



INSTITUTO POLITÉCNICO NACIONAL

UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA Y
TECNOLOGÍAS AVANZADAS

TRABAJO TERMINAL I

SISTEMA AUTOMATIZADO PARA EL
ACONDICIONAMIENTO ACÚSTICO DE
UN ESTUDIO DE AUDIO POR MEDIO
DEL MOVIMIENTO DE PANELES
ACÚSTICOS

QUE PARA OBTENER EL TÍTULO DE:

INGENIERO EN MECATRÓNICA

PRESENTAN:

BARBOSA MERCADO JOSÉ AARÓN
CAMARENA RODRÍGUEZ ALBERTO
MUÑOZ CEBALLOS TEDDY XAVIER
SÁNCHEZ TRUJILLO DANIEL

ASEORES:

ING. ERICK LÓPEZ ALARCÓN

DR. ALBERTO LUVIANO JUÁREZ

DR. RAFAEL TROVAMALA LANDA



Estados Unidos Mexicanos
Ciudad de México

2024



INSTITUTO POLITÉCNICO NACIONAL

UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA Y
TECNOLOGÍAS AVANZADAS

TRABAJO TERMINAL I

SISTEMA AUTOMATIZADO PARA EL ACONDICIONAMIENTO ACÚSTICO DE UN ESTUDIO DE AUDIO POR MEDIO DEL MOVIMIENTO DE PANELES ACÚSTICOS

PRESENTAN:

BARBOSA MERCADO JOSÉ
AARÓN

CAMARENA RODRÍGUEZ
ALBERTO

MUÑOZ CEBALLOS TEDDY
XAVIER

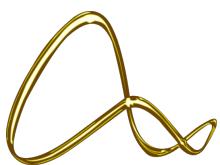
SÁNCHEZ TRUJILLO DANIEL

ASEORES:

DR. RAFAEL TROVAMALA
LANDA

DR. ALBERTO LUVIANO
JUARÉZ

ING. ERICK LÓPEZ ALARCON


upiita-ipn

Presidente del jurado

Profesor titular

DR. GUILLERMO PURIEL GIL

M.EN I. MARIBEL GUTIERREZ
ESPINOZA

Resumen

Para lograr una acústica de adecuada calidad en los estudios de audio, se suele recurrir al uso de paneles acústicos, los cuales modifican las ondas sonoras de un cuarto que no estaba inicialmente diseñado para el propósito de grabar audio, o cualquier otra función en donde la calidad sonora sea de importancia. Sin embargo, y a pesar de que el campo de la acústica está profundamente estudiado, la instalación de paneles acústicos sigue siendo un proceso mayormente empírico y que además está limitado, debido a que diferentes instrumentos requieren una acústica diferente, y el acercamiento de los paneles es buscar una acústica “aceptable” para un amplio rango de instrumentos. Este proyecto tiene como objetivo el diseño e implementación de un sistema que automatice el acondicionamiento acústico en un estudio de audio por medio del movimiento de paneles acústicos. Para este fin, se propone un sistema de paneles acústicos móviles, que cambien la superficie que está en contacto directo con las ondas generadas en el recinto y que modifiquen la respuesta de este. El sistema será inteligente, ya que ajustará la acústica del cuarto en función del instrumento que se desee tocar en el recinto. Este sistema permitirá, además de tener una acústica adecuada para cada instrumento, realizar acondicionamientos acústicos de manera automática.

Palabras clave: paneles acústicos, estudio de audio, acondicionamiento acústico, acondicionamiento automático, acústica inteligente, recinto.

Abstract

To achieve adequate quality acoustics in audio studios, it is often resorted to the use of acoustic panels, which modify the sound waves of a room that was not initially designed for the purpose of recording audio, or any other function where sound quality is important. However, even though the field of acoustics is deeply studied, the installation of acoustic panels is still a largely empirical and limited process, because different instruments require different acoustics, and the approach of the panels is to seek “acceptable” acoustics for a wide range of instruments. This project aims to design and implement a system that automates the acoustic conditioning in enclosures through the movement of sound modifying panels. For this purpose, our proposal consists of a system of movable acoustic panels, which change the surface that is in direct contact with the waves generated in the audio studio and change the response of the enclosure. The system will be intelligent, as it will adjust the acoustics of the room depending on the instrument to be played in it. The system will allow, in addition to having suitable acoustics for each instrument, to be able to perform acoustic treatment automatically.

Key words: acoustic panels, enclosure, acoustic treatment, automatic treatment, intelligent acoustics, audio studio.

Índice

	Página
I. Introducción	11
II. Planteamiento del problema	12
III. Justificación	13
IV. Objetivos	15
IV.I. Objetivo general	15
IV.II. Objetivos específicos	15
IV.II.I. Objetivos de Trabajo Terminal I	15
IV.II.II. Objetivos de Trabajo Terminal II	15
V. Antecedentes	16
1. Marco de Referencia	17
1.1. Marco Teórico	17
1.1.1. Sonido.	17
1.1.2. Generación y propagación del sonido.	18
1.1.3. Presión sonora.	18
1.1.4. Tren de pulsos.	19
1.1.5. Acústica.	19
1.1.6. Acústica Arquitectónica.	19
1.1.7. Ecos.	19
1.1.8. Reflexiones tempranas.	19
1.1.9. Absorción sonora.	19
1.1.10. Tiempos de reverberación.	21
1.1.11. Tiempos de reverberación óptimo.	21
1.1.12. Resonancias.	22
1.1.13. Claridad.	22
1.1.14. Potencia sonora	23
1.1.15. Medidas de amplitud	23
1.1.16. Parámetros relacionados con el timbre o el color tonal	24
1.1.17. Inteligibilidad del habla	25
1.2. Marco Precedimental	25
1.2.1. Definición de la metodología mecatrónica	25
1.2.2. Resolución de problemas en el micro-level	26
1.2.3. Modelo en V en el macro-level	27
1.2.4. Procesos predefinidos	27

2. Diseño del sistema	28
2.1. Diseño conceptual	28
2.1.1. Necesidades y requerimientos	28
2.1.2. Arquitectura funcional	30
2.1.3. Arquitectura física	36
2.1.4. Propuesta de solución	38
2.1.4.1. Integración de los conceptos solución	40
2.1.5. Selección diseño conceptual	41
2.2. Diseño detallado	52
2.2.1. MF1. Módulo de procesamiento	52
2.2.1.1. MF1.1. Procesamiento de la respuesta y cálculo de la acústica	52
Análisis de la respuesta al impulso	52
RayTracingImpulseResponse	58
Validación de la obtención de respuesta al impulso por simulación	70
2.2.1.2. MF1.2 Procesamiento de la estrategia de control y posiciones de los paneles	71
Cálculo de Posiciones para los paneles	71
Procesamiento de la estrategia de control	76
2.2.1.3. MF1.3 Interacciones del sistema	83
Protocolo de comunicación	83
Nube	86
Base de datos	87
Marco de trabajo para la conexión	89
2.2.2. MF2. Módulo de generación y medición de la acústica	91
2.2.3. MF3. Módulo modificador de la acústica	93
2.2.3.1. Paneles de Absorción	93
2.2.3.2. Paneles de reflexión	94
2.2.3.3. Paneles de difracción	94
Selección método de movimiento	96
2.2.3.4. Sistema de transmisión	99
Potencia del motor	99
Corona-Sinfín	102
Eje del tornillo sinfín	114
2.2.3.5. Miembro estructural	127
Cálculo de la viga A	127
Cálculo de la viga B	133
2.2.4. MF4. Módulo de interfaz de usuario	136
Experiencia de usuario y flujo de trabajo	136
2.2.4.1. Tiempos de reverberación deseados	139
Composiciones tecnológicas de la interfaz	141
2.2.5. MF5. Módulo gestor de energía	143
2.2.5.1. MF5.1 Submódulo de conversión y regulación de la tensión de entrada	143
2.2.5.2. MF5.2 Submódulo de medición del consumo energético . . .	158
2.3. Integración del sistema	161

2.4. Administración	165
2.4.1. Costos estimados	165
2.4.2. Planeación de la implementación	167
Apéndices	169
A. Códigos y scripts	169
A.1. Función para calcular los modos normales de vibración	169
A.2. Función para calcular la pared y posición de impacto	169
A.3. Función para hacer los ajustes lineales a la curva de decaimiento	170
A.4. Función para obtener el vector normal a la pared	171
A.5. Función para mostrar el cuarto y las posiciones del receptor y transmisor . .	171
A.6. Función obtener vectores esféricos uniformemente distribuidos	172
A.7. Código para obtener la respuesta al impulso del recinto por medio de Ray Tracing	172
A.8. Función para calcular los parámetros acústicos de acuerdo a ISO3382	178
A.9. Código obtener la respuesta al impulso de un recinto por medio de mediciones	180
A.10. Código para simular las activaciones de los solenoides	182
A.11. Función para cambiar los mapas de absorción y refracción	183
A.12. Código con librería para obtener la respuesta al impulso del recinto por medio de Ray Tracing	183
A.13. Programa de Simulink para el control inteligente de posición y velocidad . .	185
A.14. Programa de <i>Python</i> para la generación de las posiciones de los paneles . .	186
Anexos	190
A. Planos de manufactura	190
B. Propiedades mecánicas de aceros al carbón	208
C. Factores de concentración de esfuerzos	209
D. Dimensiones y propiedades del tubo cuadrado PHERT	210
E. Hoja de datos del MOSFET IRL2703	211
F. Hoja de datos de la Raspberry Pi 4 Model B	218
G. Hoja de datos de la Fuente de Poder MSI MAG A650BN	225

Figuras

1. Tiempo de reverberación óptimo en función del volumen de una sala (según L. L. Beranek). (a) Estudios de radiodifusión para voz. (b) Salas de conferencias. (c) Estudios de radiodifusión para música. (d) Salas de conciertos. (e) Iglesias [16].

22

2.	Ilustración de la teoría y el principio en la medición de STI o RASTI. La escala para la evolución de los valores RASTI se muestra en la parte inferior [17].	25
3.	Diagrama de flujo para la resolución de problemas en el micro-level.	26
4.	Metodología por seguir.	27
5.	Diagrama de las funciones del sistema para el tratamiento acústico.	33
6.	IDEF0 del sistema para el acondicionamiento acústico.	33
7.	IDEF0 del Módulo de medición de la acústica actual.	34
8.	IDEF0 del Módulo de modificación de la acústica.	34
9.	IDEF0 del Módulo de interacción con el usuario.	35
10.	IDEF0 del Módulo de gestión de energía.	35
11.	Diagrama del sistema para el acondicionamiento acústico	38
12.	Concepto solución elegido	51
13.	Respuesta al impulso para diferentes frecuencias	53
14.	Curva de decaimiento	55
15.	Geometría del cuarto simulado y posiciones del emisor y receptor	59
16.	Histograma dependiente de la frecuencia	66
17.	Filtros pasa-bandas para el proceso de Poisson	67
18.	RIR cuadrada obtenida de simulación	69
19.	Análisis de la respuesta al impulso simulada	70
20.	Simulación con PyRoomAcoustics	71
21.	Prueba uno de cálculo de posiciones para los paneles	75
22.	Prueba dos de cálculo de posiciones para los paneles	76
23.	Interpolación de Bezier	78
24.	Triple Interpolación con polinomios de Bezier	79
25.	Parámetros de polinomio de Bezier	79
26.	Validación mediante Simscape	81
27.	Trayectoria y movimiento simulado	81
28.	Activación de los solenoides a lo largo de la trayectoria	83
29.	Arquitectura de comunicación MF1.3	84
30.	Gráficas resultantes de generación y medición de la acústica	93
31.	Coeficientes de absorción de los paneles recubiertos	94
32.	Sección transversal de un panel QRD	95
33.	Validación torque para un panel	100
34.	Validación potencia para un panel	100
35.	Validación torque y potencia para un ensamblaje	101
36.	Geometría del tornillo sinfín	108
37.	Geometría de la corona	108
38.	Equivalencia entre las fuerzas que actúan entre el sinfín y la corona.	109
39.	Validación de esfuerzos en la corona	113
40.	Validación del factor de seguridad en la corona	114
41.	Validación de desplazamiento en la corona	114
42.	Primer segmento de tornillos sinfín.	115
43.	Diagrama de cuerpo libre del eje del tornillo sinfín.	116
44.	Diagrama de cuerpo libre en el plano X – Y	117
45.	Diagrama de la fuerza cortante en el plano X – Y	117

46. Diagrama del momento flector en el plano $X - Y$	118
47. Diagrama de cuerpo libre en el plano $X - Z$	118
48. Diagrama de la fuerza cortante en el plano $X - Z$	119
49. Diagrama del momento flector en el plano $X - Z$	119
50. Diagrama del momento torsor a lo largo del eje X	119
51. Diámetros de interés en el eje del tornillo sínfín.	120
52. Factores de acabado superficial para aceros [71].	121
53. Valores de \sqrt{a} [72].	123
54. Máximo esfuerzo de von Mises.	126
55. Deflexión máxima en el eje.	126
56. Factor de seguridad en el eje.	127
57. Representación de la viga A y el soporte d pared de dos pestañas.	128
58. Fuerza aplicada y reacciones en la chumacera.	129
59. Fuerza aplicada y reacciones en la viga A	130
60. Diagrama de cuerpo libre de la viga A	131
61. Deflexión máxima de la viga B	133
62. Diagrama de cuerpo libre de la viga B	134
63. Deflexión máxima de la viga B	136
64. Diagrama de Flujo de la aplicación	137
65. Vistas de la interfaz de usuario	139
66. Coeficientes de absorción de los paneles recubiertos	140
67. Esquemático del circuito que acciona los solenoides lineales	150
68. PCB del circuito que acciona los solenoides lineales	150
69. Modelo 3D de la PCB	151
70. PCB del circuito que acciona los solenoides lineales	152
71. Motor seleccionado[70]	152
72. Información de las características del motor proporcionadas por el vendedor[89]	153
73. Convertidor de 12Vdc a 24Vdc[90]	153
74. Driver para controlar la velocidad del motor[91]	154
75. Potenciómetro digital modelo DS1804-100[92]	155
76. Encoder seleccionado[94]	156
77. Diferentes certificaciones 80 PLUS[95]	157
78. Fuente de Poder MSI MAG A650BN, 650W[96]	158
79. Medidor de consumo eléctrico[100]	159
80. Diagrama de la conexión del medidor de consumo eléctrico	160
81. Diagrama del módulo gestor de energía	160
82. Vista isométrica del sistema integrado	162
83. Vista isométrica del mecanismo de movimiento y acoplamiento	163
84. Vista isométrica del motor y del mecanismo de movimiento y de acoplamiento	164
85. Vista explosionada del sistema integrado	165
86. Cronograma de actividades para TT2.	168

Tablas

1.	Tabla de antecedentes	17
2.	Coeficientes de absorción de diversos materiales en función de la frecuencia (según varias fuentes) [16].	21
3.	Necesidades del sistema automatizado para el acondicionamiento acústico. .	28
4.	Tabla de requerimientos.	29
5.	Matriz de trazabilidad necesidades-requerimiento.	30
6.	Matriz de trazabilidad requerimientos-funciones.	36
7.	Matriz de trazabilidad de arquitectura física	38
8.	Matriz morfológica	40
9.	Conceptos solución	41
10.	Matriz de comparación de Cr_1	41
11.	Matriz normalizada de Cr_1 y V_{Cr_1}	41
12.	Matriz de comparación de Cr_2	41
13.	Matriz normalizada de Cr_2 y V_{Cr_2}	41
14.	Matriz de comparación de Cr_3	41
15.	Matriz normalizada de Cr_3 y V_{Cr_3}	41
16.	Matriz de comparación de Cr_4	42
17.	Matriz normalizada de Cr_4 y V_{Cr_4}	42
18.	Matriz de comparación de Cr_5	42
19.	Matriz normalizada de Cr_5 y V_{Cr_5}	42
20.	Matriz de comparación de Cr_6	42
21.	Matriz normalizada de Cr_6 y V_{Cr_6}	42
22.	Matriz de comparación de Cr_7	42
23.	Matriz normalizada de Cr_7 y V_{Cr_7}	42
24.	Matriz de comparación de Cr_8	42
25.	Matriz normalizada de Cr_8 y V_{Cr_8}	42
26.	Matriz de comparación de Cr_9	42
27.	Matriz normalizada de Cr_9 y V_{Cr_9}	42
28.	Matriz de comparación de Cr_{10}	42
29.	Matriz normalizada de Cr_{10} y $V_{Cr_{10}}$	42
30.	Matriz de comparación de Cr_{11}	43
31.	Matriz normalizada de Cr_{11} y $V_{Cr_{11}}$	43
32.	Matriz de comparación de Cr_{12}	43
33.	Matriz normalizada de Cr_{12} y $V_{Cr_{12}}$	43
34.	Matriz de comparación de Cr_{13}	43
35.	Matriz normalizada de Cr_{13} y $V_{Cr_{13}}$	43
36.	Matriz de comparación de Cr_{14}	43
37.	Matriz normalizada de Cr_{14} y $V_{Cr_{14}}$	43
38.	Matriz de comparación de Cr_{21}	43
39.	Matriz normalizada de Cr_{21} y $V_{Cr_{21}}$	43
40.	Matriz de comparación de Cr_{22}	43
41.	Matriz normalizada de Cr_{22} y $V_{Cr_{22}}$	43

42. Matriz de comparación de Cr_{24}	43
43. Matriz normalizada de Cr_{24} y $V_{Cr_{24}}$	43
44. Matriz de comparación de Cr_{25}	44
45. Matriz normalizada de Cr_{25} y $V_{Cr_{25}}$	44
46. Matriz de comparación de Cr_{27}	44
47. Matriz normalizada de Cr_{27} y $V_{Cr_{27}}$	44
48. Matriz de comparación de criterios	45
49. Matriz normalizada y vector de prioridad para la comparación de criterios.	46
50. Matriz de comparación de C_3	47
51. Matriz normalizada de C_3 y V_{C_3}	47
52. Matriz de comparación de C_4	47
53. Matriz normalizada de C_4 y V_{C_4}	47
54. Matriz de comparación de C_5	47
55. Matriz normalizada de C_5 y V_{C_5}	47
56. Matriz de comparación de C_6	47
57. Matriz normalizada de C_6 y V_{C_6}	47
58. Matriz de comparación de C_7	47
59. Matriz normalizada de C_7 y V_{C_7}	47
60. Matriz de comparación de C_8	47
61. Matriz normalizada de C_8 y V_{C_8}	47
62. Matriz de comparación de C_9	48
63. Matriz normalizada de C_9 y V_{C_9}	48
64. Matriz de comparación de C_{10}	48
65. Matriz normalizada de C_{10} y $V_{C_{10}}$	48
66. Matriz de comparación de C_{11}	48
67. Matriz normalizada de C_{11} y $V_{C_{11}}$	48
68. Matriz de comparación de C_{12}	48
69. Matriz normalizada de C_{12} y $V_{C_{12}}$	48
70. Matriz de comparación de C_{13}	48
71. Matriz normalizada de C_{13} y $V_{C_{13}}$	48
72. Matriz de comparación de C_{14}	48
73. Matriz normalizada de C_{14} y $V_{C_{14}}$	48
74. Matriz de comparación de C_{15}	48
75. Matriz normalizada de C_{15} y $V_{C_{15}}$	48
76. Matriz de comparación de C_{16}	49
77. Matriz normalizada de C_{16} y $V_{C_{16}}$	49
78. Matriz de decisión	49
79. Características del concepto solución elegido	50
80. Rangos de frecuencias para filtros	53
81. Comparación de protocolos de comunicación	85
82. Comparación de proveedores de nube según criterios	87
83. Comparación de proveedores de nube según servicios [49]	87
84. Comparación de Bases de Datos SQL y NoSQL [50]	88
85. Comparación de bases de datos compatibles con GCP y noSQL	89
86. Comparación de Frameworks para Desarrollo Backend	90

87. Ventajas y desventajas de los distintos tipos de transmisión de movimiento	97
88. Ventajas y desventajas relacionadas con la utilización de un tornillo sin fin único o varios segmentos de tornillo sin fin	98
89. Ventajas y desventajas relacionadas con los diferentes métodos para obtener la configuración deseada en los paneles	99
90. Número mínimo sugerido de dientes de la corona para diseños habituales ϕ_n [71]	102
91. Resultado de la geometría del tornillo sinfín	107
92. Resultado de la geometría de la corona	108
93. Factor de materiales C_s , para coronas de bronce [71]	111
94. Factor de forma de Lewis Y en relación con el ángulo de presión ϕ_n [69]	112
95. Fuerzas en el tornillo sinfín y en la corona	113
96. Momentos resultantes	120
97. Factor de fiabilidad k_r [71].	122
98. Selección de marco de trabajo: Electron	141
99. Selección de librería de desarrollo principal: React	142
100. Selección de gestor del estado: Redux	142
101. Selección de herramienta de estilizado: Styled Components	143
102. Selección de herramienta de visualización: D3.js	143
103. Ventajas y desventajas de las opciones para solenoides lineales	145
104. Comparativa entre los diferentes tipos de encoders[93]	156
105. Lista de materias primas	166
106. Lista de componentes de fabricación	166
107. Lista de componentes de adquisición	167

I. Introducción

La grabación de sonidos puede ser un proceso complejo, pero desde que la humanidad comenzó a hacerlo a finales del siglo XIX con la invención del fonógrafo por Thomas Edison [1] no hemos parado de esforzarnos en hacerla cada vez mejor. Capturar el sonido significa conservar y reproducir a placer momentos únicos, significativos y emocionantes. Por ejemplo, se puede revivir actuaciones en vivo, recordar la cálida voz de seres queridos, descubrir música, transportarse al bullicio de la ciudad o al calmado rozar del viento bailando entre los árboles y conectar con los demás al compartir experiencias del mundo. Esta actividad relativamente moderna se convirtió rápidamente en una nueva forma de memoria en la vida de nuestra especie.

Hoy en día, la grabación de sonido es una práctica común y se utiliza en una amplia variedad de contextos, desde la producción musical, pasando por la grabación de discursos y entrevistas hasta fines de toda índole como la política, el entretenimiento, la educación, la comunicación y el arte. En cada una de estas áreas, el fenómeno de la grabación ha permitido una mayor experimentación y, por ende, innovación.

Para grabar el sonido, se captan las vibraciones invisibles del aire y se almacenan en algún medio. Reproducir el sonido es simplemente el proceso inverso. Existen dos clases principales de tecnologías de grabación, la analógica y la digital. Para la grabación digital es necesario un micrófono, nuestro instrumento de captación de ondas sonoras que a su vez las convierte en señales eléctricas, una interfaz de audio, necesaria para convertir las señales analógicas del micrófono a señales digitales almacenables en la mayoría de los dispositivos electrónicos, y un dispositivo de grabación como la computadora.

La obtención de una grabación de alta calidad es un proceso complejo que depende de múltiples factores, como la selección del equipo, la elección del espacio e incluso la habilidad técnica de los ingenieros de grabación. Los profesionales del sonido, conocidos como ingenieros de audio [2], son los responsables de asegurar la calidad del audio en grabaciones y actuaciones en vivo. Los ingenieros de audio deben de tener conocimientos en el uso del software y el equipo a emplear, además de habilidades artísticas como la composición y empleo de instrumentos musicales.

Dependiendo del propósito de la grabación, el equipo y los recursos que se tengan disponibles, hay varias opciones de lugares en los que se puede grabar audio. La más profesional y con resultados de calidad superior dado a su medio controlado es en un estudio de grabación, nombre engañoso, ya que no solo se realizan grabaciones en estos lugares, pero son comúnmente conocidos por este nombre. Este ambiente de grabación ofrece varias ventajas de entre las cuales destaca la calidad del sonido obtenido; ya que los estudios de audio están diseñados para ofrecer una acústica controlada y cuentan con equipos avanzados de grabación, reproducción, mezcla, edición, preproducción, producción y postproducción. Los estudios de grabación impactan, por tanto, a las industrias de la música, el cine y la televisión, la publicidad, los videojuegos, la radio y toda aquella en la que se desee hacer uso de

servicios especializados en audio.

Al inicio los estudios de grabación eran instalaciones muy sencillas que se limitaban a aislar el ruido del exterior, pero no eran comúnmente empleados, lo común en la música, por nombrar un ejemplo, era hacer la grabación en la misma sala de conciertos o teatros, sin embargo, los ingenieros en audio no tardaron en darse cuenta de que estas grabaciones no eran ideales debido a la acústica de los espacios en los que se realizaban, y también debido a la imposibilidad de corregir errores o ajustar la mezcla de sonido en postproducción. Hasta que con los avances tecnológicos y la grabación digital estos espacios comenzaron a popularizarse. En resumen, los estudios de grabación se comenzaron a usar como una forma de mejorar la calidad de las grabaciones de música y otros sonidos, y han evolucionado a lo largo del tiempo para convertirse en lugares altamente especializados y sofisticados que ofrecen la mejor calidad posible en la grabación de sonido.

II. Planteamiento del problema

Para modificar el ambiente acústico de un espacio cerrado, se utilizan múltiples técnicas y herramientas que modifican la reverberación, difracción, reflexión, resonancias y otros efectos acústicos. Algunos de estos métodos son el diseño de la sala, el aislamiento con el ambiente, la ventilación, el posicionamiento de los altavoces, etc. [3] dentro de los cuales, el tratamiento de las paredes es el artilugio más común que encontramos en los estudios de audio. Desde que entramos en estos espacios podemos ver que las paredes están cubiertas por una diversidad de materiales y formas irregulares, cuya función es alterar la reflexión del sonido y evitar la formación de ondas estacionarias [4]. Los elementos de control acústicos que se emplean para este tratamiento son paneles de absorción, paneles difusores y trampas de graves [5].

El procedimiento para realizar la correcta gestión acústica de un espacio con la técnica de tratamiento en paredes inicia con el análisis de la sala, con esto se detectan los problemas acústicos que puede llegar a tener el recinto y el siguiente paso es diseñar las soluciones para estos problemas encontrados, lo que involucra la instalación de los elementos de control acústicos antes mencionados en puntos estratégicos. Por último, se vuelven a hacer mediciones y se ajustan las posiciones con base en los parámetros que se desea obtener [6].

Esto supone un caso ideal, pero en la realidad, el problema es que debido al conocimiento especializado que se necesita tener sobre los cálculos acústicos y a la inaccesibilidad del instrumental con el que se realizan las mediciones, el proceso termina haciéndose a prueba y error con los elementos de control acústico en diversas posiciones hasta llegar a un resultado que se considere aceptable, lejos de ser óptimo [3]. Otro problema que se observa para este tratamiento es que la acústica tratada mediante paneles es una acústica estática que se diseña según el propósito del espacio. Para poder modificar las variables acústicas habría que regresar al paso anterior y volver a hacer el estudio por cada modificación que se desea hacer o función que se desea realizar en el espacio.

Para resolver estos problemas actualmente se utilizan equipos de procesamiento de señal como ecualizadores, compresores, reverbs, delays, filtros, moduladores, entre otros, en conjunto con software especializado, ya que es necesario tener la posibilidad de alterar las características del sonido y no siempre se cuenta con el espacio ideal para esto [7]. Aunque el uso de los aparatos y el software puede ser útil, presentan las desventajas de introducir puntos de falla y limitaciones [3]. Como son la pérdida de calidad en el sonido o generar distorsiones según la calidad de los componentes, sumado a esto, agregan una capa de complejidad al proceso, ya que para su operación se requiere capacitación para poder operar, de la misma manera que para el manejo del software.

En resumen, el tratamiento acústico por paneles que se genera en la mayoría de los casos es un proceso empírico, el cual está lleno de puntos de falla provocados por la complejidad misma del proceso, aunado a esto, incluso cuando se hace un tratamiento idea su condición es estática, por lo que la más pequeña desviación del rango de operación para el que fue diseñado el cuarto, termina nuevamente en un tratamiento acústico deficiente. El problema es la falta de un sistema automático que controle las variables acústicas sin importar la disposición o diseño del cuarto.

Un sistema que sirva para dar solución a este problema solo es posible gracias a la sinergia de la mecánica, para las piezas físicas que se moverán, la electrónica, para todas las señales que se tiene medir, el control, que permitirá la regulación del comportamiento y la computación para una interfaz moderna y fácilmente operable por el usuario.

III. Justificación

Ethan Winer advierte en su libro “The Audio Expert” que por más que se inviertan miles o millones de dólares en los equipos más precisos y de alta calidad, con un pobre tratamiento acústico del cuarto en que estos operen, no serán capaces de alcanzar su máximo potencial. “Es difícil escuchar lo que estás haciendo, lo que dificulta crear un buen sonido” afirma el autor sobre los cuartos sin un tratamiento acústico adecuado [8].

Al momento, los esfuerzos en la industria para lograr la modificación de variables acústicas se enfocan en mejorar los dispositivos y software existentes de procesamiento de audio [9]. El sistema que proponemos se enfoca en presentar una solución fuera de la oferta actual y permitir que los ingenieros de audio tengan una alternativa que les permita lograr una acústica óptima y dinámica en sus salas de grabación, lo cual permitirá tratar la acústica del recinto como un instrumento más a la hora de hacer sus funciones diarias [10], esta implementación de un nuevo elemento para el tratamiento acústico pronostica una mejora en la calidad de los resultados ligada a una experiencia de trabajo mejorada, al no tener que destinar tiempo en procesos actualmente automatizables por medio de un sistema mecatrónico.

R. Israel Quintero Tiscareño, ingeniero en audio de Espiral Estéreo, nos regala su pers-

pectiva sobre las ventajas que presenta un sistema como este y el impacto que tendría en su trabajo: *“Las ventajas que veo en un sistema de tratamiento acústico dinámico son tan variadas cómo lo pueda ser el mismo sistema, debido a que aparte de ser móvil, puede tener dos o más caras en su movilidad y textura, debido a que la reflexión de cada cara puede tener materiales y por obvias razones, densidades y factores de absorción distintas, con esto sería muy enriquecedor poder adaptar o ajustar un solo espacio con la posibilidad de emular distintas salas de grabación y ejecución... para así ecualizar con base en la o las aplicaciones deseadas! Este sistema tendría un gran impacto en el negocio de los estudios de grabación, debido a que en Latinoamérica no existen o son contados los lugares que cuentan con un sistema parecido... pero sin duda alguna no igual al que se propone”.*

La búsqueda de un sistema que pueda modificar de manera automática la acústica de un cuarto requiere de las diferentes habilidades del ingeniero mecatrónico. Se requiere de un sistema mecánico que posicione los paneles dentro del cuarto, para influenciar su acústica. Además, se necesita un sistema de control, que ayude en la precisión y repetibilidad de los movimientos de estos paneles para poder afirmar que se tiene una acústica controlable. El sistema debe contar con una interfaz computacional que le permita al usuario monitorizarlo y controlarlo. Por último, contiene una parte electrónica, la cual dirigirá la parte mecánica en función de las señales recibidas por la interfaz de usuario y procesadas por el sistema de control.

Un sistema como el que se espera obtener forma parte de los esfuerzos de diferentes equipos de investigación para dar solución a este problema. Es un concepto que recién comienza a desarrollarse y el diseño e implementación de sistemas para este propósito, desde diferentes enfoques, ayudara con el desarrollo de soluciones cada vez más complejas, precisas y confiables.

El sistema hará uso de todas las áreas de la mecatrónica, ya que conllevará la creación de una interfaz de usuario que deberá comunicarse efectivamente con el sistema, al mismo tiempo debe mostrar una gran cantidad de datos al usuario de manera digerible. Además, implicará el diseño y fabricación de los diferentes paneles acústicos en conjunto con un sistema modular que deberá moverlos por el cuarto de manera precisa, haciendo uso del control inteligente. Necesitará de un dispositivo electrónico embebido que sirva como puente entre la interfaz de usuario y el sistema, y deberá permitir que el usuario solo deba preocuparse por enchufarlo y conectarlo a su computadora. Aunado al conocimiento necesario de todas las áreas de la mecatrónica, será necesario un conocimiento de los algoritmos de aprendizaje de máquina más allá del cubierto por plan de estudios, debido a la complejidad que conlleva la creación de una relación entre las posiciones de los paneles y la acústica del cuarto. Por último, será necesario un entendimiento profundo en temas de acústica, desde la concepción del proyecto hasta las pruebas finales, debido a que gran parte de la complejidad del diseño e implementación recae en esta área externa. Por todo lo anterior, la complejidad del proyecto demanda de 4 integrantes para su desarrollo, los integrantes del equipo se especializarán en un área sin perder atención en las demás áreas, pero sí fungirán como responsables directos. Aarón Barbosa estará a cargo del control, Alberto Camarena de la programación, Teddy Muñoz de la mecánica y Daniel Sánchez de la electrónica.

IV. Objetivos

IV.I. Objetivo general

Diseñar e implementar un sistema que automatice el acondicionamiento acústico de un estudio de audio por medio del movimiento de paneles acústicos.

IV.II. Objetivos específicos

IV.II.I. Objetivos de Trabajo Terminal I

- Diseñar un módulo de medición de la respuesta del estudio de audio ante un barrido senoidal a distintas frecuencias para caracterizar su acústica.
- Calcular el tiempo de decaimiento temprano (EDT), tiempo de reverberación (T20 y T30), claridad (C50 y C80), definición (D50) y fuerza del sonido (G), a partir de la respuesta del estudio.
- Verificar la existencia de ondas estacionarias en el estudio de audio mediante el uso de las dimensiones del estudio de audio para su tratamiento.
- Diseñar un módulo de modificación de la acústica mediante el movimiento de paneles acústicos de absorción, reflexión y difusión, en dos paredes adyacentes, con control inteligente de posición y acondicionamiento estático de las demás superficies para acondicionar la acústica de un estudio de audio a diferentes instrumentos.
- Creación de relaciones entre las posiciones de los paneles y la acústica del cuarto para el control preciso de la acústica mediante el control de posición de los paneles.
- Validar los valores de los parámetros acústicos mediante simulaciones para comprobar los algoritmos de relación entre posición y acústica.
- Validar la modificación de la acústica mediante simulaciones para comprobar el correcto tratamiento acústico.

IV.II.II. Objetivos de Trabajo Terminal II

- Manufacturar e implementar el sistema energético, verificar su funcionamiento.
- Implementar un módulo de medición de la respuesta del estudio de audio ante un barrido senoidal a distintas frecuencias para caracterizar su acústica.
- Implementar un módulo de modificación de la acústica mediante el movimiento de paneles acústicos de absorción, reflexión y difusión, en dos paredes adyacentes, con control inteligente de posición y acondicionamiento estático de las demás superficies para acondicionar la acústica de un estudio de audio a diferentes instrumentos.

- Implementar una plataforma de control y monitoreo del sistema para el usuario mediante software, que muestre los parámetros acústicos calculados, mediciones, posiciones de los paneles y toda la información asociada.
- Validar el sistema mecatrónico en el ambiente de prueba mediante pruebas de acondicionamiento para distintos instrumentos para la resolución de errores y comprobación del correcto funcionamiento.

V. Antecedentes

No.	Nombre	Descripción	Características	País	Instituto	Ref
1.	Panel Acústico Variable Rotatorio. (P.A.V.R.)	Diseño y construcción de un prototipo de panel acústico variable rotatorio de 360° con control automático.(P.A.V.R)	<ul style="list-style-type: none"> ■ Difusión, reflexión o absorción del sonido. ■ Tarjeta de desarrollo Arduino Uno como para la implementación del control. ■ Medidas 113cmx113cm. 	Colombia	Universidad de San Buenaventura	[11]
2.	Acoustic Enclosure with Intelligent Controllable Noise Insulation	Recinto acústico inteligente para las máquinas que siempre están trabajando bajo cargas dinámicas.	<ul style="list-style-type: none"> ■ Aislamiento del sonido con diferentes anchos de banda. ■ El sistema de control del aislamiento consiste en transductor, PLC y servo motor. 	China	Qingdao Technological University	[12]

3.	Investigation of structural response and noise reduction of an acoustical enclosure using SEA method	<p>Modelo SEA mejorado que incluye la respuesta no resonante y el coeficiente de transmisión más preciso de paneles finitos, tomando en cuenta la comparación de los resultados medidos y la mejor concordancia entre la predicción de la respuesta estructural prevista y la reducción de ruido de un recinto acústico.</p>	<ul style="list-style-type: none"> ■ Recinto acústico de 200 m^3. ■ Recinto acústico con muchos muebles y artículos dissipadores de sonido. ■ Altoparlante generador de energía acústica a altas frecuencias. ■ Altoparlante generador de adecuada potencia de sonido a bajas y medias frecuencias. 	Australia, China	University of westerns Australia; Northwestern Polytechnical University	[13]
4.	Soundspheres: Resonant Chamber	<p>El “Soundspheres” es un proyecto acústico que conecta las interrelaciones entre material, forma espacial y sonido. Su diseño se centra en tres tipos de capas: duras, estáticas e inflexibles; físicamente manipulables; y electroacústico. Su objetivo es desarrollar un sistema interior envolvente cinético e interactivo destinado a transformar el entorno acústico a través de la dinámica espacial, materiales y tecnologías electroacústicas.</p>	<ul style="list-style-type: none"> ■ Geometrías de superficie dinámicas ■ Actuación y respuesta variables ■ Modular 	Estados Unidos de América	University of Michigan	[14]

Tabla 1: Tabla de antecedentes

1. Marco de Referencia

1.1. Marco Teórico

1.1.1. Sonido.

El sonido se puede definir de formas muy diversas. De todas ellas, las más habituales son las siguientes:

- Vibración mecánica que se propaga a través de un medio (habitualmente el aire), y que es capaz de producir una sensación auditiva.

- Sensación auditiva producida por una vibración de carácter mecánico que se propaga a través de un medio [15].

1.1.2. Generación y propagación del sonido.

El elemento generador del sonido se denomina fuente sonora (tambor, cuerda de un violín, cuerdas vocales, etc.). La generación del sonido tiene lugar cuando dicha fuente entra en vibración. Dicha vibración es transmitida a las partículas de aire adyacentes a la misma que, a su vez, la transmiten a nuevas partículas contiguas.

Las partículas no se desplazan con la perturbación, sino que simplemente oscilan alrededor de su posición de equilibrio. La manera en que la perturbación se traslada de un lugar a otro se denomina propagación de la onda sonora. La oscilación de las partículas tiene lugar en la misma dirección que la de propagación de la onda [15].

Ahora nos preguntamos qué tan rápido se aleja la onda de la fuente. La respuesta es que el sonido se propaga con una velocidad c que en el aire a 23°C vale

$$c = 345 \text{ m/s}, \quad (1)$$

o bien

$$c = 1242 \text{ km/h}, \quad (2)$$

Esta velocidad varía algo con la temperatura (un **0.17 %/ $^{\circ}\text{C}$**), por eso en diversos textos pueden encontrarse valores ligeramente diferentes [16].

1.1.3. Presión sonora.

La manera más habitual de expresar cuantitativamente la magnitud de un campo sonoro es mediante la presión sonora, o fuerza que ejercen las partículas de aire por unidad de superficie [15].

Si nos ubicamos en una posición fija, veremos que la presión atmosférica aumenta y disminuye periódicamente, conforme pasan por el lugar las sucesivas perturbaciones. Dado que nos referiremos bastante seguido a valores de presión, conviene aclarar que la unidad adoptada internacionalmente para la presión es el Pascal, abreviada Pa. Expresada en esta unidad, la presión atmosférica es del orden de 100,000 Pa. Los aumentos y las disminuciones de presión debidas a las ondas sonoras son realmente muy pequeños comparados con este valor de presión atmosférica. Los sonidos más intensos que se perciben como tales implican un aumento de unos 20 Pa. La presión sonora es lo que se debe agregar a la presión atmosférica en reposo para obtener el valor real de presión atmosférica. Las presiones sonoras audibles varían entre 0,00002 Pa y 20 Pa. El valor más pequeño, también expresado como 20 μPa se denomina umbral auditivo [16].

1.1.4. Tren de pulsos.

Es una variante de la onda cuadrada en el cual el tiempo de permanencia en cada uno de los dos niveles no es el mismo. Se suele especificar un porcentaje que corresponde a la porción del periodo en el nivel alto [16].

1.1.5. Acústica.

Es la disciplina que se ocupa para estudiar el sonido en sus diversos aspectos. Se puede dividir en una gran cantidad de sub disciplinas [16], algunas de las cuales son la acústica física, la psico acústica, la acústica musical y en la cual nos centraremos nosotros, es la acústica arquitectónica.

1.1.6. Acústica Arquitectónica.

Estudia los fenómenos vinculados con una propagación adecuada, fiel y funcional del sonido en un recinto. Las habitaciones o salas dedicadas a una aplicación determinada deben tener cualidades acústicas adecuadas para dicha aplicación. Por cualidades acústicas de un recinto entendemos una serie de propiedades relacionadas con el comportamiento del sonido en el recinto, entre las cuales se encuentran las reflexiones tempranas, la reverberación, la existencia o no de ecos y resonancias, la cobertura sonora de las fuentes, etc. [16].

1.1.7. Ecos.

El fenómeno más sencillo que tiene lugar en un ambiente con superficies reflectoras del sonido es el eco, consistente en una única reflexión que retorna al punto donde se encuentra la fuente unos 100 ms (o más) después de emitido el sonido. Se produce después de un tiempo t relacionado con la distancia d a la superficie más próxima por la expresión

$$t = \frac{2d}{c}, \quad (3)$$

donde c es la velocidad del sonido. El factor 2 se debe a que el sonido recorre de ida y de vuelta la distancia entre la fuente sonora y la superficie [16].

1.1.8. Reflexiones tempranas.

Cuando la fuente sonora está rodeada por varias superficies (piso, paredes, techo) un oyente recibirá el sonido directo, y además el sonido reflejado en cada pared. Las primeras reflexiones recibidas, que se encuentran bastante separadas en el tiempo, se denominan reflexiones tempranas [16].

