
        n_success++;
    }
    else
        printf("Test unitario - Conectar esclavo: Fail\n");
    if(unitary_test_logs_to_txt() > 0){
        printf("Test unitario - Escribir registros a archivo de texto: Success\n");
        n_success++;
    }
}

```

```
else
    printf("Test unitario - Escribir registros a archivo de texto: Fail\n");
if(unitary_test_get_estado_de_carga()){
    printf("Test unitario - Obtener estado de carga: Success\n");
    n_success++;
}
else
    printf("Test unitario - Obtener estado de carga: Fail\n");
if(unitary_test_validar_estado_de_carga()){
    printf("Test unitario - Validar estado de carga: Success\n");
    n_success++;
}
else
    printf("Test unitario - Validar estado de carga: Fail\n");
if(unitary_test_almacenar_estado_de_carga()){
    printf("Test unitario - Almacenar estado de carga: Success\n");
    n_success++;
}
else
    printf("Test unitario - Almacenar estado de carga: Fail\n");
if(unitary_test_get_voltaje_de_baterias()){
    printf("Test unitario - Obtener voltaje de las baterias: Success\n");
    n_success++;
}
else
    printf("Test unitario - Obtener voltaje de las baterias: Fail\n");
if(unitary_test_validar_voltaje_de_baterias()){
    printf("Test unitario - Validar voltaje de las baterias: Success\n");
    n_success++;
}
else
    printf("Test unitario - Validar voltaje de las baterias: Fail\n");
if(unitary_test_almacenar_voltaje_de_baterias()){
    printf("Test unitario - Almacenar voltaje de las baterias: Success\n");
    n_success++;
}
else
```

```
printf("Test unitario - Almacenar voltaje de las baterias: Fail\n");

if(unitary_test_get_temperatura_de_baterias()){
    printf("Test unitario - Obtener temperatura de las baterias: Success\n");
    n_success++;
}
else
    printf("Test unitario - Obtener temperatura de las baterias: Fail\n");
if(unitary_test_validar_temperatura_de_baterias()){
    printf("Test unitario - Validar temperatura de las baterias: Success\n");
    n_success++;
}
else
    printf("Test unitario - Validar temperatura de las baterias: Fail\n");
if(unitary_test_almacenar_temperatura_de_baterias()){
    printf("Test unitario - Almacenar temperatura de las baterias: Success\n");
    n_success++;
}
else
    printf("Test unitario - Almacenar temperatura de las baterias: Fail\n");
if(unitary_test_enviar_imagen()){
    printf("Test unitario - Enviar imagen: Success\n");
    n_success++;
}
else
    printf("Test unitario - Enviar imagen: Fail\n");
if(unitary_test_enviar_registros()){
    printf("Test unitario - Enviar registros: Success\n");
    n_success++;
}
else
    printf("Test unitario - Enviar registros: Fail\n");
printf("%d tests exitosos de %d realizados\n", n_success, total);
printf("%d tests fallidos de %d realizados\n", total-n_success, total);
}
```

```

int unitary_test_connect_slave(int slave_address){
    if (connect_slave_1(slave1_address) == 1)
        return 1;
    return 0;
}

int unitary_test_logs_to_txt(){
    FILE * datos_file;
    datos_file = fopen("/home/pi/Desktop/dataTest.txt","a");
    int write = fprintf(datos_file, "Test\n");
    fclose(datos_file);
    return write;
}

int unitary_test_get_estado_de_carga(){
    char inst[8] = "00000000";
    readBytes1 = write(deviceHandle1, inst, 8);
    readBytes1 = read(deviceHandle1, data1, 16);
    if (readBytes1 != 16)
        return 0;
    return 1;
}

int unitary_test_validar_estado_de_carga(){
    if (!validar_estado_de_carga(655355))
        return 1;
    return 0;
}

int unitary_test_almacenar_estado_de_carga(){
    int value = 0;
    timer = time(NULL);
    tm_info = localtime(&timer);
    strftime(time_buffer, 26, "%d/%m/%Y %H:%M:%S", tm_info);
    strcpy(log_estado_de_carga_bateria[nc1].id, "TN");
    strcpy(log_estado_de_carga_bateria[nc1].time_stamp, time_buffer);
    log_estado_de_carga_bateria[nc1].valor = (float)value;
}

```

```
    return 1;
}

int unitary_test_get_voltaje_de_baterias(){
    char inst[8] = "00000001";
    readBytes1 = write(deviceHandle1, inst, 8);
    readBytes1 = read(deviceHandle1, data1, 16);
    if (readBytes1 != 16)
        return 0;
    return 1;
}

int unitary_test_validar_voltaje_de_baterias(){
    if (!validar_voltaje_de_baterias(16700.0))
        return 1;
    return 0;
}

int unitary_test_almacenar_voltaje_de_baterias(){
    float value = 0.0;
    timer = time(NULL);
    tm_info = localtime(&timer);
    strftime(time_buffer, 26, "%d/%m/%Y %H:%M:%S", tm_info);
    strcpy(log_voltaje_bateria[nc2].id, "P2");
    strcpy(log_voltaje_bateria[nc2].time_stamp, time_buffer);
    log_voltaje_bateria[nc2].valor = value;
    return 1;
}

int unitary_test_get_temperatura_de_baterias(){
    char inst[8] = "00000010";
    readBytes1 = write(deviceHandle1, inst, 8);
    readBytes1 = read(deviceHandle1, data1, 16);
    if (readBytes1 != 16)
        return 0;
    return 1;
}
```

```
int unitary_test_validar_temperatura_de_baterias(){
    if (!validar_temperatura_de_baterias(168.0))
        return 1;
    return 0;
}

int unitary_test_almacenar_temperatura_de_baterias(){
    float value = 0.0;
    timer = time(NULL);
    tm_info = localtime(&timer);
    strftime(time_buffer, 26, "%d/%m/%Y %H:%M:%S", tm_info);
    strcpy(log_temperatura_bateria[nc3].id, "P3");
    strcpy(log_temperatura_bateria[nc3].time_stamp, time_buffer);
    log_temperatura_bateria[nc3].valor = value;
    return 1;
}

int unitary_test_enviar_imagen(){
    enviar_datos_a_transmitir(1);
    return 1;
}

int unitary_test_enviar_registros(){
    enviar_datos_a_transmitir(1);
    return 1;
}

int main (void)
{
    printf("Raspberry Pi I2C\n");