1.1.9. Absorción sonora.

Las superficies de un recinto reflejan solo parcialmente el sonido que incide sobre ellas; el resto es absorbido. Según el tipo de material o recubrimiento de una pared, ésta podrá absorber más o menos el sonido, lo cual lleva a definir el coeficiente de absorción sonora,

abreviado con la letra griega α (alfa), como el cociente entre la energía absorbida y la energía incidente [16]:

$$a = \frac{E_{absorbida}}{E_{incidente}} \quad (4)$$

El coeficiente de absorción tiene una gran importancia para el comportamiento acústico de un ambiente, y por esa razón se han medido y tabulado los coeficientes de absorción para varios materiales y objetos [16].

Materiales	Coeficiente de absorción α a la frecuencia					
	125	250	500	1000	2000	4000
Hormigón sin pintar	0.01	0.01	0.02	0.02	0.04	0.04
Hormigón pintado	0.01	0.01	0.01	0.02	0.02	0.02
Ladrillo visto sin pintar	0.02	0.02	0.03	0.04	0.05	0.05
Ladrillo visto pintado	0.01	0.01	0.02	0.02	0.02	0.02
Revoque de cal y arena	0.04	0.05	0.06	0.08	0.04	0.06
Placa de yeso (Durlock) 12mm a 10cm	0.29	0.10	0.05	0.04	0.07	0.09
Yeso sobre metal desplegado	0.04	0.04	0.04	0.06	0.06	0.03
Mármol o azulejo	0.01	0.01	0.01	0.01	0.02	0.02
Madera en paneles (a 5cm de la pared)	0.30	0.25	0.20	0.17	0.15	0.10
Madera aglomerada en panel	0.47	0.52	0.50	0.55	0.58	0.63
Parquet	0.04	0.04	0.07	0.06	0.06	0.07
Parquet sobre asfalto	0.05	0.03	0.06	0.09	0.10	0.22
Parquet sobre listones	0.20	0.15	0.12	0.10	0.10	0.07
Alfombra de goma 0.5cm	0.04	0.04	0.08	0.12	0.03	0.10
Alfombra de lana 1.2kg/m ²	0.10	0.16	0.11	0.30	0.50	0.47
Alfombra de lana 2.3kg/m ²	0.17	0.18	0.21	0.50	0.63	0.83
Cortina 338g/m ²	0.03	0.04	0.11	0.14	0.24	0.35
Cortina 338g/m ² fruncida al 50 %	0.07	0.31	0.49	0.75	0.70	0.60
Espuma de poliuretano (Fonac) 35mm	0.11	0.14	0.36	0.82	0.90	0.97
Espuma de poliuretano (Fonac) 50mm	0.15	0.25	0.50	0.94	0.92	0.99
Espuma de poliuretano (Fonac) 75mm	0.17	0.44	0.99	1.03	1.00	1.03
Espuma de poliuretano (Sonex) 35mm	0.06	0.20	0.45	0.71	0.95	0.89
Espuma de poliuretano (Sonex) 50mm	0.07	0.32	0.72	0.88	0.97	1.01
Espuma de poliuretano (Sonex) 75mm	0.13	0.53	0.90	1.07	1.07	1.00
Lana de vidrio (fieltró 14kg/m ³) 25mm	0.15	0.25	0.40	0.50	0.65	0.70
Lana de vidrio (fieltró 14kg/m ³) 50mm	0.25	0.45	0.70	0.80	0.85	0.85
Lana de vidrio (panel 14kg/m ³) 25mm	0.20	0.40	0.80	0.90	1.00	1.00
Lana de vidrio (panel 14kg/m ³) 50mm	0.30	0.75	1.00	1.00	1.00	1.00
Ventana abierta	1.00	1.00	1.00	1.00	1.00	1.00
Vidrio	0.03	0.02	0.02	0.01	0.07	0.04
Panel cielorraso Spanacustic (Manville) 19mm	-	0.80	0.71	0.86	0.68	-
Panel cielorraso Acustidom (Manville) 4mm	-	0.72	0.61	0.68	0.79	-
Panel cielorraso Prismatic (Manville) 19mm	-	0.70	0.61	0.70	0.78	-
Panel cielorraso Profil (Manville) 19mm	-	0.72	0.62	0.69	0.78	-
Panel cielorraso fisurado Auratone (USG) 5/8"	0.34	0.36	0.71	0.85	0.68	0.64
Panel cielorraso fisurado Cortega (AWI) 5/8"	0.31	0.32	0.51	0.72	0.74	0.77
Asiento de madera (0.8 m ² /asiento)	0.01	0.02	0.03	0.04	0.06	0.08
Asiento tapizado grueso (0.8 m ² /asiento)	0.44	0.44	0.44	0.44	0.44	0.44
Personas en asiento de madera (0.8 m ² /persona)	0.34	0.39	0.44	0.54	0.56	0.56

Personas en asiento tapizado ($0.8 \text{ m}^2/\text{persona}$)	0.53	0.51	0.51	0.56	0.56	0.59
Personas de pie ($0.8 \text{ m}^2/\text{persona}$)	0.25	0.44	0.59	0.56	0.62	0.50

Tabla 2: Coeficientes de absorción de diversos materiales en función de la frecuencia (según varias fuentes) [16].

En la tabla 2 se dan los valores de α para varios materiales típicos de construcción, objetos y personas. Se proporcionan para varias frecuencias, ya que α depende bastante de la frecuencia [16].

1.1.10. Tiempos de reverberación.

Después del periodo de las reflexiones tempranas, comienzan a aparecer las reflexiones de las reflexiones, y las reflexiones de las reflexiones de las reflexiones, y así sucesivamente, dando origen a una situación muy compleja en la cual las reflexiones se densifican cada vez más. Esta permanencia del sonido aun después de interrumpida la fuente se denomina reverberación. Para medir cuánto demora este proceso de extinción del sonido se introduce el concepto de tiempo de reverberación, T, técnicamente definido como el tiempo que demora el sonido en bajar 60 dB por debajo de su nivel inicial [16].

La propiedad anterior se puede expresar por medio de una fórmula, denominada fórmula de Sabine, en honor al físico norteamericano que la obtuvo a principios de este siglo. Según dicha fórmula, el tiempo de reverberación T puede calcularse como:

$$T = 0,161 \frac{V}{\alpha S} \quad (5)$$

donde V es el volumen de la habitación en m^3 , S es el área de su superficie interior total en m^2 , y α es el coeficiente de absorción sonora [16].

1.1.11. Tiempos de reverberación óptimo.

Varias investigaciones realizadas evaluando las acústicas de las mejores salas del mundo (según la opinión de las audiencias o usuarios y de expertos) han revelado que para cada finalidad existe un tiempo de reverberación óptimo, que aumenta al aumentar el volumen en m^3 de la sala.

En la ilustración 1 se muestra el resultado de uno de estos estudios. Debe aclararse que no hay coincidencia entre los resultados presentados por diversos investigadores, aunque cualitativamente son similares.

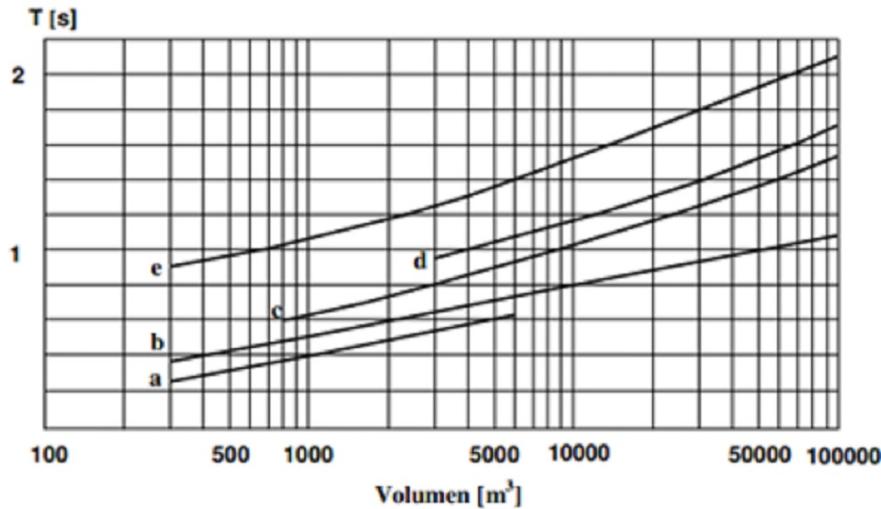


Figura 1: Tiempo de reverberación óptimo en función del volumen de una sala (según L. L. Beranek). (a) Estudios de radiodifusión para voz. (b) Salas de conferencias. (c) Estudios de radiodifusión para música. (d) Salas de conciertos. (e) Iglesias [16].

1.1.12. Resonancias.

Las resonancias o modos normales de vibración suceden como consecuencia de las reflexiones sucesivas en paredes opuestas. Una onda estacionaria es una onda que va y viene una y otra vez entre dos paredes, por lo que, si la distancia entre las dos paredes es L , la longitud de tal onda es $2L$ y por consiguiente deberá cumplir que

$$2 \cdot L = \frac{c}{f} \quad (6)$$

Donde c es la velocidad del sonido, y f la frecuencia del sonido resultante. De aquí se puede obtener la frecuencia, que resulta ser

$$f = \frac{c}{2 \cdot L} \quad (7)$$

Para las frecuencias de resonancia el tiempo de reverberación es mucho más prolongado, por lo cual dicha nota se prolongará más que las otras. Esto se considera un defecto acústico importante. Entre las posibles soluciones, están: a) evitar las superficies paralelas, que favorecen las resonancias, b) agregar absorción acústica que reduzca el tiempo de reverberación, c) ecualizar el sistema de sonido de modo de atenuar las frecuencias próximas a la resonancia o resaltar las otras frecuencias [16].

1.1.13. Claridad.

La claridad describe el grado en que cada detalle de las actuaciones puede ser percibido en lugar de que todo se difumine por los componentes de sonido reverberantes que llegan más tarde. Por lo tanto, la claridad es en gran medida una propiedad complementaria a la reverberancia.

Cuando las reflexiones se retrasan no más de 50-80 ms en relación con el sonido directo, el oído integrará estas contribuciones y el sonido directo, lo que significa que percibimos principalmente el efecto como si el sonido claro y original se hubiera amplificado en relación con la energía reverberante posterior. Por lo tanto, se ha encontrado que un parámetro objetivo que compara la relación entre la energía en la respuesta al impulso antes y después de 80 ms es un descriptor razonablemente bueno de claridad.

$$C = 10 \log_{10} \left[\frac{\int_0^{80ms} h^2(t)dt}{\int_{80ms}^{\infty} h^2(t)dt} \right] \quad (8)$$

Cuanto mayor sea el valor de C, más dominará el sonido inicial y mayor será la impresión de claridad [17].

1.1.14. Potencia sonora

La influencia de la sala en la sonoridad percibida es otro aspecto importante de la acústica de la sala. Una medida relevante de esta propiedad es simplemente la diferencia en dB entre el nivel de una fuente de sonido continua y calibrada medido en la habitación y el nivel que genera la misma fuente a 10 m de distancia en un entorno anecoico. Esta medida objetiva, denominada fuerza (relativa) G, también puede obtenerse a partir de registros de respuesta al impulso a partir de la relación entre la energía total de la respuesta al impulso y la energía del sonido directo, y esta última se registra a una distancia fija (10 m) de la fuente de sonido impulsivo:

$$G = 10 \log_{10} \left[\frac{\int_0^{\infty} h^2(t)dt}{\int_0^{t_{dir}} h_{10m}^2(t)dt} \right] \quad (9)$$

En este caso, el límite superior de integración en el denominador t_{dir} debe limitarse a la duración del impulso de sonido directo (que en la práctica dependerá de la anchura de banda seleccionada). Se puede utilizar una distancia diferente de 10 m, si también se aplica una corrección para la atenuación de la distancia.

El valor esperado de G según la teoría de campos difusos se convierte en una función de T, así como del volumen de la habitación, V [17]:

$$G_{exp} = 10 \log_{10} \left(\frac{T}{V} \right) + 45dB \quad (10)$$

1.1.15. Medidas de amplitud

La amplitud es la sensación de que el sonido llega desde muchas direcciones diferentes en contraste con una impresión monofónica de todo el sonido que llega al oyente a través de una abertura estrecha. Ahora está claro que hay dos aspectos de la amplitud, los cuales son atractivos, especialmente cuando se escucha música:

Anchura aparente de la fuente (ASW): la impresión de que la imagen sonora es más amplia que la extensión visual y física de la(s) fuente(s) en el escenario. Envolvente del oyente (LEV): la impresión de estar dentro y rodeado por el campo sonoro reverberante de la habitación.

Se ha encontrado que ambos aspectos dependen de la dirección de incidencia de las reflexiones de respuesta al impulso. Cuando una mayor parte de la energía de reflexión temprana (hasta unos 80 ms) llega desde direcciones laterales (desde los lados), el ASW aumenta. Cuando el nivel de las reflexiones laterales tardías es alto, se produce un fuerte LEV.

Los componentes laterales de la energía de respuesta al impulso se pueden grabar utilizando un micrófono en forma de ocho con el eje sensible mantenido horizontal y perpendicular a la dirección hacia la fuente de sonido (de modo que la fuente se encuentre en el plano sordo del micrófono). Para la medición de la fracción de energía lateral (LEF), la parte inicial (hasta 80 ms) de esta energía sonora lateral se compara con la energía del sonido directo más todas las reflexiones tempranas captadas por un micrófono omnidireccional ordinario

$$LEF = \frac{\int_{t=5ms}^{t=80ms} h_1^2(t)dt}{\int_{t=0ms}^{t=80ms} h^2(t)dt} \quad (11)$$

donde h_1 es la presión de respuesta al impulso registrada con un micrófono en forma de ocho, mientras que h se captura a través del micrófono omnidireccional (habitual). Es principalmente la energía a frecuencias bajas y medias la que contribuye a la amplitud. En consecuencia, LEF se promedia normalmente en las cuatro octavas 125-1000 Hz. Cuanto mayor sea el valor de LEF, más amplio será el ASW. En un campo completamente difuso, LEF sería constante, con un valor de 0,33, que es superior al que normalmente se encuentra en las salas reales. La diferencia subjetiva para LEF es de aproximadamente el 5 %. El aspecto ASW de la amplitud no solo depende de la relación entre el sonido lateral temprano y el sonido temprano total; pero también en el nivel total del sonido. Cuanto mayor sea el valor G (y más fuerte sea la fuente de sonido), más amplia será la imagen acústica de la fuente. Sin embargo, en el momento de escribir este artículo, no existe una forma sólida de incorporar la influencia del nivel en la medida objetiva. La envolvente del oyente parece estar determinada principalmente por la distribución espacial y el nivel de las reflexiones tardías (que llegan después de 80 ms) [17].

1.1.16. Parámetros relacionados con el timbre o el color tonal

El timbre describe el grado en que la habitación influye en el equilibrio de frecuencias entre las frecuencias altas, medias y bajas, es decir, si el sonido es áspero, brillante, hueco, cálido o cualquier otro adjetivo que se use para describir el color tonal. Tradicionalmente, se ha utilizado un gráfico de la variación de frecuencia de T (por 1/1 o 1/3 de octava) para indicar esta cualidad; pero se ha sugerido un conveniente parámetro de un solo número destinado a medir la calidez de la sala: la relación de graves (BR) dada por:

$$BR = \frac{T_{125Hz} + T_{250Hz}}{T_{500Hz} + T_{1000Hz}} \quad (12)$$

Del mismo modo, una relación de agudos (TR) se puede formar como:

$$BR = \frac{T_{2000Hz} + T_{4000Hz}}{T_{500Hz} + T_{1000Hz}} \quad (13)$$

Sin embargo, en algunas salas, se experimenta una falta de sonido de graves a pesar de los altos valores de T en las frecuencias bajas. Por lo tanto, EDT o tal vez G versus frecuencia sería un parámetro mejor, e intuitivamente más lógico, para la medición del timbre. Del mismo modo, BR o TR podrían basarse en valores G en lugar de T [17].

1.1.17. Inteligibilidad del habla

Todos los parámetros objetivos mencionados anteriormente (excepto el parámetro básico T), son principalmente relevantes en auditorios más grandes destinados a la interpretación de música. En los auditorios utilizados para el habla, como las salas de conferencias o los teatros, la influencia de la acústica en la inteligibilidad es un problema importante. Actualmente, la forma más común de evaluar objetivamente la inteligibilidad del habla en las salas es mediante la medición del índice de transmisión del habla STI [17].

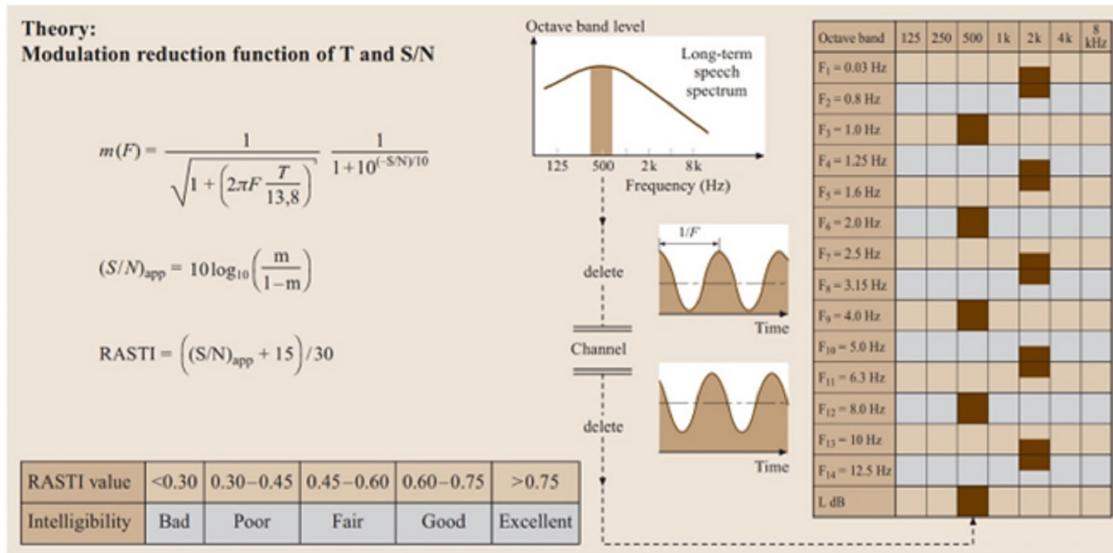


Figura 2: Ilustración de la teoría y el principio en la medición de STI o RASTI. La escala para la evolución de los valores RASTI se muestra en la parte inferior [17].

1.2. Marco Precedimental

1.2.1. Definición de la metodología mecatrónica

La mecatrónica es una ingeniería interdisciplinaria y, por tanto, es necesario buscar metodologías de diseño que nos permita especificar las necesidades del sistema, planear actividades, dividir tareas y validar resultados.

En [18] se muestra la metodología mecatrónica ‘VDI 2206’. En ella se sigue un modelo procedimental flexible basado en tres elementos principales:

- El ciclo general de resolución de problemas a pequeña escala (micro-level)
- El modelo en forma de V en la gran escala (macro-level)
- Módulo de procesos predefinidos para pasos de operación que se repiten durante el diseño de los sistemas mecatrónicos.

1.2.2. Resolución de problemas en el micro-level

La resolución de problemas en el micro-level comienza de dos maneras: analizando la situación actual y analizando la relación de la condición actual con la condición meta. Lo anterior nos lleva a una síntesis y análisis que permite generar, rechazar y elegir soluciones. El análisis de la solución y la evaluación nos puede llevar a replantear la meta o a volver a analizar la situación. Si el resultado es satisfactorio, nos sirve como herramienta de aprendizaje y se planean las acciones futuras.

El flujo de esta metodología de resolución de problemas se puede ver en la figura 3:

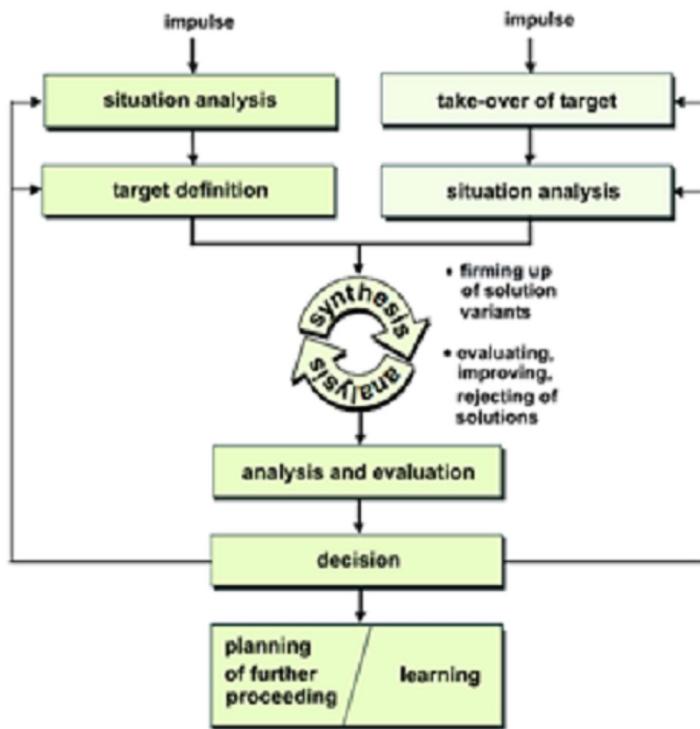


Figura 3: Diagrama de flujo para la resolución de problemas en el micro-level.

1.2.3. Modelo en V en el macro-level

En modelo en V, en donde la primera parte es un procedimiento de arriba hacia abajo y la segunda parte es de abajo hacia arriba, nos permitirá separar las tareas a gran escala y realizar validaciones constantes de los sistemas.

El modelo V busca partir de los requerimientos y realizar un diseño del sistema conceptual, el cual es multidominio y describe las características esenciales del producto. A continuación, hace diseños específicos para cada dominio, los cuales integra en una etapa posterior y analiza sus interrelaciones. A lo largo de este proceso se debe hacer una verificación y validación constante de que el diseño cumpla con los requerimientos y el diseño conceptual del producto, además de apoyarse en el modelado y la simulación por computadora para hacer estas validaciones. En caso de cumplirse con esto se cuenta con un producto que cumple con los requisitos.

El modelo V es un modelo iterativo, es decir, es poco probable que el primer producto al que se llegue después de la integración sea totalmente funcional o incluso el óptimo. Por lo anterior, se deben llevar a cabo múltiples ciclos de diseño y validación, para que el producto alcance un mayor grado de madurez. Cada etapa tiene un “contraparte” a la que se puede volver en caso de que no se cumpla alguna verificación, esto lleva a mejorar el diseño y analizar múltiples opciones de solución.

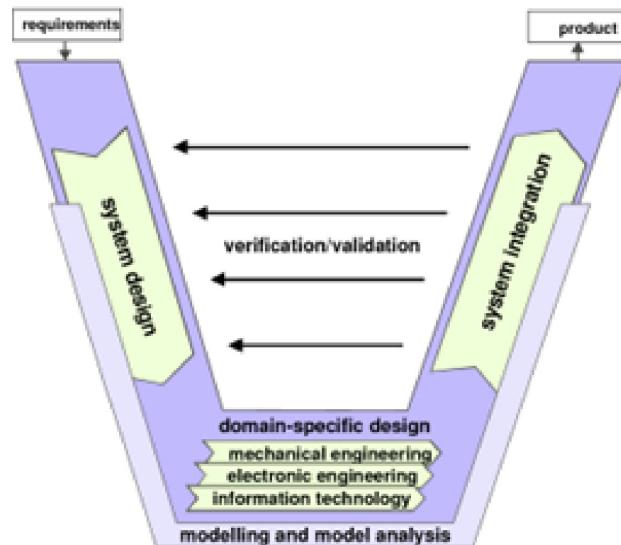


Figura 4: Metodología por seguir.

1.2.4. Procesos predefinidos

En cada una de las etapas de diseño, se tienen procesos predefinidos que se repiten regularmente durante estas etapas. Cada uno de estos procesos están mejor definidos en [19] y se utilizarán en el desarrollo de este proyecto para la generación de soluciones.

2. Diseño del sistema

2.1. Diseño conceptual

2.1.1. Necesidades y requerimientos

La metodología VDI 2206 emplea los requerimientos como entrada para comenzar el proceso, por tanto, estos deben definirse con anterioridad. Para este propósito se utiliza la ingeniería de requerimientos.

La ingeniería de requerimientos consiste en definir primero las necesidades funcionales y no funcionales y transformarlas en requerimientos que son simplemente propiedades medibles.

ID	Necesidades	Clasificación
N1	Medición de la acústica	Funcional
N2	Movimiento de paneles	Funcional
N3	Modificación de la acústica	Funcional
N4	Repetibilidad en acondicionamiento acústico	Funcional
N5	Fiabilidad en acondicionamiento acústico	Funcional
N6	Ingreso de datos por medio de interfaz de usuario	Funcional
N7	Control local de la posición de los paneles	Funcional
N8	Instalación posible en paredes	Funcional
N9	Hacer acondicionamiento a partir de tipo de instrumento a grabar	Funcional
N10	Condiciones de operación propias de un estudio de audio	No funcional
N11	No requerimiento de conocimientos en acondicionamiento acústico para la operación	No funcional
N12	Despliegue de información por medio de interfaz	No funcional

Tabla 3: Necesidades del sistema automatizado para el acondicionamiento acústico.

ID	Requerimiento	Valor
R1	Función principal	Hacer el acondicionamiento acústico de un estudio de audio de manera automática para diferentes instrumentos
R2	Ubicación	Cuarto destinado a estudio de audio en <19.509021,-99.096411>
R3	Dimensiones del estudio de audio	6m x 4m x 3.6m
R4	Medición del tiempo de decaimiento temprano (EDT)	Si
R5	Medición del tiempo de reverberación (T20 y T30)	Si

R6	Medición de la claridad (C50 y C80)	Si
R7	Medición de la fuerza del sonido (G)	Si
R8	Medición de la definición (D50)	Si
R9	Disminución de ondas estacionarias	Si
R10	Modificación mínima en tiempo de reverberación	0.1 s
R11	Número de parámetros acústicos calculados	7
R12	Error de acondicionamiento	< 5 %
R13	Diferencia entre acondicionamientos sucesivos	< 5 %
R14	Tensión de alimentación	127 VAC
R15	Corriente máxima	30 A
R16	Sistema de selección de instrumento	Si
R17	Soporte para paredes	Si
R18	Información en interfaz para usuario	Si

Tabla 4: Tabla de requerimientos.

Se realizó la validación de los requerimientos acorde con las necesidades planteadas. Para ello, se elaboró la matriz de trazabilidad mostrada en la tabla 4, la cual permite visualizar de manera gráfica cómo se relacionan, así como corroborar que los requerimientos satisfacen todas las necesidades.

	N1	N2	N3	N4	N5	N6	N7	N8	N9	N10	N11	N12
R1												
R2												
R3												
R4												
R5												
R6												
R7												
R8												
R9												
R10												
R11												
R12												
R13												
R14												
R15												
R16												
R17												
R18												

Tabla 5: Matriz de trazabilidad necesidades-requerimiento.

2.1.2. Arquitectura funcional

Siguiendo con la metodología, se decidió definir las funciones y subfunciones que debe desarrollar el sistema, para así poder comenzar su diseño.

Función principal: Hacer un acondicionamiento acústico a un cuarto en función de la acústica óptima para un cierto instrumento.

F1. Medición de la acústica actual

- C1. El sistema debe generar un sonido en el estudio de audio.
- C2. El sistema debe grabar la respuesta del estudio de audio al sonido generado.
- C3. El sistema debe calcular el tiempo de decaimiento temprano (EDT) a partir de la respuesta del estudio de audio.
- C4. El sistema debe calcular el tiempo de reverberación (T20 y T30) a partir de la respuesta del estudio de audio.
- C5. El sistema debe calcular la claridad (C50 y C80) a partir de la respuesta del estudio de audio.
- C6. El sistema debe calcular la definición (D50) a partir de la respuesta del estudio de audio.
- C7. El sistema debe calcular la fuerza del sonido (G) a partir de la respuesta del estudio de audio.
- C8. El sistema debe calcular los modos normales de vibración del estudio de audio.

F2. Modificación de la acústica

- C1. El sistema debe conocer la diferencia entre la acústica actual y la acústica deseada.
- C2. El sistema debe calcular la posición de los paneles que generan la acústica deseada.
- C3. El sistema debe calcular la diferencia entre la posición actual de los paneles y la posición deseada.
- C4. El sistema debe mover los paneles a la posición deseada.
- C5. El sistema debe disminuir la generación de ondas estacionarias.

F3. Interacción con el usuario

- C1. El sistema debe recibir el tipo de instrumento que se va a grabar.
- C2. El sistema debe saber la acústica óptima para el instrumento que se va a grabar.

- C3. El sistema debe mostrar al usuario los parámetros acústicos que definen la acústica actual.
- C4. El sistema debe mostrar al usuario las frecuencias problemáticas del estudio de audio (que generan ondas estacionarias).
- C5. El sistema debe dar control al usuario de cuándo hacer un acondicionamiento acústico.
- C6. El sistema debe dar a elegir al usuario para qué tipo de instrumento hacer el acondicionamiento acústico.
- C7. El sistema debe mostrar la acústica ideal para dicho instrumento.
- C8. El sistema debe mostrar la acústica después del acondicionamiento y compararla con la acústica ideal.

F4. Gestión de energía

- C1. El sistema tiene un bajo consumo energético cuando no se está moviendo.
- C2. El sistema gestiona el movimiento de los paneles para evitar un uso excesivo de corriente.
- C3. El sistema debe transformar y controlar la energía que obtiene de la red eléctrica para entregar la tensión de alimentación correcta a cada componente.

1. Medición de la acústica actual

- 1.1. Generación de onda de sonido
 - 1.1.1. Selección de ondas necesarias
 - 1.1.2. Gestión de secuencia de ondas
 - 1.1.3. Generación de secuencia de ondas
- 1.2. Obtención de la respuesta
 - 1.2.1. Sensado de la respuesta
- 1.3. Procesamiento de la respuesta
 - 1.3.1. Cálculo de los parámetros acústicos
 - 1.3.1.1. Cálculo del tiempo de decaimiento temprano (EDT)
 - 1.3.1.2. Cálculo del tiempo de reverberación (T20 y T30)
 - 1.3.1.3. Cálculo de la claridad (C50 y C80)
 - 1.3.1.4. Cálculo de la definición (D50)
 - 1.3.1.5. Cálculo de la fuerza del sonido (G)
 - 1.3.2. Cálculo de modos normales de vibración
 - 1.3.2.1. Obtención de dimensiones del estudio de audio
 - 1.3.2.2. Obtención de posición de la fuente
 - 1.3.2.3. Obtención de posición del receptor

2. Modificación de la acústica

- 2.1. Cálculo de posiciones deseadas
- 2.2. Compensación para disminución de modos normales de vibración
- 2.3. Movimiento de paneles
 - 2.3.1. Medición del error de posición
 - 2.3.1.1. Medición de la posición actual
 - 2.3.1.2. Cálculo del error de posición
 - 2.3.2. Procesamiento por estrategia de control
3. Interacción con el usuario
 - 3.1. Recepción de instrucciones de acondicionamiento
 - 3.1.1. Selección de instrumento
 - 3.1.1.1. Despliegue de acústica ideal
 - 3.2. Despliegue de acústica actual
 - 3.2.1. Despliegue de las variables acústicas calculadas
 - 3.2.2. Despliegue de las frecuencias de vibración problemáticas
 - 3.3. Despliegue de la acústica posterior al acondicionamiento
 - 3.3.1. Despliegue de las variables acústicas calculadas
 - 3.3.2. Despliegue de las ondas estacionarias
 - 3.4. Control de apagado y encendido
 4. Gestión de energía
 - 4.1. Conversión y regulación de tensión
 - 4.1.1. Conversión de tensión
 - 4.1.2. Regulación de tensión
 - 4.2. Medición del consumo energético
 - 4.2.1. Medición de la tensión de entrada
 - 4.2.2. Medición de la corriente de entrada

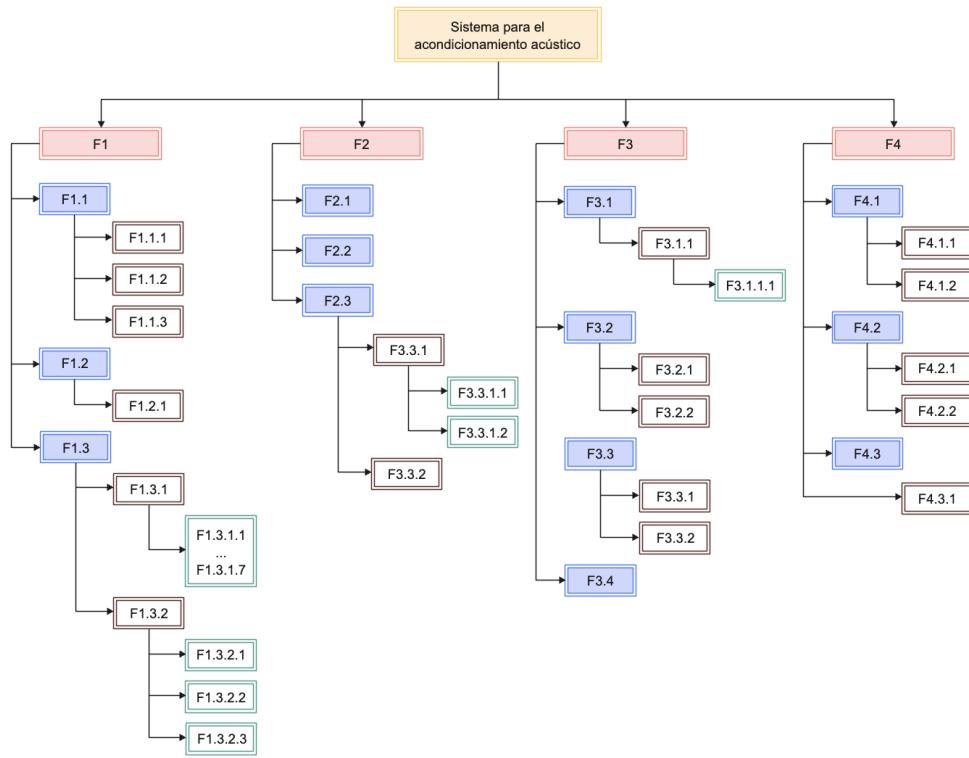


Figura 5: Diagrama de las funciones del sistema para el tratamiento acústico.

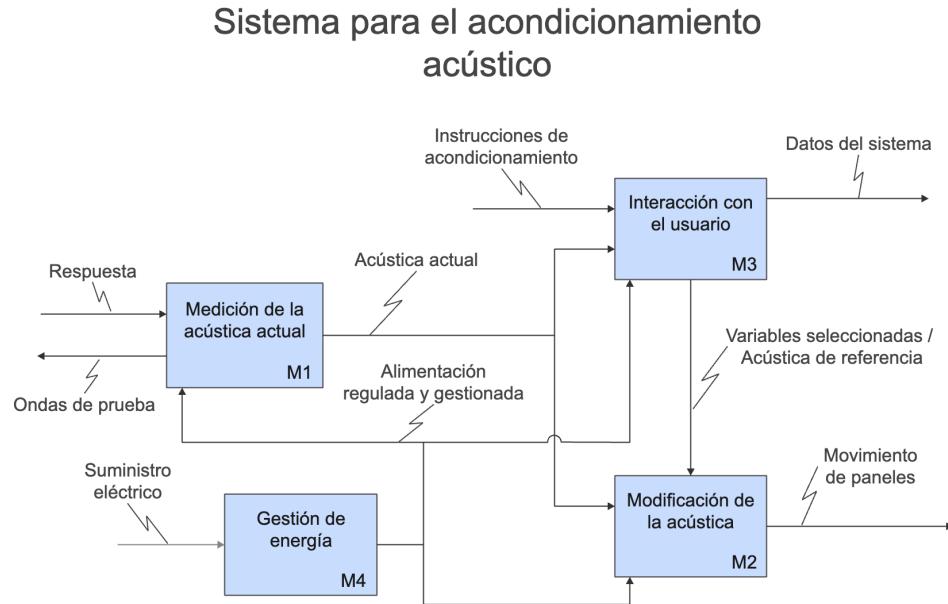


Figura 6: IDEF0 del sistema para el acondicionamiento acústico.

(M1) Medición de la Acústica Actual

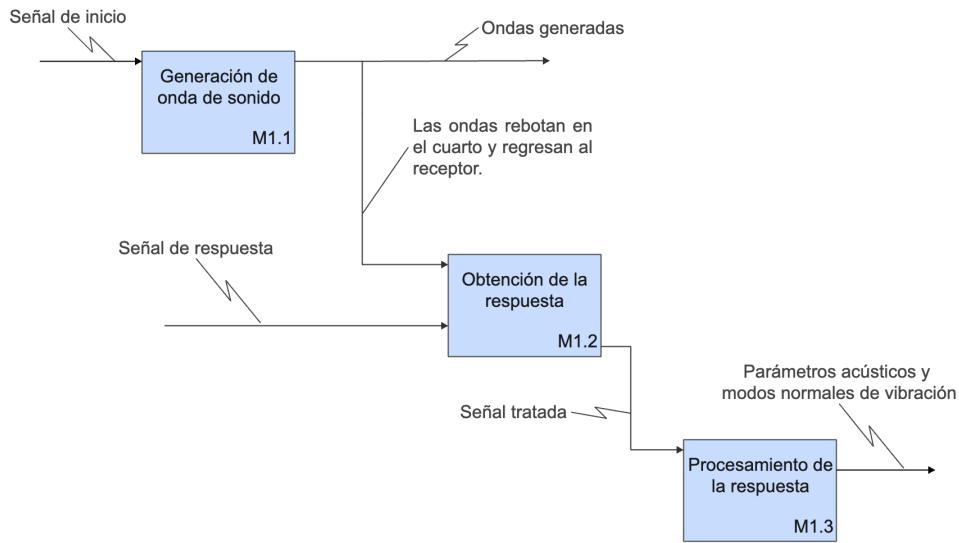


Figura 7: IDEF0 del Módulo de medición de la acústica actual.

(M2) Modificación de la Acústica

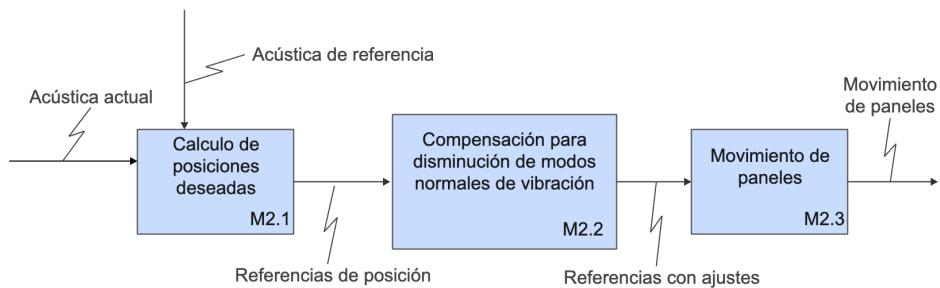


Figura 8: IDEF0 del Módulo de modificación de la acústica.

(M3) Interacción con el usuario

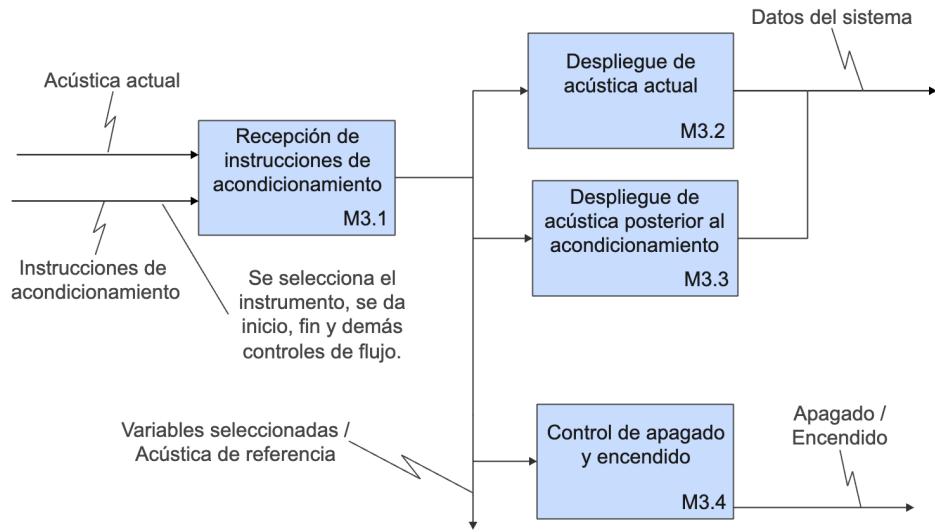


Figura 9: IDEF0 del Módulo de interacción con el usuario.

(M4) Gestión de energía

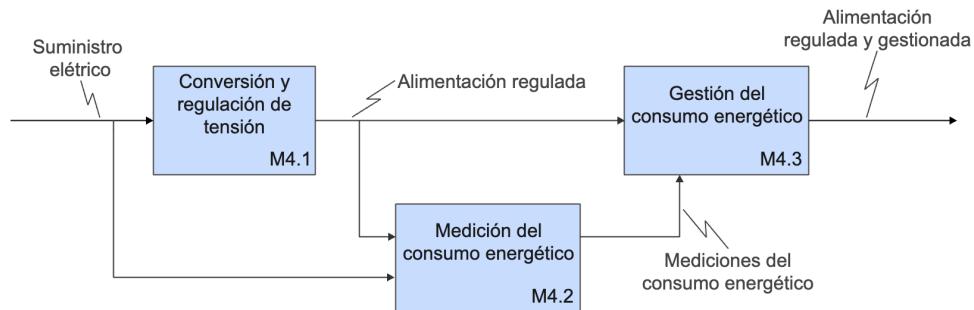


Figura 10: IDEF0 del Módulo de gestión de energía.