    // inicializar buffers
    buffer1[0] = 0x00;
    buffer2[0] = 0x00;

    time_t start, end;
```

```

double elapsed;
start = time(NULL);
int terminate = -1;

int toca120 = 0;

if (connect_slave_1(slave1_address) == 1){
    while(terminate){
        end = time(NULL);
        elapsed = difftime(end, start);
        if (elapsed > 5400.0) // test 2 - 1h 30 minutos
            //if (elapsed > 20.0) // test 1 - 20 s
                terminate = 0;
        else {
            if(toca120%2 == 0){
                estado_de_carga();
                voltaje_de_baterias();
                nc1++;
                nc2++;
                logs_to_txts_1();
            }
            temperatura_de_baterias();
            nc3++;
            toca120++;
            logs_to_txts_2();
            usleep(6000000); // test 2 - 60s
            //usleep(2000000); // test 1 - 2s
        }
    }
    close(deviceHandle1);
    //logs_to_txts();
}

/* if (connect_slave_2(slave2_address) == 1){
    // drive some tests
    //send_picture();
    enviar_datos_a_transmitir(0);
}

```

```

    //enviar_datos_a_transmitir(1);
    // close connection and return
    close(deviceHandle2);
} */

return 0;
}

int binTwosComplementToInt(char binary[], int bits){
    int potencia = pow(2, bits-1);
    int res = 0;
    for(int i=0; i<bits; i++){
        if (i==0 && binary[i]!='0')
            res += potencia * -1;
        else
            res += (binary[i]-'0')*potencia;
        potencia /= 2;
    }
    return res;
}

int connect_slave_1(int slave_address){
    // open device on /dev/i2c-0
    if ((deviceHandle1 = open("/dev/i2c-1", O_RDWR)) < 0) {
        printf("Error: no se pudo abrir la interfaz I2C! %d\n", deviceHandle1);
        return 0;
    }

    // connect to arduino as i2c slave
    if (ioctl(deviceHandle1, I2C_SLAVE, slave_address) < 0) {
        printf("Error: no hay respuesta del esclavo!\n");
        return 0;
    }

    // begin transmission and request acknowledgement
    readBytes1 = write(deviceHandle1, buffer1, 1);
    if (readBytes1 != 1)

```

```

    {
        printf("Error: No se recibio el ACK-Bit, no se pudo establecer la conexion!\n");
        return 0;
    }
    printf("Esclavo %d conectado\n", slave_address);
    return 1;
}

int connect_slave_2(int slave_address){
    // open device on /dev/i2c-0
    if ((deviceHandle2 = open("/dev/i2c-1", O_RDWR)) < 0) {
        printf("Error: no se pudo abrir la interfaz I2C! %d\n", deviceHandle2);
        return 0;
    }

    // connect to arduino as i2c slave
    if (ioctl(deviceHandle2, I2C_SLAVE, slave_address) < 0) {
        printf("Error: no hay respuesta del esclavo!\n");
        return 0;
    }

    // begin transmission and request acknowledgement
    readBytes2 = write(deviceHandle2, buffer2, 1);
    if (readBytes2 != 1)
    {
        printf("Error: No se recibio el ACK-Bit, no se pudo establecer la conexion!\n");
        return 0;
    }
    printf("Esclavo %d conectado\n", slave_address);
    return 1;
}

void logs_to_txts_1(){
    FILE * datos_file;

    datos_file = fopen("/home/pi/Desktop/dataC1.txt", "a");
    for(int i=nc1-1; i<nc1; i++){

```

```

char entry[36] = "";
strcpy(entry, log_estado_de_carga_bateria[i].id);
strcat(entry, ",");
strcat(entry, log_estado_de_carga_bateria[i].time_stamp);
strcat(entry, ",%f\n");
fprintf(datos_file, entry, log_estado_de_carga_bateria[i].valor);
}
fclose(datos_file);

datos_file = fopen("/home/pi/Desktop/dataC2.txt","a");
for(int i=nc2-1; i<nc2; i++){
char entry[36] = "";
strcpy(entry, log_voltaje_bateria[i].id);
strcat(entry, ",");
strcat(entry, log_voltaje_bateria[i].time_stamp);
strcat(entry, ",%f\n");
fprintf(datos_file, entry, log_voltaje_bateria[i].valor);
}
fclose(datos_file);
}

void logs_to_txts_2(){
FILE * datos_file;

datos_file = fopen("/home/pi/Desktop/dataC3.txt","a");
for(int i=nc3-1; i<nc3; i++){
char entry[36] = "";
strcpy(entry, log_temperatura_bateria[i].id);
strcat(entry, ",");
strcat(entry, log_temperatura_bateria[i].time_stamp);
strcat(entry, ",%f\n");
fprintf(datos_file, entry, log_temperatura_bateria[i].valor);
}
fclose(datos_file);
}

void estado_de_carga(){

```

```

int value = get_estado_de_carga();
value = (validar_estado_de_carga(value)==1) ? value : -1;
timer = time(NULL);
tm_info = localtime(&timer);
strftime(time_buffer, 26, "%d/%m/%Y %H:%M:%S", tm_info);
strcpy(log_estado_de_carga_bateria[nc1].id, "P1");
strcpy(log_estado_de_carga_bateria[nc1].time_stamp, time_buffer);
log_estado_de_carga_bateria[nc1].valor = (float)value;
}

```

```

int get_estado_de_carga(){
    printf("Obteniendo valores de BQ28Z610\n");
    char inst[8] = "00000000";
    int value = 0;
    readBytes1 = write(deviceHandle1, inst, 8);

    // read success
    readBytes1 = read(deviceHandle1, data1, 16);
    if (readBytes1 != 16)
    {
        printf("Error: Datos no recibidos!\n");
        value = -1;
    }
    else
    {
        for(int i=0; i<readBytes1; i++){
            int a = (int)pow(2,i);
            int b = data1[i]-'0';
            value += a*b;
        }
    }
    printf("BQ28Z610 respuesta: %d\n", value);
    return value;
}

```

```

int validar_estado_de_carga(int valor){
    return ((valor>=0)&&(valor<=65535));
}

```

```

}

void voltaje_de_baterias(){
    float value = get_voltaje_de_baterias();
    value = (validar_voltaje_de_baterias(value)==1) ? value : -1.0;
    timer = time(NULL);
    tm_info = localtime(&timer);
    strftime(time_buffer, 26, "%d/%m/%Y %H:%M:%S", tm_info);
    strcpy(log_voltaje_bateria[nc2].id, "P2");
    strcpy(log_voltaje_bateria[nc2].time_stamp, time_buffer);
    log_voltaje_bateria[nc2].valor = value;
}

float get_voltaje_de_baterias(){
    printf("Obteniendo valores de INA3221\n");
    char inst[8] = "00000001";
    float value = 0.0;
    readBytes1 = write(deviceHandle1, inst, 8);

    // read success
    readBytes1 = read(deviceHandle1, data1, 16);
    if (readBytes1 != 16)
    {
        printf("Error: Datos no recibidos!\n");
        value = -1;
    }
    else
    {
        int comp2_value = binTwosComplementToInt(data1, 16);
        int f_comp2_value = (float) comp2_value;
        value = f_comp2_value/200.0;
    }
    printf("INA3221 respuesta: %f\n", value);
    return value;
}

int validar_voltaje_de_baterias(float valor){

```

```

    return ((valor>=-16400.0)&&(valor<=16400.0));
}

void temperatura_de_baterias(){
    float value = get_temperatura_de_baterias();
    value = (validar_temperatura_de_baterias(value)==1) ? value : -1.0;
    timer = time(NULL);
    tm_info = localtime(&timer);
    strftime(time_buffer, 26, "%d/%m/%Y %H:%M:%S", tm_info);
    strcpy(log_temperatura_bateria[nc3].id, "P3");
    strcpy(log_temperatura_bateria[nc3].time_stamp, time_buffer);
    log_temperatura_bateria[nc3].valor = value;
}

float get_temperatura_de_baterias(){
    printf("Obteniendo valores de TMP101\n");
    char inst[8] = "00000010";
    float value = 0.0;
    readBytes1 = write(deviceHandle1, inst, 8);

    // read success
    readBytes1 = read(deviceHandle1, data1, 16);
    if (readBytes1 != 16)
    {
        printf("Error: Datos no recibidos!\n");
        value = -1;
    }
    else
    {
        int comp2_value = binTwosComplementToInt(data1, 12);
        int f_comp2_value = (float) comp2_value;
        value = f_comp2_value/16.0;
    }
    printf("TMP101 respuesta: %f\n", value);
    return value;
}

```

```

int validar_temperatura_de_baterias(float valor){
    return ((valor>=-128.0)&&(valor<=128.0));
}

void enviar_img(char *fileName){
    FILE *file = fopen(fileName, "r");
    int c;
    char img_chr[1];

    if (file == NULL) return NULL; //could not open file

    while ((c = fgetc(file)) != EOF) {
        img_chr[0] = (char)c;
        readBytes2 = write(deviceHandle2, img_chr, 1);
    }
}

void enviar_datos_a_transmitir(int tipo){
    printf("Conectando con modulo de comunicaciones\n");
    char inst[8] = "00000000";
    readBytes2 = write(deviceHandle2, inst, 8);

    char value[64];
    char pixel[32] = "11010101110101011101010111010101";

    // read success
    readBytes2 = read(deviceHandle2, data2, 8);
    if (readBytes2 != 8)
    {
        printf("Error: Datos no recibidos!\n");
    }
    else
    {
        printf("Resuesta del modulo de comunicaciones: %s\n", data2);
        if(strcmp(data2, "00000001\3")==0){
            switch(tipo){
                // Logs
            }
        }
    }
}

```

case 0:

```

printf("Enviando 6.61 KBytes de logs\n");
timer = time(NULL);
tm_info = localtime(&timer);
strftime(time_buffer, 26, "%d/%m/%Y %H:%M:%S", tm_info);
printf("At time: ");
puts(time_buffer);

for(int i=0; i<n_estado_carga; i++){
    readBytes2 = write(deviceHandle2, log_estado_de_carga_bateria[i].id, 3);
    readBytes2 = write(deviceHandle2, log_estado_de_carga_bateria[i].time_stamp, 26);
    snprintf(value, sizeof value, "%f", log_estado_de_carga_bateria[i].valor);
    readBytes2 = write(deviceHandle2, value, 8);
}

for(int i=0; i<n_voltaje; i++){
    readBytes2 = write(deviceHandle2, log_voltaje_bateria[i].id, 3);
    readBytes2 = write(deviceHandle2, log_voltaje_bateria[i].time_stamp, 26);
    snprintf(value, sizeof value, "%f", log_voltaje_bateria[i].valor);
    readBytes2 = write(deviceHandle2, value, 8);
}

for(int i=0; i<n_temperatura; i++){
    readBytes2 = write(deviceHandle2, log_temperatura_bateria[i].id, 3);
    readBytes2 = write(deviceHandle2, log_temperatura_bateria[i].time_stamp, 26);
    snprintf(value, sizeof value, "%f", log_temperatura_bateria[i].valor);
    readBytes2 = write(deviceHandle2, value, 8);
}

printf("Transmision finalizada\n");
timer = time(NULL);
tm_info = localtime(&timer);
strftime(time_buffer, 26, "%d/%m/%Y %H:%M:%S", tm_info);
printf("At time: ");
puts(time_buffer);
break;

```

case 1:

```

printf("Enviando imagen de 900 KBytes\n");
timer = time(NULL);
tm_info = localtime(&timer);
strftime(time_buffer, 26, "%d/%m/%Y %H:%M:%S", tm_info);
printf("At time: ");
puts(time_buffer);

enviar_img(img_path);

printf("Transmision finalizada\n");
timer = time(NULL);
tm_info = localtime(&timer);
strftime(time_buffer, 26, "%d/%m/%Y %H:%M:%S", tm_info);
printf("At time: ");
puts(time_buffer);
break;
default: break;
}
}
}
}
}

```

11. Código para mecanismo de redundancia

```

/*
Daniel Orozco
@ UVG
*/

#include <stdio.h>
#include <unistd.h> //Used for UART
#include <fcntl.h> //Used for UART
#include <termios.h> //Used for UART
#include <time.h>
#include <wiringPi.h>
#include <unistd.h>
#include <sys/types.h>

```

```
#define uart_device "/dev/ttyS0"
#define n_retry_bus 4
#define n_retry_read 10
#define n_retry_read_bus_master 5
#define max_time_retry_bus 60
#define max_time_retry_read 2
#define max_time_retry_read_bus_master 2
#define max_buffer_size 20
#define alive_char '1'
#define bus_master '1'

#define ledBusMaster 0

#define i2c_path "/home/pi/Desktop/./master"

int uart0_filestream = -1;

//----- TX BYTES -----
unsigned char tx_buffer[20];
unsigned char *p_tx_buffer;
//----- RX BYTES -----
unsigned char rx_buffer[max_buffer_size];

void unitary_tests(){
    int n_success = 0;
    int total = 8;
    if(unitary_test_uart()){
        printf("Test unitario - Conectar UART: Success\n");
        n_success++;
    }
    else
        printf("Test unitario - Conectar UART: Fail\n");
    if(unitary_test_send_alive_signal()){
        printf("Test unitario - Enviar señal: Success\n");
        n_success++;
    }
}
```