Se procedió a validar la relación entre los requerimientos y las funciones, para lo cual se elaboró la matriz de trazabilidad presentada en la Tabla “Matriz de trazabilidad requerimientos-funciones”.

	F1.1	F1.2	F1.3	F2.1	F2.2	F2.3	F3.1	F3.2	F3.3	F3.4	F4.1	F4.2	F4.3
R1													
R2													
R3													
R4													

R5												
R6												
R7												
R8												
R9												
R10												
R11												
R12												
R13												
R14												
R15												
R16												
R17												
R18												
R19												
R20												
R21												

Tabla 6: Matriz de trazabilidad requerimientos-funciones.

2.1.3. Arquitectura física

Para dar forma a la parte física del proyecto se definen los módulos que darán integridad a la propuesta de solución. La arquitectura física del sistema es representada por la transformación de las funciones en módulos, que tomarán parte en los ensambles y componentes físicos. La arquitectura física está planteada para ser capaz de realizar las funciones descritas en la arquitectura funcional y cumplir con los requerimientos del sistema para lograr el funcionamiento deseado.

MF1 Módulo de procesamiento: Es un módulo cuya función principal es el procesamiento de las ondas de sonido, el cálculo de las posiciones de los paneles y el procesamiento de los comandos ingresados por el usuario. El módulo se define en submódulos, donde el submódulo **MF_{1,1}** es el encargado de procesar la respuesta del estudio y realizar los cálculos de los parámetros acústicos y los modos normales de vibración. En el submódulo **MF_{1,2}** se generan los cálculos de las posiciones deseadas de los paneles, así como el procesamiento de la estrategia de control para el movimiento de los mismos, contemplando la medición y el cálculo del error de posición, para que la localización de los paneles sea fiable. El submódulo **MF_{1,3}** se encarga de procesar todas las instrucciones de acondicionamiento ingresadas por el usuario, así como de todos los parámetros acústicos que se despliegan en la interfaz de muestreo.

MF2 Módulo de generación y medición de la acústica: Tiene la función de generar las ondas que excitan el estudio (**MF_{2,1}**) y medir la interacción de las ondas con el recinto (**MF_{2,2}**). El submódulo de la generación de las ondas de sonido funciona a partir de un algoritmo que genera una señal de excitación que es transmitida al estudio mediante una bocina, las ondas que son transmitidas se encargan de describir las características

del estudio. Así, las señales que interactúan con el estudio posteriormente son recibidas por el submódulo de medición que consta de un micrófono que capta todas las señales que han interactuado con el recinto.

- MF3 Módulo modificador de la acústica: Este módulo tiene la función de generar el movimiento de los paneles acústicos considerando la compensación para disminuir los modos normales de vibración (**MF_{3,1}**). El movimiento se lleva a cabo mediante un mecanismo que permite su colocación en un punto donde se requiere modificar la superficie que entra en contacto con las ondas sonoras, este mecanismo es capaz de sensar el error de posición con el propósito de que el lugar en el que el módulo requiere la localización del panel sea específicamente la solicitada.
- MF4 Módulo interfaz de usuario: Además de gestionar el encendido y apagado del sistema, el submódulo de interacción directa (**MF_{4,1}**) se encarga de recibir las instrucciones por parte del usuario para el acondicionamiento del estudio, es decir, la selección del instrumento. Al seleccionar el instrumento, la consola muestra a través del submódulo de muestreo de datos (**MF_{4,2}**) la acústica ideal para dicho instrumento y posteriormente de que el módulo dos realiza el acondicionamiento se muestran los nuevos valores de las variables acústicas. De igual forma la interfaz muestra las variables acústicas antes de realizar el proceso de acondicionamiento y las frecuencias de vibración que son un problema para el sistema.
- MF5 Módulo gestor de energía: Se encarga de gestionar el suministro eléctrico requerido por cada uno de los módulos y submódulos. El submódulo de conversión y regulación de la tensión de entrada (**MF_{5,1}**) convierte y regula la señal de corriente alterna que es suministrada al sistema dependiendo de las características de alimentación de cada uno de los componentes del sistema. El módulo es capaz de medir el consumo energético del sistema y sus componentes, a través del submódulo de medición del consumo energético (**MF_{5,2}**), es decir, la medición de la tensión de entrada y la corriente que demanda el sistema.

De igual manera se realizó la validación de los módulos y submódulos acorde a las funciones planteadas para el sistema, haciendo uso de la matriz de trazabilidad que se muestra a continuación.

	MF ₁			MF ₂		MF ₃	MF ₄		MF ₅	
	MF _{1,1}	MF _{1,2}	MF _{1,3}	MF _{2,1}	MF _{2,2}	MF _{3,1}	MF _{4,1}	MF _{4,2}	MF _{5,1}	MF _{5,2}
F_{1,1}										
F_{1,2}										
F_{1,3}										
F_{2,1}										
F_{2,2}										
F_{2,3}										
F_{3,1}										
F_{3,2}										
F_{3,3}										
F_{3,4}										

F_{4,1}											F_{4,2}	
F_{4,2}												
F_{4,3}												F_{4,3}

Tabla 7: Matriz de trazabilidad de arquitectura física

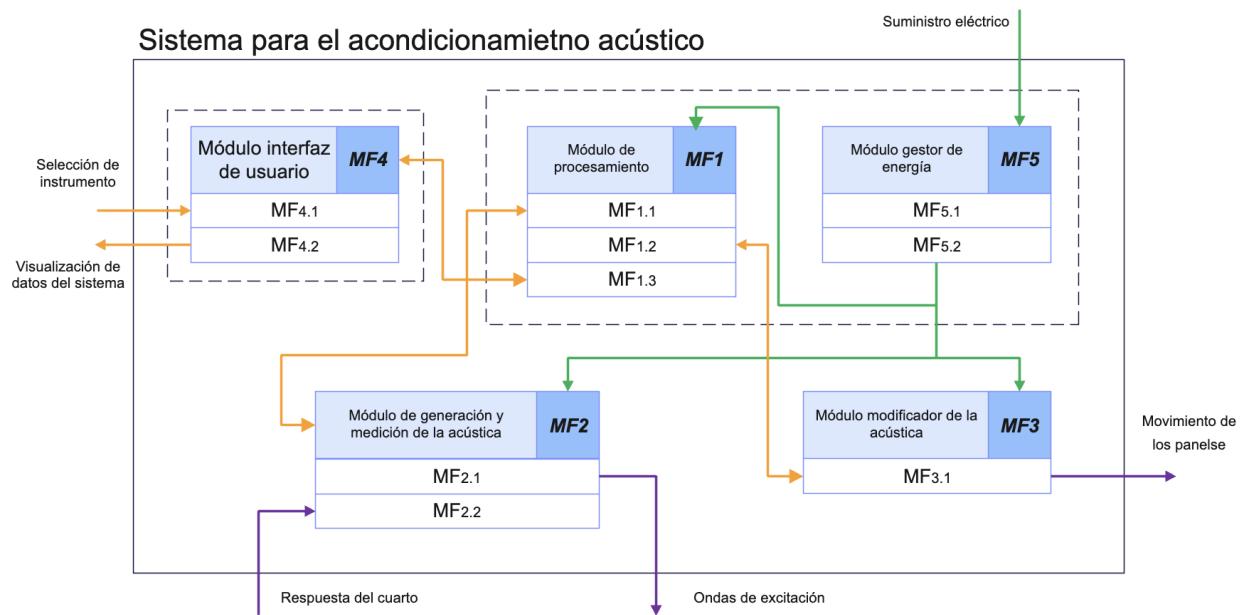


Figura 11: Diagrama del sistema para el acondicionamiento acústico

2.1.4. Propuesta de solución

Para desarrollar la propuesta de solución, primero se hizo uso de la matriz morfológica para observar las diferentes alternativas que se tienen para cada uno de los sistemas. La matriz morfológica toma los aspectos más relevantes del diseño físico y las diferentes soluciones. Lo anterior nos permitirá generar diferentes conceptos solución, que sean combinaciones de las alternativas de solución para los diferentes sistemas. Es necesario hacer una valoración de los conceptos solución, y para este propósito, es necesario contar con criterios que nos permitan separar los unos de los otros. De acuerdo a la metodología, se hará uso del Proceso Analítico Jerárquico (AHP por sus siglas en inglés).

- C_{r1} Velocidad de actuación
- C_{r2} Peso
- C_{r3} Complejidad del mecanismo
- C_{r4} Cantidad de actuadores

- C_{r5} Costo de la actuación
- C_{r6} Resolución acústica
- C_{r7} Repetibilidad
- C_{r8} Consumo energético
- C_{r9} Variedad de potencias consumidas
- C_{r10} Facilidad del control de la actuación
- C_{r11} Tamaño
- C_{r12} Confiabilidad del sistema de actuación
- C_{r13} Complejidad de manufactura del mecanismo de actuación
- C_{r14} Error de posición
- C_{r15} Cantidad de hardware externo al equipo de cómputo personal
- C_{r16} Relación entre señal y ruido
- C_{r17} Sensibilidad del micrófono
- C_{r18} Potencia de procesado de la tarjeta embebida
- C_{r19} Potencia de procesado del equipo de cómputo
- C_{r20} Simplicidad de la medición en el error de posición
- C_{r21} Robustez necesaria del anclaje
- C_{r22} Propensión a fallas estructurales
- C_{r23} Propensión a fallas eléctricas
- C_{r24} Facilidad para saltar entre disposiciones
- C_{r25} Modularidad
- C_{r26} Costo de la electrónica embebida
- C_{r27} Cantidad de paneles utilizados

La mayoría de los criterios enlistados anteriormente están íntimamente relacionados con la forma en la que el sistema mueve los paneles a lo largo del cuarto, por lo que la generación de conceptos solución, tendrán como eje principal, las diferentes formas de movimiento de los paneles. Algunas de las alternativas de solución para sistemas específicos, son compatibles con diferentes propuestas de solución, por lo que se estudiaran aparte y se adjuntaran al concepto solución elegido con base en los criterios que sí son propios del mismo.

	Características	Alternativa 1	Alternativa 2	Alternativa 3	Alternativa 4
C_1	Forma de movimiento de los paneles	Rotacional en base prismática	Tipo pick and place	Construcción y movimiento de líneas	Paneles en configuración plegable
C_2	Método para la reflexión	Paneles de reflexión	Pared de fondo	-	-
C_3	Grabación en varias posiciones	No	Manual	Automático	Con múltiples micrófonos
C_4	Generación de ondas de sonido	Bocina de equipo de cómputo	Bocina externa	Bocina para circuitos	-
C_5	Dispositivo de grabación	Micrófono de equipo de cómputo	Micrófono externo	Micrófono para circuitos	-
C_6	Plataforma de cálculo para las posiciones deseadas	Equipo de cómputo	Microcontrolador embebido	FPGA	-
C_7	Plataforma de procesado del control	Equipo de cómputo	Microcontrolador embebido	FPGA	-
C_8	Plataforma para procesado de la acústica	Equipo de cómputo	Microcontrolador embebido	FPGA	-
C_9	Sensado de las posiciones de los paneles	Encoders	Sensores de contacto	Visión artificial	Sensor de flexión
C_{10}	Plataforma de visualización de información	Equipo de cómputo	Pantalla OLED	Pantalla LCD	-
C_{11}	Control de interfaz	Pantalla táctil	Botonera	Ratón y teclado	-
C_{12}	Encapsulado de sistemas embebidos	Si	No	-	-
C_{13}	Inclusión del sistema energético en el sistema embebido	Si	No	-	-
C_{14}	Anclaje al suelo	Si	No	-	-
C_{15}	Método de anclaje	Tornillos	Colgados	Adhesivos	Magnéticos
C_{16}	Método de comunicación	Serial	Wi-fi	Bluetooth	-

Tabla 8: Matriz morfológica

2.1.4.1 Integración de los conceptos solución

Con base en las diferentes alternativas, se crearon 4 conceptos de solución, cuya mayor diferencia es el método de movimiento para los paneles. Debido a que hay alternativas que no son dependientes del método de movimiento, la selección de estas se hará independientemente.

	Características	<i>CS₁</i>	<i>CS₂</i>	<i>CS₃</i>	<i>CS₄</i>
<i>C₁</i>	Forma de movimiento de los paneles	Rotacional en base prismática	Tipo pick and place	Construcción y movimiento de líneas	Paneles en configuración plegable
<i>C₂</i>	Tipo de actuación	Con motor y acopladores magnéticos	Con motor y actuador magnético	Con motores	Motores lineales
<i>C₃</i>	Sensado de las posiciones de los paneles	Encoders, sensores de contacto o visión artificial	Encoders o visión artificial	Encoders	Sensores de flexión
<i>C₄</i>	Posicionamiento de paneles	Anclados a la base prismática	Acoplamiento en pared	Acoplado a un cable	Acoplados entre si
<i>C₅</i>	Sistema de alimentación de paneles	No	Si	Si	No

Tabla 9: Conceptos solución

2.1.5. Selección diseño conceptual

Primero se hará la comparación de los diferentes conceptos solución con base en cada uno de los criterios. Es importante notar que no todos los criterios se pueden aplicar para discernir entre los diferentes conceptos solución, por tanto, esos criterios se aplicaran a las alternativas que se analizaran individualmente.

	<i>CS₁</i>	<i>CS₂</i>	<i>CS₃</i>	<i>CS₄</i>
<i>CS₁</i>	1	10	5	2
<i>CS₂</i>	0.10	1	0.50	0.20
<i>CS₃</i>	0.20	2	1	0.40
<i>CS₄</i>	0.50	5	2.50	1

Tabla 10: Matriz de comparación de *Cr₁*

	<i>CS₁</i>	<i>CS₂</i>	<i>CS₃</i>	<i>CS₄</i>	<i>V_{Cr₁}</i>
<i>CS₁</i>	0.56	0.56	0.56	0.56	0.56
<i>CS₂</i>	0.06	0.06	0.06	0.06	0.06
<i>CS₃</i>	0.11	0.11	0.11	0.11	0.11
<i>CS₄</i>	0.28	0.28	0.28	0.28	0.28

Tabla 11: Matriz normalizada de *Cr₁* y *V_{Cr₁}*

	<i>CS₁</i>	<i>CS₂</i>	<i>CS₃</i>	<i>CS₄</i>
<i>CS₁</i>	1	0.38	0.38	1
<i>CS₂</i>	2.67	1	1	2.67
<i>CS₃</i>	2.67	1	1	2.67
<i>CS₄</i>	1	0.38	0.38	1

Tabla 12: Matriz de comparación de *Cr₂*

	<i>CS₁</i>	<i>CS₂</i>	<i>CS₃</i>	<i>CS₄</i>	<i>V_{Cr₂}</i>
<i>CS₁</i>	0.14	0.14	0.14	0.14	0.14
<i>CS₂</i>	0.36	0.36	0.36	0.36	0.36
<i>CS₃</i>	0.36	0.36	0.36	0.36	0.36
<i>CS₄</i>	0.14	0.14	0.14	0.14	0.14

Tabla 13: Matriz normalizada de *Cr₂* y *V_{Cr₂}*

	<i>CS₁</i>	<i>CS₂</i>	<i>CS₃</i>	<i>CS₄</i>
<i>CS₁</i>	1	2.67	4	4
<i>CS₂</i>	0.38	1	1.50	1.50
<i>CS₃</i>	0.25	0.67	1	1
<i>CS₄</i>	0.25	0.67	1	1

Tabla 14: Matriz de comparación de *Cr₃*

	<i>CS₁</i>	<i>CS₂</i>	<i>CS₃</i>	<i>CS₄</i>	<i>V_{Cr₃}</i>
<i>CS₁</i>	0.53	0.53	0.53	0.53	0.53
<i>CS₂</i>	0.20	0.20	0.20	0.20	0.20
<i>CS₃</i>	0.13	0.13	0.13	0.13	0.13
<i>CS₄</i>	0.13	0.13	0.13	0.13	0.13

Tabla 15: Matriz normalizada de *Cr₃* y *V_{Cr₃}*

A continuación, y siguiendo la metodología, se deben comparar los criterios de modo que nos permitan obtener un vector de prioridad. Lo anterior es resultado de la diferencia en la importancia que se le asigna a cada criterio.

	CS_1	CS_2	CS_3	CS_4
CS_1	1	1.17	1.40	3.50
CS_2	0.86	1	1.20	3.00
CS_3	0.71	0.83	1	2.50
CS_4	0.29	0.33	0.40	1

Tabla 16: Matriz de comparación de Cr_4

	CS_1	CS_2	CS_3	CS_4	V_{Cr_4}
CS_1	0.35	0.35	0.35	0.35	0.35
CS_2	0.30	0.30	0.30	0.30	0.30
CS_3	0.25	0.25	0.25	0.25	0.25
CS_4	0.10	0.10	0.10	0.10	0.10

Tabla 17: Matriz normalizada de Cr_4 y V_{Cr_4}

	CS_1	CS_2	CS_3	CS_4
CS_1	1	0.89	1.14	4
CS_2	1.13	1	1.29	4.50
CS_3	0.88	0.78	1	3.50
CS_4	0.25	0.22	0.29	1

Tabla 18: Matriz de comparación de Cr_5

	CS_1	CS_2	CS_3	CS_4	V_{Cr_5}
CS_1	0.31	0.31	0.31	0.31	0.31
CS_2	0.35	0.35	0.35	0.35	0.35
CS_3	0.27	0.27	0.27	0.27	0.27
CS_4	0.08	0.08	0.08	0.08	0.08

Tabla 19: Matriz normalizada de Cr_5 y V_{Cr_5}

	CS_1	CS_2	CS_3	CS_4
CS_1	1	0.90	0.90	2.25
CS_2	1.11	1	1	2.50
CS_3	1.11	1	1	2.50
CS_4	0.44	0.40	0.40	1

Tabla 20: Matriz de comparación de Cr_6

	CS_1	CS_2	CS_3	CS_4	V_{Cr_6}
CS_1	0.27	0.27	0.27	0.27	0.27
CS_2	0.30	0.30	0.30	0.30	0.30
CS_3	0.30	0.30	0.30	0.30	0.30
CS_4	0.12	0.12	0.12	0.12	0.12

Tabla 21: Matriz normalizada de Cr_6 y V_{Cr_6}

	CS_1	CS_2	CS_3	CS_4
CS_1	1	5	3.33	1.43
CS_2	0.20	1	0.67	0.29
CS_3	0.30	1.50	1	0.43
CS_4	0.70	3.50	2.33	1

Tabla 22: Matriz de comparación de Cr_7

	CS_1	CS_2	CS_3	CS_4	V_{Cr_7}
CS_1	0.45	0.45	0.45	0.45	0.45
CS_2	0.09	0.09	0.09	0.09	0.09
CS_3	0.14	0.14	0.14	0.14	0.14
CS_4	0.32	0.32	0.32	0.32	0.32

Tabla 23: Matriz normalizada de Cr_7 y V_{Cr_7}

	CS_1	CS_2	CS_3	CS_4
CS_1	1	3	2.25	4.50
CS_2	0.33	1	0.75	1.50
CS_3	0.44	1.33	1	2
CS_4	0.22	0.67	0.50	1

Tabla 24: Matriz de comparación de Cr_8

	CS_1	CS_2	CS_3	CS_4	V_{Cr_8}
CS_1	0.50	0.50	0.50	0.50	0.50
CS_2	0.17	0.17	0.17	0.17	0.17
CS_3	0.22	0.22	0.22	0.22	0.22
CS_4	0.11	0.11	0.11	0.11	0.11

Tabla 25: Matriz normalizada de Cr_8 y V_{Cr_8}

	CS_1	CS_2	CS_3	CS_4
CS_1	1	1	0.88	0.70
CS_2	1	1	0.88	0.70
CS_3	1.14	1.14	1	0.80
CS_4	1.43	1.43	1.25	1

Tabla 26: Matriz de comparación de Cr_9

	CS_1	CS_2	CS_3	CS_4	V_{Cr_9}
CS_1	0.22	0.22	0.22	0.22	0.22
CS_2	0.22	0.22	0.22	0.22	0.22
CS_3	0.25	0.25	0.25	0.25	0.25
CS_4	0.31	0.31	0.31	0.31	0.31

Tabla 27: Matriz normalizada de Cr_9 y V_{Cr_9}

	CS_1	CS_2	CS_3	CS_4
CS_1	1	9	4.50	1.29
CS_2	0.11	1	0.50	0.14
CS_3	0.22	2	1	0.29
CS_4	0.78	7	3.50	1

Tabla 28: Matriz de comparación de Cr_{10}

	CS_1	CS_2	CS_3	CS_4	$V_{Cr_{10}}$
CS_1	0.47	0.47	0.47	0.47	0.47
CS_2	0.05	0.05	0.05	0.05	0.05
CS_3	0.11	0.11	0.11	0.11	0.11
CS_4	0.37	0.37	0.37	0.37	0.37

Tabla 29: Matriz normalizada de Cr_{10} y $V_{Cr_{10}}$

	CS_1	CS_2	CS_3	CS_4
CS_1	1	1.60	2	0.89
CS_2	0.63	1	1.25	0.56
CS_3	0.50	0.80	1	0.44
CS_4	1.13	1.80	2.25	1

Tabla 30: Matriz de comparación de Cr_{11}

	CS_1	CS_2	CS_3	CS_4	$V_{Cr_{11}}$
CS_1	0.31	0.31	0.31	0.31	0.31
CS_2	0.19	0.19	0.19	0.19	0.19
CS_3	0.15	0.15	0.15	0.15	0.15
CS_4	0.35	0.35	0.35	0.35	0.35

Tabla 31: Matriz normalizada de Cr_{11} y $V_{Cr_{11}}$

	CS_1	CS_2	CS_3	CS_4
CS_1	1	5	5	2.50
CS_2	0.20	1	1	0.50
CS_3	0.20	1	1	0.50
CS_4	0.40	2	2	1

Tabla 32: Matriz de comparación de Cr_{12}

	CS_1	CS_2	CS_3	CS_4	$V_{Cr_{12}}$
CS_1	0.56	0.56	0.56	0.56	0.56
CS_2	0.11	0.11	0.11	0.11	0.11
CS_3	0.11	0.11	0.11	0.11	0.11
CS_4	0.22	0.22	0.22	0.22	0.22

Tabla 33: Matriz normalizada de Cr_{12} y $V_{Cr_{12}}$

	CS_1	CS_2	CS_3	CS_4
CS_1	1	2	3	2
CS_2	0.50	1	1.50	1
CS_3	0.33	0.67	1	0.67
CS_4	0.50	1	1.50	1

Tabla 34: Matriz de comparación de Cr_{13}

	CS_1	CS_2	CS_3	CS_4	$V_{Cr_{13}}$
CS_1	0.43	0.43	0.43	0.43	0.43
CS_2	0.21	0.21	0.21	0.21	0.21
CS_3	0.14	0.14	0.14	0.14	0.14
CS_4	0.21	0.21	0.21	0.21	0.21

Tabla 35: Matriz normalizada de Cr_{13} y $V_{Cr_{13}}$

	CS_1	CS_2	CS_3	CS_4
CS_1	1	10	2	1.25
CS_2	0.10	1	0.20	0.13
CS_3	0.50	5	1	0.63
CS_4	0.80	8	1.60	1

Tabla 36: Matriz de comparación de Cr_{14}

	CS_1	CS_2	CS_3	CS_4	$V_{Cr_{14}}$
CS_1	0.42	0.42	0.42	0.42	0.42
CS_2	0.04	0.04	0.04	0.04	0.04
CS_3	0.21	0.21	0.21	0.21	0.21
CS_4	0.33	0.33	0.33	0.33	0.33

Tabla 37: Matriz normalizada de Cr_{14} y $V_{Cr_{14}}$

	CS_1	CS_2	CS_3	CS_4
CS_1	1	2	2	1
CS_2	0.50	1	1	0.50
CS_3	0.50	1	1	0.50
CS_4	1	2	2	1

Tabla 38: Matriz de comparación de Cr_{21}

	CS_1	CS_2	CS_3	CS_4	$V_{Cr_{21}}$
CS_1	0.33	0.33	0.33	0.33	0.33
CS_2	0.17	0.17	0.17	0.17	0.17
CS_3	0.17	0.17	0.17	0.17	0.17
CS_4	0.33	0.33	0.33	0.33	0.33

Tabla 39: Matriz normalizada de Cr_{21} y $V_{Cr_{21}}$

	CS_1	CS_2	CS_3	CS_4
CS_1	1	7	7	1
CS_2	0.14	1	1	0.14
CS_3	0.14	1	1	0.14
CS_4	1	7	7	1

Tabla 40: Matriz de comparación de Cr_{22}

	CS_1	CS_2	CS_3	CS_4	$V_{Cr_{22}}$
CS_1	0.44	0.44	0.44	0.44	0.44
CS_2	0.06	0.06	0.06	0.06	0.06
CS_3	0.06	0.06	0.06	0.06	0.06
CS_4	0.44	0.44	0.44	0.44	0.44

Tabla 41: Matriz normalizada de Cr_{22} y $V_{Cr_{22}}$

	CS_1	CS_2	CS_3	CS_4
CS_1	1	10	5	1.25
CS_2	0.10	1	0.50	0.13
CS_3	0.20	2	1	0.25
CS_4	0.80	8	4	1

Tabla 42: Matriz de comparación de Cr_{24}

	CS_1	CS_2	CS_3	CS_4	$V_{Cr_{24}}$
CS_1	0.48	0.48	0.48	0.48	0.48
CS_2	0.05	0.05	0.05	0.05	0.05
CS_3	0.10	0.10	0.10	0.10	0.10
CS_4	0.38	0.38	0.38	0.38	0.38

Tabla 43: Matriz normalizada de Cr_{24} y $V_{Cr_{24}}$

	CS_1	CS_2	CS_3	CS_4
CS_1	1	4	2.67	1.33
CS_2	0.25	1	0.67	0.33
CS_3	0.38	1.50	1	0.50
CS_4	0.75	3	2	1

Tabla 44: Matriz de comparación de Cr_{25}

	CS_1	CS_2	CS_3	CS_4	$V_{Cr_{25}}$
CS_1	0.42	0.42	0.42	0.42	0.42
CS_2	0.11	0.11	0.11	0.11	0.11
CS_3	0.16	0.16	0.16	0.16	0.16
CS_4	0.32	0.32	0.32	0.32	0.32

Tabla 45: Matriz normalizada de Cr_{25} y $V_{Cr_{25}}$

	CS_1	CS_2	CS_3	CS_4
CS_1	1	0.57	0.57	1
CS_2	1.75	1	1	1.75
CS_3	1.75	1	1	1.75
CS_4	1	0.57	0.57	1

Tabla 46: Matriz de comparación de Cr_{27}

	CS_1	CS_2	CS_3	CS_4	$V_{Cr_{27}}$
CS_1	0.18	0.18	0.18	0.18	0.18
CS_2	0.32	0.32	0.32	0.32	0.32
CS_3	0.32	0.32	0.32	0.32	0.32
CS_4	0.18	0.18	0.18	0.18	0.18

Tabla 47: Matriz normalizada de Cr_{27} y $V_{Cr_{27}}$

	Cr_1	Cr_2	Cr_3	Cr_4	Cr_5	Cr_6	Cr_7	Cr_8	Cr_9	Cr_{10}	Cr_{11}	Cr_{12}	Cr_{13}	Cr_{14}	Cr_{21}	Cr_{22}	Cr_{24}	Cr_{25}	Cr_{27}
Cr_1	1	5	0.75	3	3	1	3	5	10	5	3	2	5	2	8	2	1	5	4
Cr_2	0.20	1	6.67	1.67	1.67	5	1.67	1	0.50	1	1.67	2.50	1	2.50	0.63	2.50	5	1	1.25
Cr_3	1.33	0.15	1	0.25	0.25	0.75	0.25	0.15	0.08	0.15	0.25	0.38	0.15	0.38	0.09	0.38	0.75	0.15	0.19
Cr_4	0.33	0.60	4	1	1	3	1	0.60	0.30	0.60	1	1.50	0.60	1.50	0.38	1.50	3	0.60	0.75
Cr_5	0.33	0.60	4	1	1	3	1	0.60	0.30	0.60	1	1.50	0.60	1.50	0.38	1.50	3	0.60	0.75
Cr_6	1	0.20	1.33	0.33	0.33	1	0.33	0.20	0.10	0.20	0.33	0.50	0.20	0.50	0.13	0.50	1	0.20	0.25
Cr_7	0.33	0.60	4	1	1	3	1	0.60	0.30	0.60	1	1.50	0.60	1.50	0.38	1.50	3	0.60	0.75
Cr_8	0.20	1	6.67	1.67	1.67	5	1.67	1	0.50	1	1.67	2.50	1	2.50	0.63	2.50	5	1	1.25
Cr_9	0.10	2	13.33	3.33	3.33	10	3.33	2	1	2	3.33	5	2	5	1.25	5	10	2	2.50
Cr_{10}	0.20	1	6.67	1.67	1.67	5	1.67	1	0.50	1	1.67	2.50	1	2.50	0.63	2.50	5	1	1.25
Cr_{11}	0.33	0.60	4	1	1	3	1	0.60	0.30	0.60	1	1.50	0.60	1.50	0.38	1.50	3	0.60	0.75
Cr_{12}	0.50	0.40	2.67	0.67	0.67	2	0.67	0.40	0.20	0.40	0.67	1	0.40	1	0.25	1	2	0.40	0.50
Cr_{13}	0.20	1	6.67	1.67	1.67	5	1.67	1	0.50	1	1.67	2.50	1	2.50	0.63	2.50	5	1	1.25
Cr_{14}	0.50	0.40	2.67	0.67	0.67	2	0.67	0.40	0.20	0.40	0.67	1	0.40	1	0.25	1	2	0.40	0.50
Cr_{21}	0.13	1.60	10.67	2.67	2.67	8	2.67	1.60	0.80	1.60	2.67	4	1.60	4	1	4	8	1.60	2
Cr_{22}	0.50	0.40	2.67	0.67	0.67	2	0.67	0.40	0.20	0.40	0.67	1	0.40	1	0.25	1	2	0.40	0.50
Cr_{24}	1	0.20	1.33	0.33	0.33	1	0.33	0.20	0.10	0.20	0.33	0.50	0.20	0.50	0.13	0.50	1	0.20	0.25
Cr_{25}	0.20	1	6.67	1.67	1.67	5	1.67	1	0.50	1	1.67	2.50	1	2.50	0.63	2.50	5	1	1.25
Cr_{27}	0.25	0.80	5.33	1.33	1.33	4	1.33	0.80	0.40	0.80	1.33	2	0.80	2	0.50	2	4	0.80	1

Tabla 48: Matriz de comparación de criterios

	Cr_1	Cr_2	Cr_3	Cr_4	Cr_5	Cr_6	Cr_7	Cr_8	Cr_9	Cr_{10}	Cr_{11}	Cr_{12}	Cr_{13}	Cr_{14}	Cr_{21}	Cr_{22}	Cr_{24}	Cr_{25}	Cr_{27}	V_{Cr}	
Cr_1	0.116	0.270	0.008	0.117	0.117	0.015	0.117	0.270	0.596	0.270	0.117	0.056	0.270	0.056	0.486	0.056	0.015	0.270	0.191	0.133	
Cr_2	0.023	0.054	0.073	0.065	0.065	0.073	0.065	0.054	0.030	0.054	0.065	0.065	0.070	0.054	0.070	0.038	0.070	0.073	0.054	0.060	0.064
Cr_3	0.154	0.008	0.011	0.010	0.010	0.011	0.010	0.008	0.004	0.008	0.010	0.010	0.008	0.010	0.006	0.010	0.011	0.008	0.009	0.010	
Cr_4	0.039	0.032	0.044	0.039	0.039	0.044	0.039	0.032	0.018	0.032	0.039	0.042	0.032	0.042	0.023	0.042	0.044	0.032	0.036	0.038	
Cr_5	0.039	0.032	0.044	0.039	0.039	0.044	0.039	0.032	0.018	0.032	0.039	0.042	0.032	0.042	0.023	0.042	0.044	0.032	0.036	0.038	
Cr_6	0.116	0.011	0.015	0.013	0.013	0.015	0.013	0.011	0.006	0.011	0.013	0.014	0.011	0.014	0.008	0.014	0.015	0.011	0.012	0.013	
Cr_7	0.039	0.032	0.044	0.039	0.039	0.044	0.039	0.032	0.018	0.032	0.039	0.042	0.032	0.042	0.023	0.042	0.044	0.032	0.036	0.038	
Cr_8	0.023	0.054	0.073	0.065	0.065	0.073	0.065	0.054	0.030	0.054	0.065	0.070	0.054	0.070	0.038	0.070	0.073	0.054	0.060	0.064	
Cr_9	0.012	0.108	0.146	0.130	0.130	0.145	0.130	0.108	0.060	0.108	0.130	0.139	0.108	0.139	0.076	0.139	0.145	0.108	0.119	0.128	
Cr_{10}	0.023	0.054	0.073	0.065	0.065	0.073	0.065	0.054	0.030	0.054	0.065	0.070	0.054	0.070	0.038	0.070	0.073	0.054	0.060	0.064	
Cr_{11}	0.039	0.032	0.044	0.039	0.039	0.044	0.039	0.032	0.018	0.032	0.039	0.042	0.032	0.042	0.023	0.042	0.044	0.032	0.036	0.038	
Cr_{12}	0.058	0.022	0.029	0.026	0.026	0.029	0.026	0.022	0.012	0.022	0.026	0.028	0.022	0.028	0.015	0.028	0.029	0.022	0.024	0.026	
Cr_{13}	0.023	0.054	0.073	0.065	0.065	0.073	0.065	0.054	0.030	0.054	0.065	0.070	0.054	0.070	0.038	0.070	0.073	0.054	0.060	0.064	
Cr_{14}	0.058	0.022	0.029	0.026	0.026	0.029	0.026	0.022	0.012	0.022	0.026	0.028	0.022	0.028	0.015	0.028	0.029	0.022	0.024	0.026	
Cr_{21}	0.014	0.086	0.117	0.104	0.104	0.116	0.104	0.086	0.048	0.086	0.104	0.111	0.086	0.111	0.061	0.111	0.116	0.086	0.096	0.102	
Cr_{22}	0.058	0.022	0.029	0.026	0.026	0.029	0.026	0.022	0.012	0.022	0.026	0.028	0.022	0.028	0.015	0.028	0.029	0.022	0.024	0.026	
Cr_{24}	0.116	0.011	0.015	0.013	0.013	0.015	0.013	0.011	0.006	0.011	0.013	0.014	0.011	0.014	0.008	0.014	0.015	0.011	0.012	0.013	
Cr_{25}	0.023	0.054	0.073	0.065	0.065	0.073	0.065	0.054	0.030	0.054	0.065	0.070	0.054	0.070	0.038	0.070	0.073	0.054	0.060	0.064	
Cr_{27}	0.029	0.043	0.059	0.052	0.052	0.058	0.052	0.043	0.024	0.043	0.052	0.056	0.043	0.056	0.030	0.056	0.058	0.043	0.048	0.051	

Tabla 49: Matriz normalizada y vector de prioridad para la comparación de criterios.

Ahora, se analizan las alternativas independientes para la decisión usando la misma metodología.

	Alt_1	Alt_2	Alt_3	Alt_4
Alt_1	1	1	2.67	1.33
Alt_2	1	1	2.67	1.33
Alt_3	0.38	0.38	1	0.50
Alt_4	0.75	0.75	2	1

Tabla 50: Matriz de comparación de C_3

	Alt_1	Alt_2	Alt_3	Alt_4	V_{C_3}
Alt_1	0.32	0.32	0.32	0.32	0.32
Alt_2	0.32	0.32	0.32	0.32	0.32
Alt_3	0.12	0.12	0.12	0.12	0.12
Alt_4	0.24	0.24	0.24	0.24	0.24

Tabla 51: Matriz normalizada de C_3 y V_{C_3}

	Alt_1	Alt_2	Alt_3	Alt_4
Alt_1	1	0.50	1.67	-
Alt_2	2	1	3.33	-
Alt_3	0.60	0.30	1	-
Alt_4	-	-	-	-

Tabla 52: Matriz de comparación de C_4

	Alt_1	Alt_2	Alt_3	Alt_4	V_{C_4}
Alt_1	0.28	0.28	0.28	-	0.28
Alt_2	0.56	0.56	0.56	-	0.56
Alt_3	0.17	0.17	0.17	-	0.17
Alt_4	-	-	-	-	-

Tabla 53: Matriz normalizada de C_4 y V_{C_4}

	Alt_1	Alt_2	Alt_3	Alt_4
Alt_1	1	0.80	2	-
Alt_2	1.25	1	2.50	-
Alt_3	0.50	0.40	1	-
Alt_4	-	-	-	-

Tabla 54: Matriz de comparación de C_5

	Alt_1	Alt_2	Alt_3	Alt_4	V_{C_5}
Alt_1	0.36	0.36	0.36	-	0.36
Alt_2	0.45	0.45	0.45	-	0.45
Alt_3	0.18	0.18	0.18	-	0.18
Alt_4	-	-	-	-	-

Tabla 55: Matriz normalizada de C_5 y V_{C_5}

	Alt_1	Alt_2	Alt_3	Alt_4
Alt_1	1	0.89	1.33	-
Alt_2	1.13	1	1.50	-
Alt_3	0.75	0.67	1	-
Alt_4	-	-	-	-

Tabla 56: Matriz de comparación de C_6

	Alt_1	Alt_2	Alt_3	Alt_4	V_{C_6}
Alt_1	0.35	0.35	0.35	-	0.35
Alt_2	0.39	0.39	0.39	-	0.39
Alt_3	0.26	0.26	0.26	-	0.26
Alt_4	-	-	-	-	-

Tabla 57: Matriz normalizada de C_6 y V_{C_6}

	Alt_1	Alt_2	Alt_3	Alt_4
Alt_1	1	0.89	1.33	-
Alt_2	1.13	1	1.50	-
Alt_3	0.75	0.67	1	-
Alt_4	-	-	-	-

Tabla 58: Matriz de comparación de C_7

	Alt_1	Alt_2	Alt_3	Alt_4	V_{C_7}
Alt_1	0.35	0.35	0.35	-	0.35
Alt_2	0.39	0.39	0.39	-	0.39
Alt_3	0.26	0.26	0.26	-	0.26
Alt_4	-	-	-	-	-

Tabla 59: Matriz normalizada de C_7 y V_{C_7}

	Alt_1	Alt_2	Alt_3	Alt_4
Alt_1	1	0.89	1.33	-
Alt_2	1.13	1	1.50	-
Alt_3	0.75	0.67	1	-
Alt_4	-	-	-	-

Tabla 60: Matriz de comparación de C_8

	Alt_1	Alt_2	Alt_3	Alt_4	V_{C_8}
Alt_1	0.35	0.35	0.35	-	0.35
Alt_2	0.39	0.39	0.39	-	0.39
Alt_3	0.26	0.26	0.26	-	0.26
Alt_4	-	-	-	-	-

Tabla 61: Matriz normalizada de C_8 y V_{C_8}

	Alt_1	Alt_2	Alt_3	Alt_4
Alt_1	1	3	4.50	-
Alt_2	0.33	1	1.50	-
Alt_3	0.22	0.67	1	-
Alt_4	-	-	-	-

Tabla 62: Matriz de comparación de C_9

	Alt_1	Alt_2	Alt_3	Alt_4	V_{C_9}
Alt_1	0.64	0.64	0.64	-	0.64
Alt_2	0.21	0.21	0.21	-	0.21
Alt_3	0.14	0.14	0.14	-	0.14
Alt_4	-	-	-	-	-

Tabla 63: Matriz normalizada de C_9 y V_{C_9}

	Alt_1	Alt_2	Alt_3	Alt_4
Alt_1	1	1.25	1.43	-
Alt_2	0.80	1	1.14	-
Alt_3	0.70	0.88	1	-
Alt_4	-	-	-	-

Tabla 64: Matriz de comparación de C_{10}

	Alt_1	Alt_2	Alt_3	Alt_4	$V_{C_{10}}$
Alt_1	0.40	0.40	0.40	-	0.40
Alt_2	0.32	0.32	0.32	-	0.32
Alt_3	0.28	0.28	0.28	-	0.28
Alt_4	-	-	-	-	-

Tabla 65: Matriz normalizada de C_{10} y $V_{C_{10}}$

	Alt_1	Alt_2	Alt_3	Alt_4
Alt_1	1	1.14	0.80	-
Alt_2	0.88	1	0.70	-
Alt_3	1.25	1.43	1	-
Alt_4	-	-	-	-

Tabla 66: Matriz de comparación de C_{11}

	Alt_1	Alt_2	Alt_3	Alt_4	$V_{C_{11}}$
Alt_1	0.32	0.32	0.32	-	0.32
Alt_2	0.28	0.28	0.28	-	0.28
Alt_3	0.40	0.40	0.40	-	0.40
Alt_4	-	-	-	-	-

Tabla 67: Matriz normalizada de C_{11} y $V_{C_{11}}$

	Alt_1	Alt_2	Alt_3	Alt_4
Alt_1	1	5	-	-
Alt_2	0.20	1	-	-
Alt_3	-	-	-	-
Alt_4	-	-	-	-

Tabla 68: Matriz de comparación de C_{12}

	Alt_1	Alt_2	Alt_3	Alt_4	$V_{C_{12}}$
Alt_1	0.83	0.83	-	-	0.83
Alt_2	0.17	0.17	-	-	0.17
Alt_3	-	-	-	-	-
Alt_4	-	-	-	-	-

Tabla 69: Matriz normalizada de C_{12} y $V_{C_{12}}$

	Alt_1	Alt_2	Alt_3	Alt_4
Alt_1	1	1.60	-	-
Alt_2	0.63	1	-	-
Alt_3	-	-	-	-
Alt_4	-	-	-	-

Tabla 70: Matriz de comparación de C_{13}

	Alt_1	Alt_2	Alt_3	Alt_4	$V_{C_{13}}$
Alt_1	0.62	0.62	-	-	0.62
Alt_2	0.38	0.38	-	-	0.38
Alt_3	-	-	-	-	-
Alt_4	-	-	-	-	-

Tabla 71: Matriz normalizada de C_{13} y $V_{C_{13}}$

	Alt_1	Alt_2	Alt_3	Alt_4
Alt_1	1	1.50	-	-
Alt_2	0.67	1	-	-
Alt_3	-	-	-	-
Alt_4	-	-	-	-

Tabla 72: Matriz de comparación de C_{14}

	Alt_1	Alt_2	Alt_3	Alt_4	$V_{C_{14}}$
Alt_1	0.60	0.60	-	-	0.60
Alt_2	0.40	0.40	-	-	0.40
Alt_3	-	-	-	-	-
Alt_4	-	-	-	-	-

Tabla 73: Matriz normalizada de C_{14} y $V_{C_{14}}$

	Alt_1	Alt_2	Alt_3	Alt_4
Alt_1	1	1.50	2.25	4.50
Alt_2	0.67	1	1.50	3
Alt_3	0.44	0.67	1	2
Alt_4	0.22	0.33	0.50	1

Tabla 74: Matriz de comparación de C_{15}

	Alt_1	Alt_2	Alt_3	Alt_4	$V_{C_{15}}$
Alt_1	0.43	0.43	0.43	0.43	0.43
Alt_2	0.29	0.29	0.29	0.29	0.29
Alt_3	0.19	0.19	0.19	0.19	0.19
Alt_4	0.10	0.10	0.10	0.10	0.10

Tabla 75: Matriz normalizada de C_{15} y $V_{C_{15}}$

	Alt_1	Alt_2	Alt_3	Alt_4
Alt_1	1	0.78	0.88	-
Alt_2	1.29	1	1.13	-
Alt_3	1.14	0.89	1	-
Alt_4	-	-	-	-

Tabla 76: Matriz de comparación de C_{16}

	Alt_1	Alt_2	Alt_3	Alt_4	$V_{C_{16}}$
Alt_1	0.29	0.29	0.29	-	0.29
Alt_2	0.38	0.38	0.38	-	0.38
Alt_3	0.33	0.33	0.33	-	0.33
Alt_4	-	-	-	-	-

Tabla 77: Matriz normalizada de C_{16} y $V_{C_{16}}$

Una vez que se cuenta con los valores del cumplimiento de cada alternativa de solución con respecto a cada criterio, se utiliza en conjunto con el vector de prioridad para obtener la puntuación de cada alternativa de solución. La alternativa elegida sera la de mayor valor.

	CS_1	CS_2	CS_3	CS_4	V_{Cr}
V_{Cr_1}	0.56	0.06	0.11	0.28	0.133
V_{Cr_2}	0.14	0.36	0.36	0.14	0.064
V_{Cr_3}	0.53	0.20	0.13	0.13	0.010
V_{Cr_4}	0.35	0.30	0.25	0.10	0.038
V_{Cr_5}	0.31	0.35	0.27	0.08	0.038
V_{Cr_6}	0.27	0.30	0.30	0.12	0.013
V_{Cr_7}	0.45	0.09	0.14	0.32	0.038
V_{Cr_8}	0.50	0.17	0.22	0.11	0.064
V_{Cr_9}	0.22	0.22	0.25	0.31	0.128
$V_{Cr_{10}}$	0.47	0.05	0.11	0.37	0.064
$V_{Cr_{11}}$	0.31	0.19	0.15	0.35	0.038
$V_{Cr_{12}}$	0.56	0.11	0.11	0.22	0.026
$V_{Cr_{13}}$	0.43	0.21	0.14	0.21	0.064
$V_{Cr_{14}}$	0.42	0.04	0.21	0.33	0.026
$V_{Cr_{21}}$	0.33	0.17	0.17	0.33	0.102
$V_{Cr_{22}}$	0.44	0.06	0.06	0.44	0.026
$V_{Cr_{24}}$	0.48	0.05	0.10	0.38	0.013
$V_{Cr_{25}}$	0.42	0.11	0.16	0.32	0.064
$V_{Cr_{27}}$	0.18	0.32	0.32	0.18	0.051
Total	0.376	0.174	0.191	0.259	1.000

Tabla 78: Matriz de decisión

Por ultimo, la siguiente tabla hace una recopilación de la alternativa de solución elegida en conjunto con las alternativas independientes que fueron elegidas.

Características	Concepto solución
Forma de movimiento de los paneles	Rotacional en base prismática
Tipo de actuación	Con motor y acopladores magnéticos
Método para la reflexión	Paneles reflectivos
Sensado de las posiciones de los paneles	Encoders, sensores de contacto o visión artificial
Posicionamiento de paneles	Anclados a la base prismática
Sistema de alimentación de paneles	No
Grabación en varias posiciones	No

Generación de ondas de sonido	Bocina externa
Dispositivo de grabación	Micrófono externo
Plataforma para el cálculo de las posiciones deseadas	Microcontrolador embebido
Plataforma de procesado del control	Microcontrolador embebido
Plataforma de procesado de la acústica	Microcontrolador embebido
Sensado de las posiciones de los paneles	Encoders
Plataforma de visualización de información	Equipo de cómputo
Control de interfaz	Ratón y teclado
Encapsulado de sistemas embebidos	Si
Inclusión del sistema energético en sistema embebido	Si
Anclaje al suelo	Si
Método de anclaje	Tornillos
Método de comunicación	Wi-fi

Tabla 79: Características del concepto solución elegido

Se desarrolló el diseño del concepto solución elegido en un software CAD, es a partir de esta propuesta inicial que se comenzaran a desarrollar los cálculos y validaciones para el diseño final.

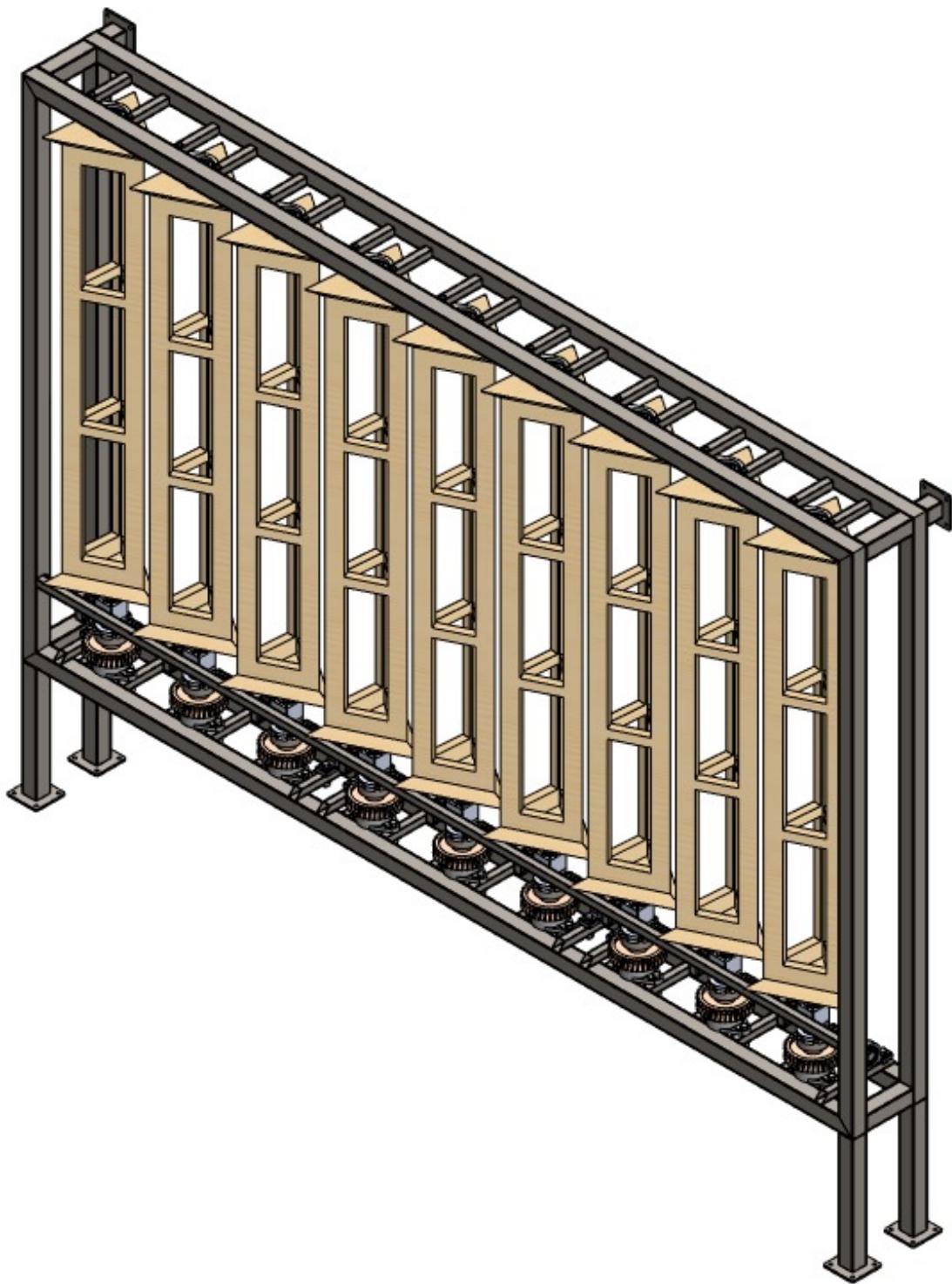


Figura 12: Concepto solución elegido

2.2. Diseño detallado

2.2.1. MF1. Módulo de procesamiento

2.2.1.1 MF1.1. Procesamiento de la respuesta y cálculo de la acústica

Análisis de la respuesta al impulso A partir de la respuesta al impulso de un recinto, se pueden derivar distintos parámetros acústicos, como los establecidos en la norma ISO3381 [20]. Siendo que la respuesta al impulso la podemos obtener de mediciones físicas, de simulaciones del recinto, e incluso de bancos de datos con respuestas de múltiples recintos para fines de validación, fue necesario tener una metodología general que nos permitiera analizar datos de diferentes fuentes.

El programa de generación y medición de la acústica, tiene como salida una estructura con los datos de la respuesta al impulso y el tiempo de muestreo; para las simulaciones y los bancos de datos, las respuestas al impulso se guardan como un archivo de tipo *wav*. Se iniciará con una línea de lectura de archivos de este tipo, que se desactivara en caso de usarse consecutivo al programa de generación y medición de la acústica. Dicho código también se encargará de combinar las señales en caso de leer archivos *wav* con múltiples canales.

```

1 [ temp_y , ImpulseResponse . fs ] = audioread ([ 'C:\Users\User\Desktop\RIR_Database\' ...
2 ...
3 if size(temp_y ,2) > 1
4     ImpulseResponse . y = mean(temp_y ,2);
5 end

```

De acuerdo con la norma ISO-3382 [20], los parámetros acústicos que caracterizan un cuarto son:

- EDT , tiempo de decaimiento temprano
- T_{20} tiempo de reverberación referido a -20 Db
- T_{30} tiempo de reverberación referido a -30 Db
- C_{50} claridad a 50 ms
- C_{80} claridad a 80 ms
- D_{50} ratio de energía útil
- G fuerza del sonido

Todos los parámetros anteriores, deben calcularse en las frecuencias centrales de las bandas de frecuencia pertenecientes octavas de 125 Hz a 4000 Hz. Por esto, primero se deben pasar por filtros pasa-bandas, distribuidos a lo largo de las octavas [21].

Se pasó la respuesta al impulso por 6 filtros, con frecuencias centrales y rangos, como se muestra en la tabla 80:

Frecuencia central	Límite inferior	Límite superior
125 Hz	88.39 Hz	176.8 Hz
250 Hz	176.8 Hz	353.6 Hz
500 Hz	353.6 Hz	707.1 Hz
1000 Hz	707.1 Hz	1414 Hz
2000 Hz	1414 Hz	2828 Hz
4000 Hz	2828 Hz	5657 Hz

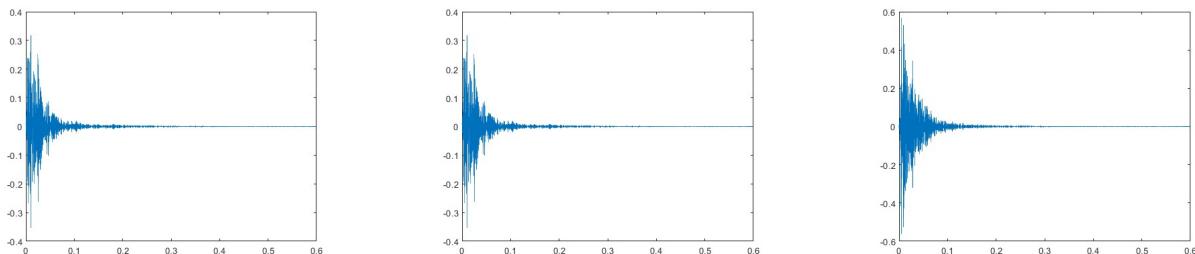
Tabla 80: Rangos de frecuencias para filtros

```

1 RIR = struct( 'General' , ImpulseResponse );
2 RIR.f125.y = bandpass( ImpulseResponse.y ,[ 88.39 176.8 ] , ImpulseResponse.fs );
3 RIR.f125.fs = ImpulseResponse.fs ;
4 RIR.f250.y = bandpass( ImpulseResponse.y ,[ 176.8 353.6 ] , ImpulseResponse.fs );
5 RIR.f250.fs = ImpulseResponse.fs ;
6 RIR.f500.y = bandpass( ImpulseResponse.y ,[ 353.6 707.1 ] , ImpulseResponse.fs );
7 RIR.f500.fs = ImpulseResponse.fs ;
8 RIR.f1k.y = bandpass( ImpulseResponse.y ,[ 707.1 1414 ] , ImpulseResponse.fs );
9 RIR.f1k.fs = ImpulseResponse.fs ;
10 RIR.f2k.y = bandpass( ImpulseResponse.y ,[ 1414 2828 ] , ImpulseResponse.fs );
11 RIR.f2k.fs = ImpulseResponse.fs ;
12 RIR.f4k.y = bandpass( ImpulseResponse.y ,[ 2828 5657 ] , ImpulseResponse.fs );
13 RIR.f4k.fs = ImpulseResponse.fs ;

```

Se pueden observar algunas la respuesta del recinto para ciertas frecuencias.



(a) Respuesta al impulso a 500 Hz (b) Respuesta al impulso a 1000 Hz (c) Respuesta al impulso a 2000 Hz

Figura 13: Respuesta al impulso para diferentes frecuencias

A continuación podemos hacer el análisis de la respuesta al impulso en sus diferentes frecuencias, además de la general, lo que nos dará los parámetros acústicos actuales del recinto. Este proceso se hace mediante otro *script* llamado *RIR_Analysis*.

La función toma como primera entrada el método de empaquetado de la señal, se debe ingresar un 1 en caso de que se quiera analizar un archivo de tipo *wav*, y cualquier otro número en caso de que la señal ya venga empaquetada en una estructura con el tiempo de

muestreo y la amplitud. A continuación se introduce la señal correspondiente, además de las dimensiones del cuarto en el que se grabó la respuesta al impulso y una variable booleana que indica al programa si debe mostrar la gráfica de decaimiento de la energía, junto con los ajustes lineales y los parámetros.

La función retorna la respuesta al impulso cuadrada, la gráfica de decaimiento, el vector de tiempo correspondiente y una estructura con los parámetros acústicos.

```
1 function [RIRsq,DC,t,AcousticParams] = RIR_Analisys(method, input,
room_dimensions, printFlag)
```

La gráfica del decaimiento se obtiene mediante la integral de Schroeder, la cual es una integración hacia atrás, que se hace desde un punto donde la señal ya solo contiene ruido de fondo [22].

```
1 RIRsq = y.^2;
2 schroeder_cumsum = cumsum(flipud(RIRsq));
3 schroeder_normalized = schroeder_cumsum / max(schroeder_cumsum);
4 DC = flipud(10*log10(schroeder_normalized));
```

La gráfica se muestra en decibelios y va de 0 a $-\infty$.

La curva de decaimiento de la energía nos permite ver el decaimiento exponencial del sonido como un decaimiento lineal, y así, poder ajustar rectas que se apeguen más al decaimiento de la señal sin verse interferida por el ruido de fondo. Adicionalmente, podemos usar esta curva para calcular los otros parámetros acústicos como D_{50} o C_{50} .

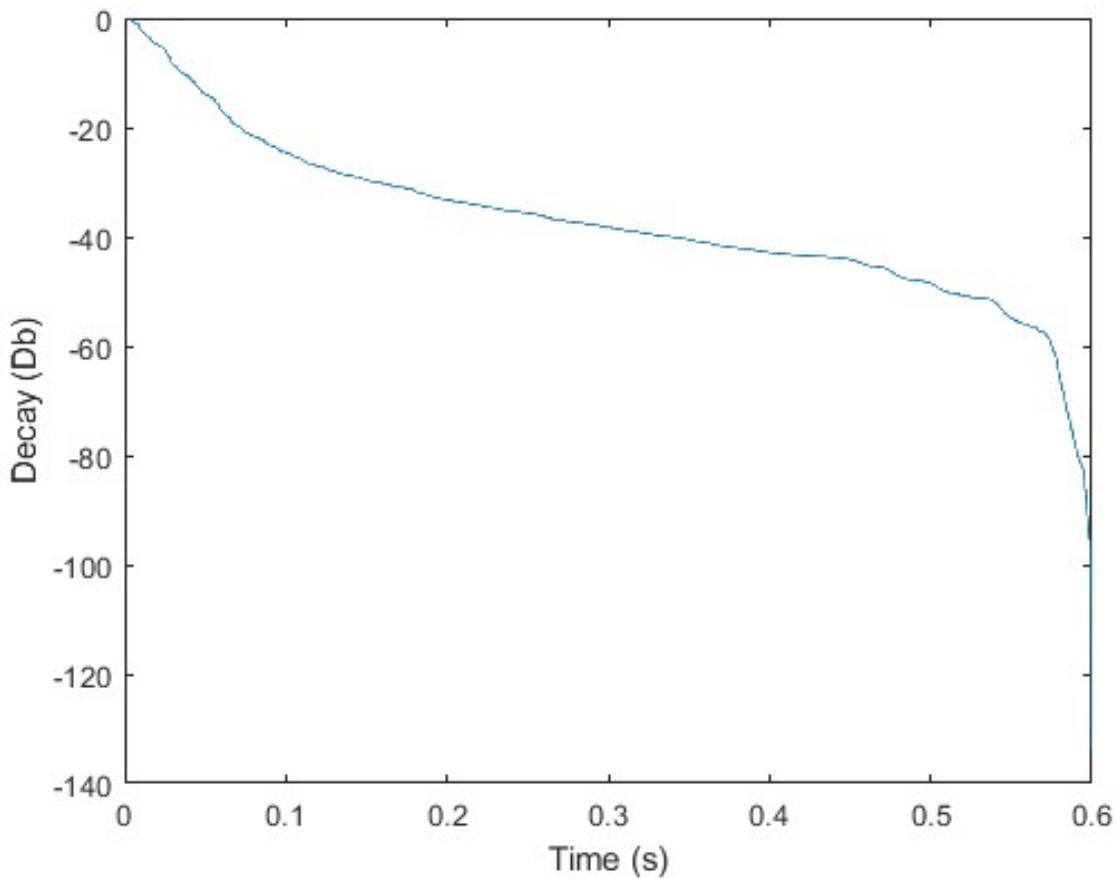


Figura 14: Curva de decaimiento

El comportamiento de la curva deja de ser lineal cuando la señal comienza a verse afectada por el ruido de fondo. Idealmente, el ruido de fondo debería presentarse como un valle horizontal, y la señal con un decaimiento lo más recto posible, hasta curvarse al final para convertirse en el valle del ruido de fondo[23].

El análisis de esta curva revela la importancia de tener una excitación lo bastante fuerte como para superar al ruido de fondo y ver el decaimiento de la curva por más tiempo antes de comenzar a perder información. Es por esto también que, a pesar de que el tiempo de reverberación está referido a -60 Db, no suele medirse el decaimiento de esta manera, sino que se recurre a medir un decaimiento de -20 Db o -30 Db y se hace una extrapolación hasta los -60 Db. La norma ISO-3382 recomienda comenzar a medir el decaimiento hasta los -5 Db, para evitar interferencias de armónicos y entonces disminuir 20 o 30 Db. Intentar observar el decaimiento hasta -65 Db, implica que para un ruido de fondo de unos 30 Db, requeriríamos excitar el cuarto con una señal de al menos 95 Db, lo cual escapa de las posibilidades dentro de un recinto relativamente pequeño.

Posteriormente, se ajustan rectas tomando en cuenta los intervalos de -20 Db, -30 Db y

del EDT.

```

1 [~, EDT_Idx1] = min(abs(LF_EDT+10));
2 [~, EDT_Idx2] = min(abs(LF_EDT));
3 T60delEDT = 6*(t(EDT_Idx1)-t(EDT_Idx2));
4
5 [~, EDT_Idx1] = min(abs(LF_T20+20));
6 [~, EDT_Idx2] = min(abs(LF_T20));
7 T60delT20 = 3*(t(EDT_Idx1)-t(EDT_Idx2));
8
9 [~, EDT_Idx1] = min(abs(LF_T30+30));
10 [~, EDT_Idx2] = min(abs(LF_T30));
11 T60delT30 = 2*(t(EDT_Idx1)-t(EDT_Idx2));

```

La relación de energía útil (D) se define como la relación entre la energía total y la energía de la señal con arribo temprano. Se utilizan D_{50} y D_{80} para definir este parámetro usando 50 y 80 ms respectivamente como el límite para el arribo temprano [20]. Cabe resaltar que este tiempo se toma a partir del momento que llega la señal directa (en línea recta de la fuente al receptor).

$$D(50, 80) = \frac{\int_0^{(50,80) \text{ ms}} IR^2(t) dt}{\int_0^{\infty} IR^2(t) dt} \quad (14)$$

```

1 [~, maxIdx] = max(RIRsq);
2 RIRsq_trunc = RIRsq(maxIdx:end);
3 Energy0_50 = trapz(RIRsq_trunc(1:round(0.05*fs)));
4 Energy0_80 = trapz(RIRsq_trunc(1:round(0.08*fs)));
5 Energy0_end = trapz(RIRsq_trunc(1:end));
6
7 D50 = Energy0_50/Energy0_end;
8 D80 = Energy0_80/Energy0_end;

```

Los índices de claridad C_{50} y C_{80} siguen la misma definición que el índice D , pero, en lugar de representar un ratio o porcentaje, estos se presentan en Db como escala logarítmica. El índice C_{50} se puede entender como la claridad del habla y C_{80} como la claridad de la música [17]. Ambos se pueden calcular a partir del índice D como:

$$C_{50/80} = 10 \log \left(\frac{D_{50/80}}{1 - D_{50/80}} \right) \quad (15)$$

```

1 C50 = 10*log(D50/(1-D50));
2 C80 = 10*log(D80/(1-D80));

```

Por último, la fuerza del sonido G se define como la relación entre el nivel de presión del sonido en el recinto y el nivel de presión del mismo sonido en un campo libre, es decir, sin reflexiones, únicamente el sonido directo [17].

La medición de este parámetro es un poco más complicada, ya que se requiere aislar de todas las reflexiones, al pico en la amplitud producido por el sonido directo. Por esto, se decidió hacer una estimación validada por [17], que relaciona la fuerza del sonido con el tiempo de reverberación en un recinto y sus dimensiones.

$$G = 10 \log_{10} \left(\frac{RT_{60}}{V} \right) \quad (16)$$

El último análisis que se hace a la respuesta al impulso es respecto a los modos de vibración. Un recinto con paredes ortogonales, presenta distancias entre paredes que es un múltiplo de la longitud de onda de distintas frecuencias. Lo anterior ocasiona que las ondas oscilen entre paredes como si estuvieran ancladas a las paredes [24]. Cuando el sonido es persistente o el tiempo de reverberación alto, un oyente puede encontrar que existen zonas del cuarto donde el sonido es muy fuerte y otras donde el sonido es muy bajo y se puede medir el cambio en el volumen si se desplaza por el recinto.

Los modos de vibración pueden ser axiales, tangenciales y oblicuos, dependiendo del número de superficies en las que rebota. Los modos de vibración axiales suelen ser los más significativos y por suerte, son los más fáciles de predecir. [24]

Podemos calcular los modos de vibración calculando las frecuencias en donde las dimensiones del cuarto son múltiplos de sus respectivas longitudes de onda. El programa cuenta con un *script* externo llamado **CalcRoomModes** que realiza esta operación. El programa retorna los primeros n modos axiales de vibración.

```

1 c = 343; % m/s
2 f_length = c/room_dim(1);
3 f_width = c/room_dim(2);
4 f_depth = c/room_dim(3);
5 modes_length = zeros(1,num_modes);
6 modes_width = zeros(1,num_modes);
7 modes_depth = zeros(1,num_modes);
8
9 for i = 1:num_modes
10    modes_length(i) = f_length*(0.5*i);
11    modes_width(i) = f_width*(0.5*i);
12    modes_depth(i) = f_depth*(0.5*i);
13 end

```

Adicionalmente, el programa se encarga de calcular la frecuencia de Schroeder, la cual nos indica la frecuencia a la que el sonido dentro de un recinto deja de estar dominado por los modos de vibración y pasa a comportarse como un campo difuso. Se calcula con la siguiente fórmula [25]:

$$f_s = 2000 \sqrt{\frac{RT_{60}}{V}} \quad (17)$$

El cálculo de este parámetro es importante, debido a que el resto de parámetros acústicos solo son válidos si el sonido presente en el recinto se comporta mayormente como un campo difuso. Como se ve en la fórmula, la variable libre de la que depende esta frecuencia es el tiempo de reverberación; esto nos ayudara a introducir cotas para el tiempo de reverberación que nos asegure no tener modos de vibración significativos durante la reproducción de música en el estudio. La ecuación de Sabine [26] relaciona el tiempo de reverberación con la absorción de las superficies del cuarto y su geometría.

$$RT_{60} = \frac{0,161V}{S\alpha} \quad (18)$$

Podemos utilizar esta ecuación para tener una relación entre el tiempo de reverberación del cuarto y la superficie de absorción. Debido a que conocemos las absorciones tanto de los paneles como de las paredes, eso deja a la superficie como parámetro libre. Entonces, el control acústico consistirá en encontrar la cantidad de absorción específica para alcanzar un tiempo de reverberación deseado, y con ella, obtener la cantidad de superficie de paneles acústicos que debemos añadir en el cuarto para alcanzar dicho tiempo de reverberación.

```

1 V = prod(room_dim);
2 Current_abs = 0.161*V/AcousticParams.General.T60delT20;
3 DesiredT60 = 0.27
4 Needed_abs = 0.161*V/DesiredT60;
5 Offset_abs = Needed_abs-Current_abs;
6 MaterialAbs.Abs_Panels = 0.8;
7 MaterialAbs.Wall = 0.02;
8 Surface = Offset_abs/(MaterialAbs.Abs_Panels-MaterialAbs.Wall)

```

RayTracingImpulseResponse

En este programa se simula la respuesta al impulso en una sala con el objetivo de modelar las propiedades reverberantes de un espacio sin tener que realizar mediciones acústicas. La metodología usada presente en Real-Time Auralization [27], consiste en tratar el sonido como rayos que viajan en un recinto, los cuales se emiten de manera aleatoria y uniforme, y chocan contra diferentes superficies, perdiendo energía en el proceso. Además, cuando los rayos chocan con una superficie, esta refleja los rayos de manera difusa, por lo que una parte de la energía termina en el receptor. La simulación consiste en seguir el camino de los rayos para observar como pierden energía y calcular cuanta energía llega al receptor en cada reflexión, esto para construir un histograma dependiente de la frecuencia y después, obtener la respuesta al impulso pesando un proceso aleatorio de Poisson con el histograma.

El programa está basado en una implementación derivada de *Room Impulse Response Simulation with Stochastic Ray Tracing* [28], la cual no nos permite tener características

localizadas, sino que las toma uniformes en toda la pared, lo cual entra en conflicto con la colocación de paneles acústicos.

Para empezar se colocan las dimensiones de la sala, para posteriormente colocar las posiciones del emisor y del receptor y colocar el radio del micrófono, el cual en este caso es de 8.75cm.

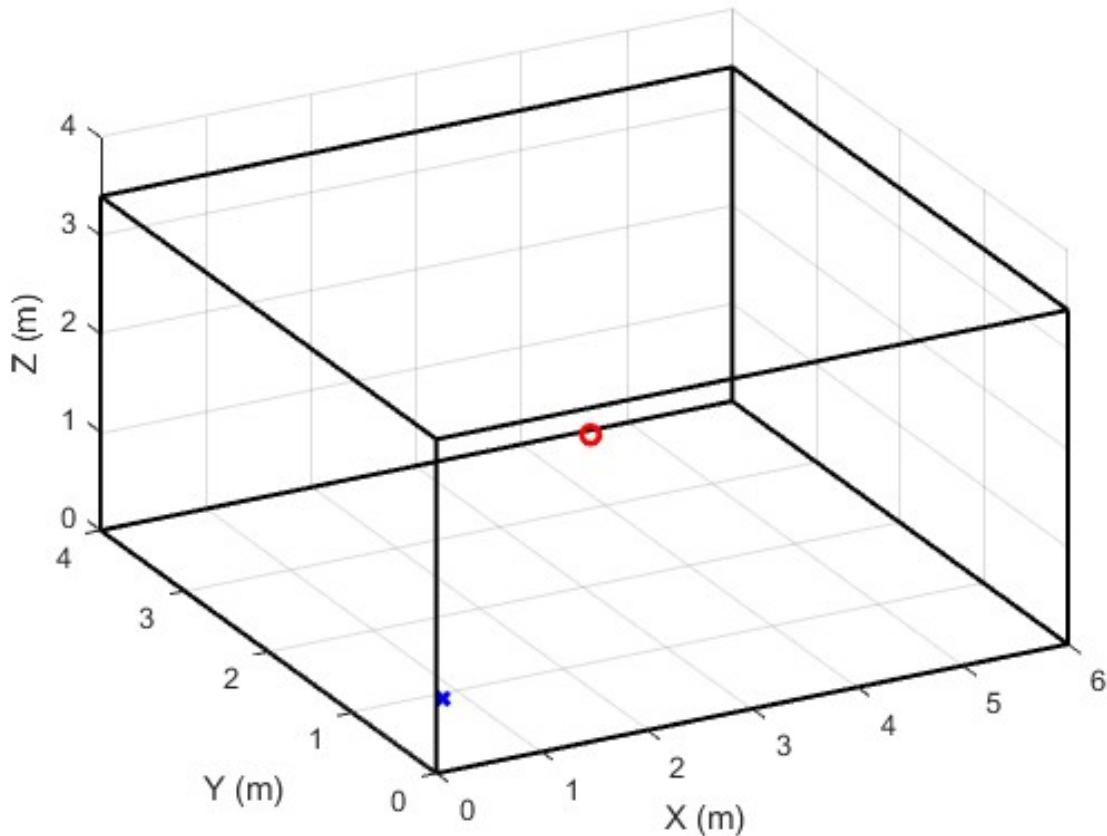


Figura 15: Geometría del cuarto simulado y posiciones del emisor y receptor

En la siguiente sección se generan los rayos, los cuales son emanados de la fuente en direcciones aleatorias. Para generar los rayos se utiliza la función RandSampleSphere, dichos rayos son una matriz N por 3 y cada fila de rayos mantiene la dirección del vector de rayos tridimensional.

Posteriormente en el código se definen los coeficientes de reflexión y dispersión. Un rayo de sonido se refleja cuando incide sobre una superficie. La reflexión es una combinación de un componente especular y un componente difuso. El coeficiente de absorción es una medida de cuánto sonido se absorbe (en lugar de reflejarse) al golpear una superficie, mientras que el coeficiente de difusión indica que tan especular o difusa es la reflexión.

Debido a que los parámetros acústicos se calculan para diferentes bandas de frecuencia, se tiene que hacer un análisis en diferentes frecuencias, dadas por FVect.

```

1 clear; close all; clc;
2
3 %% SetUp
4 SetUpStruct.room = [10 8 4];
5 SetUpStruct.src_pos = [2 2 2];
6 SetUpStruct.mic_pos = [5 5 1.8];
7 SetUpStruct.mic_radius = 0.0875;
8 impResTime = 10;
9
10 plotRoom(SetUpStruct.room, SetUpStruct.mic_pos, SetUpStruct.src_pos, 1)
11
12 %% Generate Rays
13 N = 5000;
14 rng(0)
15 rays = RandSampleSphere(N);
16
17 %% Reflections and Scattering Coefficients
18 FVect = [125 250 500 1000 2000 4000];
19
20 abs_coeffs = [];
21 abs_coeffs(:,1) = [0.02, 0.02, 0.03, 0.03, 0.04, 0.05, 0.05]; %Concrete
22 abs_coeffs(:,2) = [0.70, 0.45, 0.65, 0.60, 0.75, 0.65, 0.65]; %AbsPanels
23 abs_coeffs(:,3) = [0.14, 0.10, 0.06, 0.08, 0.10, 0.10, 0.10]; %Door
24
25 scatt_coeffs = [];
26 scatt_coeffs(:,1) = [0.30, 0.50, 0.60, 0.60, 0.70, 0.70, 0.70];
27 scatt_coeffs(:,2) = [0.30, 0.50, 0.60, 0.60, 0.70, 0.70, 0.70];
28 scatt_coeffs(:,3) = [0.30, 0.50, 0.60, 0.60, 0.70, 0.70, 0.70];

```

A continuación se hace la modificación al programa, creando un mapa de cada una de las paredes, que contiene valores diferentes de absorción y difusión para cada centímetro cuadrado de la pared.

```

1 abs_map = {};
2 abs_map{1} = zeros(round(room_dim(2)*100), round(room_dim(3)*100), num_FBands);
3 abs_map{2} = zeros(round(room_dim(2)*100), round(room_dim(3)*100), num_FBands);
4 abs_map{3} = zeros(round(room_dim(1)*100), round(room_dim(3)*100), num_FBands);
5 abs_map{4} = zeros(round(room_dim(1)*100), round(room_dim(3)*100), num_FBands);
6 abs_map{5} = zeros(round(room_dim(1)*100), round(room_dim(2)*100), num_FBands);
7 abs_map{6} = zeros(round(room_dim(1)*100), round(room_dim(2)*100), num_FBands);
8
9 scatt_map = {};
10 scatt_map{1} = zeros(round(room_dim(2)*100), round(room_dim(3)*100), num_FBands)
11 ;
11 scatt_map{2} = zeros(round(room_dim(2)*100), round(room_dim(3)*100), num_FBands)
11 ;
12 scatt_map{3} = zeros(round(room_dim(1)*100), round(room_dim(3)*100), num_FBands)
12 ;

```

```

13 scatt_map{4} = zeros(round(room_dim(1)*100),round(room_dim(3)*100),num_FBands)
    ;
14 scatt_map{5} = zeros(round(room_dim(1)*100),round(room_dim(2)*100),num_FBands)
    ;
15 scatt_map{6} = zeros(round(room_dim(1)*100),round(room_dim(2)*100),num_FBands)
    ;

```

A continuación, la función *updateAbsScattCoefs* nos ayuda a actualizar los mapas de absorción y difusión, colocando los valores de absorción y difusión correspondientes a cada banda de frecuencia, en los índices del mapa correspondientes a sus posiciones en la pared.

```

1 [abs_map, scatt_map] = updateAbsScattCoeffs(abs_map, scatt_map, 1, abs_coeffs
    (:,1), ...
2     scatt_coeffs(:,1), [0,0], [room_dim(2), room_dim(3)]) ; %SmallWall
3 [abs_map, scatt_map] = updateAbsScattCoeffs(abs_map, scatt_map, 2, abs_coeffs
    (:,1), ...
4     scatt_coeffs(:,1), [0,0], [room_dim(2), room_dim(3)]) ; %OpSmallWall
5 [abs_map, scatt_map] = updateAbsScattCoeffs(abs_map, scatt_map, 3, abs_coeffs
    (:,1), ...
6     scatt_coeffs(:,1), [0,0], [room_dim(1), room_dim(3)]) ; %LargeWall
7 [abs_map, scatt_map] = updateAbsScattCoeffs(abs_map, scatt_map, 4, abs_coeffs
    (:,1), ...
8     scatt_coeffs(:,1), [0,0], [room_dim(1), room_dim(3)]) ; %OpLargeWall
9 [abs_map, scatt_map] = updateAbsScattCoeffs(abs_map, scatt_map, 5, abs_coeffs
    (:,1), ...
10    scatt_coeffs(:,1), [0,0], [room_dim(1), room_dim(2)]) ; %Floor
11 [abs_map, scatt_map] = updateAbsScattCoeffs(abs_map, scatt_map, 6, abs_coeffs
    (:,1), ...
12    scatt_coeffs(:,1), [0,0], [room_dim(1), room_dim(2)]) ; %Ceiling

```

Como los mapas están vacíos al crearse, primero se definen todas las paredes del cuarto con los respectivos coeficientes de absorción y difusión, y posteriormente ya se pueden actualizar ciertas zonas, con los coeficientes correspondientes a los paneles.

El mapa de reflexiones se obtiene a partir del mapa de absorción siguiendo la siguiente fórmula [29]:

$$Ref = \sqrt{1 - Abs} \quad (19)$$

```

1 ref_map = cell(1,num_FBands);
2 for i = 1:numel(abs_map)
3     ref_map{i} = sqrt(1-abs_map{i});
4 end

```

Se definen también los parámetros del histograma.

```

1 histTimeStep = 0.0010;
2 nTBins = round(impResTime/histTimeStep);

```

```

3| nFBins = length(FVect);
4| TFHist = zeros(nTBins, nFBins);

```

El proceso de *Ray-Tracing* comienza tomando un rayo dentro de una banda de frecuencia, tomando su posición, dirección, el tiempo del rayo y su energía, y posteriormente, calcular en donde va a colisionar. Esto se hace observando los signos de la dirección del rayo, los cuales nos dice con cuál, de entre dos paredes paralelas, va a colisionar el rayo. A continuación, se calcula el desplazamiento necesario para llegar a la coordenada constante de una pared y resaltando que chocara con la pared para la que necesite el menor desplazamiento. Este cálculo se hace dentro de la función *GetImpactWall*

```

1 function [surfaceofimpact, displacement] = getImpactWall(ray_xyz, ray_dxyz,
   roomDims)
2 % GETIMPACTWALL Determine which wall the ray encounters
3 surfaceofimpact = -1;
4 displacement = 1000;
5 % Compute time to intersection with x-surfaces
6 if (ray_dxyz(1) < 0)
7     displacement = -ray_xyz(1) / ray_dxyz(1);
8     if displacement==0
9         displacement=1000;
10    end
11    surfaceofimpact = 1; %"SmallWall"
12 elseif (ray_dxyz(1) > 0)
13     displacement = (roomDims(1) - ray_xyz(1)) / ray_dxyz(1);
14     if displacement==0
15         displacement=1000;c
16     end
17     surfaceofimpact = 2; %"OpSmallWall"
18 end
19 % Compute time to intersection with y-surfaces
20 if ray_dxyz(2)<0
21     t = -ray_xyz(2) / ray_dxyz(2);
22     if (t<displacement) && t>0
23         surfaceofimpact = 3; %"LargeWall";
24         displacement = t;
25     end
26 elseif ray_dxyz(2)>0
27     t = (roomDims(2) - ray_xyz(2)) / ray_dxyz(2);
28     if (t<displacement) && t>0
29         surfaceofimpact = 4; %"OpLargeWall";
30         displacement = t;
31     end
32 end
33 % Compute time to intersection with z-surfaces
34 if ray_dxyz(3)<0
35     t = -ray_xyz(3) / ray_dxyz(3);
36     if (t<displacement) && t>0
37         surfaceofimpact = 5; %"Floor";
38         displacement = t;
39     end
40 elseif ray_dxyz(3)>0

```

```

41 t = (roomDims(3) - ray_xyz(3)) / ray_dxyz(3);
42 if (t<displacement) && t>0
43     surfaceofimpact = 6; %> Ceiling";
44     displacement = t;
45 end
46 end
47
48 displacement = displacement * ray_dxyz;
49
50 end

```

El desplazamiento y la pared de impacto, nos permite obtener las coordenadas en las que choca el rayo y así, consultar nuestro mapa de reflexión y difusión, para calcular el decaimiento de la energía con base en la reflexión y una nueva dirección para el rayo, la cual es combinación de la reflexión especular y la difusa.

Adicionalmente, se usa la difusión en ese punto, para calcular que cantidad de la energía restante del rayo va a capturar el receptor, la cual disminuye conforme más se aleja del vector normal de la pared.