```
else
    printf("Test unitario - Enviar señal: Fail\n");
if(unitary_test_read_alive_signal()){
    printf("Test unitario - Leer señal: Success\n");
    n_success++;
}
else
    printf("Test unitario - Leer señal: Fail\n");
if(unitary_test_encender_led()){
    printf("Test unitario - Encender LED: Success\n");
    n_success++;
}
else
    printf("Test unitario - Encender LED: Fail\n");
if(unitary_test_apagar_led()){
    printf("Test unitario - Apagar LED: Success\n");
    n_success++;
}
else
    printf("Test unitario - Apagar LED: Fail\n");
if(unitary_test_activar_i2c()){
    printf("Test unitario - Activar I2C: Success\n");
    n_success++;
}
else
    printf("Test unitario - Activar I2C: Fail\n");
if(unitary_test_determinar_estado()){
    printf("Test unitario - Determinar estado: Success\n");
    n_success++;
}
else
    printf("Test unitario - Determinar estado: Fail\n");
if(unitary_test_cambiar_estado()){
    printf("Test unitario - Cambiar estado: Success\n");
    n_success++;
}
else
```

```

    printf("Test unitario - Cambiar estado: Fail\n");
    printf("%d tests exitosos de %d realizados\n", n_success, total);
    printf("%d tests fallidos de %d realizados\n", total-n_success, total);
}

int unitary_test_uart(){
    uart0_filestream = open("/dev/ttyS0", O_RDWR | O_NOCTTY | O_NDELAY);    //Open in non
    blocking read/write mode
    if (uart0_filestream != -1)
        return 1;
    return 0;
}

int unitary_test_send_alive_signal(){
    return send_alive_signal(uart0_filestream);
}

int unitary_test_read_alive_signal(){
    return !read_alive_signal(uart0_filestream);
}

int unitary_test_encender_led(){
    digitalWrite(ledBusMaster, HIGH);
    return 1;
}

int unitary_test_apagar_led(){
    digitalWrite(ledBusMaster, LOW);
    return 1;
}

int unitary_test_activar_i2c(){
    pid_t pid = fork();
    if(pid==0){
        static char *argv[] = {"", "", NULL};
        execlv(i2c_path, argv);
        return 1;
    }
}

```

```

    }
    return 0;
}

int unitary_test_determinar_estado(){
    return read_bus_master(uart0_filestream);
}

int unitary_test_cambiar_estado(){
    return read_bus_master(uart0_filestream);
}

void launch_i2c(){
    pid_t pid = fork();
    if(pid==0){
        static char *argv[] = {"", "", NULL};
        execv(i2c_path, argv);
    }
}

void set_bus_master(char c){
    tx_buffer[1] = c;
}

void as_main(){
    digitalWrite(ledBusMaster, HIGH);
    set_bus_master(bus_master);
    printf("Running as Main\n");
    // run i2c
    launch_i2c();
    int alive = -1;
    while(1){
        usleep(1000000);
        int s = send_alive_signal(uart0_filestream);
    }
}

```

```

void as_backup(){
    digitalWrite(ledBusMaster, LOW);
    printf("Running as Backup\n");
    int alive = -1;
    while(alive){
        printf("Main alive\n");
        rx_buffer[0] = '\0';
        rx_buffer[1] = '\0';
        alive = read_alive_signal(uart0_filestream);
    }
    as_main();
}

int main (void){

    //int uart0_filestream = -1;
    uart0_filestream = open("/dev/ttyS0", O_RDWR | O_NOCTTY | O_NDELAY);    //Open in non
blocking read/write mode

    if (uart0_filestream == -1){
        //ERROR - CAN'T OPEN SERIAL PORT
        printf("Error - Unable to open UART. Ensure it is not in use by another application\n");
    }

    if(wiringPiSetup() == -1){ //when initialize wiringPi failed, print message to screen
        printf("wiringPi setup failed !\n");
    }

    pinMode(ledBusMaster, OUTPUT);

    struct termios options;
    tcgetattr(uart0_filestream, &options);
    options.c_cflag = B9600 | CS8 | CLOCAL | CREAD;    //<Set baud rate
    options.c_iflag = IGNPAR;
    options.c_oflag = 0;
    options.c_lflag = 0;
    tcflush(uart0_filestream, TCIFLUSH);

```

```

tcsetattr(uart0_filestream, TCSANOW, &options);

p_tx_buffer = &tx_buffer[0];
*p_tx_buffer++ = alive_char;
*p_tx_buffer++ = bus_master;

if (read_bus_master(uart0_filestream))
    as_main();
else
    as_backup();
}

int reopen_uart(int uart_filestream){
    int try = 0;
    uart_filestream = open(uart_device, O_RDWR | O_NOCTTY | O_NDELAY);    //Open in non blocking
read/write mode
    if (uart_filestream != -1)
        return uart_filestream;
    while(try<n_retry_bus){
        if (try == n_retry_bus-1){
            printf("Last try to re open UART\n");
            usleep(max_time_retry_bus * 1000000);
            close(uart_filestream);
            uart_filestream = open(uart_device, O_RDWR | O_NOCTTY | O_NDELAY);    //Open in non
blocking read/write mode
            if (uart_filestream != -1)
                return uart_filestream;
        }
        if (uart_filestream == -1){
            printf("Trying to re open UART\n");
            if (try == n_retry_bus){
                printf("Unable to open UART\n");
            }
            close(uart_filestream);
            uart_filestream = open(uart_device, O_RDWR | O_NOCTTY | O_NDELAY);    //Open in non
blocking read/write mode
            if (uart_filestream != -1)

```

```

        return uart_filestream;
    }
    try++;
}
return -1;
}

int send_alive_signal(int uart_filestream){
    if (uart_filestream == -1)
        uart_filestream = reopen_uart(uart_filestream);
    if (uart_filestream == -1){
        printf("Unable to send data\n");
        return 0;
    }
    rx_buffer[0] = '\0';
    rx_buffer[1] = '\0';
    int count = write(uart_filestream, &tx_buffer[0], (p_tx_buffer - &tx_buffer[0]));    //Filestream, bytes
    to write, number of bytes to write
    if (count < 0)
        return 0;
    //printf("Main ALIVE!\n");
    return 1;
}

int read_alive_signal(int uart_filestream){
    if (uart_filestream == -1)
        uart_filestream = reopen_uart(uart_filestream);
    if (uart_filestream == -1){
        printf("Unable to read data\n");
        return 0;
    }
    int retry = 0;
    while(retry < n_retry_read){
        int rx_length = read(uart_filestream, (void*)rx_buffer, max_buffer_size-1);    //Filestream, buffer to
        store in, number of bytes to read (max)
        rx_buffer[rx_length] = '\0';
        if(rx_buffer[0] == alive_char){

```

```

        rx_buffer[0] = '\0';
        rx_buffer[1] = '\0';
        return 1;
    }
    rx_buffer[0] = '\0';
    rx_buffer[1] = '\0';
    //send_alive_signal(uart_filestream);
    usleep(max_time_retry_read * 1000000);
    retry++;
}
return 0;
}

int read_bus_master(int uart_filestream){
    printf("Reading bus master\n");
    if (uart_filestream == -1)
        uart_filestream = reopen_uart(uart_filestream);
    if (uart_filestream == -1){
        printf("Unable to read data\n");
        return 0;
    }
    int retry = 0;
    while(retry < n_retry_read_bus_master){
        int rx_length = read(uart_filestream, (void*)rx_buffer, max_buffer_size-1); //Filestream, buffer to
store in, number of bytes to read (max)
        rx_buffer[rx_length] = '\0';
        if(rx_buffer[1] == '2')
            return 0;
        rx_buffer[0] = '\0';
        rx_buffer[1] = '\0';
        usleep(max_time_retry_read_bus_master * 1000000);
        retry++;
    }
    return 1;
}

```

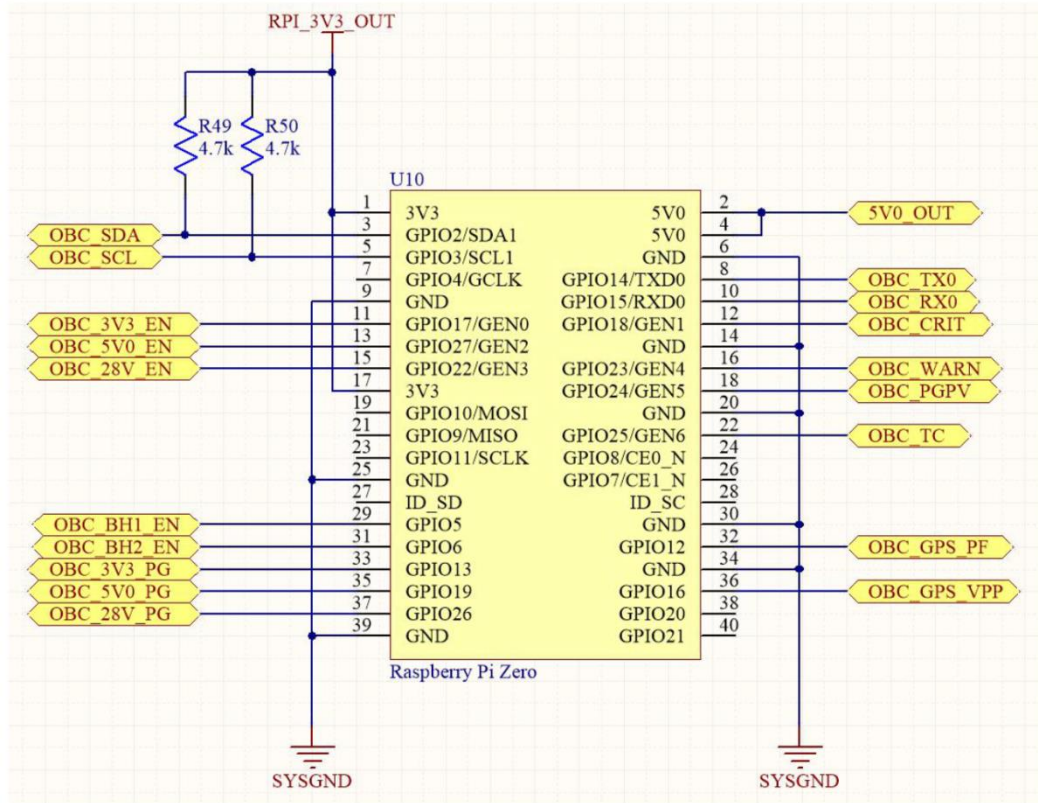
12. Código desarrollado para reiniciar computadoras para pruebas del mecanismo de redundancia

```
/*
Daniel Orozco
@ UVG
*/

#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main (void){
    srand(time(NULL));
    int seconds = (rand() % 1500) + 300; // random between 5-30 mins [300 - 1800 in seconds]
    FILE * datos_file;
    datos_file = fopen("/home/pi/Desktop/reboots.txt", "a");
    fprintf(datos_file, "%d\n", seconds);
    fclose(datos_file);
    usleep(seconds*1000000);
    int reboot = system("sudo reboot");
    return 0;
}
```

Figura 162. Interfaces de Raspberry Pi Zero.



D. CÓDIGO DE SIMULACIÓN DEL SISTEMA NO LINEAL Y CONTROLADORES

1. Programa principal - controladores PID y LQR

```

% -----
% -----
% Simulacion CubeSat
%
% Este codigo toma en cuenta principalmente el control del un satelite
% tipo
% Cubesat, este hace uso de algunas perturbaciones para hacer de la
% simulacion una mas realista
%
% Hecho por:
% Diego Alejandro Rodriguez Garcia
%
% con colaboracion de:
% Msc. Miguel Enrique Zea Arenales
% -----
% -----
    
```

```

% -----
% -----
% Definicion de tiempo de ejecucion del programa
% -----
% -----

% Tiempo de paso de la simulacion
dt = 0.0005;
t = 0:dt:1;
% Cantidad de orbitas a analizar
simtime = 2;
N = length(t)-1;
% Numero de iteraciones a realizar
K = simtime*N;

% -----
% -----
% Parametros de la orbita
% -----
% -----

% Tiempo de duracion de una sola orbita (mins)
T = 90;

% Tiempo real de simulacion (s)
dT = dt*T*60;

% Radio de la tierra (km)
rE = 6371;

% Radio de orbita del satelite [RadioTierra + Altura orbita] (km)
rS = rE + 408;

% Normalizacion de la orbita para posteriormente ser graficable
r = rS/rE;

% -----
% -----
% Especificaciones de inercia del satelite a estudiar (kg * m^2)
% -----
% -----

% Momento de Inercia en el eje X
Ixx = 0.002136;

% Momento de Inercia en el eje Y
Iyy = 0.000935;

% Momento de Inercia en el eje Z
Izz = 0.001996;

% Matriz de inercia, Es un diagonal por asumir un cuerpo rigido
I = diag([Ixx, Iyy, Izz]);

```

```

% -----
----
% Se hace uso de cuaterniones para manejar las rotaciones del satélite.
La
% razón del mismo es por ser valores constantes y por ser una fuente de
ma-
% nejo de datos sin singularidades.
%
% Nota:
% MATLAB hace uso de los cuaterniones de la siguiente manera:
%  $q = q_0 + q_1 i + q_2 j + q_3 k$ 
% Esto quiere decir que eta y los episilons son:
%  $q = n + e_1 x + e_2 y + e_3 z$ 
% -----
----

% -----
----
% Condición inicial de los ángulos de Euler
% -----
----
% roll (X)
IniRoll = 0*pi/180;

% pitch (Y)
IniPitch = 0*pi/180;

% yaw (Z)
IniYaw = 0*pi/180;

InEul = [IniRoll, IniPitch, IniYaw];

% Transformador a cuaterniones
Inq = eul2quat(InEul);

% -----
----
% Ángulos de Euler de referencia
% -----
----
% roll (X)
RefRoll = 0*pi/180;