Este proceso se repite para la nueva posición, dirección, tiempo y energía del rayo; hasta que se supere el tiempo de simulación o la energía del rayo disminuya hasta ser despreciable.

```

1 for iBand = 1:nFBins
2     fprintf(" Calculating rays for band %d\n",iBand)
3 % Perform ray tracing independently for each frequency band.
4     for iRay = 1:size(rays,1)
5         % Select ray direction
6         ray = rays(iRay,:);
7         % All rays start at the source/transmitter
8         ray_xyz = source_pos;
9         % Set initial ray direction. This direction changes as the ray is
10        % reflected off surfaces.
11        ray_dxyz = ray;
12        % Initialize ray travel time. Ray tracing is terminated when the
13        % travel time exceeds the impulse response length.
14        ray_time = 0;
15        % Initialize the ray energy to a normalized value of 1.      Energy
16        % decreases when the ray hits a surface.
17        ray_energy = 1;
18
19        while (ray_time <= impResTime)
20
21            % Determine the surface that the ray encounters
22            [surfaceofimpact,displacement] = getImpactWall(ray_xyz, ...
23                                              ray_dxyz,room_dim);
24
25            % Determine the distance traveled by the ray
26            distance = sqrt(sum(displacement.^2));
27
28            % Determine the coordinates of the impact point

```

```

29 impactCoord = ray_xyz+displacement;
30
31 if surfaceofimpact > 4
32     pointOfImpact = [ impactCoord(1) ,impactCoord(2) ];
33 elseif surfaceofimpact < 3
34     pointOfImpact = [ impactCoord(2) ,impactCoord(3) ];
35 else
36     pointOfImpact = [ impactCoord(1) ,impactCoord(3) ];
37 end
38
39 % Update ray location/source
40 ray_xyz = impactCoord;
41
42 % Update cumulative ray travel time
43 c = 343; % speed of light (m/s)
44 ray_time = ray_time+distance/c;
45
46 % Apply surface reflection to ray's energy
47 % This is the amount of energy that is not lost through
48 % absorption .
49
50 ReflectionAtPoint = ref_map{surfaceofimpact}( ceil(100*
pointOfImpact(1)) ,ceil(100*pointOfImpact(2)) ,iBand );
51 ray_energy = ray_energy*ReflectionAtPoint ;
52
53 % Apply diffuse reflection to ray energy
54 % This is the fraction of energy used to determine what is
55 % detected at the receiver
56 DifussionAtPoint = scatt_map{surfaceofimpact}( ceil(100*
pointOfImpact(1)) ,ceil(100*pointOfImpact(2)) ,iBand );
57 rayrecv_energy = ray_energy*DifussionAtPoint ;
58
59 % Determine impact point-to-receiver direction .
60 rayrecvvector = mic_pos-impactCoord ;
61
62 % Determine the ray's time of arrival at receiver .
63 distance = sqrt(sum(rayrecvvector.*rayrecvvector));
64 recv_timeofarrival = ray_time+distance/c;
65
66 if recv_timeofarrival>impResTime
67     break
68 end
69
70 if ray_energy < 0.000001
71     break
72 end
73
74 % Determine amount of diffuse energy that reaches the receiver .
75 % See (5.20) in [2].
76
77 % Compute received energy
78 N = getWallNormalVector(surfaceofimpact);
79 cosTheta = sum(rayrecvvector.*N)/(sqrt(sum(rayrecvvector.^2)));
80 cosAlpha = sqrt(sum(rayrecvvector.^2)-mic_radius^2)/sum(

```

```
rayrecvvector.^2);
81 E = (1-cosAlpha)*2*cosTheta*rayrecv_energy;
82
83 % Update energy histogram
84 tbin = floor(recv_timeofarrival/histTimeStep + 0.5);
85 TFHist(tbin,iBand) = TFHist(tbin,iBand) + E;
86
87 % Compute a new direction for the ray.
88 % Pick a random direction that is in the hemisphere of the
89 % normal to the impact surface.
90 d = rand(1,3);
91 d = d/norm(d);
92 if sum(d.*N)<0
93     d = -d;
94 end
95
96 % Derive the specular reflection with respect to the incident
97 % wall
98 ref = ray_dxyz -2*(sum(ray_dxyz.*N))*N;
99
100 % Combine the specular and random components
101 d = d/norm(d);
102 ref = ref/norm(ref);
103 ray_dxyz = DifussionAtPoint*d+(1-DifussionAtPoint)*ref;
104 ray_dxyz = ray_dxyz/norm(ray_dxyz);
105 end
106 end
107 end
```

Este proceso nos deja con un histograma dependiente de la frecuencia, el cual representa la envoltura de la respuesta al impulso.

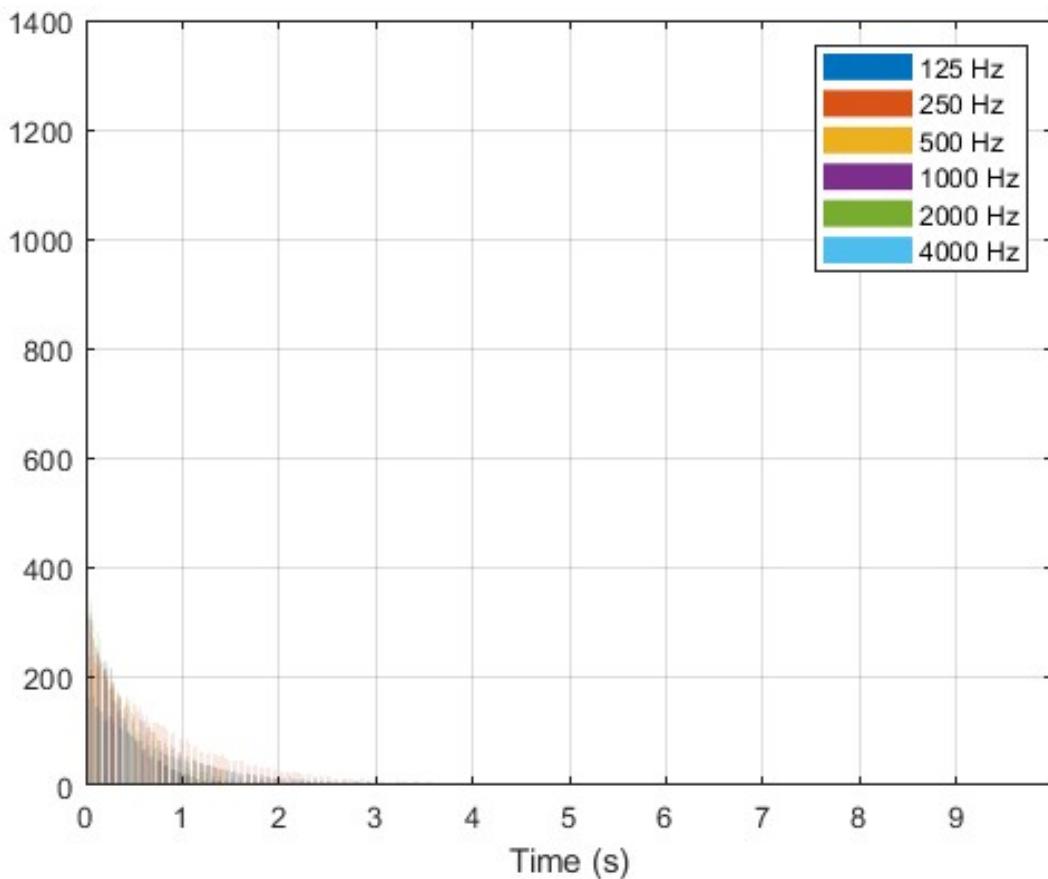


Figura 16: Histograma dependiente de la frecuencia

Para la construcción de la respuesta al impulso, es necesario modelar la estructura detallada a partir del histograma. Esto se puede hacer mediante ruido aleatorio con una distribución de Poisson, y se construye tomando la reflexión de un rayo como evento [30].

Es importante pasar el proceso aleatorio de Poisson por filtros, pasa bandas que empaten con las frecuencias del histograma, para así, poder multiplicarlos.

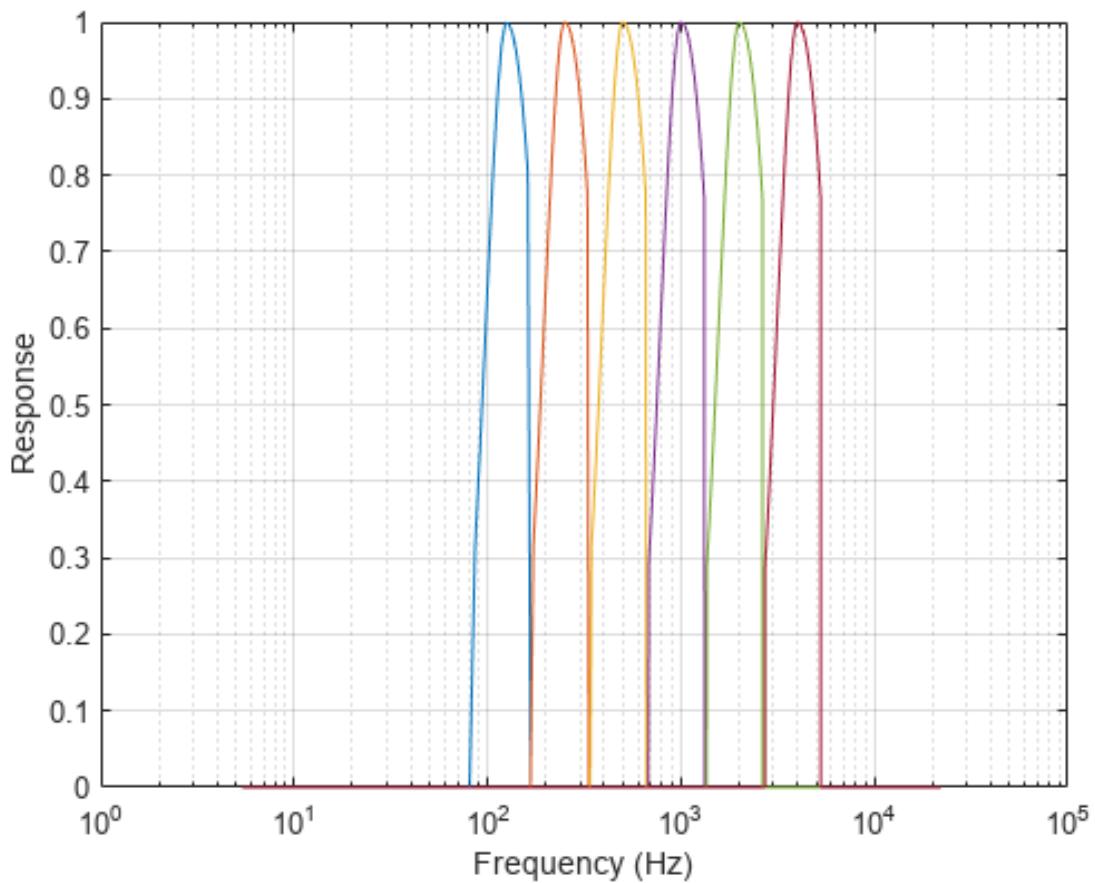


Figura 17: Filtros pasa-bandas para el proceso de Poisson

```

1 fs = 44100;
2 V = prod(room_dim);
3 t0 = ((2*V*log(2))/(4*pi*c^3))^(1/3); % eq 5.45 in [2]
4 poissonProcess = [];
5 timeValues = [];
6 t = t0;
7 while (t<impResTime)
8     timeValues = [timeValues t]; %#ok
9     % Determine polarity.
10    if (round(t*fs)-t*fs) < 0
11        poissonProcess = [poissonProcess 1]; %#ok
12    else
13        poissonProcess = [poissonProcess -1];%#ok
14    end
15    % Determine the mean event occurrence (eq 5.44 in [2])
16    mu = min(1e4,4*pi*c^3*t^2/V);
17    % Determine the interval size (eq. 5.44 in [2])
18    deltaTA = (1/mu)*log(1/rand); % eq. 5.43 in [2]
19    t = t+deltaTA;
20 end

```

```

21 randSeq = zeros(ceil(impResTime*fs),1);
22 for index=1:length(timeValues)
23     randSeq(round(timeValues(index)*fs)) = poissonProcess(index);
24 end
25 flow = [115 225 450 900 1800 3600 7200];
26 fhigh = [135 275 550 1100 2200 4400 8800];
27 NFFT = 8192;
28 win = hann(882,"symmetric");
29 sfft = dsp.STFT(Window = win, OverlapLength=441, FFTLength=NFFT, FrequencyRange="onesided");
30 isfft = dsp.ISTFT(Window=win, OverlapLength=441, FrequencyRange="onesided");
31 F = sfft.getFrequencyVector(fs);
32 RCF = zeros(length(FVect),length(F));
33 for index0 = 1:length(FVect)
34     for index=1:length(F)
35         f = F(index);
36         if f < FVect(index0) && f >= flow(index0)
37             RCF(index0,index) = .5*(1+cos(2*pi*f/FVect(index0)));
38         end
39         if f < fhigh(index0) && f >= FVect(index0)
40             RCF(index0,index) = .5*(1-cos(2*pi*f/(FVect(index0)+1)));
41         end
42     end
43 end
44 frameLength = 441;
45 numFrames = length(randSeq)/frameLength;
46 y = zeros(length(randSeq),numel(FVect));
47 for index=1:numFrames
48     x = randSeq((index-1)*frameLength+1:index*frameLength);
49     X = sfft(x);
50     X = X.*RCF.';
51     y((index-1)*frameLength+1:index*frameLength,:) = isfft(X);
52 end
53 impTimes = (1/fs)*(0:size(y,1)-1);
54 hisTimes = histTimeStep/2 + histTimeStep*(0:nTBins);
55 W = zeros(size(impTimes,2),numel(FVect));
56 BW = fhigh - flow;
57 for k=1:size(TFHist,1)
58     gk0 = floor((k-1)*fs*histTimeStep)+1;
59     gk1 = floor(k*fs*histTimeStep);
60     yy = y(gk0:gk1,:).^2;
61     val = sqrt(TFHist(k,:)/sum(yy,1)).*sqrt(BW/(fs/2));
62     for iRay=gk0:gk1
63         W(iRay,:)= val;
64     end
65 end

```

Por último, podemos crear la respuesta al impulso con los pesos y el histograma. Es importante notar que se agregó ruido de fondo, que nos permita observar el decaimiento en la integral de Schroeder.

```

1 y_2 = y.*W;
2 ip = sum(y_2,2);

```

```
3 ip = ip ./max( abs(ip) );
4 ip = ip + rand(length(ip),1)/1000;
5 vectorTiempo = (1/fs)*(0: numel(ip)-1);
6 figure
7 plot(vectorTiempo, ip.^2)
8 grid on
9 xlabel("Time (s)")
10 ylabel("Impulse Response")
```

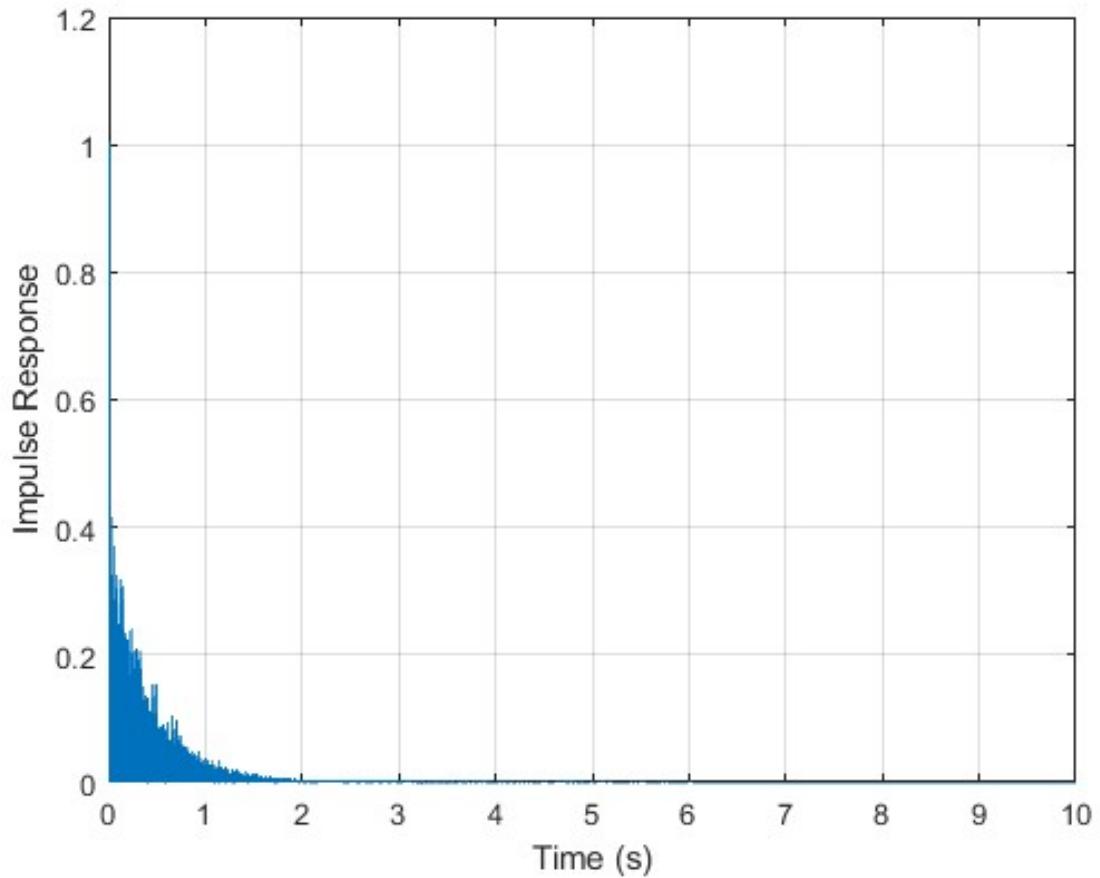


Figura 18: RIR cuadrada obtenida de simulación

A esta respuesta al impulso podemos aplicarle los mismos análisis que hacíamos con una respuesta al impulso obtenida mediante una medición.

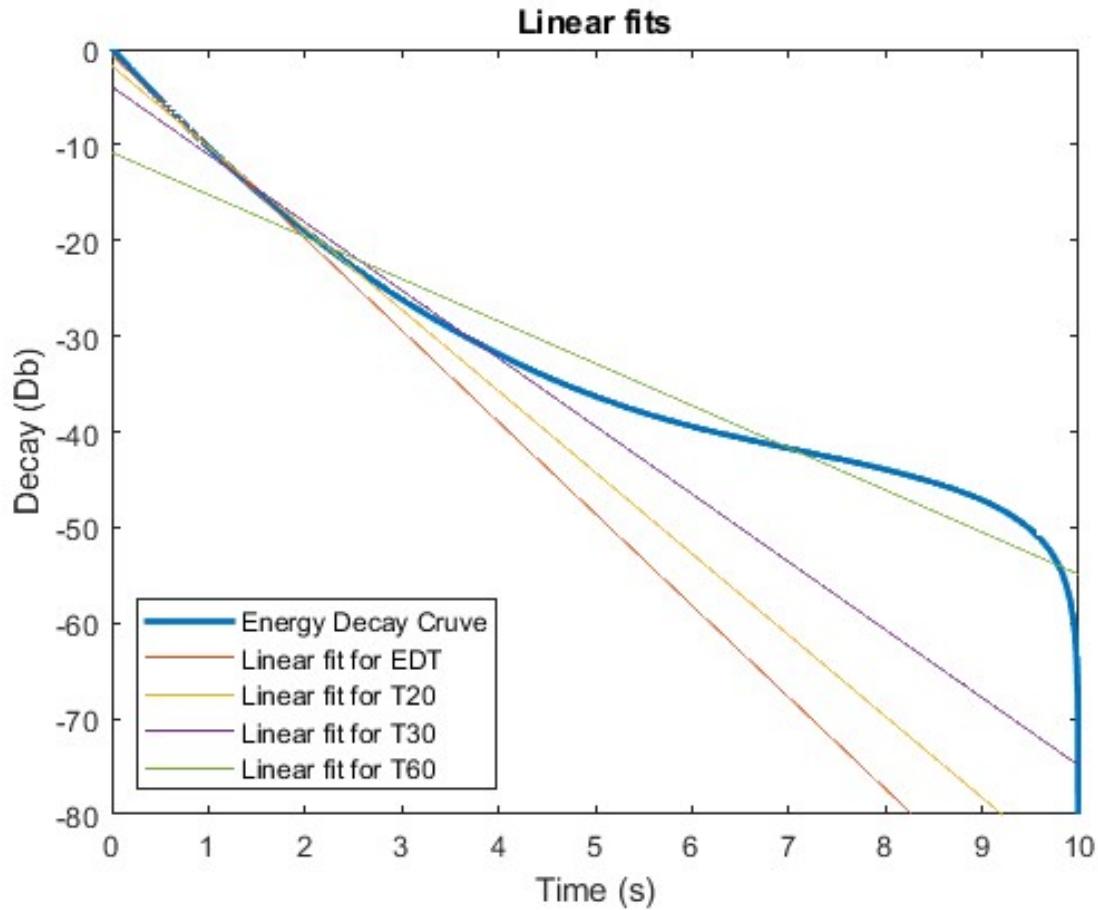


Figura 19: Análisis de la respuesta al impulso simulada

Validación de la obtención de respuesta al impulso por simulación Para comprobar el funcionamiento de nuestra simulación, recurrimos al uso de una paquetería de Python llamada PyRoomAcoustics, la cual permite hacer la simulación tal como la hicimos en MATLAB, pero de manera más eficiente y con métodos más completos. Sin profundizar en el funcionamiento, se creó un cuarto igual al de la implementación de MATLAB y se simuló la respuesta al impulso (Se adjunta el código utilizado en los apéndices).

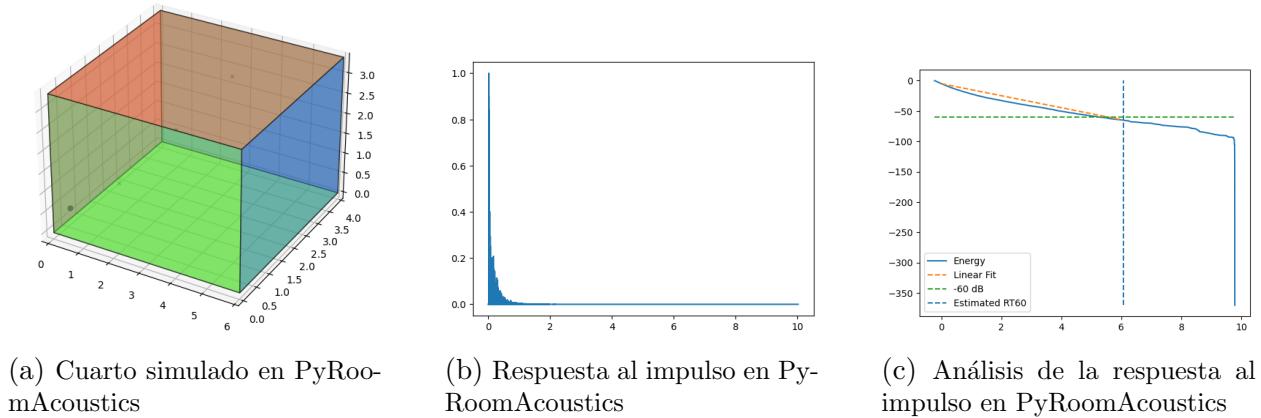


Figura 20: Simulación con PyRoomAcoustics

Como se puede observar, la simulación en Python arroja resultados muy similares a los obtenidos con la implementación en MATLAB. En el caso de MATLAB, el tiempo de reverberación calculado fue de 6,19 segundos, mientras que en Python fue de 6,024 segundos, un tiempo muy cercano considerando que se pierde resolución conforme aumenta el tiempo de reverberación debido a la naturaleza exponencial del decaimiento.

Una vez comprobada la implementación en MATLAB, podemos utilizarla con características localizadas de los materiales en las paredes.

2.2.1.2 MF1.2 Procesamiento de la estrategia de control y posiciones de los paneles

Cálculo de Posiciones para los paneles Del cálculo de la acústica obtuvimos el tiempo de reverberación, de cuál se puede calcular la absorción estimada el cuarto. La absorción calculada nos permite calcular, de acuerdo a la absorción de los paneles, la cantidad de superficie que se tiene que añadir al cuarto para alcanzar un tiempo de reverberación deseado.

Entonces, resta obtener un programa que, a partir de las posiciones de los paneles y la superficie necesaria, nos entregue las nuevas posiciones de los paneles que resulten en la acústica deseada.

La implementación se hizo en *Python* y la lógica es la siguiente:

- Encontrar el número de paneles que se deben de cambiar de la posición de reflexión a la posición de absorción, con base en la absorción efectiva del panel y la diferencia entre tiempos de reverberación (actual y deseado)
- Si el número de paneles supera al número de paneles disponibles, solo mover los disponibles

- Preguntar al usuario cuantos de los paneles que no se cambiaron a la posición de absorción quiere cambiar a la posición de difusión. Se deja este proceso al usuario, ya que la especialidad no es un parámetro ideal, sino que varía en función al deseo artístico del usuario.
- Dada una jerarquía de paneles (primero mover los de en medio, e ir moviéndose hacia los extremos para después terminar los huecos) cambiar el indicador de tipo de panel a absorción y los deseados a difracción.

Para encontrar el número de paneles faltantes, utilizamos la fórmula de la reverberación [26]:

$$RT = \frac{0,161V}{S\alpha} \quad (20)$$

Ya que la frecuencia central utilizada por la mayoría de instrumentos y tratamientos acústicos es a los 500 Hz, se tomara la absorción de los paneles a esta frecuencia [31].

```

1 def PanelsToChange( Current_RT60 , Needed_RT60 , room_dim ) :
2     V = np.prod(room_dim)
3     Panel_Surface = np.prod(PanelSize)
4     Current_Abs = 0.161*V/Current_RT60
5     Needed_Abs = 0.161*V/Needed_RT60
6     AbsFromSinglePanel = Panel_Surface*(AbsCoeffs[ 'AbsPanels '][6] - AbsCoeffs[ 'Concrete '][6])
7     PanelsToChange = (Needed_Abs-Current_Abs)//AbsFromSinglePanel
8     return PanelsToChange
9
10 x = PanelsToChange(0.9,0.4,room_dim)
11 print(f'Changing {int(x)} panels to Absorption')

```

Se puede observar que se colocaron los coeficientes de absorción en diccionarios, que se pueden acceder por medio del indicador *string*.

```

1 if x > 18:
2     print('Only 18 panels available')
3     x = 18
4
5 DPanels = int(input(f"Enter the number of diffusion panels to put ({18-x} Available): "))
6 if DPanels > 18-x :
7     print("Not a valid number, using 0")
8     DPanels = 0

```

Después, se hace la validación de la cantidad de paneles disponibles y, en función de los restantes, la cantidad de paneles de difusión a cambiar.

Para la modificación del tipo de panel en función a la jerarquía se decidió utilizar un tipo de estructura llamada *linked list*, ya que nos permitirá guardar los datos del panel (Qué tipo

de superficie se pondrá y qué posición ocupa en el arreglo de paneles) en un nodo, además de poder indicar, cuál es el siguiente nodo a visitar. De esta manera, podemos crear una programa que va visitando los paneles en función de la lista jerárquica, la cual será, el orden de los paneles en la lista. Para el movimiento en sí, se puede visitar la lista y obtener los datos del nodo, lo que nos permitirá asociar una posición en arreglo con cierto tipo de panel.

```

1 class Node:
2     def __init__(self, data, position):
3         self.Type = data
4         self.Position = position
5         self.next = None

```

El nodo solo contiene el tipo de panel (1:'Reflexion',2:'Absorcion',3:'Difraccion'), la posición que ocupa en el arreglo, existen 18 posiciones que indican los 9 paneles que hay por pared, y la referencia al siguiente nodo, el cual nos da la lista jerárquica.

El constructor de la lista solo indica cuál es la *cabeza* o inicio de la lista y después, se van añadiendo nodos conforme a la lista jerárquica.

```

1 PanelOrder = LinkedList()
2 PanelOrder.insertAtBegin(NameToKey['Refle'], 14)
3 PanelOrder.insertAtEnd(NameToKey['Refle'], 5)
4 PanelOrder.insertAtEnd(NameToKey['Refle'], 12)
5 PanelOrder.insertAtEnd(NameToKey['Refle'], 16)
6 PanelOrder.insertAtEnd(NameToKey['Refle'], 3)
7 PanelOrder.insertAtEnd(NameToKey['Refle'], 7)
8 PanelOrder.insertAtEnd(NameToKey['Refle'], 10)
9 PanelOrder.insertAtEnd(NameToKey['Refle'], 18)
10 PanelOrder.insertAtEnd(NameToKey['Refle'], 1)
11 PanelOrder.insertAtEnd(NameToKey['Refle'], 9)
12 PanelOrder.insertAtEnd(NameToKey['Refle'], 13)
13 PanelOrder.insertAtEnd(NameToKey['Refle'], 15)
14 PanelOrder.insertAtEnd(NameToKey['Refle'], 4)
15 PanelOrder.insertAtEnd(NameToKey['Refle'], 6)
16 PanelOrder.insertAtEnd(NameToKey['Refle'], 11)
17 PanelOrder.insertAtEnd(NameToKey['Refle'], 17)
18 PanelOrder.insertAtEnd(NameToKey['Refle'], 2)
19 PanelOrder.insertAtEnd(NameToKey['Refle'], 8)

```

En la lista jerárquica la prioridad es la siguiente:

- (1) Panel central (Posición 5)
- (2) Panel izquierdo con uno de separación del central (Posición 3)
- (2) Panel derecho con uno de separación del central (Posición 7)
- (3) Panel del extremo izquierdo (Posición 1)
- (3) Panel del extremo derecho (Posición 9)

- (4) Panel izquierdo pegado al central (Posición 4)
- (4) Panel derecho pegado al central (Posición 6)
- (5) Panel izquierdo pegado al del extremo (Posición 2)
- (5) Panel derecho pegado al del extremo (Posición 8)

Esta lista jerárquica nos permite mantener el balance de paneles en el cuarto y los paneles con el mismo número se consideran un conjunto. Cabe resaltar que esta lista es solo para una pared, los de las demás paredes siguen la misma lista jerárquica, con la diferencia que al terminar un conjunto, se modifica ese mismo conjunto en la siguiente pared y posteriormente se vuelve a modificar el siguiente conjunto de la primera pared.

El cambio de tipo de los paneles se hace con el siguiente código:

```

1 for i in range(18):
2     if i < x:
3         PanelOrder.updateNode(NameToKey[ 'Abs' ], i)
4     elif i < x + DPanels:
5         PanelOrder.updateNode(NameToKey[ 'Scatt' ], i)
6     else:
7         PanelOrder.updateNode(NameToKey[ 'Refle' ], i)

```

Se puede ver como va por la lista jerárquica y cambia los absorbentes, luego los de difracción indicados y los restantes los mantiene en reflexivos.

Por último, para acceder a los paneles en orden, se guardan en un diccionario, donde la *key* es la posición en el arreglo (No la lista), y el valor es el tipo de panel en esa posición.

```

1 def GetPanels(PanelOrder):
2     PanelArray = {}
3     for i in range(18):
4         currentPanel_Type, currentPanel_Pos = PanelOrder.returnAtIndex(i)
5         PanelArray[currentPanel_Pos] = currentPanel_Type
6     return PanelArray
7
8 PanelArray = GetPanels(PanelOrder)
9 for i in range(1,19):
10    print(PanelArray[i])

```

Iterar a lo largo del diccionario nos da el tipo de panel de cada posición y es el utilizado para el control.

Para validar esto, se realizaron varios ejemplos. En el primer caso, se cambiará el tiempo de reverberación de 0.9 s a 0.85 s. El programa calcula que es necesario cambiar dos paneles a absorción y en este caso, se indicó que agregara cinco paneles de difusión. La lista de paneles entregada es: (1, 1, 3, 1, 2, 1, 3, 1, 1)(3, 1, 3, 1, 2, 1, 3, 1, 1). Se puede observar que solo los paneles centrales se cambiaron a absorción, que los de difracción están balanceados en el arreglo derecho y que el panel de difusión extra en la derecha está en el extremo izquierdo.

```
Changing 2 panels to Absorption
Enter the number of diffusion panels to put (16.0 Available): 5
1
1
3
1
2
1
3
1
1
3
1
3
1
2
1
3
1
1
```

Figura 21: Prueba uno de cálculo de posiciones para los paneles

En el segundo caso, se cambiará el tiempo de reverberación de 0.9 s a 0.7 s. El programa calcula que es necesario cambiar doce paneles a absorción y en este caso, se indicó que agregara 2 paneles de difusión. La lista de paneles entregada es: $(2, 1, 2, 3, 2, 3, 2, 1, 2)(2, 1, 2, 2, 2, 2, 1, 2)$. Se puede observar que en ambos arreglos, los paneles impares se cambiaron a absorción, y los dos sobrantes se colocaron junto al central, y se puede ver también que los de difracción están balanceados en el arreglo izquierdo.

```
Changing 12 panels to Absorption
Enter the number of diffusion panels to put (6.0 Available): 2
2
1
2
3
2
3
2
1
2
2
2
1
2
2
2
2
1
2
```

Figura 22: Prueba dos de cálculo de posiciones para los paneles

El programa completo se encuentra en los apéndices para su revisión.

Procesamiento de la estrategia de control El control del sistema se hará mediante una ley de control de lazo cerrado de tipo PID. Se decidió hacer uso de esta estrategia básica debido a la transmisión de engrane de corona que tiene el sistema.

El sistema de transmisión permite un desacople de inercias, esto gracias a que en caso de que el tornillo no se mueve, el eje perpendicular que contiene la corona no puede moverse por su propia inercia, ya que sus dientes se detienen con los dientes del tornillo (que no cuentan con movimiento en el eje perpendicular al de la corona). Por lo anterior, se puede hacer el control del eje del tornillo, tomando en cuenta para el control, únicamente la inercia y amortiguamiento efectivo de este. El peso del eje de la corona, se puede ver como una fuerza externa que se aplica sobre el sistema principal y que se sopesa mediante el control PID.

El propósito del control es hacer que el eje del prisma triangular, se mueva un tercio de vuelta por vez, y que permita que en cada tercio de vuelta, los solenoides puedan desacoplar los paneles que deben quedarse en esa posición.

Se hará uso de un control por trayectoria, con una interpolación con polinomios de Bézier, ya que nos permitirá un movimiento suave de aceleración y desaceleración del motor [32].

El polinomio interpolador de grado $2n - 1$ tiene la siguiente fórmula [33]:

$$\mu(t, t_1, t_2) = \mu(\delta) = \delta^3 (\gamma_1 - \gamma_2\delta + \gamma_3\delta^2 - \gamma_4\delta^3 + \cdots + (-1)^{n-1} \gamma_n\delta^{n-1}) \quad (21)$$

Donde δ es igual a $\frac{t-t_1}{t_2-t_1}$

En este caso utilizaremos un polinomio de grado 11, entonces $n = 6$.

$$\mu(\delta) = \delta^6 (\gamma_1 - \gamma_2\delta + \gamma_3\delta^2 - \gamma_4\delta^3 + \gamma_5\delta^4 - \gamma_6\delta^5) \quad (22)$$

Los polinomios cuentan con la condición de frontera, $\mu(1) = 1$, y, por tanto, las múltiples derivadas, son iguales a cero [33]. Se pueden utilizar estas condiciones de frontera para encontrar los parámetros γ , los cuales son únicos para un polinomio de cierto grado. En nuestro caso, las γ son:

- $\gamma_1 = 175,63$
- $\gamma_2 = 548,18$
- $\gamma_3 = 601,36$
- $\gamma_4 = 216,36$
- $\gamma_5 = -45,81$
- $\gamma_6 = -34,86$

Por último, se puede escribir la interpolación completa como:

$$q = \begin{cases} q_0 & t \leq t_0 \\ q_0 + (q_1 - q_0)\mu(\delta) & t_0 \leq t \leq t_1 \\ q_1 & t \geq t_1 \end{cases} \quad (23)$$

Se pueden utilizar las ecuaciones para la posición y las primeras dos derivadas, la velocidad y aceleración, para pasar la trayectoria al controlador PID.

Se hizo la validación de la trayectoria por medio de Simulink.

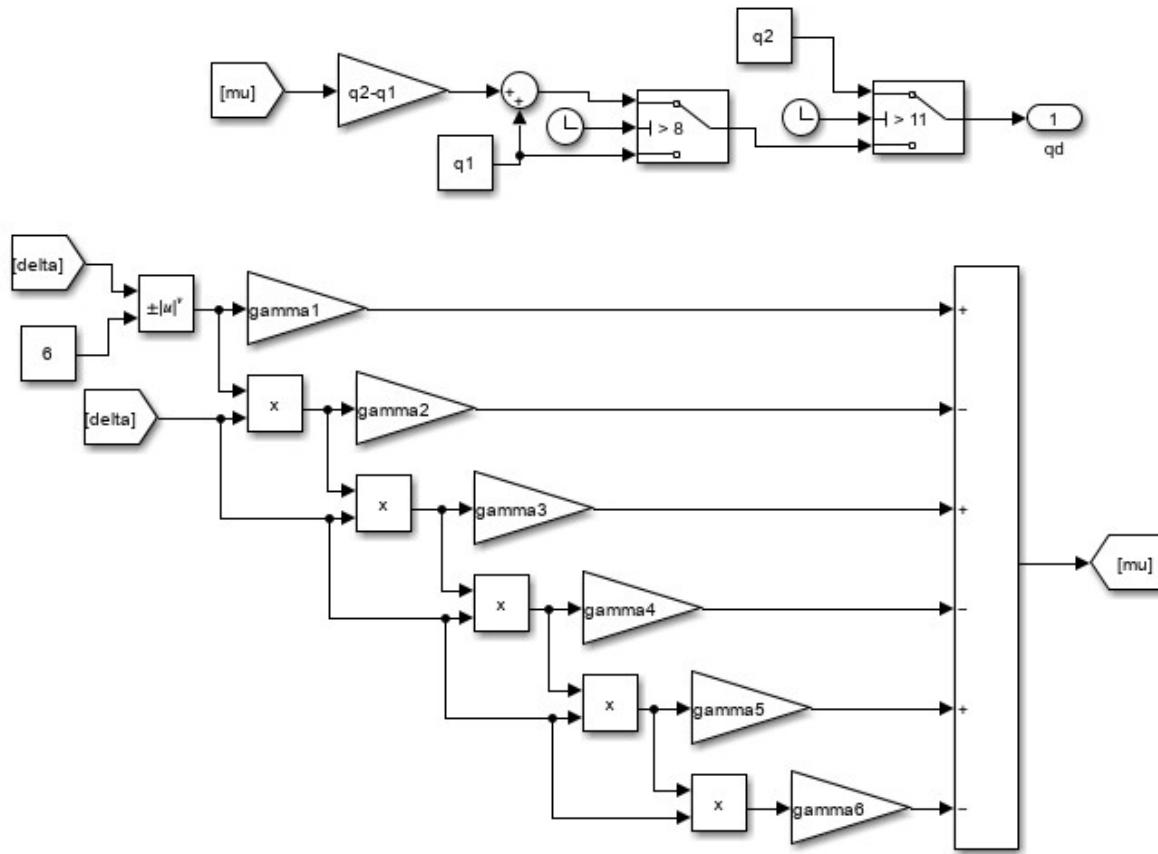


Figura 23: Interpolación de Bezier

Es importante notar que esto hace una interpolación de una posición a otra, pero nuestra implementación requiere que se haga una interpolación de una posición a otra, esperar en esa posición y posteriormente hacer otra interpolación de esa posición a la siguiente. Debido a esto se utilizarán tres interpolaciones, las cuales se seleccionaran en función del tiempo transcurrido.

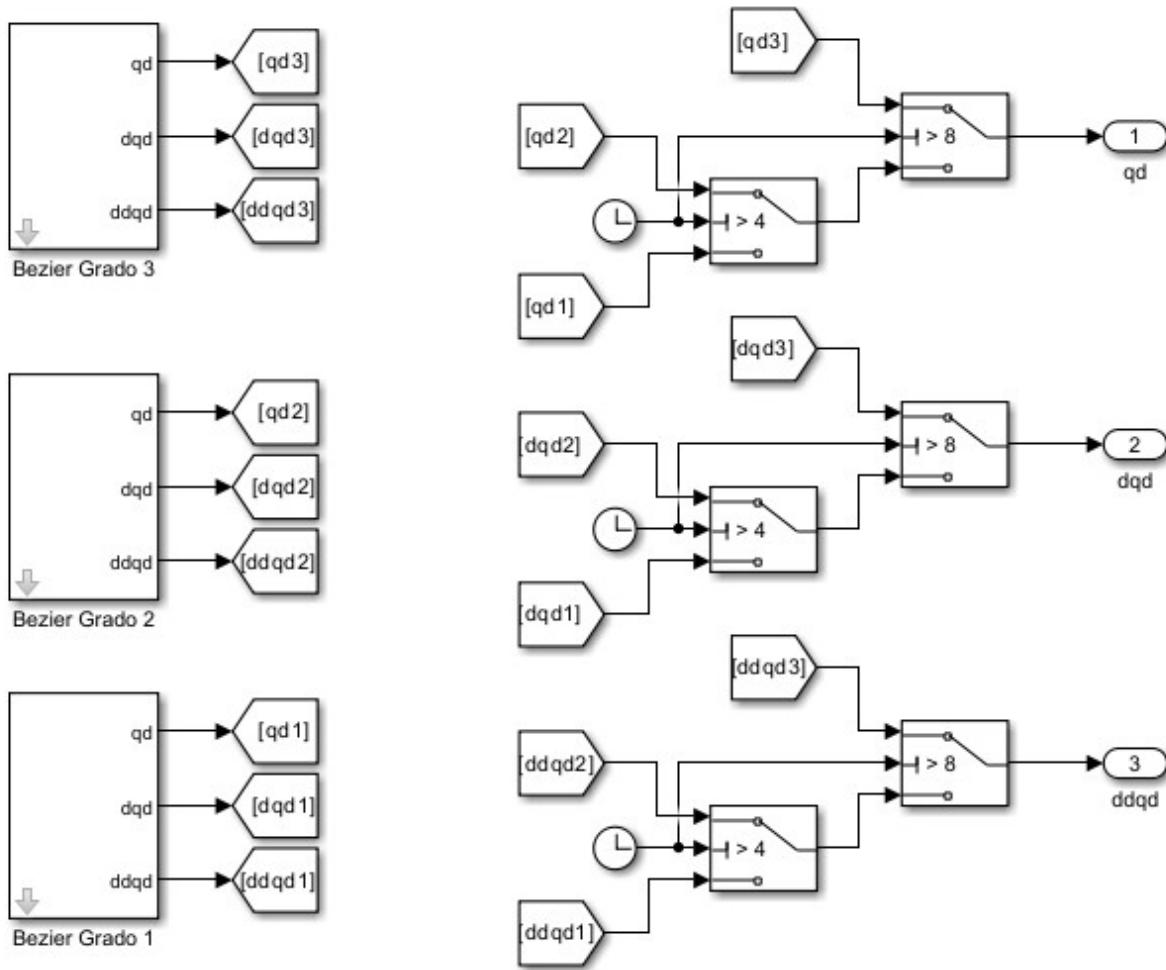


Figura 24: Triple Interpolación con polinomios de Bezier

Los únicos parámetros que tomara entonces la interpolación de entre dos posiciones, será el tiempo inicial y final de la interpolación, además de las posiciones iniciales y finales.

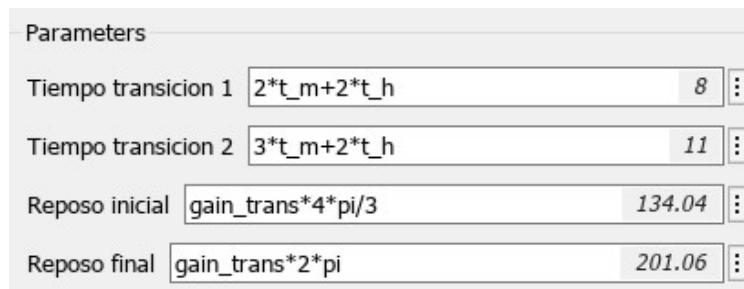


Figura 25: Parámetros de polinomio de Bezier

Estos parámetros son fijos, y dependen de la implementación. En nuestro caso, ya que queremos que el eje de la corona de un tercio de vuelta en cada trayectoria, entonces las posiciones del eje del tornillo irán desde cero hasta $\tau \cdot n \cdot 2 \cdot \pi/3$. Donde τ es la ganancia

debido a la transmisión de corona y n va de 1 a 3, que son las posiciones que va a tomar.

Externamente, la interpolación triple solo tomará tres parámetros, el tiempo que le toma al sistema realizar un tercio de vuelta, el tiempo que esperará para permitir el desacoplamiento de los paneles y la relación de transmisión de la transmisión de corona.

La ley de control de un PID es la siguiente [34]:

$$u = J \left(-k_p(q - q^*) - k_i \int (q - q^*) dt - k_d \left(\frac{\partial q}{\partial t} - \frac{\partial q^*}{\partial t} \right) \right) + B \frac{\partial q^*}{\partial t} + J \frac{\partial^2 q^*}{\partial t^2} \quad (24)$$

Donde q es la posición y dq y ddq , la velocidad y la aceleración, respectivamente. La notación q^* indica los parámetros de la trayectoria. Adicionalmente, para la calibración de las ganancias debemos calcular los parámetros ζ y ω_n , mediante las siguientes fórmulas:

$$\zeta = \frac{\log \left(\frac{M_p}{100} \right)}{\sqrt{\pi^2 + \left(\log \left(\frac{M_p}{100} \right) \right)^2}} \quad (25)$$

$$\omega_n = \frac{3}{\zeta t_s} \quad (26)$$

Donde t_s es el tiempo de asentamiento y M_p es el máximo sobre impulso.

La ganancia integral se define estocásticamente, mientras que la ganancia derivativa se define como $k_d = 2\zeta\omega_n$ y la ganancia proporcional como $k_p = \omega_n^2$ [35].

Los parámetros externos de la interpolación y del control PID se hicieron mediante máscaras para poder cambiar rápidamente los parámetros.

Para la validación del control de trayectorias, se usará el paquete de Simscape para simular la inercia y el amortiguamiento del tornillo sifín. Se utilizará una fuente de torque ideal como el motor y un sensor rotacional para medir la posición y la velocidad del eje.

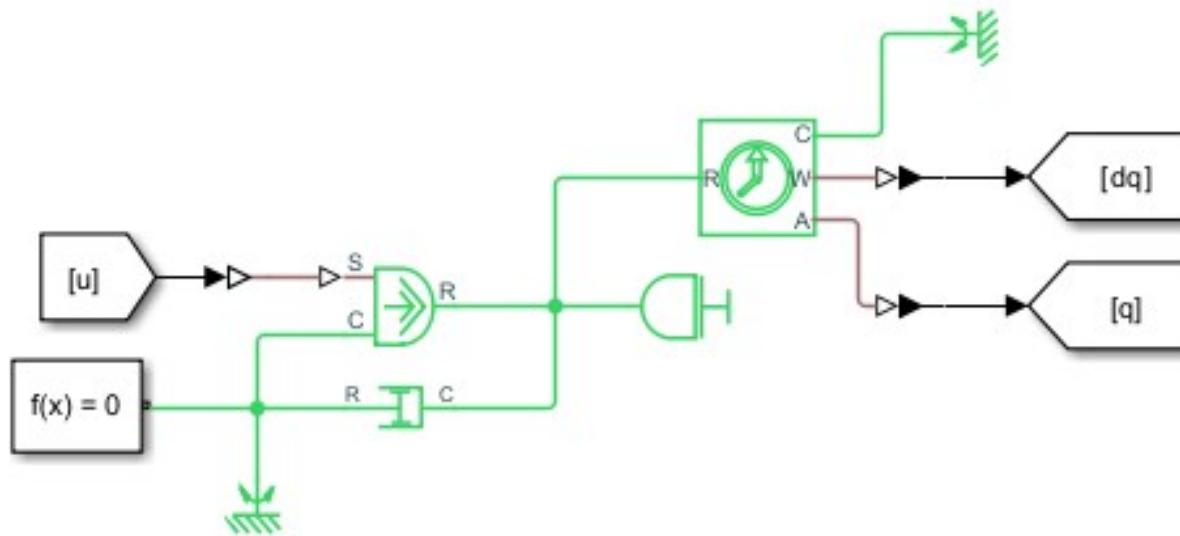


Figura 26: Validación mediante Simscape

Por último, se hizo la simulación para comprobar el seguimiento del sistema de la trayectoria planeada.

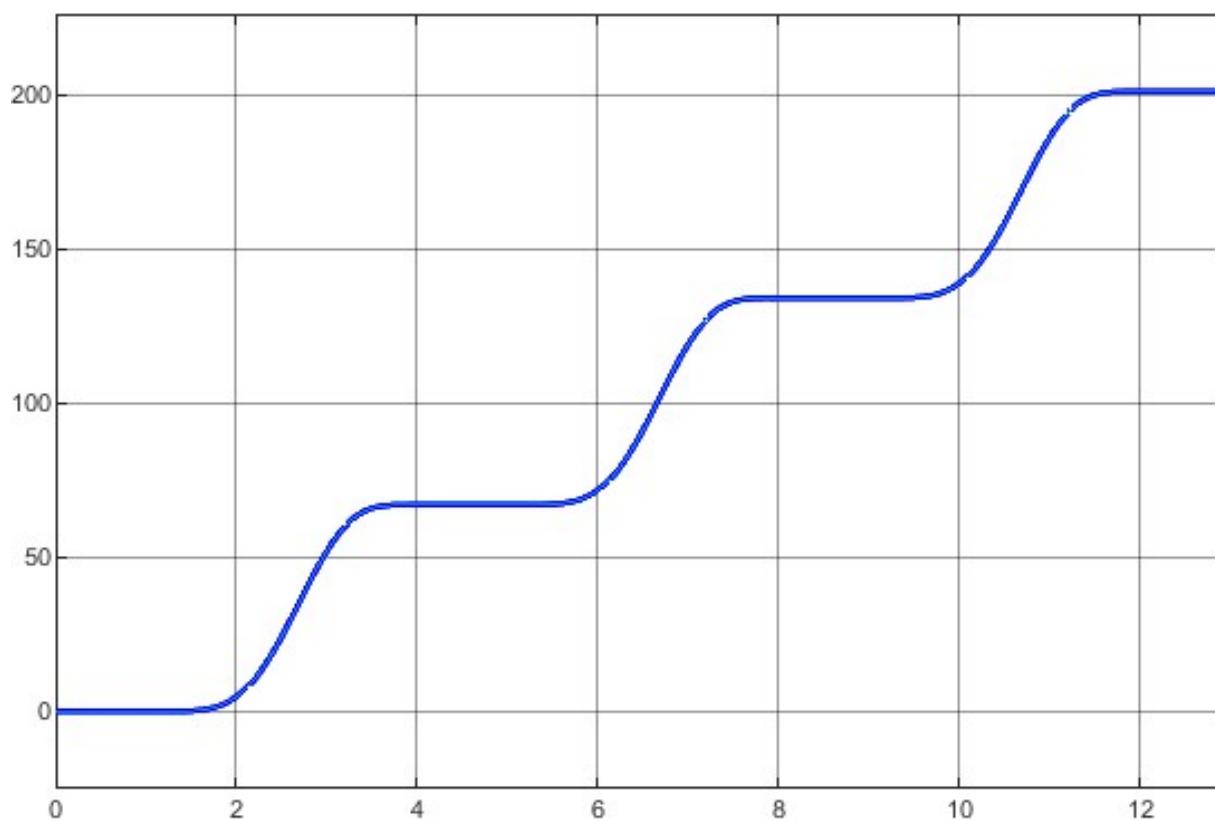


Figura 27: Trayectoria y movimiento simulado

Se realizó un código que simularía la lógica detrás de los disparos de los solenoides para desacoplar cada panel al llegar a la posición deseada.

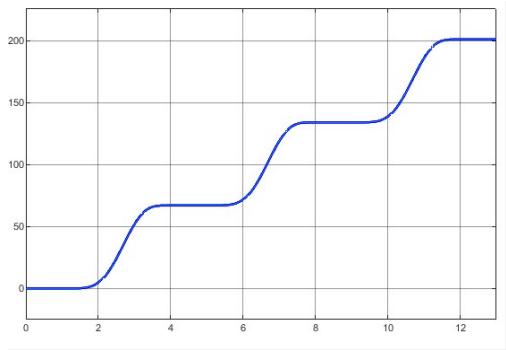
El código genera el vector de tiempo en función de los parámetros externos de la trayectoria para cada uno de los paneles y cuando se llega al tiempo de espera entre las trayectorias, el programa dispara los solenoides correspondientes a la posición en la que se encuentre (1,2,3). En el código, el vector *arrange* contiene las posiciones de cada uno de los paneles, y *Signals* es la matriz que contiene el estado en cada instante del tiempo de cada uno de los paneles.

```

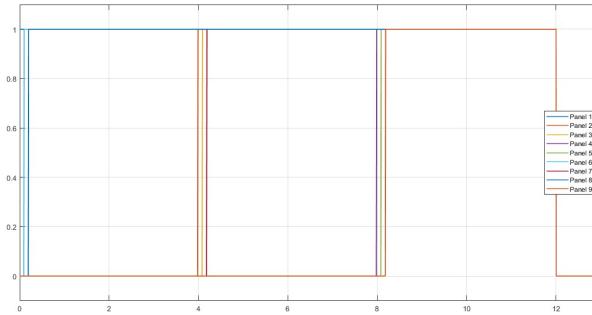
1 for i = 1:num_panels
2   if arrange(i) == 2
3     Signals(i, offset1+((time_of_motion+time_of_hold)/step):end-1/step) =
4       1;
5     offset1 = offset1 + 10;
6   elseif arrange(i) == 3
7     Signals(i, offset2+(2*time_of_motion+2*time_of_hold)/step:end-1/step) =
8       1;
9     offset2 = offset2 + 10;
10    elseif arrange(i) == 1
11      Signals(i, offset3:end-1/step) = 1;
12      offset3 = offset3 + 10;
13    end
14 end

```

Para exemplificar el funcionamiento tomemos el vector *arrange* como [1, 2, 3, 2, 1, 3]. Al iniciar la simulación se dispararán inmediatamente los solenoides de los paneles 1 y 5, y permanecerán activados por el resto de la simulación. Cuando pasen *tiempodemovimiento* + *tiempodeespera* segundos, el sistema ya habrá acabado el primer tercio de vuelta, en ese momento se activan los solenoides de los paneles 2 y 4. De igual manera, al haber pasado *2*tiempodemovimiento* + *2*tiempodeespera* segundos, el sistema habrá dado dos tercios de vuelta y activará los solenoides 3 y 6. Es importante notar que la trayectoria lleva a los paneles a la posición inicial (posición 1), y cuando ya se completó toda la vuelta, se desactivan todos los solenoides para llevar el sistema al estado de espera. Podemos ver una simulación de los disparos con las siguientes posiciones al lado de la trayectoria. *arrange* = [1, 2, 2, 3, 3, 1, 2, 1, 3]



(a) Trayectoria simulada



(b) Activaciones de los solenoides

Figura 28: Activación de los solenoides a lo largo de la trayectoria

Se puede observar que al menos en la simulación, lo único que rige el movimiento del tornillo sinfín es la inercia y el amortiguamiento, y, debido a que el controlador toma en cuenta la inercia efectiva y el amortiguamiento efectivo para el cálculo de la ley de control; el seguimiento presenta muy poco error. Además, al hacer un seguimiento de trayectoria en lugar de una referencia constante como en un impulso, se reduce en gran medida el sobre impulso y el tiempo de asentamiento. Se espera que en las pruebas físicas, se presente un error un poco mayor al previsto por la simulación debido al efecto de la masa del prisma triangular actuando como fuerza externa.

2.2.1.3 MF1.3 Interacciones del sistema

Este módulo se encarga de recibir las instrucciones de acondicionamiento emitidas por el usuario en la interfaz, así como de disponer la información necesaria para que pueda visualizarse en la interfaz, es donde los diferentes programas que componen al proyecto interactúan bajo la arquitectura descrita en la figura 29.

Protocolo de comunicación La integración entre el dispositivo físico seleccionado para el procesamiento (Raspberry Pi) y el PC de escritorio propio del usuario, en el que vivirá el programa de interfaz de usuario, se comunica mediante el protocolo de transferencia de

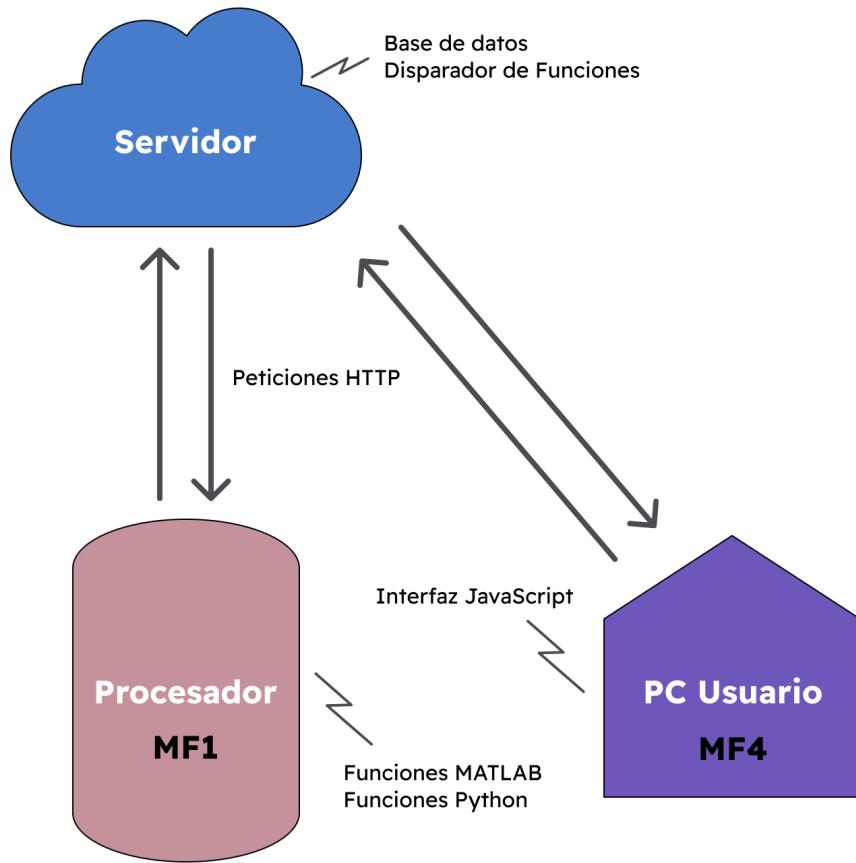


Figura 29: Arquitectura de comunicación MF1.3

hipertexto (HTTP). Considerando la oferta de protocolos disponibles, y sus distintas ventajas y desventajas, se presenta a continuación una tabla 81 de comparación que destaca las características clave de otros protocolos relevantes. Esta tabla sirve como referencia para evaluar la idoneidad de HTTP en comparación con alternativas como MQTT, CoAP, Web-Sockets y gRPC, proporcionando una visión general de las opciones disponibles y ayudando a respaldar la elección final del protocolo HTTP en el contexto específico de este proyecto.

Protocolo	Ventajas	Desventajas
MQTT [36] (Message Queuing Telemetry Transport)	Es un protocolo ligero diseñado para conexiones remotas con dispositivos con recursos limitados y redes de ancho de banda restringido. Ideal para redes de sensores y aplicaciones de telemetría debido a su bajo consumo de datos.	Menor compatibilidad con aplicaciones web estándar. Requiere un broker MQTT adicional para gestionar la comunicación.

CoAP [37] (Constrained Application Protocol)	Diseñado para dispositivos con recursos limitados en redes de baja potencia. Es similar a HTTP pero más ligero, utilizando UDP en lugar de TCP.	Menos soporte y herramientas en comparación con HTTP. No es tan ampliamente adoptado, lo que puede limitar la interoperabilidad.
WebSockets [38]	Permite comunicación bidireccional en tiempo real entre el cliente y el servidor. Es ideal para aplicaciones que requieren actualizaciones en tiempo real como chats y juegos en línea.	Puede ser más complejo de implementar y gestionar en comparación con HTTP. Requiere soporte tanto en el servidor como en el cliente para mantener la conexión activa.
gRPC [39] (gRPC Remote Procedure Calls)	Protocolo de alto rendimiento basado en HTTP/2 que permite la llamada a procedimientos remotos (RPC). Ideal para comunicaciones eficientes entre servicios.	Puede ser más complejo de implementar y requiere que ambos extremos (cliente y servidor) soporten gRPC. No es tan adecuado para aplicaciones con interfaces web estándar.

Tabla 81: Comparación de protocolos de comunicación

La elección de HTTP como protocolo de comunicación en este proyecto se debe los siguientes criterios:

- **Interoperabilidad:** HTTP es ampliamente compatible con servicios y aplicaciones web, lo que facilita la integración con otros sistemas y plataformas. Esto asegura que los diferentes componentes del sistema puedan comunicarse de manera efectiva sin problemas de compatibilidad.
- **Herramientas y Bibliotecas:** Existe una vasta cantidad de bibliotecas y herramientas disponibles para HTTP en Python, facilitando el desarrollo y la implementación. Esto permite a los desarrolladores aprovechar recursos existentes y reducir el tiempo de desarrollo.
- **Simplicidad y Conocimientos Previos:** HTTP es un protocolo conocido y ampliamente utilizado, lo que reduce la curva de aprendizaje y facilita la implementación y mantenimiento del sistema. Muchos desarrolladores ya están familiarizados con HTTP, lo que puede acelerar el proceso de desarrollo.
- **Seguridad y Escalabilidad:** HTTPS ofrece una capa de seguridad adicional mediante la encriptación de datos, protegiendo la información sensible durante la transmisión. Además, la integración con la nube proporciona escalabilidad y resiliencia, permitiendo que el sistema maneje un mayor volumen de datos y usuarios sin comprometer el rendimiento.
- **Ecosistema de Servicios:** Utilizar HTTP permite aprovechar una amplia gama de servicios en la nube y herramientas web. Esto incluye opciones para almacenamiento,

procesamiento de datos y otros servicios que pueden integrarse fácilmente en el sistema, ofreciendo una solución completa y robusta.

Tras el análisis, en comparación con los distintos protocolos disponibles, se seleccionó HTTP debido a su amplio soporte, interoperabilidad con tecnologías web estándar y su capacidad para facilitar la transferencia de datos de manera segura y eficiente entre el procesador y el PC del usuario [40], lo que contribuye significativamente a la implementación exitosa de nuestro sistema con las tecnologías previamente seleccionadas que son MATLAB, Python, JavaScript y a la simbiótica integración que tienen estas con una arquitectura ampliamente utilizada en la industria como es el uso de la nube.

Nube La integración de un servidor en nube desempeña un papel fundamental en la infraestructura y el funcionamiento integral de nuestro proyecto. Debido a la naturaleza del mismo, tenemos que contar con recepción, análisis y transmisión de datos entre el dispositivo Raspberry Pi y el usuario final en su interfaz gráfica, estos servicios requieren de un entorno robusto y escalable que la nube puede proporcionar de manera efectiva, rápida y alcanzable para nuestro tiempo de desarrollo [41].

En primer lugar, la nube actúa como un repositorio centralizado para almacenar los datos generados en una base de datos. Esto garantiza la disponibilidad y la integridad de los datos, además de facilitar su acceso y gestión desde cualquier ubicación. La base de datos permite el almacenamiento estructurado y eficiente de la información relevante para el usuario. Otro aspecto crucial es la capacidad para ejecutar funciones y servicios de forma remota, lo que permite una interacción dinámica entre el usuario y el dispositivo Raspberry Pi. A través de esta infraestructura, el usuario puede enviar instrucciones y solicitudes que viajan por internet, y activan acciones específicas en el dispositivo, como puede ser una consulta de los últimos datos de lectura, o iniciar un nuevo acondicionamiento.

La utilización de la nube, es de ayuda para garantizar la escalabilidad, seguridad e interactividad necesarias para ofrecer una solución completa y eficiente a nuestras necesidades [42]. La elección de un proveedor de nube adecuado se basa en los criterios de disponibilidad y confiabilidad en el servicio (uptime), ubicación del centro de datos, escalabilidad y flexibilidad, seguridad, servicios ofrecidos, costo. A continuación, se presenta una tabla de comparación 82 entre diferentes ofertas de proveedores de servicios en la nube de acuerdo a estos criterios que nos ayudaron en la selección de nuestro proveedor.

Proveedor de Nube	Uptime (%)	Ubicación del Centro de Datos	Escalabilidad y Flexibilidad
Amazon AWS [43]	99.95	Querétaro, México	Alta
Google Cloud Platform GCP [44]	99.95	Los Angeles, US	Alta
Microsoft Azure [45]	99.99	Querétaro, México	Alta
IBM Z Cloud [46]	99.95	Dallas, US	Media-Alta
Oracle Cloud [47]	99.95	Querétaro, México	Media-Alta

Digital Ocean [48]	99.99	San Francisco, US	Alta
--------------------	-------	-------------------	------

Tabla 82: Comparación de proveedores de nube según criterios

Y una comparación entre la disponibilidad de servicios 83.

Servicio	AWS	GCP	Azure	IBM Cloud	Oracle Cloud	Digital Ocean
Almacenamiento	Sí	Sí	Sí	Sí	Sí	Sí
Computación	Sí	Sí	Sí	Sí	Sí	Sí
Red	Sí	Sí	Sí	Sí	Sí	Sí
Bases de Datos	Sí	Sí	Sí	Sí	Sí	Sí
IA/ML	Sí	Sí	Sí	No	No	No

Tabla 83: Comparación de proveedores de nube según servicios [49]

Entre los proveedores revisados, Google Cloud Platform (GCP) se destacó como la mejor opción por varias razones, ya que nos ofrece un uptime del 99.95, lo que garantiza una alta disponibilidad, además, contando con el hecho de que si bien no cuenta con el centro de datos más cercano actualmente, se anunció el lanzamiento de uno en Querétaro en un futuro próximo, por lo que la latencia, incluso con el actual en Los Angeles no representa un problema. Otro aspecto importante es el sólido soporte técnico que ofrece Google Cloud y su documentación clara. Es importante mencionar también que encontramos en este proveedor una amplia gama de servicios que si bien para los alcances actuales del proyecto solo requerimos de la computación, almacenamiento y bases de datos, permiten escalar las limitaciones actuales con herramientas de inteligencia artificial y machine learning, visión computacional, IoT, entre otros posibles servicios actualmente disponibles. Por último, en temas de costos, GCP nos facilita como estudiantes créditos para el desarrollo, y un costo por uso accesible para un futuro despliegue al ámbito productivo. En conjunto, todos estos aspectos hacen que Google Cloud Platform sea la opción más sólida y completa para satisfacer nuestras necesidades de infraestructura en la nube para este proyecto.

Base de datos Para la selección de la base de datos, consideramos principalmente la facilidad de acceso a los datos, y la flexibilidad sobre el tiempo, teniendo en cuenta que actualmente existen dos grandes opciones disponibles, SQL y noSQL, realizamos una comparación 84 de las ventajas, desventajas y casos de uso documentados y utilizados actualmente en la industria para entender que opción se ajusta mejor a nuestro proyecto.

	SQL	NoSQL
Ventajas	Estructura definida, transacciones ACID, escalabilidad vertical	Esquema flexible, escalabilidad horizontal, alto rendimiento

Desventajas	Esquema rígido, escalabilidad horizontal limitada, menor flexibilidad en datos no estructurados	Menor consistencia en algunos modelos, menor soporte para transacciones complejas, menor adopción en algunos casos
Casos de Uso	Aplicaciones con integridad de datos y transacciones seguras, proyectos con estructura de datos predefinida	Aplicaciones web escalables, manejo de datos no estructurados, análisis de big data

Tabla 84: Comparación de Bases de Datos SQL y NoSQL [50]

Con esto, rápidamente entendemos que el uso de una base de datos no relacional (noSQL) nos proporciona lo que buscamos, para entonces ahora definir el uso de alguno de los productos de esta categoría agregamos un criterio más, que es la integración con nuestro proveedor de nube. Bajo esta condición seguimos encontrando una oferta amplia, desde bases de datos propias de Google hasta integraciones con terceros, entonces recurrimos a definir un último criterio de selección, que es el tipo de base de dato no relacional. En la siguiente tabla 85 podemos encontrar una comparación entre las diferentes opciones, todas con integración para CGP.

Base de Datos	Tipo	Descripción	Casos de uso
Bigtable [51]	Clave-valor	Servicio de bases de datos NoSQL de alto rendimiento y totalmente gestionado, pensado para grandes cargas de trabajo analíticas y operativas. Procesa picos de más de 5000 millones de solicitudes por segundo y puede gestionar más de 10 exabytes de datos.	Motores de recomendaciones Detección de fraudes
Firestore [52]	Documentos	Servicio de bases de datos de documentos altamente escalable y muy popular, diseñado para el desarrollo móvil, web y de servidores, que permite hacer consultas más rápidas y fructíferas.	Aplicaciones móviles, web y del Internet de las cosas Sincronización sin conexión

Memory store [53]	En memoria	Acceso a los datos en un tiempo inferior a un milisegundo con Redis y Memcached totalmente gestionados.	Almacenamiento en caché Chats de redes sociales o feeds de noticias
MongoDB [54]	Documentos	De código abierto que se destaca por su flexibilidad y escalabilidad. Su flexibilidad y capacidad para manejar datos no estructurados o semiestructurados lo hacen adecuado para proyectos donde la estructura de los datos puede cambiar con el tiempo o es difícil de modelar en un esquema relacional tradicional.	Aplicaciones móviles, web y del Internet de las cosas Gestión de contenido

Tabla 85: Comparación de bases de datos compatibles con GCP y noSQL

La decisión entre Firebase (Firestore) y MongoDB para el proyecto generó una discusión en torno a varios factores. Si bien ambas opciones son altamente escalables y populares en el desarrollo IoT, lo cual las hacen excelentes candidatas, MongoDB fue elegida como la mejor opción por las siguientes razones. En primer lugar, MongoDB tiene una mejor integración con Python, que es uno de los lenguajes de programación utilizados en el proyecto, esta estrecha integración facilita el desarrollo y la gestión de la base de datos dentro de nuestro entorno de trabajo. Además, MongoDB es una base de datos de código abierto, lo que nos da la confianza de uso por miles de usuarios en temas de seguridad y dirección de la tecnología a futuro.

Marco de trabajo para la conexión Es necesario un programa que haga la gestión de las funciones de cálculo y procesamiento, que realice las distintas operaciones en base de datos, y exponga los puntos de acceso para las solicitudes HTTP. Es decir, una pieza de software que integre las herramientas mencionadas hasta el momento en este módulo, este programa es conocido como el backend del proyecto [55].

Características	Django [56]	Flask [57]	FastAPI [58]	Spring Boot [59]
-----------------	-------------	------------	--------------	------------------

Estructura y Modularidad	Estructura completa, incluye ORM y autenticación	Microframework, altamente flexible	API-first, basado en OpenAPI	Framework completo, incluye muchas herramientas
Facilidad de Gestión de CRUD	Muy fácil, incluye un ORM potente	Necesita extensiones como SQLAlchemy	Rápido y sencillo con pydantic y SQLAlchemy	Incluye JPA y Hibernate para gestión de datos
Integración con Servicios en la Nube	Amplia integración con proveedores como AWS, GCP	Depende de extensiones y bibliotecas adicionales	Integración nativa con muchos servicios en la nube	Alta integración con servicios de nube empresarial
Seguridad	Incluye muchas características de seguridad por defecto	Depende del desarrollador implementar seguridad	Incluye características de seguridad, pero personalizables	Incluye muchas características de seguridad
Escalabilidad	Buena, adecuada para proyectos grandes	Mejor para proyectos pequeños a medianos	Alta escalabilidad gracias a su diseño asíncrono	Muy alta, adecuada para proyectos empresariales
Velocidad de Desarrollo	Rápido gracias a muchas herramientas integradas	Rápido para proyectos pequeños	Muy rápido para desarrollar APIs	Moderado, depende del tamaño del proyecto
Documentación y Soporte	Extensa documentación y gran comunidad	Buena documentación, comunidad activa	Buena documentación, creciendo en popularidad	Extensa documentación y soporte empresarial
Comunidad y Ecosistema	Muy grande, con muchas bibliotecas y herramientas	Grande, con muchas extensiones disponibles	Creciendo rápidamente, buena comunidad	Muy grande, con muchas herramientas empresariales
Herramientas y Bibliotecas Predefinidas	Incluye muchas bibliotecas útiles por defecto	Minimalista, depende de extensiones	Incluye herramientas para desarrollo rápido de APIs	Incluye muchas bibliotecas y herramientas

Tabla 86: Comparación de Frameworks para Desarrollo Backend

En Python existen diversos marcos de trabajo con los cuales se puede desarrollar herramientas como esta, siendo Django nuestra mejor opción debido a su robustez y capacidad para manejar múltiples tareas de manera eficiente. Django ofrece una estructura clara y modular que facilita la gestión de la ejecución de códigos existentes, la realización de operaciones CRUD (crear, leer, actualizar, eliminar) en la base de datos, y esto viviendo desde la

nube de GCP. Además, Django proporciona una gran cantidad de herramientas y bibliotecas predefinidas que simplifican la creación de endpoints (puntos de acceso) HTTP para la comunicación con la interfaz de usuario y la Raspberry Pi. Comparado con otras alternativas como Flask o FastAPI, Django destaca por su enfoque en la seguridad, la escalabilidad y la rápida velocidad de desarrollo, haciendo que sea una solución ideal para un proyecto que requiere una integración compleja y una gestión eficiente de recursos.

2.2.2. MF2. Módulo de generación y medición de la acústica

Para este módulo, se hizo uso del entorno de MATLAB. El programa consiste en un dispositivo de reproducción y grabación simultánea, que envían una señal de excitación hacia el cuarto por medio de un altavoz, mientras un micrófono recoge tanto el sonido del altavoz como las reflexiones de las superficies del recinto. Posteriormente, el programa se encarga de utilizar la señal de excitación y la capturada, para estimar la respuesta al impulso del estudio.

Dentro del *Audio Toolbox* de MATLAB [60] existe el objeto *AudioPlayerRecorder*, que nos permite escribir y leer paquetes de audio al dispositivo de la computadora.

Las primeras líneas del código están referidas a la inicialización y configuración del dispositivo de lectura y escritura.

```

1 clc; clear; close all;
2
3 fs = 44100;
4 room_dim = [6, 5, 3.5];
5 device = "ASIO4ALL v2";
6 aPR = audioPlayerRecorder("SampleRate", fs, "Device", device);

```

El dispositivo “ASIO4ALL v2”[61] es un *driver* para Windows, que nos permite utilizar los dispositivos de audio conectados a la computadora como tarjetas ASIO (*Audio Stream Input-Output*) con la posibilidad de combinar dispositivos de entrada y salida.

Existen distintos métodos para medir la respuesta al impulso de un recinto, sin embargo, por la poca calibración que necesita para obtener resultados óptimos, además de que permite obtener un excelente ratio entre señal y sonido, el mejor método es el barrido senoidal. [62]

El barrido senoidal consiste en una señal senoidal, que aumenta de frecuencia conforme avanza el tiempo. El dispositivo ASIO emite la señal mientras va leyendo la respuesta del recinto. Se puede estimar la respuesta al impulso al realizar una deconvolución entre la señal emitida y la capturada[63]. El siguiente paso en el código es definir los parámetros del barrido senoidal, como la amplitud, rango y duración.

```

1 sweep_dur = 5;
2 duration_per_run = 8;
3 start_silence = 1;
4 silence_dur = duration_per_run - sweep_dur - start_silence;
5 sweep_range = [10 8000];
6 percentage = 100;

```

```

7 outputLevel = 20*log10(percentage/100);
8 sweepsine = sweepTone(sweep_dur,silence_dur,fs,"SweepFrequencyRange", ...
9     sweep_range,"ExcitationLevel",outputLevel);

```

Para asegurar la simultaneidad de las señales de entrada es salida, es necesario utilizar *buffers* que se encargaran de gestionar el flujo de datos. Además, se incluyen tiempos de silencio al antes y después de la señal, que permita compensar por pequeños desfases.

```

1 excitation = [zeros(start_silence*fs,1); sweepsine];
2 sampling = 1024;
3 excitation_length = length(excitation);
4 bufOut = dsp.AsyncBuffer(excitation_length);
5 bufIn = dsp.AsyncBuffer(excitation_length);
6 write(bufOut, excitation);

```

Se leerán y escribirán datos a la tarjeta de audio mediante paquetes de 1024 muestras, hasta que se acaben los datos del *buffer*.

```

1 while bufOut.NumUnreadSamples > 0
2     exci = read(bufOut, sampling);
3     [rec, num_under, num_over] = aPR(exci);
4     write(bufIn, rec);
5     if num_under > 0 || num_over > 0
6         fprintf("Underrun by %d frames, overrun by %d frames.\n", num_under,
7             num_over)
8     end
9 end
10 release(aPR);
11 read(bufIn, start_silence*fs);
12 audioFromDevice = read(bufIn);

```

Por último, se hace el proceso de deconvolución mediante la función de matlab *impzest*, se normaliza para tener amplitud de 1 y se crea el vector de tiempo. Los datos de la respuesta al impulso se guardan en una estructura junto con el tiempo de muestreo f_s .

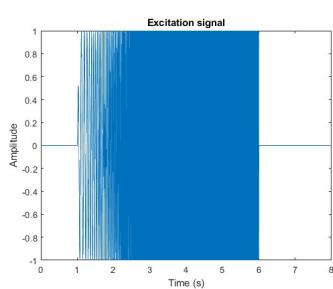
```

1 time = (1:length(audioFromDevice))/fs;
2 RIR = impzest(sweepsine, audioFromDevice);
3 [max_RIR, Idx] = max(abs(RIR));
4 n_RIR = RIR(Idx-round(fs/100):end)/max_RIR;
5 time_RIR = (1:length(n_RIR))/fs;
6 ImpulseResponse.fs = fs;
7 ImpulseResponse.y = n_RIR;
8
9 plot(time, audioFromDevice)
10 title('Audio recorded')
11 xlabel('Time(s)')
12 ylabel('Amplitude')
13
14 plot(time_RIR, n_RIR)
15 title('Estimated impulse response')
16 xlabel('Time(s)')

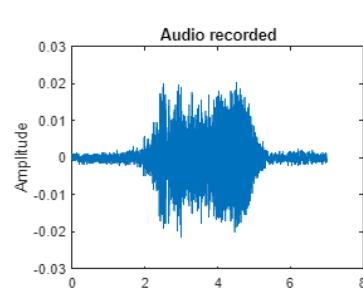
```

```
17| ylabel('Amplitude')
```

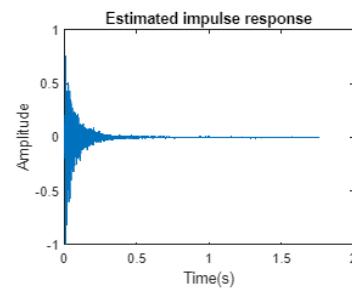
Las gráficas resultantes muestran la señal de excitación, la grabación y la estimación de la respuesta al impulso.



(a) Señal de excitación (Barrido senoidal)



(b) Audio grabado



(c) Respuesta al impulso estimada

Figura 30: Gráficas resultantes de generación y medición de la acústica

El programa completo se encuentra como en los apéndices.

2.2.3. MF3. Módulo modificador de la acústica

2.2.3.1 Paneles de Absorción

Los paneles de absorción son utilizados para controlar la reverberación dentro de espacios cerrados y confinados. Su función es la de transformación superficies reflejantes en superficies absorbentes con el fin de controlar la energía del sonido proveniente de las reflexiones que llega al receptor [64].

La elección de nuestros paneles de absorción dependerá en gran medida del coeficiente de absorción que se necesitará para poder hacer modificaciones medibles al tiempo de reverberación. Para este propósito se utilizó el programa *PanelPositions.py* que hace la relación entre la absorción efectiva y el tiempo de reverberación.

```
1| x = PanelsToChange(1, 0.7, room_dim)
```

Por ejemplo, para reducir el tiempo de reverberación de 1 segundo a 0.7 segundos, un coeficiente de absorción de 0.9 a 500 Hz, da un total de 16 paneles. En cambio, los mismos parámetros, pero con un coeficiente de absorción de 0.6 da un total de 24 paneles.

Este ejemplo nos ayuda a mostrar el efecto de la calidad de los paneles de absorción, la cantidad que se necesitarán de ellos. Debido a que el costo de incluir un panel más es mayor al costo de adquirir un panel de mejor calidad, ya que implica la instalación del sistema de movimiento extra, además del aumento en las dimensiones del miembro estructural, flecha y potencia del motor; se decidió optar por paneles de mejor calidad, que nos permitan usar menor infraestructura.

En la base de datos del módulo de *Python* para acústica, podemos encontrar los coeficientes de absorción de diversos materiales comúnmente encontrados en recintos acústicos [65]. Entre ellos, en la sección de paneles de absorción de pared, podemos encontrar coeficientes que van desde 0.2, hasta 1. Los que tienen el coeficiente de absorción más alto son los paneles cubiertos con tela con un núcleo de lana mineral y los plafones llenos de lana mineral. Ya que los plafones no se ajustan a nuestra implementación en las paredes, se decidió optar por los paneles cubiertos, específicamente los de 6 pcf (libras por pie cuadrado), ya que son menos densos que los 8 pcf, y, por tanto, tienen coeficientes de absorción mayores.

Wall absorbers

keyword	descrip-tion	125 Hz	250 Hz	500 Hz	1 kHz	2 kHz	4 kHz	8 kHz
panel_fabric_covered_6pcf	Fabric-covered panel, 6 pcf rock-wool core	0.46	0.93	1	1	1	1	1.00

Figura 31: Coeficientes de absorción de los paneles recubiertos

2.2.3.2 Paneles de reflexión

De la misma base de datos utilizada para los paneles absorbentes, se puede elegir el material para los paneles de reflexión.

En este caso, hay muchos materiales con coeficientes de absorción muy bajos que serían excelentes paneles de reflexión, sin embargo, se debe tener en cuenta, y como se ve en la figura 31, los paneles suelen trabajar mal a frecuencias bajas.

Debido a la longitud de onda de las frecuencias más bajas, las geometrías y densidades de los paneles acústicos rara vez son suficientes para absorberlas [66]. Por tanto, es necesario que los paneles de reflexión presentes, contribuyan a la absorción de estas frecuencias.

De los materiales más comunes que pueden ser fácilmente montados, encontramos la madera contrachapada, la cual tiene coeficientes de absorción muy bajos a medias y altas frecuencias, pero un coeficiente de absorción de 0.46 a bajas frecuencias. Sumado a su bajo costo y peso, se decidió que era el material más conveniente para nuestra aplicación.

2.2.3.3 Paneles de difracción

Los paneles de difracción están caracterizados por su capacidad de reflejar las ondas de sonido en muchas direcciones, lo que permite una mejor distribución de la energía acústica en el cuarto. Los difusores permiten que los oyentes reciban muchas más reflexiones de menor energía y que, por tanto, tengan una mayor sensación de profundidad y de volumen. [67]

Existen diversos tipos de difusores, a continuación se listan los más importantes [68]:

- Difusores de tipo *Skyline*. Consta de agrupaciones de bloques a diferentes alturas que recuerdan a los rascacielos de una ciudad (De ahí su nombre). Las diferentes alturas permiten que las ondas reboten en diferentes instantes del tiempo, además, la diferencia en la dirección de las superficies permite que se reflejen hacia diversas direcciones.
- Difusores de residuo cuadráticos. Consta de diferentes canales que corren paralelos en el panel, a diferentes profundidades. Las profundidades no son aleatorias, sino que están diseñadas para ser eficientes a ciertas frecuencias.
- Difusores de residuo cuadráticos en 2D. Son muy similares a los de tipo *Skyline* con la diferencia que están cuidadosamente planificados para ser eficientes a ciertas frecuencias, además, los bloques se separan entre ellos.
- Difusores de barril. Se asemejan a un cilindro cortado a la mitad, y se colocan convexos al cuarto. Su geometría permite reflejos especulares que varían de acuerdo al punto de impacto. Se diferencian del resto por su capacidad de albergar materiales absorbentes en el interior, lo que permite absorber frecuencias que no se difractan.
- Difusores piramidales. Constan de pirámides cuadrangulares de diferentes alturas, lo que modifica los ángulos de incidencia.