% pitch (Y)
RefPitch = 0*pi/180;

% yaw (Z)
RefYaw = 44.99*pi/180;

RefEul = [RefRoll, RefPitch, RefYaw];
% Transformador a cuaterniones
Refq = eul2quat(RefEul);

% -----
----

```

```

% Inclusion cuaternion de error
% -----
----
Actq = quatinv(Inq);
qError = quatmultiply(Refq,Actq);
qEeta = qError(1);
qEeps = qError(2:4)';

% -----
----
% Velocidades angulares iniciales al momento de lanzarse al espacio
% -----
----
w = 50*[1; -1; 1];

% -----
----
% Condiciones iniciales
% -----
----
x0 = [qEeps; qEeta; w];
x = x0;

% -----
----
% Accion de control inicial
% -----
----
u0 = [0;0;0];
u = u0;

% -----
----
% Errores acumulados y actuales
% -----
----
e = 0; E = 0;

% -----
----
% Matriz K LQR
% -----
----
KM = [1, 0, 0, 0, 0.025, 0, 0;...
      0, 1, 0, 0, 0, 0.025, 0;...
      0, 0, 1, 0, 0, 0, 0.025];

% Storage arrays for the time evolution of the state and control inputs
X = zeros(length(x), K);
U = zeros(length(u0), K);
P = zeros(length(u0), K);
TS = zeros(3, K);
TA = zeros(3, K);
TM = zeros(3, K);
TG = zeros(3, K);

```

```

% Vectors that define the orbit's plane -----
-----
%
a = [1,-2,1];
b = [-a(3),0,a(1)];
wo = 2*pi; % orbit frequency
xt = r*cos(wo*t)*a(1) + r*sin(wo*t)*b(1); % orbit's relative x
coordinate
yt = r*cos(wo*t)*a(2) + r*sin(wo*t)*b(2); % orbit's relative y
coordinate
zt = r*cos(wo*t)*a(3) + r*sin(wo*t)*b(3); % orbit's relative z
coordinate

% Creacion de vectores para las perturbaciones
Ts = [0;0;0];
Ta = [0;0;0];
Tm = [0;0;0];
Tg = [0;0;0];

% Simulate the dynamical system for the given simulation time using the
RK4
% discretization method.
for k = 0:K-1

    % Entrada de control LQR
    u = -KM * x;
    % habilita el control PID
    % [u, e, E] = threeaxis_pid(x, e, E, dt);
    % Entrada de perturbaciones
    % creacion de variacion de area para presiones
    Arear_x = 0.01;
    Area_y = 0.01;
    Area_z = 0.01;
    Area_press = [Arear_x,0,0; 0,Area_y,0; 0,0,Area_z];
    Area_press2 = [Arear_x; Area_y; Area_z];
    % coeficiente de reflexion
    Qreflection = [1;1;1];

    % multiplicador de campo magnetico
    multmag = 2;

    % Angulo de rotacion por gravedad
    tetha = pi()/4;

    % inicializacion de perturbaciones
    Ts = perturbacion_solar(Area_press, Qreflection);
    Ta = perturbacion_aerodinamica(Area_press2);
    Tm = perturbacion_magnetica(rS, multmag);
    Tg = perturbacion_gravitacional(rS, Izz, Iyy, tetha);

    Tp = (Ts + Ta + Tm + Tg)*1E5;

    % introduccion de variables de estado, torques de control, torques
de
    % perturbacion al modelado del sistema

```

```

x = modelado_del_satelite(x, u, Tp, I, dt);

% Store state and inputs for further analysis
X(:,k+1) = x;
U(:,k+1) = u;
P(:,k+1) = Tp;
TS(:,k+1) = Ts;
TA(:,k+1) = Ta;
TM(:,k+1) = Tm;
TG(:,k+1) = Tg;
% Extract the current pose and angular velocity from the state
vector
eps = x(1:3);
eta = x(4);
w = x(5:7);
end
whitebg('white');

figure;
t = 0:dt:simtime;
subplot(2,2,1);
plot(t*T, [TS(:,1), TS]', 'LineWidth', 1);
title('Torque por presión de radiación solar', 'FontSize', 18)
xlabel('$t$ (min)', 'Interpreter', 'latex', 'FontSize', 16);
ylabel('$Torque$ (N m)', 'Interpreter', 'latex', ...
'FontSize', 16);
l = legend('$\tau_x$', '$\tau_y$', '$\tau_z$', ...
'Location', 'northoutside', 'Orientation', 'horizontal');
set(l, 'Interpreter', 'latex', 'FontSize', 14);

subplot(2,2,2);
plot(t*T, [TA(:,1), TA]', 'LineWidth', 1);
title('Torque aerodinámico', 'FontSize', 18)
xlabel('$t$ (min)', 'Interpreter', 'latex', 'FontSize', 16);
ylabel('$Torque$ (N m)', 'Interpreter', 'latex', ...
'FontSize', 16);
l = legend('$\tau_x$', '$\tau_y$', '$\tau_z$', ...
'Location', 'northoutside', 'Orientation', 'horizontal');
set(l, 'Interpreter', 'latex', 'FontSize', 14);

subplot(2,2,3);
plot(t*T, [TM(:,1), TM]', 'LineWidth', 1);
title('Torque magnético', 'FontSize', 18)
xlabel('$t$ (min)', 'Interpreter', 'latex', 'FontSize', 16);
ylabel('$Torque$ (N m)', 'Interpreter', 'latex', ...
'FontSize', 16);
set(l, 'Interpreter', 'latex', 'FontSize', 14);

subplot(2,2,4);
plot(t*T, [TG(:,1), TG]', 'LineWidth', 1);
title('Torque gravitacional', 'FontSize', 18)
xlabel('$t$ (min)', 'Interpreter', 'latex', 'FontSize', 16);
ylabel('$Torque$ (N m)', 'Interpreter', 'latex', ...
'FontSize', 16);
set(l, 'Interpreter', 'latex', 'FontSize', 14);

```

```

u_max = 6; % maximum torque per axis (in N m)

figure;
t = 0:dt:simtime;

subplot(2,2,1);

plot(t*T, [x0(1:4), X(1:4,:)'], 'LineWidth', 1);
title('Observación de estados de cuaterniones', 'FontSize', 18);
xlabel('$t$ (min)', 'Interpreter', 'latex', 'FontSize', 16);
ylabel('$Cuaternion$', 'Interpreter', 'latex', ...
    'FontSize', 16);
l = legend('$\epsilon_x$', '$\epsilon_y$', '$\epsilon_z$', '$\eta$', ...
    'Location', 'northoutside', 'Orientation', 'horizontal');
set(l, 'Interpreter', 'latex', 'FontSize', 14);

subplot(2,2,3);

plot(t*T, [P(:,1), P]'), 'LineWidth', 1);
title('Torque de perturbaciones (total)', 'FontSize', 18);
xlabel('$t$ (min)', 'Interpreter', 'latex', 'FontSize', 16);
ylabel('$Torques$ (N m)', 'Interpreter', 'latex', ...
    'FontSize', 16);
l = legend('$\tau_{xEx}(t)$', '$\tau_{yEx}(t)$', '$\tau_{zEx}(t)$', ...
    'Location', 'northoutside', 'Orientation', 'horizontal');
set(l, 'Interpreter', 'latex', 'FontSize', 14);

subplot(2,2,2);
% subplot(2,1,1);

plot(t*T, [x0(5:7), X(5:7,:)'], 'LineWidth', 1);
%title('Observación de las velocidades angulares del satélite',
'FontSize', 18);
xlabel('$t$ (min)', 'Interpreter', 'latex', 'FontSize', 16);
ylabel('${\boldmath}\omega\}$ (rad/s)', 'Interpreter', 'latex', ...
    'FontSize', 16);
l = legend('$\omega_x(t)$', '$\omega_y(t)$', '$\omega_z(t)$', ...
    'Location', 'northoutside', 'Orientation', 'horizontal');
set(l, 'Interpreter', 'latex', 'FontSize', 14);
subplot(2,2,4);
% subplot(2,1,2);
%plot(t*T, [u0, U]'), 'LineWidth', 1);

plot(t*T, [U(:,1), U]'), 'LineWidth', 1);
% hold on;
%title('Torque de control', 'FontSize', 18);
% hline = reffline([0 u_max]);
% hline.Color = 'r';
% hline2 = reffline([0 -u_max]);
% hline2.Color = 'r';
%plot(t*T, u_max*[ones(length(t)), -ones(length(t))], '--b',
'LineWidth', 1);
xlabel('$t$ (min)', 'Interpreter', 'latex', 'FontSize', 16);
ylabel('${\boldmath}\tau\}$ (N m)', 'Interpreter', 'latex', ...
    'FontSize', 16);

```

```

l = legend('$\tau_x(t)$', '$\tau_y(t)$', '$\tau_z(t)$', ...
    'Location', 'northoutside', 'Orientation','horizontal');
set(l, 'Interpreter', 'latex', 'FontSize', 14);
% hold off;

figure;
% Set-up the animated plot of the spacecraft orbiting the Earth

original = imread('mundo.jpg');
[T,W,U] = sphere();
mundo = surf(T,W,-U);
axis('square');
h = findobj('Type','surface');
set(h, 'CData', original, 'FaceColor', 'texturemap')
view(100, 20);
xlim([-2, 2]);
ylim([-2.5, 2.5]);
zlim([-2, 2]);
hold on;
plot3(xt,yt,zt, '--w', 'Linewidth', 1);
orgn = zeros(3,1); % spacecraft's COM position
size = 0.1; % size of satellite (illustrative)
whitebg('white');
% Plot and animate the time evolution of the satellite's attitude
during
% the simulation

for k = 0:K-1

    rotate(mundo, [0 0 1], 0.0625);
    % Update the spacecraft's COM position within the orbit
    orgn(1) = xt(mod(k,N)+1);
    orgn(2) = yt(mod(k,N)+1);
    orgn(3) = zt(mod(k,N)+1);

    x = X(:,k+1); % get the state
    % Get the current attitude as unit quaternion
    eps = x(1:3);
    eta = x(4);
    %    w = x(5:7);

    R = quat2rotm([eta, eps]); % get the attitude's rotation matrix

    % 3D plot the spacecraft in orbit
    if(k == K-1)
        draw_cube(orgn, size, R, 1);
    else
        draw_cube(orgn, size, R, 0);
    end
end

hold off;

```

2. Función para el controlador PID

```

function [u, e, E] = threeaxis_pid(x, e_old, E_old, h)
% THREEAXIS_PID PID para controlar un satélite
% Es un controlador que se le puede modificar sus parámetros los cuales
% controlan las velocidades angulares de un satélite. A través de esto
se
% hace una operación con errores actuales, pasados y promedios de los
% mismos para llevar a cabo el PID, se regresa esto en una matriz de 3
% líneas.
%
=====
==
% A MODIFICAR
%
=====
==
% Constantes PID eje x:
% -----
----
kPx = 0.01;
kIx = 0.002;
kDx = 0;
% -----
----
% Constantes PID eje y:
% -----
----
kPy = 0.01;
kIy = 0.001;
kDy = 0;
% -----
----
% Constantes PID eje z:
% -----
----
kPz = 0.01;
kIz = 0.005;
kDz = 0;
% -----
----

% Combina los valores y los normaliza
KP = [kPx, kPy, kPz]';
KI = h*[kIx, kIy, kIz]';
KD = (1/h)*[kDx, kDy, kDz]';

% Calcula el error acumulado y el actual
rw = zeros(3,1);
e = (rw - x(5:7));
E = E_old + e;

% Desarrolla el control PID
uP = e;
uI = E;
uD = e - e_old;

```

```

u = KP.*uP + KI.*uI + KD.*uD;

end

```

3. Función del modelo no lineal del satélite

```

function xn = modelado_del_satelite(x, u, Tp, I, h)
% función para modelar el sistema del satélite que es el no lineal,
también
% tiene el método de Runge Kutta de grado 4 para hacer la aproximación
de
% la solución de la ecuación diferencial del sistema.
f = @(x,u,Tp) [0.5*skew2(x(5:7))*x(1:4);...
    I^(-1)*(-skew(x(5:7))*I*x(5:7)+ u + Tp)];

k1 = f(x,u,Tp);
k2 = f(x+(h/2)*k1,u,Tp);
k3 = f(x+(h/2)*k2,u,Tp);
k4 = f(x+h*k3,u,Tp);
y = x + (h/6)*(k1+2*k2+2*k3+k4);

xn = [y(1:4)/norm(y(1:4)); y(5:7)];

end

```

4. Función de la perturbación aerodinámica

```

function PA = perturbacion_aerodinamica(Area_press)
% Modelo simplificado de la perturbación aerodinámica que afecta al
CubeSat
VelOrb = 7504;
DragC = 2;
AtmosDensity = 1E-13;
Center_of_pressure = 0.05;

cons = 0.5* AtmosDensity* DragC*Center_of_pressure*(VelOrb)^2;

PA = cons * Area_press;

end

```

5. Función de la perturbación gravitacional

```

function PG = perturbacion_gravitacional(rS, Izz, Iyy, tetha)
% Modelo simplificado de la perturbación gravitacional que afecta al
CubeSat
Gravconst = 3.986E14;

```

```

cons = 1.5 * Gravconst / ((rS*1000)^3);
res = abs(Izz - Iyy);
PGs = cons * res * sin(2*tetha);
PG = PGs * ones(3,1);

```