Para nuestra aplicación, se decidió utilizar los Difusores de residuo cuadrático (también llamados *Schorroeder Diffusers* o QRD), ya que, además de permitirnos atacar ciertas frecuencias importantes para nosotros, también son fáciles de manufacturar.

La sección transversal de un panel QRD es la siguiente [67]:

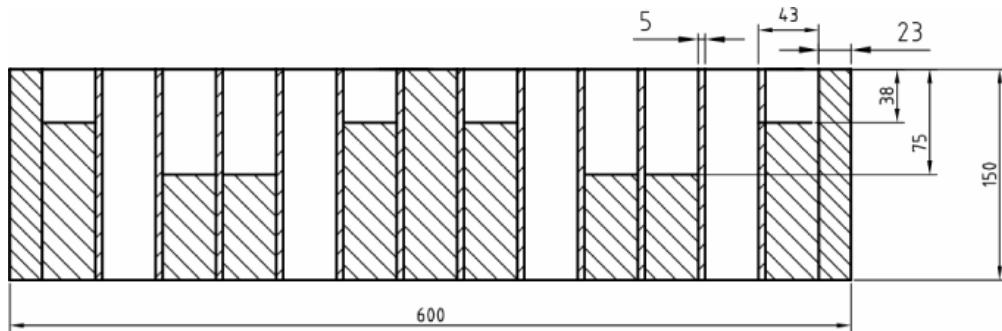


Figura 32: Sección transversal de un panel QRD

La manufactura constará de una operación con la fresadora para realizar cada uno de los canales a partir de un bloque sólido, la altura de la fresadora se ajustará para las diferentes profundidades.

Los paneles QRD pueden colocarse paralelos o rotados 90 grados uno respecto del otro. Si se colocan todos paralelos, la difracción de las ondas ocurrirá solamente en el eje perpendicular a la dirección de los canales. La rotación de los difusores QRD permite que las ondas sean difractadas en ambos ejes.

Selección método de movimiento Para realizar la modificación de la acústica en el estudio se debe de cambiar la superficie que interactúa con las ondas de sonido generadas, para poder realizar el cambio de las superficies debemos de girar los prismas triangulares de nuestro trabajo terminal, para lo cual debemos de trasmitir dicho movimiento desde un motor hasta dichos prismas. Existen varias formas de transmitir dichos movimientos, en la Tabla 87 se pueden

Tipo	Descripción	Ventajas	Desventajas
Bandas y poleas	Utilizar poleas y bandas dentadas para poder realizar el movimiento giratorio de los prismas triangulares	<ul style="list-style-type: none"> ■ Ruido mínimo ■ Vibraciones mínimas ■ Movimiento preciso con una sincronización exacta ■ Resistencia a la abrasión, al óxido, productos químicos y contaminantes. 	<ul style="list-style-type: none"> ■ Se necesitan bandas de largas dimensiones ■ Mayor costo ■ Ideal para transferir a una potencia relativamente baja ■ La potencia de transferencia está a una distancia relativamente menor en comparación con otras bandas de transmisión
Tornillo sin fin y corona	Utilizar un tornillo sin fin que mueva coronas, las cuales están acopladas a los prismas triangulares	<ul style="list-style-type: none"> ■ Elevada capacidad de carga ■ Ruido mínimo ■ Movimiento preciso con una sincronización exacta ■ Compacto 	<ul style="list-style-type: none"> ■ Bajo rendimiento en las etapas de reducción ■ Requiere mantenimiento periódico debido a que sufren desgaste por fricción ■ Costo de mantenimiento elevado si se llegara a requerir reparaciones.

Movimiento individual	Colocar un motor por cada uno de los prismas triangulares para controlarlos individualmente	<ul style="list-style-type: none"> ■ Control personalizado a cada prisma sin la necesidad de ningún otro mecanismo ■ Movimiento preciso con una sincronización exacta 	<ul style="list-style-type: none"> ■ Elevado costo debido a que se necesitan comprar una gran cantidad de motores ■ Mayor consumo energético comparado con las otras opciones.
-----------------------	---	---	--

Tabla 87: Ventajas y desventajas de los distintos tipos de transmisión de movimiento

Después de observar las ventajas y desventajas de cada uno de los métodos de transmisión de movimiento nos decidimos por la transmisión por medio de un tornillo sinfín y corona debido a que se tiene un buen control de la posición de los prismas que portan los paneles, tiene una gran capacidad de carga, además de que el ruido producido es bajo. Al elegir este tipo de transmisión surgieron problemas en cuanto las dimensiones del tornillo sin fin, ya que si optábamos por utilizar un solo tornillo sin fin para mover todos los prismas este debía de ser de un largo bastante considerable, y realizar la fabricación de este sería una tarea difícil, por lo que en la Tabla 88 se comparan las opciones que se tienen para realizar la transmisión de tornillo sin fin y corona.

Tipo	Descripción	Ventajas	Desventajas
Un solo tornillo	Utilizar un tornillo para mover todos los prismas triangulares	<ul style="list-style-type: none"> ■ No se necesita ningún acople o unión para mover todos los paneles utilizando un solo motor 	<ul style="list-style-type: none"> ■ Difícil fabricación de un tornillo sin fin de esas dimensiones ■ Posible flexión debido a las dimensiones ■ Posible torsión debido a las dimensiones ■ Elevado costo de fabricación
Un tornillo por cada dos prismas triangulares	Utilizar tornillos sin fin de dimensiones más pequeñas para que cada uno de estos tornillos mueva dos prismas triangulares	<ul style="list-style-type: none"> ■ Fácil de fabricar ■ Menor margen de error al momento de fabricar ■ Costos de fabricación bajos 	<ul style="list-style-type: none"> ■ Se necesitan acoples para poder realizar el movimiento de todos los paneles utilizando un solo motor ■ Número elevado de tornillos sin fin que se deben de fabricar

Un tornillo por cada tres prismas triangulares	Utilizar tornillos sin fin de dimensiones más pequeñas para que cada uno de estos tornillos mueva tres prismas triangulares	<ul style="list-style-type: none"> ■ Fácil de fabricar ■ Menor margen de error al momento de fabricar ■ Costos de fabricación bajos 	<ul style="list-style-type: none"> ■ Se necesitan acoplos para poder realizar el movimiento de todos los paneles utilizando un solo motor
--	---	--	--

Tabla 88: Ventajas y desventajas relacionadas con la utilización de un tornillo sin fin único o varios segmentos de tornillo sin fin

La mejor opción es la de utilizar un tornillo sin fin de dimensiones pequeñas, el cual esté encargado de mover tres prismas triangulares. Para poder utilizar un solo motor para mover los paneles se optó por utilizar acoplos entre cada unos de los tornillos sin fin, el problema que se tiene ahora es como realizar las combinaciones necesarias para poder cambiar los parámetros acústicos como se desea, para ellos se optaron por tres opciones para girar los prismas independientemente uno de otro, estas opciones se pueden observar en la Tabla.

Tipo	Descripción	Ventajas	Desventajas
Combinación binaria	Hacer que un prisma gire una vuelta completa cada vez que el prisma anterior de tres vueltas	<ul style="list-style-type: none"> ■ No hay necesidad de desacoplar ninguna parte del mecanismo ■ Menor cantidad de espacio ocupado 	<ul style="list-style-type: none"> ■ Giros excesivos de los primeros prismas para poder mover los últimos ■ Larga duración del proceso para obtener la configuración necesaria en los paneles ■ Desgaste excesivo en algunos componentes
Utilización de guías lineales	Utilizar guías lineales para acoplar y desacoplar los tornillos, la corona y el prisma triangular del tornillo sin fin	<ul style="list-style-type: none"> ■ Corta duración del proceso para obtener la configuración necesaria en los paneles 	<ul style="list-style-type: none"> ■ Costo elevado en la construcción de las guías lineales. ■ Posible desgaste de los dientes del tornillo sin fin y la corona al momento de acoplar y desacoplar ■ Mayores en el diseño.

Utilizar un acople lineal	Acoplar y desacoplar el prisma triangular de la corona sin fin por medio de un acople.	<ul style="list-style-type: none"> ■ Corta duración del proceso para obtener la configuración necesaria en los paneles ■ Menor costo de fabricación ■ Menores dimensiones en el diseño 	<ul style="list-style-type: none"> ■ Complicado de fabricar
---------------------------	--	---	--

Tabla 89: Ventajas y desventajas relacionadas con los diferentes métodos para obtener la configuración deseada en los paneles

2.2.3.4 Sistema de transmisión

Habiendo elegido el tipo de transmisión que genera la rotación del prisma triangular y los demás elementos que conforman el sistema de transmisión, se desarrolla el cálculo de cada uno de los componentes asociados a ella. De acuerdo con Budynas y Keith[69], no existe una secuencia precisa de pasos para algún proceso de diseño, Por su naturaleza, el diseño es un proceso iterativo en el que es necesario realizar algunas selecciones tentativas y construir un esquema previo para determinar las partes cruciales del mismo.

Potencia del motor

Siguiendo la secuencia de diseño para transmisiones de potencia [69], como primer paso es conocer los requisitos de potencia y par de torsión, ya que determinará las necesidades globales de dimensionamiento de todo el sistema.

Los parámetros conocidos son la velocidad angular y aceleración angular de la corona, estos parámetros vienen dados por las necesidades de la trayectoria propuesta. En este caso $\omega = 2,094 \frac{\text{rad}}{\text{s}}$ y $\alpha = 2,094 \frac{\text{rad}}{\text{s}^2}$.

El momento necesario para romper la inercia estática del ensamblaje de la corona es:

$$M_g = I \cdot \alpha = 0,02615532 \cdot 2,094 = 0,05477 \text{Nm} \quad (27)$$

El momento de inercia se obtuvo del diseño CAD.

A continuación se puede calcular la potencia necesaria con base en la siguiente formula:

$$P = M \cdot \omega = 0,05477 \cdot 2,094 = 0,1147 \text{W} \quad (28)$$

que es la potencia para mover un solo ensamblaje del panel.

Se puede hacer lo mismo para el ensamblaje del tornillo sinfín.

$$Mg = 0,00189 \frac{\text{kg}}{\text{m}^2} \cdot 67,02 \frac{\text{rad}}{\text{s}^2} = 0,127 \text{Nm} \quad (29)$$

$$P = 0,127 \text{Nm} \cdot 67,02 \frac{\text{rad}}{\text{s}} = 8,5 \text{W} \quad (30)$$

De SolidWorks se hizo un estudio de movimiento en el que se validaron estos parámetros.

- $T_{panel} = 0,05477 \text{Nm}$
- $P_{panel} = 0,11 \text{W}$
- $T_{tornillo} = 0,127 \text{Nm}$
- $P_{tornillo} = 8,2 \text{W}$

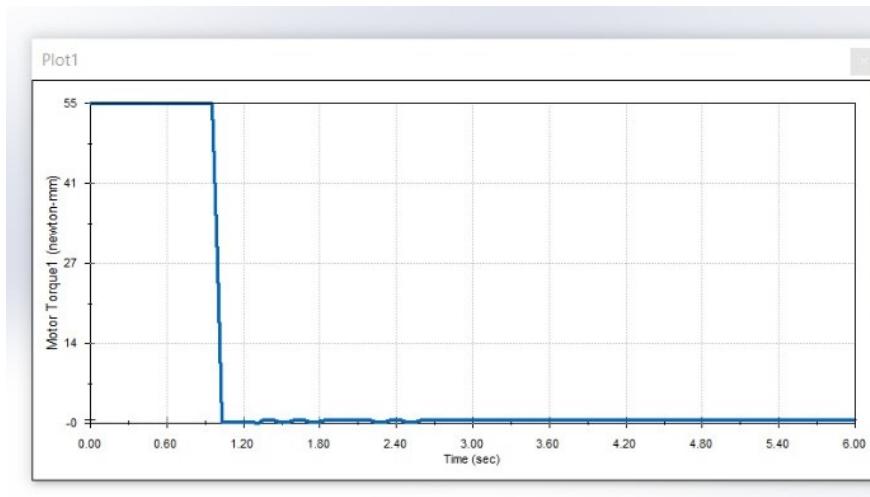


Figura 33: Validación torque para un panel

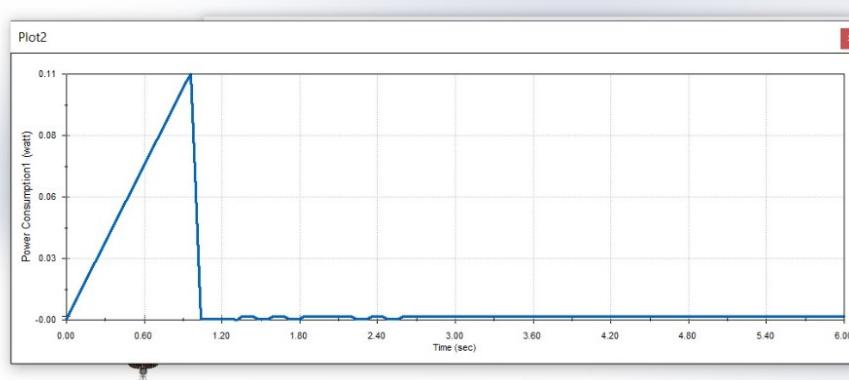


Figura 34: Validación potencia para un panel

Se hizo la suma del torque necesario para mover un ensamblaje de panel y de tornillo

$$T_{ensamblaje} = 0,127 \text{Nm} + 0,05477 \text{Nm} = 0,171 \text{Nm} \quad (31)$$

se utilizara este torque para hacer el calculo del torque necesario en el motor y de la potencia del mismo. Ya que existen nueve ensamblajes la potencia total por 9, $T_{total} = 9 \cdot 0,171 \text{Nm} =$

$1,539Nm$.

Por ultimo se hizo el calculo de la potencia

$$P = T \cdot \omega = 1,539Nm \cdot 67,02 \frac{rad}{s} = 103,14W \quad (32)$$

La validación se encuentra en la figura 35, en donde se muestran el torque y la potencia para un ensamblaje, si se multiplican los resultados de la gráfica por los 9 ensamblajes resultan en los valores calculados totales.

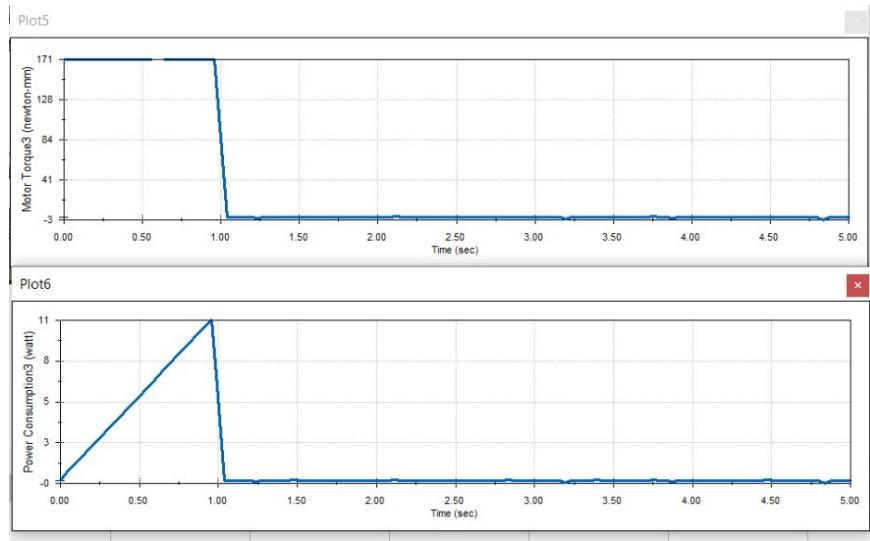


Figura 35: Validación torque y potencia para un ensamblaje

Motor

Al obtener la potencia necesaria para el correcto funcionamiento del sistema se procede a hacer la selección del motor.

Con base en el sistema de control se hizo notar que el motor debía ser preferiblemente de CC, ya que si fuera de CA, se requeriría el uso de un variador de frecuencia para el control de la velocidad, los cuales son costosos y difíciles de manejar electrónicamente.

El motor finalmente elegido fue el motor de CC de la marca Yuntaike [70], debido a su bajo costo y a que cumple con bastante seguridad los requisitos de funcionamiento. Es importante notar que el motor es de 300 W, lo cual es mayor a los 100 W calculados, esto para tener un factor de seguridad de al menos 3, y que pueda ser capaz de seguir moviendo el sistema aun si se llegan a hacer cambios con los paneles y el ensamblaje. Por su parte, el motor solo es de $0,833Nm$, lo cual es menor al torque necesario. Sin embargo, y como ya estaba contemplado anteriormente, el motor es de altas revoluciones, y el sistema no requiere de una velocidad angular tan alta, lo que nos permite colocar una relación de transmisión mediante una catarina para reducir la velocidad y aumentar el torque. Ya que el torque necesario es de $1,539Nm$, se puede utilizar una relación de transmisión de 1 : 2 lo que aumentaría el torque del motor a $1,666Nm$, suficiente para mover el sistema, mientras que la velocidad angular baja de $1680rpm$ (bajo carga) a $890rpm$, que siguen siendo mayores a las $280rpm$ necesarias

para cumplir con la trayectoria.

Esta configuración permite además, el cambio en la relación de transmisión, en caso que los parámetros reales no se ajusten a los calculados o se tengan que hacer ajustes o cambios en los componentes estructurales y sus materiales.

Corona-Sinfín

Con la finalidad de generar el movimiento de rotación de los paneles, se optó por añadir un mecanismo de corona-sinfín con los siguientes requerimientos. Para el diseño se consideró una velocidad en la corona (acoplada al eje de los paneles) n_g de $20 \frac{\text{rev}}{\text{min}}$, la velocidad nominal del eje del tornillo sinfín n_w de $640 \frac{\text{rev}}{\text{min}}$, el número de entradas del tornillo $N_w = 1$, el módulo $m = 3$ y un ángulo de presión $\phi_n = 20^\circ$.

Cálculo de la geometría del tornillo sinfín y de la corona

Para las ecuaciones que se usan a continuación se sigue la metodología que utiliza Hamrock en su libro. [71] Al tener la velocidad angular de entrada se obtiene la relación de transmisión.

$$Z = \frac{n_w}{n_g} \quad (33)$$

$$Z = \frac{640}{20} = 32$$

Por lo tanto, el número de dientes de la corona está dado por

$$N_g = Z \cdot N_w \quad (34)$$

$$N_g = 32 \cdot 1 = 32$$

Por lo cual se corrobora que el número de dientes esté dentro del mínimo que es permitido por el ángulo de presión, como lo muestra la Tabla 90.

Ángulo de presión ϕ_n	Número mínimo de dientes en la corona
14.5	40
17.5	27
20	21
22.5	17
25	14
27.5	12
30	10

Tabla 90: Número mínimo sugerido de dientes de la corona para diseños habituales ϕ_n [71]

Luego para obtener el paso circular P_c , el paso base P_b y el paso diametral P_d se utilizan las siguientes fórmulas

$$P_c = \pi \cdot m \quad (35)$$

$$P_c = 3\pi = 9,42mm$$

$$P_b = P_c \cos \phi_n \quad (36)$$

$$P_b = 3\pi \cos 20^\circ = 8,86mm$$

$$P_d = \frac{1}{m} \quad (37)$$

$$P_d = \frac{1}{3} = 0,33mm^{-1} = 333m^{-1}$$

Una parte importante del diseño del mecanismo es la determinación de la geometría del diente, por lo que se calcula mediante las siguientes fórmulas. [71]

$$a = \frac{P_b}{\pi} \quad (38)$$

$$a = \frac{8,86}{\pi} = 2,82mm$$

Donde a es el Addendum.

El Dedendum está dado por [71]

$$b = \begin{cases} \frac{1,157P_b}{\pi} & P_b > 4,06mm \\ \frac{1,200P_b}{\pi} & P_b < 4,06mm \end{cases} \quad (39)$$

Por lo que al P_b ser mayor que $4.06mm$, se tiene que el Dedendum es igual a

$$b = \frac{1,157 \cdot P_b}{\pi} \quad (40)$$

$$b = \frac{1,157(8,86)}{\pi} = 3,26mm$$

Para calcular la altura de trabajo 41 y la altura total del diente 42 es mediante las siguientes fórmulas

$$h_k = 2 \cdot a \quad (41)$$

$$h_k = 2(2,82) = 5,64mm$$

$$h_t = a + b \quad (42)$$

$$h_t = 2,82 + 3,26 = 6,1mm$$

Teniendo la geometría del diente se procede a hacer el cálculo de las dimensiones de la corona, empezando con el diámetro de paso de la corona d_g .

$$d_g = m \cdot N_g \quad (43)$$

$$d_g = 3(32) = 96mm$$

Diámetro exterior de la corona d_{og} .

$$d_{og} = d_g + 2a \quad (44)$$

$$d_{og} = 96 + 2(2,82) = 101,64mm$$

Diámetro de raíz de la corona d_{rg} .

$$d_{og} = d_g - 2b \quad (45)$$

$$d_{og} = 96 - 2(3,26) = 89,48mm$$

Paso axial de la corona P_x

$$P_x = \frac{\pi \cdot d_g}{N_g} \quad (46)$$

$$P_x = \frac{\pi \cdot 96}{32} = 3\pi$$

El espesor de filete está dado por

$$e = \frac{P_x}{2} \quad (47)$$

$$e = \frac{3\pi}{2} = 1,5\pi$$

El ancho de cara está dado por [71]

$$b_{wg} = \begin{cases} 6 + 2,38P_x & P_x \geq 4,06mm \\ 1,125\sqrt{(d_{og} + 2c)^2 - (d_{og} - 4a)^2} & P_x < 4,06mm \end{cases} \quad (48)$$

Donde d_w es igual al diámetro exterior del sinfín dado por la ecuación 53 y c es la holgura. Para este caso $P_x \geq 4.06 \text{ mm}$ por lo que el ancho de cara b_{wg} es

$$b_{wg} = 6 + 2,38P_x \quad (49)$$

$$b_{wg} = 6 + 2,38(3\pi) = 20mm$$

De igual forma para el diámetro mayor y sobre aristas d_1 , para sinfín de una y dos entradas se tiene que

$$d_1 = d_{og} + 0,4775P_c \quad (50)$$

$$d_1 = 101,64 + 0,4775(3\pi) = 106,14mm$$

Para la concavidad periférica

$$R = 0,5d_w - m \quad (51)$$

$$R = 0,5(30) - 3 = 12mm$$

Como siguiente paso se determina la geometría del tornillo sinfín, en este punto se propone una distancia entre centros $C_d=63\text{mm}$. Las buenas proporciones indican que el diámetro de paso del tornillo sinfín d_w se ubica en el intervalo [71]

$$\frac{C_d^{0,875}}{2} \leq d_w \leq \frac{C_d^{0,875}}{1,07} \quad (52)$$

$$18,77 \leq d_w \leq 35,8$$

En este caso se eligió $d_w = 30 \text{ mm}$, diámetro dentro de ese intervalo.

El diámetro exterior y el diámetro de raíz para la geometría del tornillo sinfín se calcula de igual forma que la ecuación 44 y para la ecuación 45, pero considerando el diámetro de paso del sinfín anteriormente calculado.

$$\begin{aligned} d_{ow} &= d_w + 2a & (53) \\ d_{ow} &= 30 + 2(2,82) = 35,64\text{mm} \end{aligned}$$

$$\begin{aligned} d_{rw} &= d_w - 2b & (54) \\ d_{rw} &= 30 - 2(3,26) = 23,48\text{mm} \end{aligned}$$

Un término importante en el tornillo sinfín es el avance L y su ángulo λ , los cuales se calculan de la siguiente manera [71]

$$\begin{aligned} L &= N_w \cdot P_c & (55) \\ L &= 1(3\pi) = 3\pi\text{mm} \end{aligned}$$

$$\begin{aligned} \lambda &= \tan^{-1} \left[\frac{L}{\pi d_w} \right] & (56) \\ \lambda &= \tan^{-1} \left[\frac{3\pi}{\pi(30)} \right] = 5,71^\circ \end{aligned}$$

Y a partir del ángulo de avance se obtiene el ángulo de hélice ψ .

$$\begin{aligned} \psi &= 90^\circ - \lambda & (57) \\ \psi &= 90^\circ - 5,71^\circ = 84,29^\circ \end{aligned}$$

El ancho de cara del tornillo sinfín se calcula a partir de la ecuación [71]

$$b_{ww} = 2\sqrt{\left(\frac{d_{og}}{2}\right)^2 - \left(\frac{d_g}{2} - a\right)^2} \quad (58)$$

$$b_{ww} = 2\sqrt{\left(\frac{101,64}{2}\right)^2 - \left(\frac{96}{2} - 2,82\right)^2} = 64,82mm$$

El radio de la cabeza está dado por

$$\begin{aligned} r &= 0,05(P_c) \\ r &= 0,05(3\pi) = 0,47mm \end{aligned} \quad (59)$$

Para la corrección de la distancia entre centros se utiliza la siguiente fórmula [71]

$$\begin{aligned} C_d &= \frac{d_g + d_w}{2} \\ C_d &= \frac{128 + 40}{2} = 84mm \end{aligned} \quad (60)$$

En la Tabla 91 se muestran los resultados de la geometría del tornillo sinfín.

Parámetros del tornillo sinfín	
Número de entradas	1
Paso axial	$3\pi mm$
Diámetro exterior	35.64 mm
Diámetro de paso	30 mm
Diámetro de raíz	23.48 mm
Addendum	2.82 mm
Dedendum	4.3.26 mm
Altura de trabajo	5.64 mm
Altura total	6.1 mm
Longitud de cara	64.82 mm
Avance	$3\pi mm$
Ángulo de presión	20°
Ángulo de avance	5.71°
Ángulo de Hélice	84.29°
Radio de la cabeza	0.47 mm

Tabla 91: Resultado de la geometría del tornillo sinfín

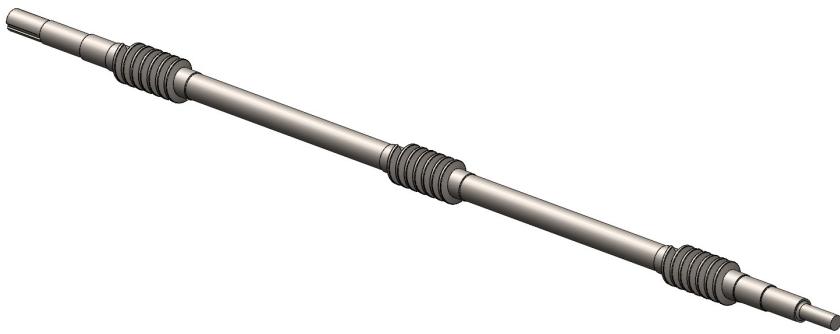


Figura 36: Geometría del tornillo sinfín

Y de igual forma en la Tabla 92 se muestran los resultados de la geometría de la corona.

Parámetros de la corona	
Número de dientes	32
Paso circular	$3\pi \text{ mm}$
Módulo	3
Diámetro exterior	101.64 mm
Diámetro de paso	96 mm
Diámetro de raíz	89.48 mm
Diámetro sobre aristas	106.14 mm
Addendum	2.82 mm
Dedendum	3.26 mm
Espesor del filete	$1.5\pi \text{ mm}$
Altura de trabajo	5.64 mm
Altura total	6.1 mm
Ancho de cara	20 mm
Concavidad periférica	12 mm
Ángulo de presión	20°

Tabla 92: Resultado de la geometría de la corona

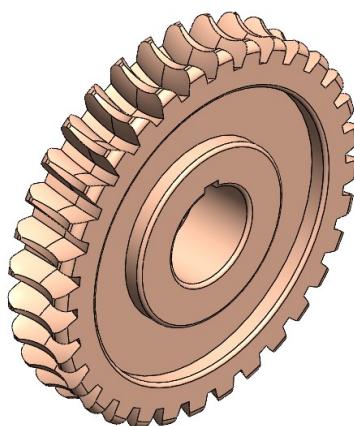


Figura 37: Geometría de la corona

$$v_t = \frac{(67,021)(0,03)}{2 \cos 5,71} = 1,01 \frac{m}{s}$$

Donde ω es la velocidad angular del sinfín en $\frac{rad}{s}$ y d_{wm} es el diámetro de paso del sinfín pero en metros. Para el coeficiente de fricción se utiliza la siguiente expresión dada por la AGMA [71]

$$\mu = \begin{cases} 0,150 & v_t = 0 \frac{m}{s} \\ 0,124e^{(-2,233v_t^{0,645})} & 0 < v_t \leq 0,05 \frac{m}{s} \\ 0,103e^{(-1,185v_t^{0,450})} & 0,05 \frac{m}{s} < v_t \end{cases} \quad (62)$$

Por lo que al tener una velocidad de $1,35 \frac{m}{s}$ se tiene que

$$\begin{aligned} \mu &= 0,103e^{(-1,185v_t^{0,450})} \\ \mu &= 0,103e^{(-1,185(1,01)^{0,450})} = 0,0433 \end{aligned} \quad (63)$$

Para las fuerzas que actúan sobre el tornillo y la corona hay igualdades entre algunas fuerzas como se observa en la figura 38.

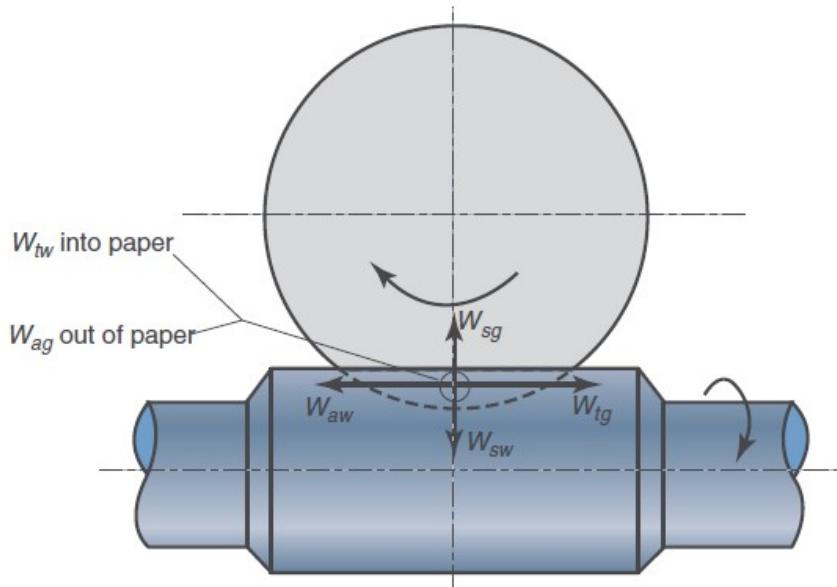


Figura 38: Equivalencia entre las fuerzas que actúan entre el sinfín y la corona.

Para el cálculo de la fuerza tangencial del tornillo sinfín se tiene la siguiente ecuación [71]

$$W_{tw} = \frac{2\tau_w}{d_{wm}} \quad (64)$$

$$W_{tw} = \frac{2(0,171)}{0,03} = 11,4N$$

Fuerza tangencial de la corona a partir de la W_{tw} [69]

$$W_{tg} = \frac{W_{tw}}{\left(\frac{\cos \phi_n \sin \lambda + \mu \cos \lambda}{\cos \phi_n \cos \lambda - \mu \sin \lambda} \right)} \quad (65)$$

$$W_{tg} = \frac{11,4}{\left(\frac{\cos(20)\sin(5,71) + 0,0433\cos(5,71)}{\cos(20)\cos(5,71) - 0,0433\sin(5,71)} \right)} = 77,69N$$

Para la fuerza radial o también conocida como fuerza de empuje se utilizó la siguiente fórmula.[71]

$$W_{rg} = \frac{W_{tg} \tan \phi_n}{\cos \lambda} \quad (66)$$

$$W_{rg} = \frac{77,69 \tan(20)}{\cos(5,71)} = 28,42N$$

Como se mencionó anteriormente, en el sistema corona-sinfín las fuerzas son equivalentes (véase Figura 38) por lo que se llega a lo siguiente:

Fuerza axial de la corona

$$W_{ag} = W_{tw} = 23,45N$$

Fuerza axial del tornillo sinfín

$$W_{aw} = W_{tg} = 11,4N$$

Fuerza radial del tornillo sinfín

$$W_{rw} = W_{rg} = 28,42N$$

La fuerza de fricción está dada por

$$W_f = \frac{\mu W_{tg}}{\cos \lambda \cos \phi} \quad (67)$$

$$W_f = \frac{0,0385(77,69)}{\cos(5,71)\cos(20)} = 3,6N$$

Para el cálculo de la fuerza tangencial permisible W_{tPERM} , se requieren algunos factores como lo son el factor de materiales C_s , el factor de corrección por relación C_m y el factor de velocidad C_v .

El factor de materiales se obtiene mediante la siguiente tabla 93, tomando en cuenta el diámetro de paso de la corona y considerando el proceso de manufactura de la misma.

Proceso de manufactura	Diámetro de paso d_g	Factor de materiales C_s
Fundición en arena	$d_g \leq 64mm$ $d_g \geq 64mm$	$10001859 - 476,5 \log d_g$
Fundición por enfriamientoestático o forjado	$d_g \leq 200mm$ $d_g \geq 200mm$	$10002052 - 455,8 \log d_g$
Fundición centrífuga	$d_g \leq 625mm$ $d_g \geq 625mm$	$10001504 - 179,8 \log d_g$

Tabla 93: Factor de materiales C_s , para coronas de bronce [71]

Por lo tanto, se tiene que el proceso de manufactura de la corona es mediante fundición en arena y el diámetro d_g es mayor a $64mm$, entonces se tiene

$$C_s = 1859 - 476,5 \log d_g \quad (68)$$

$$C_s = 1859 - 476,5 \log(96) = 914,45$$

El factor de corrección por relación se obtiene con el siguiente criterio [71], en donde se considera el valor de la relación de transmisión Z , como se muestra a continuación

$$C_m = \begin{cases} 0,0200(-Z^2 + 40Z - 76)^{0,5} + 0,46 & 3 \leq Z \leq 20 \\ 0,0107(-Z^2 + 56Z + 5145)^{0,5} & 20 \leq Z \leq 76 \\ 1,1483 - 0,00658Z & 76 \leq Z \end{cases} \quad (69)$$

Por lo que al tener una relación de transmisión de $Z=32$

$$C_m = 0,0107(-Z^2 + 56Z + 5145)^{0,5} \quad (70)$$

$$C_m = 0,0107(-32^2 + 56(32) + 5145)^{0,5} = 0,823$$

El factor de corrección de velocidad está dado por

$$C_v = \begin{cases} 0,659e^{(-0,2165v_t)} & 0 < v_t \leq 3,5 \frac{m}{s} \\ 0,6520v_t^{-0,571} & 3,5 < v_t \leq 15 \frac{m}{s} \\ 1,098v_t^{-0,774} & 15 \frac{m}{s} < v_t \end{cases} \quad (71)$$

Con el valor de $v_s = 1.5 \frac{m}{s}$ se obtiene

$$\begin{aligned} C_v &= 0,659e^{(-0,2165v_t)} \\ C_v &= 0,659e^{(-0,2165(1,01))} = 0,53 \end{aligned} \quad (72)$$

Al tener el valor de los factores se sustituyen en la siguiente fórmula y se obtiene

$$\begin{aligned} W_{tPERM} &= \frac{C_s d_g^{0,8} b_{wg} C_m C_v}{75,948} \\ W_{tPERM} &= \frac{(914,45)(96)^{0,8}(20)(0,823)(0,53)}{75,948} = 4047,31 N \end{aligned} \quad (73)$$

De esa forma como la fuerza tangencial permisible ($W_{tPERM} = 4047,31 N$) es más grande que la fuerza tangencial sobre la corona ($W_{tg} = 77,69 N$), se considera que el diseño es seguro.

Cálculo del esfuerzo de flexión en los dientes de la corona

Para calcular el esfuerzo de flexión de los dientes de la corona, es necesario conocer el factor de forma de Lewis Y , el cual está relacionado con el ángulo de presión ϕ_n de la siguiente forma.

Ángulo de presión ϕ_n	Factor de forma de Lewis Y
14,5°	0.100
20°	0.125
25°	0.150
30°	0.175

Tabla 94: Factor de forma de Lewis Y en relación con el ángulo de presión ϕ_n [69]

Por lo que el factor de Lewis $Y = 0,125$ debido a $\phi_n = 20^\circ$ y el esfuerzo de flexión en el diente de la corona se define como [71]

$$\begin{aligned} \sigma &= \frac{W_{tg} P_d}{b_{wg} Y} \\ \sigma &= \frac{(77,69)(333)}{(0,020)(0,125)} = 16,3 MPa \end{aligned} \quad (74)$$

Cabe mencionar que P_d es en m^{-1} y b_{wg} en m .

Validación del mecanismo corona-sinfín

De acuerdo a los resultados de los cálculos de las fuerzas del tornillo sinfín y la corona, en la Tabla 95, se muestran los valores de las fuerzas.

Parámetros	Tornillo Sinfín	Corona
Fuerza tangencial	11.4 N	77.69 N
Fuerza Radial	28.42 N	28.42 N
Fuerza Axial	77.69 N	11.4 N

Tabla 95: Fuerzas en el tornillo sinfín y en la corona

Se hizo un estudio estático para validar los esfuerzos permisibles en la corona.

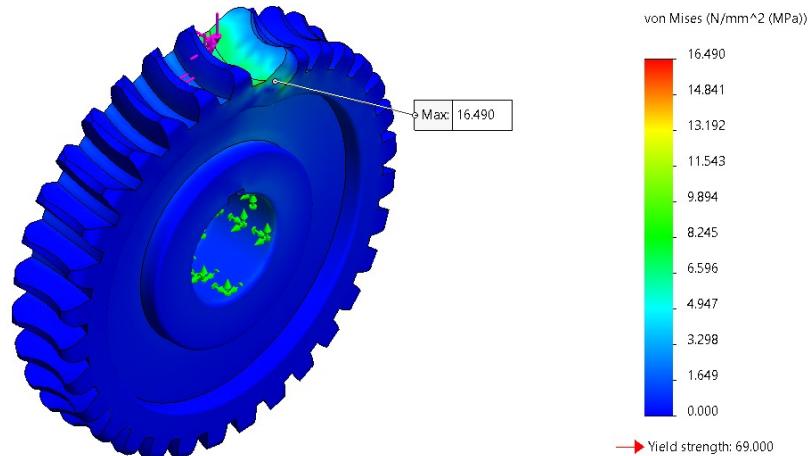


Figura 39: Validación de esfuerzos en la corona

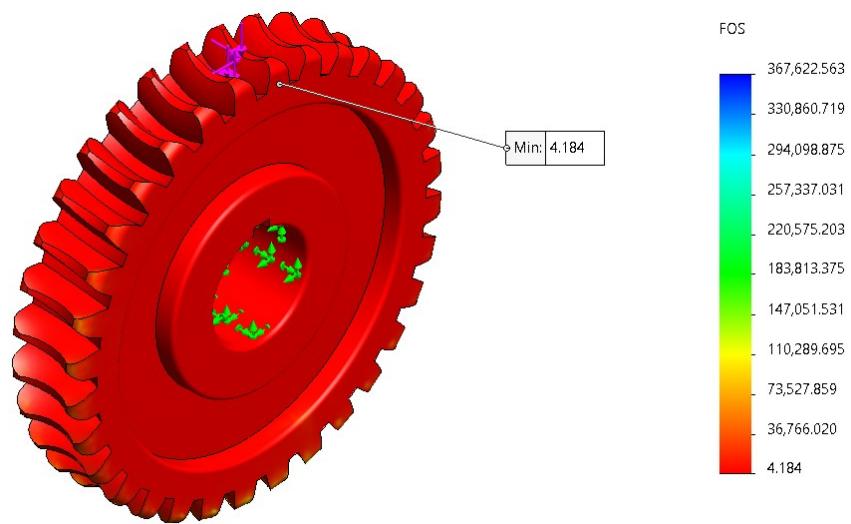


Figura 40: Validación del factor de seguridad en la corona

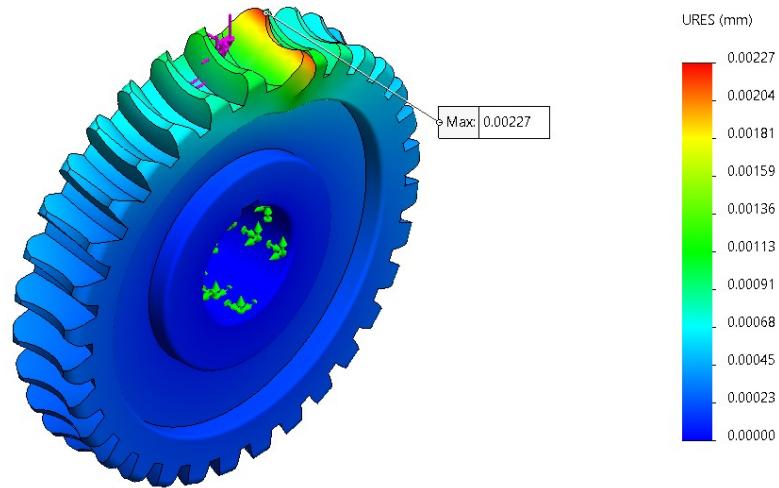


Figura 41: Validación de desplazamiento en la corona

Eje del tornillo sinfín

Con el propósito de transmitir la potencia del motor a los paneles de forma segura, se diseñó un eje, el cual sea lo suficientemente resistente para soportar todas las cargas que están actuando sobre él. Al tratarse de un elemento que está sometido a esfuerzos altos, se utilizan los aceros de bajo carbono como los AISI 1020, 1117, 8620 O 4320 [72], por tal motivo se propuso utilizar un acero AISI 1020 CD con una resistencia a la fluencia $S_y = 350 MPa$ y una resistencia última de tensión $S_u = 420 MPa$. Véase el anexo B.