```
end
```

6. Función de la perturbación de presión por radiación solar

```

function PS = perturbacion_solar(Area_press, Qreflection)
% Modelo simplificado de la perturbación de presión por radiación solar
% que afecta al CubeSat
Solar_const = 1367;
Speed_of_light = 3E8;
Center_of_pressure = 0.05;

```

```

cons = (Solar_const/Speed_of_light)*Center_of_pressure;

```

```

PS = cons * Area_press * (ones(3,1) + Qreflection);

```

```
end
```

7. Función de la perturbación magnética

```

function PM = perturbacion_magnetica(rS, multmag)
% Modelo simplificado de la perturbación aerodinámica que afecta al
CubeSat
Dipolo = 0.5;
Magnconst = 7.8E15;

```

```

cons = (Dipolo * Magnconst) / ((rS * 1000)^3);

```

```

PMs = cons * multmag;

```

```

PM = PMs * ones(3,1);

```

```
end
```

8. Función para animación del CubeSat

```

function draw_cube(origin, size, R, stopcond)
% función hecha para dibujar el cubo en la simulación
x = ([0 1 1 0 0 0; 1 1 0 0 1 1; 1 1 0 0 1 1; 0 1 1 0 0 0] - 0.5) *
size;
y = ([0 0 1 1 0 0; 0 1 1 0 0 0; 0 1 1 0 1 1; 0 0 1 1 1 1] - 0.5) *
size;
z = ([0 0 0 0 0 1; 0 0 0 0 0 1; 1 1 1 1 0 1; 1 1 1 1 0 1] - 0.5) *
size;
H = [];

```

```
for i=1:6
```

```

    A = R*[x(:,i)'; y(:,i)'; z(:,i)'];
    X = A(1,:) + origin(1);
    Y = A(2,:) + origin(2);
    Z = A(3,:) + origin(3);
    h = patch(X,Y,Z,'b');
    set(h,'edgecolor','k');
    H = [H, h];
end
delay_us = 1;
pause(delay_us/1e6);

if(~stopcond)
    for i=1:6
        delete(H(i));
    end
end
end

```

9. Matriz skew 3D

```

function S = skew(s)
% S = SKEW(s) computes the skew symmetric matrix of a 3d vector
S = [0, -s(3), s(2); s(3), 0, -s(1); -s(2), s(1), 0];

```

10. Matriz skew 4D

```

function S = skew2(s)
% S = SKEW(s) computes the skew symmetric matrix of a 4d vector
S = [0, s(3),

```

11. Programa para desarrollo de la matriz del control LQR

```

%
=====
==
% Desarrollo de controlador LQR para el satélite CubeSat
%
=====
==
clear all;

% Momento de Inercia en el eje X
Ix = 0.002136;

% Momento de Inercia en el eje Y
Iy = 0.000935;

% Momento de Inercia en el eje Z
Iz = 0.001996;

```

```

A = [0,0,0,0.5,0,0;...
      0,0,0,0,0.5,0;...
      0,0,0,0,0,0.5;...
      0,0,0,0,0,0;...
      0,0,0,0,0,0;...
      0,0,0,0,0,0];

B = [0,0,0;...
      0,0,0;...
      0,0,0;...
      1/Ix,0,0;...
      0,1/Iy,0;...
      0,0,1/Iz];

C = eye(6);
% -----
----
% Diseño de control por LQR (comentar si se quiere usar pole placement)
% -----
----
% Definimos nuestras matrices Q y R:
Q = eye(6);
R = eye(3);
% % Resolvemos la ecuación algebraica de Ricatti (ARE):
P = are(A, B*inv(R)*B', Q);
% % Encontramos K
Klqr = inv(R) * B' * P;
% Otra opción es usar la función de Matlab
% K = lqr(A,B,Q,R);

% -----
----
% Parámetros de simulación
dt = 0.001; % período de muestreo (step size)
t0 = 0;
tf = 25;
N = (tf-t0) / dt;

% Discretización por ZOH
sys = c2d(ss(A,B,C,0), dt, 'zoh');
Ad = double(sys.A);
Bd = double(sys.B);
Cd = double(sys.C);

% Simulación del sistema
x0 = 0.1 .* ones(6,1);
x = x0; % Inicialización del vector de estado
X = x; % Array para guardar trayectorias de variables de estado
y = Cd*x; % Inicialización de la salida real
Y = y; % Array para guardar las mediciones del sistema
U = zeros(3,1); % Array para guardar entradas al sistema
W = zeros(3,1);

```

E. CÓDIGO FUENTE DE LA PLATAFORMA FÍSICA DE CONTROL

```

#include <Adafruit_Sensor.h>
#include <Adafruit_BNO055.h>

/* Set the delay between fresh samples */
#define BNO055_SAMPLERATE_DELAY_MS (100)

Adafruit_BNO055 bno = Adafruit_BNO055();

int i = 0;
const int pinA = 5;
const int pinB = 6;
float biasZ = 0.0;

void setup(void)
{
  pinMode(pinA, OUTPUT);
  pinMode(pinB, OUTPUT);
  digitalWrite(pinA, LOW);
  digitalWrite(pinB, LOW);

  Serial.begin(9600);

  /* Initialise the sensor */
  if(!bno.begin())
  {
    /* There was a problem detecting the BNO055 ... check your connections */
    Serial.print("Oops, no BNO055 detected ... Check your wiring or I2C ADDR!");
    while(1);
  }
  delay(1000);

  bno.setExtCrystalUse(true);

  imu::Vector<3> gyro = bno.getVector(Adafruit_BNO055::VECTOR_GYROSCOPE);
  Serial.println("Inicio");
  //Calculate el desfase de error de la IMU

```

```

for (i = 0; i < 100; i++){
  biasZ = biasZ + gyro.z();
  delay(10);
}
biasZ = biasZ / 100;
Serial.println(biasZ);

}

void loop(void)
{
  // obtiene lo datos de la IMU
  imu::Vector<3> gyro = bno.getVector(Adafruit_BNO055::VECTOR_GYROSCOPE);

  // le resta el error de ruido
  float WZ = gyro.z() - biasZ;

  // aplica un control proporcional
  float k = 10;
  float velocidadmotor = k * WZ;

  // cuantifica el valor obtenido para volverlo un valor PWN
  float vel = min(1, abs(velocidadmotor));
  int salidamot = (int)(255*vel);

  // decide a que dirección girar el motor
  if (velocidadmotor < 0){
    analogWrite(pinA, salidamot);
    digitalWrite(pinB, LOW);
  }else{
    analogWrite(pinB, salidamot);
    digitalWrite(pinA, LOW);
  }

  // imprime a través del serial el valor de la IMU para monitoreo serial
  Serial.println(WZ);
}

```