El cálculo del tornillo sinfín se realizó sobre uno de los tres segmentos que se acopla a lo largo de los paneles 42.



Figura 42: Primer segmento de tornillos sinfín.

Para el diseño del eje se consideró una velocidad angular de $67.02 \frac{rad}{s}$ (640 rpm) y un factor de seguridad $n_s = 4$, con un torque a la entrada de 1.539 Nm . Anteriormente en la sección de corona-sinfín se obtuvieron las fuerzas que actúan sobre el eje del tornillo sinfín ($W_t = 11,4\text{N}$, $W_a = 77,69\text{N}$, $W_r = 28,42\text{N}$), las cuales se representan de manera gráfica en el diagrama de cuerpo libre en la Figura 43.

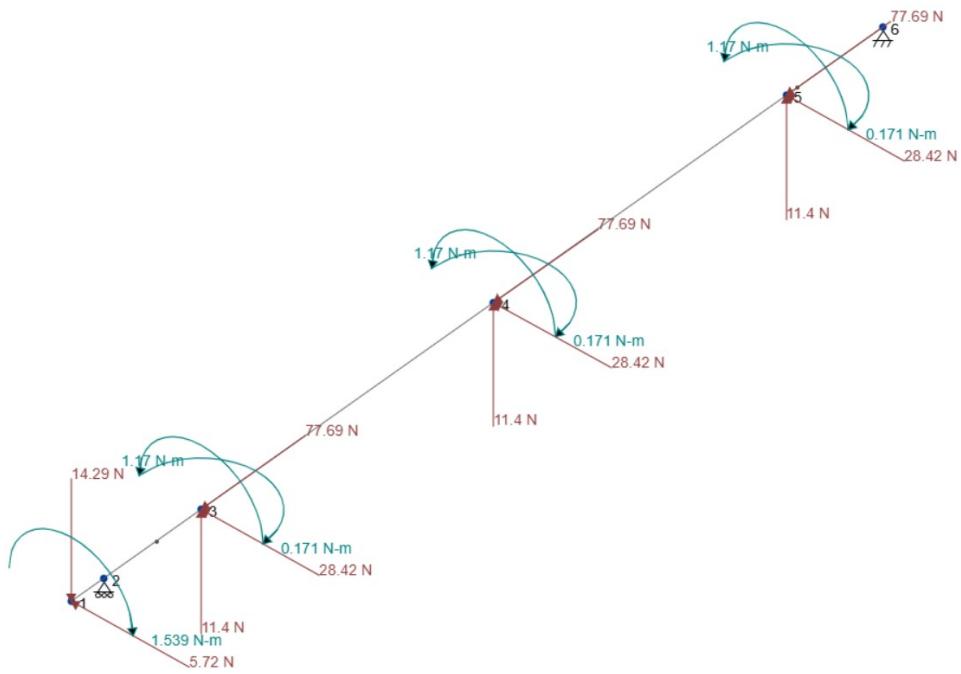


Figura 43: Diagrama de cuerpo libre del eje del tornillo sinfín.

Cálculo de las reacciones en los apoyos y diagramas cortante, flector y torsor

Al existir una carga que actúa axialmente sobre el diente se genera un momento, el cual se calculó de la siguiente forma

$$M_a = W_a \frac{d_w}{2} \quad (75)$$

$$M_a = 77,69 \frac{0,03}{2} = 1,17 \text{ Nm}$$

Teniendo todas la fuerzas y los momentos involucrados, se obtuvieron las reacciones en los planos $X - Y$ (Figura 44) y $X - Z$ (Figura 47).

Para $X - Y$

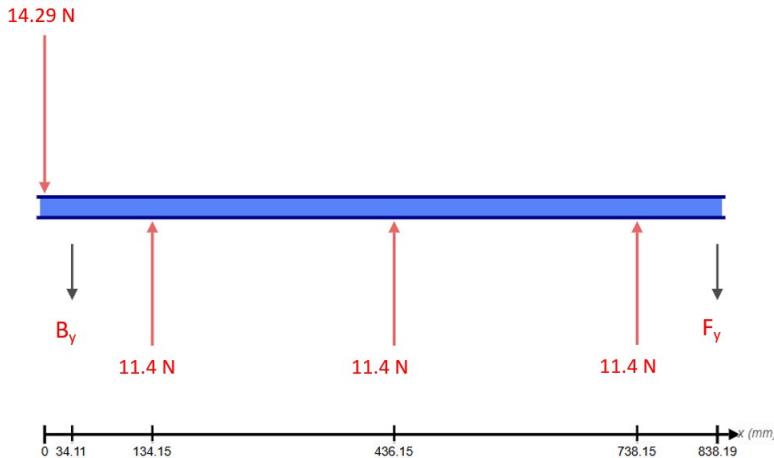


Figura 44: Diagrama de cuerpo libre en el plano $X - Y$.

Suma de momentos al rededor de B igual con cero

$$\begin{aligned}
 + \circlearrowleft M_B &= 0 \\
 14,29N(0,03411m) + 11,4N(0,10004m) + 11,4N(0,40204m) \\
 + 11,4N(0,70404m) - F_y(0,80408m) &= 0 \\
 F_y &= 17,71N
 \end{aligned} \tag{76}$$

Suma de fuerzas a lo largo del eje Y igual con cero para equilibrio estático

$$\begin{aligned}
 + \uparrow F_y &= 0 \\
 B_y + 11,4N - 14,29N + 11,4N + 11,4N - 17,71N &= 0 \\
 B_y &= -2,2N \downarrow
 \end{aligned} \tag{77}$$

Luego se obtuvieron los diagramas de la fuerza cortante y momento flector

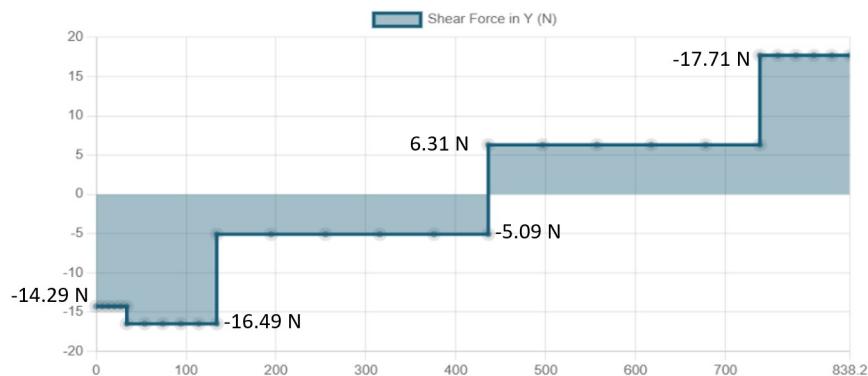


Figura 45: Diagrama de la fuerza cortante en el plano $X - Y$.

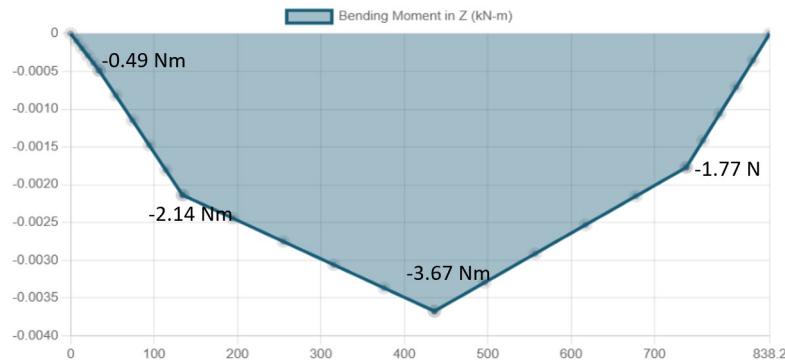


Figura 46: Diagrama del momento flector en el plano $X - Y$.

Para $X - Z$

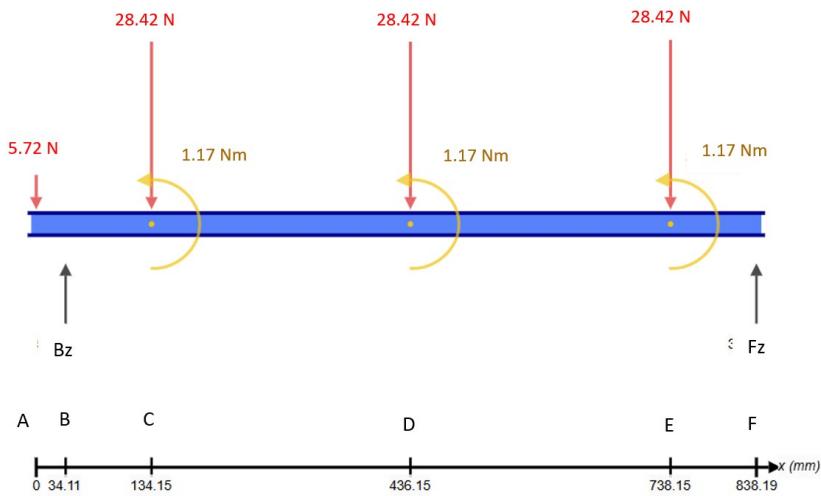


Figura 47: Diagrama de cuerpo libre en el plano $X - Z$.

Suma de fuerzas a lo largo del eje Y igual con cero para equilibrio estático

$$+ \uparrow F_z = 0$$

$$B_z + F_z - 5,72N - 3(28,42N) = 0$$

$$B_z + F_z = 90,98N$$

Suma de momentos al rededor de B igual con cero

$$+ \circlearrowleft M_B = 0$$

$$\begin{aligned} F_z(0,80408m) + 5,72N(0,03411m) - 28,42N(0,10004m) - 28,42N(0,43615m) \\ - 28,42N(0,73815m) + 3(1,17Nm) = 0 \end{aligned}$$

$$F_z = 38,022 \text{ N}$$

$$B_z + F_z = 90,98 \text{ N}$$

$$B_z = 52,958 \text{ N}$$

Los diagramas de la fuerza cortante y momento flector se muestran la Figura 48 y 49 respectivamente.

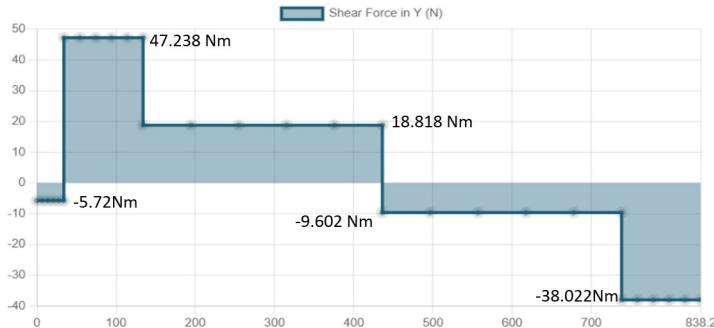


Figura 48: Diagrama de la fuerza cortante en el plano $X - Z$.

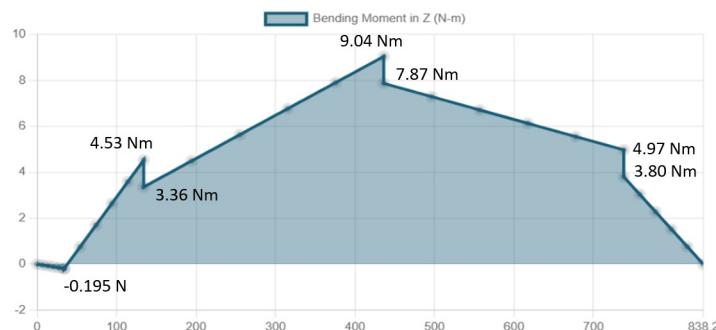


Figura 49: Diagrama del momento flector en el plano $X - Z$.

También se obtuvo el diagrama del momento torsor T_x a lo largo del eje X , como se muestra en la figura 50.

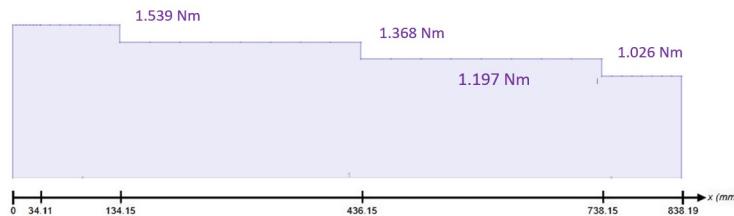


Figura 50: Diagrama del momento torsor a lo largo del eje X .

Luego se calcularon los momentos resultantes M_T en cada punto del eje, partiendo de la Ecuación 78 [71].

$$M_T = \sqrt{(M_{X-Y})^2 + (M_{X-Z})^2} \quad (78)$$

Punto	Momento resultante
A	0 Nm
B	0.53 Nm
C	5.01 Nm
D	9.76 Nm
E	5.28 Nm
F	0 Nm

Tabla 96: Momentos resultantes

Cálculo de los diámetros en los puntos críticos

Para el cálculo de los diámetros se consideran las secciones en donde el eje está mayormente comprometido estructuralmente, en este caso se consideró el punto de estudio *D*, Puesto que ahí se concentra el momento resultante más grande y en consecuencia la flexión es mayor.

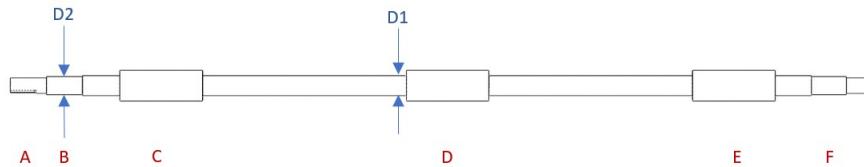


Figura 51: Diámetros de interés en el eje del tornillo sinfín.

Mediante el método de diseño de ejes por fatiga se comienza por calcular el límite de resistencia a la fatiga S'_e

$$S'_e = 0,5S_u \quad (79)$$

$$S'_e = 0,5(420 MPa) = 210 MPa$$

Para así calcular el límite a la fatiga corregido S_e , dado por

$$S_e = S'_e \cdot k_f \cdot k_s \cdot k_r \cdot k_t \cdot k_m \quad (80)$$

Donde

- k_f factor de acabado superficial
- k_s factor tamaño
- k_r factor de fiabilidad
- k_t factor de temperatura
- k_m factor diverso

Para el factor de superficie k_f , se elige un proceso de maquinado y mediante la Figura 52 se tiene que $k_f = 0,8$.

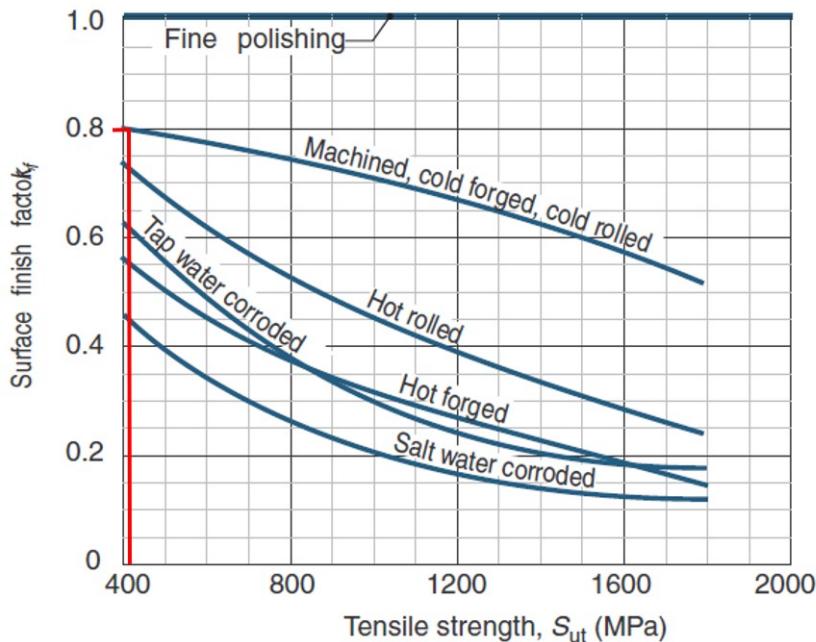


Figura 52: Factores de acabado superficial para aceros [71].

Para el factor de tamaño se propone un diámetro de 20 mm y el resultado del factor de tamaño se obtiene mediante la siguiente formula 81

$$k_s = 1,128d^{-0,112} \quad (81)$$

$$k_s = 1,128(20)^{-0,112} = 0,89$$

Para el factor de fiabilidad se propone una fiabilidad del 99,9 % y mediante la Tabla 97

Porcentaje de probabilidad de supervivencia	Factor de fiabilidad k_r
50	1.00 Nm
90	0.90 Nm

95	0.87 Nm
99	0.82 Nm
99.9	0.75 Nm
99.99	0.70 Nm

Tabla 97: Factor de fiabilidad k_r [71].

Y para k_t y k_m se considera un valor de 1. Por lo tanto, el límite a la fatiga corregido S_e es

$$S_e = (210)(0.8)(0.89)(0.75) = 112,14 MPa$$

El factor de concentración de esfuerzo por fatiga K_t está dado por

$$K_f = 1 + q_n(K_c - 1) \quad (82)$$

En donde q_n es la sensibilidad de la muesca y K_c es la concentración de esfuerzos teórica para la geometría y carga en particular. q_n está dada por

$$q_n = \frac{1}{1 + \frac{\sqrt{a}}{\sqrt{r}}} \quad (83)$$

Para la ecuación anterior se necesitan proponer la relación $\frac{D}{d}$ y $\frac{r}{d}$, pero anteriormente se mencionó que se propuso un diámetro $d = 20mm$, una relación del 10 % de aumento y se propone un $r = 0,5mm$ por lo que las relaciones resultan

$$\begin{aligned} \frac{D}{d} &= 1,1 \\ \frac{r}{d} &= 0,025 \end{aligned}$$

Punto D

Para el punto D se tiene el momento resultante máximo $M_{TD} = 9,76Nm$ y un momento torsor $T_D = 1,368Nm$, por lo que se calcula K_f para flexión y K_{fs} para torsión.

Para la sensibilidad de la muesca el valor de \sqrt{a} se obtiene de la siguiente de la Figura 53

S_u		\sqrt{a}			
(ksi)	(MPa)	Carga axial y flexión		Torsión	
		(in ^{0.5})	(mm ^{0.5})	(in ^{0.5})	(mm ^{0.5})
50	345	0.130	0.66	0.093	0.47
55	380	0.118	0.59	0.087	0.44
60	415	0.108	0.54	0.080	0.40
70	485	0.093	0.47	0.070	0.35
80	550	0.080	0.40	0.062	0.31
90	620	0.070	0.35	0.055	0.28
100	690	0.062	0.31	0.049	0.25
110	760	0.055	0.28	0.044	0.22
120	825	0.049	0.25	0.039	0.20
130	895	0.044	0.22	0.035	0.18
140	965	0.039	0.20	0.031	0.16
160	1100	0.031	0.16	0.024	0.12
180	1240	0.024	0.12	0.018	0.09
200	1380	0.018	0.09	0.013	0.07
220	1515	0.013	0.07	0.009	0.05
240	1655	0.009	0.05		

Figura 53: Valores de \sqrt{a} [72].

Como el valor de S_u no se encuentra en la tabla se hace una interpolación lineal y se obtiene que para flexión $\sqrt{a} = 0,535$ y para torsión $\sqrt{a} = 0,396$, teniendo el valor de \sqrt{a} se obtiene el valor de q_{nM} para flexión y q_{nT} para torsión.

$$q_{nM} = \frac{1}{1 + \frac{0,535}{\sqrt{0,5}}} = 0,57$$

$$q_{nT} = \frac{1}{1 + \frac{0,396}{\sqrt{0,5}}} = 0,64$$

Para el factor geométrico de concentración de esfuerzos K_c para un eje con filete de hombro a flexión [72] (véase Anexo C) se tiene

$$K_c \approx A \left(\frac{r}{d} \right)^b \quad (84)$$

Donde $A = 0,95120$ y $b = -0,23757$

$$K_c \approx (0,95120) \left(\frac{0,5}{20} \right)^{-0,23757} = 2,28$$

Y para el factor geométrico de concentración de esfuerzos K_c para un eje con filete de hombro a torsión [72] (véase Anexo C) se realizó una interpolación lineal que dio como resultado $A = 0,89708$ y $b = -0,13506$

$$K_c \approx (0,89708) \left(\frac{0,5}{20} \right)^{-0,13506} = 1,48$$

Por lo tanto, el resultado de K_f para flexión es

$$K_f = 1 + (0,57)(2,28 - 1) = 1,7296$$

Y el factor de concentración de esfuerzos K_{fs} para torsión es

$$K_{fs} = 1 + (0,64)(1,5215 - 1) = 1,33376$$

Por lo tanto, el diámetro mínimo para el eje en el punto D está dado por la teoría del esfuerzo cortante máximo (MSST) y la línea de Soderberg. Equación 85 [71].

$$d_1 = \left[\frac{32n_s}{\pi S_y} \sqrt{\left(M_m + \frac{S_y}{S_e} K_f M_a \right)^2 + \left(T_m + \frac{S_y}{S_e} K_{fs} T_a \right)^2} \right]^{1/3} \quad (85)$$

En este caso, al ser un ciclo de flexión invertida el momento medio $M_m = 0$ y al ser un torque constante en la sección el momento torsor $T_a = 0$, entonces

$$d_1 = \left[\frac{32(4)}{\pi 350x10^6} \sqrt{\left(\frac{350x10^6}{112,14x10^6} (1,7296)(9,76) \right)^2 + (1,368)^2} \right]^{1/3}$$

$$d = 18,30mm$$

Recalcando el diámetro alrededor del punto D

Tomando el punto D , se hace una nueva iteración, debido a que el diámetro anterior solo fue un diámetro propuesto y es necesario recalcular para dar certeza a que el diámetro se encuentra en el rango. Nótese que el factor de acabado superficial k_f , el factor de fiabilidad k_r , el factor de temperatura k_t y el factor de factores diversos k_m no cambian, solamente el factor de tamaño k_s .

Con $d = 18,30$ se tiene

$$k_s = 0,90$$

Por lo tanto,

$$S_e = (210)(0,8)(0,90)(0,75) = 113,4 MPa$$

La relación $\frac{D}{d}$ se mantiene igual, pero la relación $\frac{r}{d} = 0,027$. Por lo tanto, K_c se redefine como

$$K_c \approx (0,95120) \left(\frac{0,5}{18,3} \right)^{-0,23757} = 2,23$$

Por tal razón K_f se obtiene como

$$K_f = 1 + (0,57)(2,23 - 1) = 1,7056$$

En consecuencia d es igual a

$$d_1 = \left[\frac{32(4)}{\pi 350x10^6} \sqrt{\left(\frac{350x10^6}{113,4x10^6} (1,70)(9,76) \right)^2 + (1,368)^2} \right]^{1/3}$$

$$d_1 = 18,13 mm$$

Al no variar mucho el diámetro se toma como aceptable y se redondea al valor cerrado inmediato superior, por lo tanto $d_1 = 19 mm$.

Para validar que el diámetro calculado es capaz de trabajar bajo las condiciones de carga establecidas en el problema, se realizó un análisis estático del elemento siendo modelado en el software de SolidWorks®. Las condiciones del estudio fueron restricciones de tipo geometría fija en las caras en donde se monta el eje sobre la chumacera y una mallado basado en curvatura.

Las fuerzas que se aplicaron fueron las obtenidas en la sección del cálculo de la corona y el sinfín, las fuerzas tienen un valor de $77.69 N$ para la fuerza axial del sinfín, $28.42 N$ para la fuerza radial y $11.4 N$ para la fuerza tangencial del tornillo sinfín. Los resultados obtenidos fueron un esfuerzo de von Mises máximo de $24.120 MPa$ (Figura 54), una deflexión máxima de $0.0305 mm$ (Figura 55) y un factor de seguridad de 10 (Figura 56). Se concluye que el eje es capaz de transmitir la potencia desde el motor hasta los paneles.

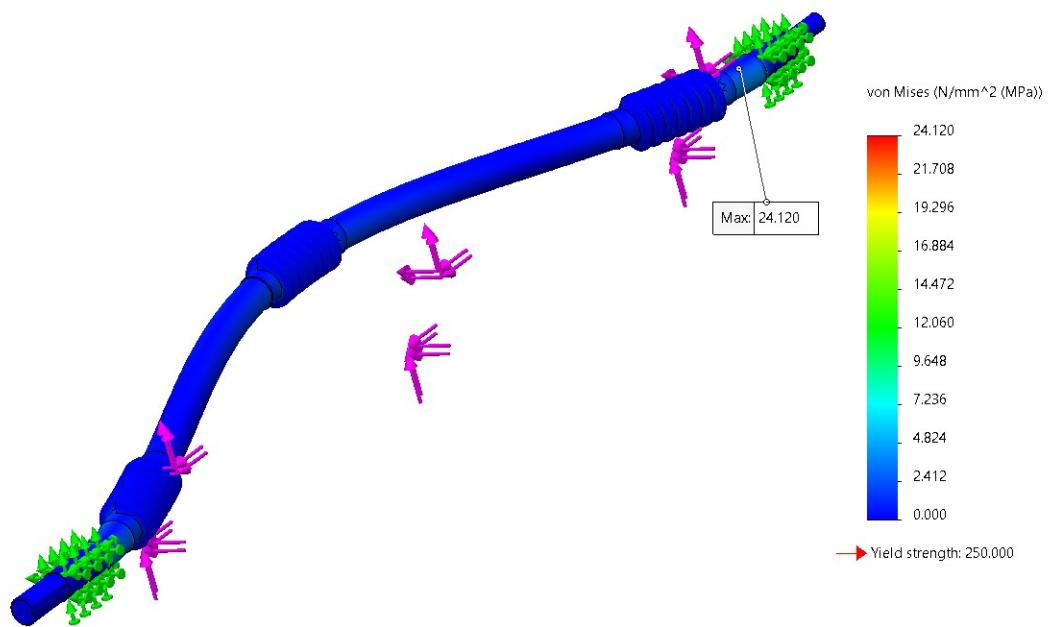


Figura 54: Máximo esfuerzo de von Mises.

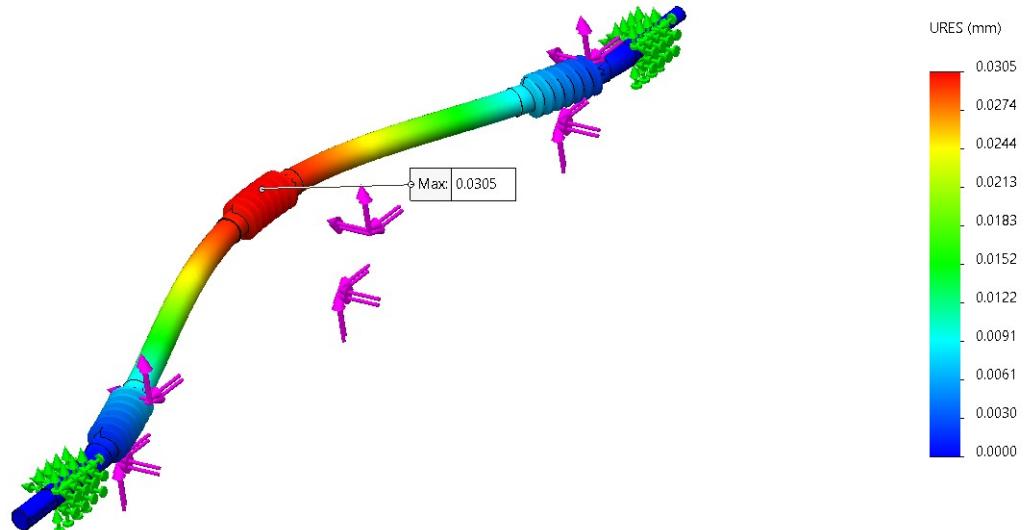


Figura 55: Deflexión máxima en el eje.

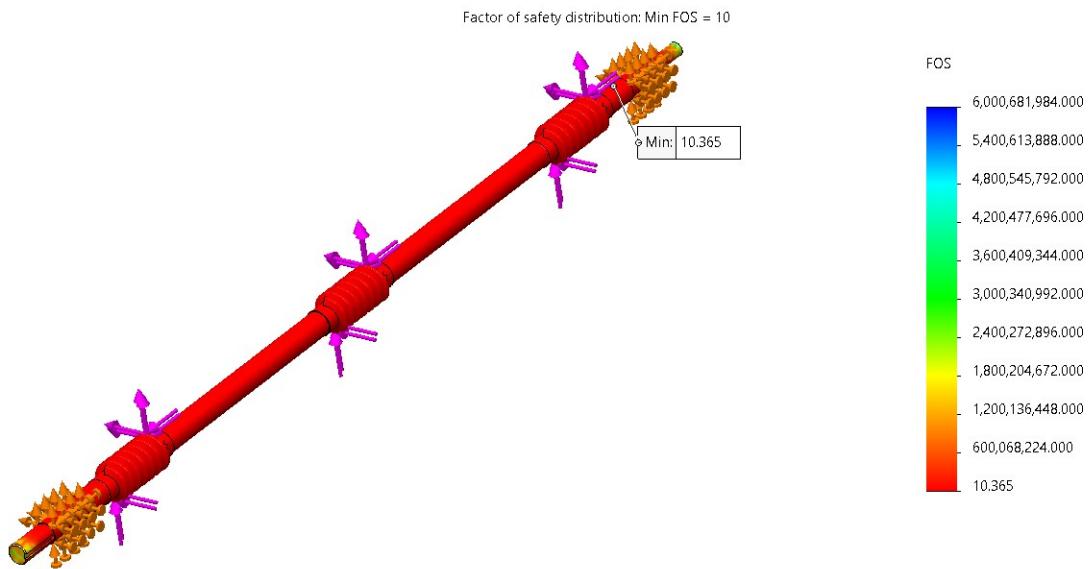


Figura 56: Factor de seguridad en el eje.

Selección del rodamiento para los apoyos del tornillo sinfín

Al haberse calculado el diámetro de la sección mayor se utilizó la relación de $\frac{D}{d} = 1,1$ para asignarle un valor al diámetro en donde va el rodamiento del eje, es importante considerar que el rodamiento va montado sobre una chumacera, por lo tanto, para el diámetro del apoyo se tiene $d_2 = 17,27\text{mm}$, pero considerando la relación de transmisión dada por el motor, se buscó tener un diámetro mayor en la sección en donde se monta la catarina, entonces la dimensión del eje en donde se monta el rodamiento se estableció en 20 mm . El diámetro calculado, respetando la relación de incremento y decremento del 10% quedó de 22 mm .

2.2.3.5 Miembro estructural

Para el debido funcionamiento del sistema, es necesario contar con una estructura que sea capaz de resistir todas las cargas que se generen cuando el sistema esté operando, por lo que se realizó el cálculo de las vigas que se consideran mayormente comprometidas estructuralmente.

Cálculo de la viga A

Para realizar el cálculo en la viga A (Figura 57), se debe de conocer la carga que se aplica sobre ella. El peso del prisma triangular de los paneles está sujetado por un soporte de pared de dos pestañas (chumacera) (Figura 57), el cual está actuando como una viga con una carga puntual simétrica, en donde la reacción que está generando en una de las pestañas es la carga que está aplicada sobre el elemento de interés.

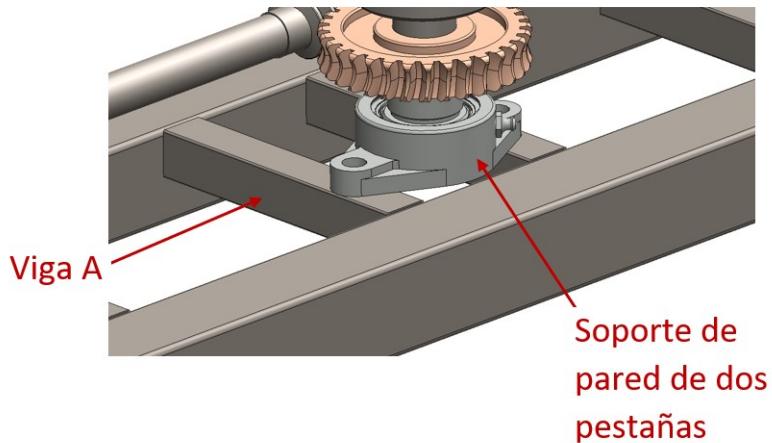


Figura 57: Representación de la viga *A* y el soporte d pared de dos pestañas.

Cálculo de las reacciones en el soporte de pared

La fuerza que actúa sobre la chumacera se obtiene a partir de la segunda ley de Newton

$$F = m \cdot a$$

En donde m es la masa que está actuando sobre el elemento y a es la aceleración debido al campo gravitatorio, la masa del prisma triangular se obtuvo mediante un modelo en el software de CAD Y CAM, SolidWorks, obteniendo una masa $m = 8,588Kg$, por lo tanto, la carga aplicada sobre la chumacera es igual a

$$F = (8,588kg)(9,81\frac{m}{s^2}) = 84,2467N$$

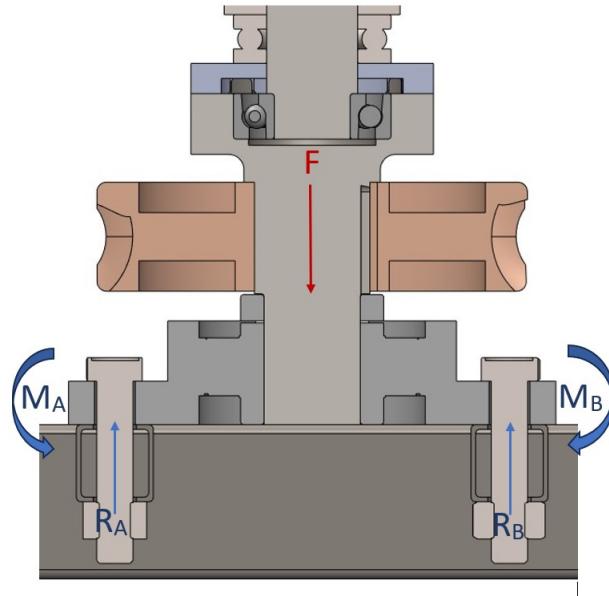


Figura 58: Fuerza aplicada y reacciones en la chumacera.

Para este tipo de vigas en las que se tiene una carga puntual F es muy simple encontrar el valor de las reacciones, puesto que los valores de las fuerzas para ambos lados es simétrica, por lo tanto $R_A = \frac{F}{2} = 42,123N$ y $R_A = R_B$. Para los momentos, al tratarse de un elemento del cual se espera que no tenga desplazamientos se considera como una viga hiper estática, entonces, por el método de integración doble se obtiene [72]

$$\frac{d^2y}{dx^2} = \frac{M}{EI} \quad (86)$$

$$EI \frac{d^2y}{dx^2} = -M_A + R_A(x) - F(x - 65,09) \quad (87)$$

Al integrar una vez se obtiene la ecuación para encontrar la pendiente

$$EI \frac{dy}{dx} = -M_Ax + R_A \frac{x^2}{2} - F \frac{(x - 65,09)^2}{2} + C_1 \quad (88)$$

Y al integrar nuevamente se obtiene la ecuación para el desplazamiento

$$EIy = -M_A \frac{x^2}{2} + R_A \frac{x^3}{6} - F \frac{(x - 65,09)^3}{6} + C_1(x) + C_2 \quad (89)$$

Los valores de las constantes de integración son cero, debido a las condiciones de frontera, donde para $x = 0$, se tiene que no hay desplazamiento, por lo que $y = 0$, por lo tanto $C_2 = 0$, también se considera que al no haber desplazamiento la pendiente tiene inclinación cero, por

lo que $\frac{dy}{dx} = 0$ y $C_1 = 0$. Para encontrar el valor del momento se observa que están situados en los extremos de la viga, por lo que en ese punto la $y = 0$ y $x = 130,18$, ahora sustituyendo esos valores en la ecuación 89 se obtiene

$$EI(0) = -M_A \frac{130,18^2}{2} + 42,123 \frac{130,18^3}{6} - 82,25 \frac{(130,18 - 65,09)^3}{6}$$

$$0 = -130,18M_A - 8473,42(42,123) - 178464,97$$

$$0 = -130,18M_A - 8473,42(42,123) - 178464,97$$

Por lo tanto se obtiene que

$$M_a = 1370,88 Nmm$$

Y de igual forma por simetría se tiene que $M_A = M_B = 1370,88 Nmm$.

Cálculo de las reacciones y del momento de inercia de la viga A

Como se mencionó anteriormente, las reacciones son las que están actuando directamente sobre la viga A, por lo que al saber el valor de la reacción es posible calcular el tamaño necesario para que la viga sea lo suficientemente rígida.

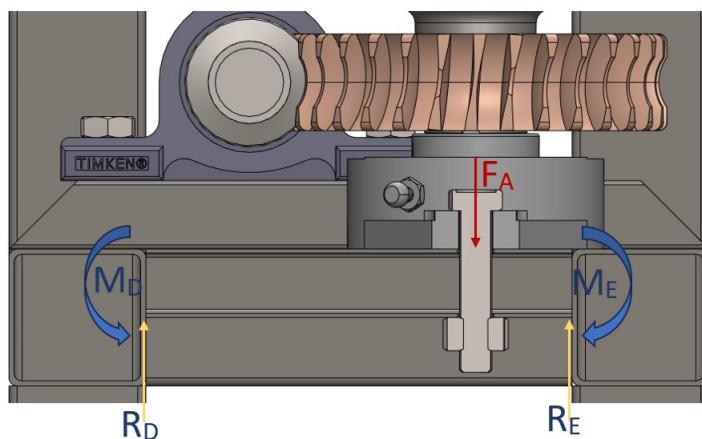


Figura 59: Fuerza aplicada y reacciones en la viga A.

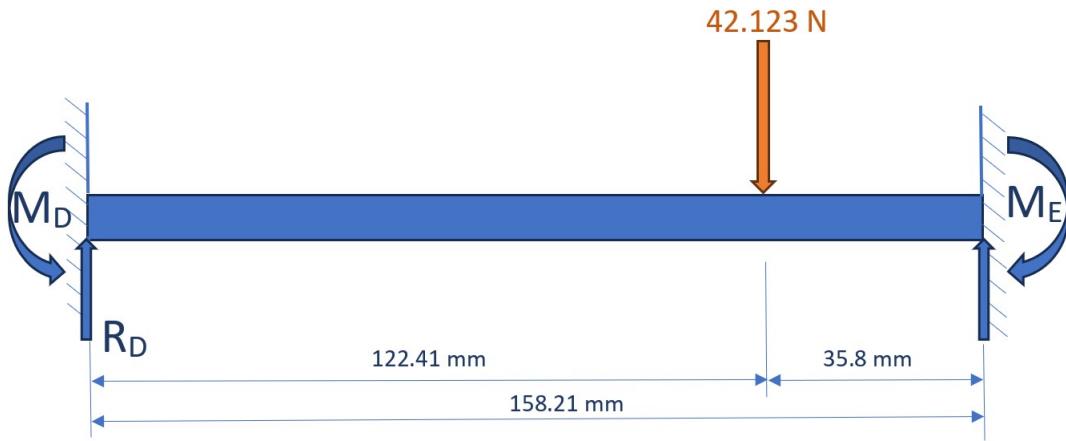


Figura 60: Diagrama de cuerpo libre de la viga A .

Para encontrar las reacciones de igual forma mediante la ecuación del momento de la viga y el método de la doble integración.

$$EI \frac{d^2y}{dx^2} = -M_D + R_D(x) - F_A(x - 112,41) \quad (90)$$

Al integrar una vez se obtiene la ecuación para encontrar la pendiente

$$EI \frac{dy}{dx} = -M_D x + R_D \frac{x^2}{2} - F_A \frac{(x - 122,41)^2}{2} + C_1 \quad (91)$$

Y al integrar nuevamente se obtiene la ecuación para el desplazamiento

$$EIy = -M_D \frac{x^2}{2} + R_D \frac{x^3}{6} - F_A \frac{(x - 122,41)^3}{6} + C_1(x) + C_2 \quad (92)$$

De igual forma, las constantes de integración son cero debido a las condiciones de frontera. Cuando $x = 158,21\text{mm}$ $y = 0$

$$0 = -12515,20M_D + 660010,04R_D - 322119,58 \quad (93)$$

Cuando $x = 158,21\text{mm}$ $\frac{dy}{dx} = 0$

$$0 = -158,21M_D + 12515,2R_D - 26993,23 \quad (94)$$

Despejando R_D en 94 y sustituyendo en 93 se tiene que

$$M_D = \frac{1423534,79 - 322119,58}{4171,73} = 264,02\text{Nm}$$

Se sustituye M_D en 94 y se obtiene

$$R_D = \frac{41770,6 + 26993,26}{12515,20} = 5,5N$$

Con sumatoria de fuerzas en y se obtiene el valor de R_E

$$R_E = 42,123 - 5,5 = 36,623N$$

Y para el momento M_E

$$M_E = -M_D + R_Dx - F_A(x - 122,41) = 901,87Nmm$$

Para encontrar el valor del momento de inercia se considera el valor de x como 122.41 mm debido a que ahí se encuentra la carga que causa la deflexión en la viga. Las condiciones que se establecen para la viga son un acero ASTM A36 que cuenta con un módulo de elasticidad $E = 200000MPa$ y se propone una deflexión máxima en la viga de 0.0002 mm.

De la ecuación 92

$$EIy = -264,02Nmm \frac{122,41mm^2}{2} + 5,5N \frac{122,41mm^3}{6} - 42,123 \frac{(122,41 - 122,41)^3}{6}$$

$$EIy = 296699,81Nmm^3$$

Despejando I

$$I = \frac{296699,81Nmm^3}{(200000 \frac{N}{mm^2} 0,0002mm)} = 7417,5mm^4 = 0,74175cm^4$$

Una vez obtenido el valor del momento de inercia I , se consulta el perfil con el momento de inercia inmediato superior en los catálogos de los fabricantes de acero [73]. Con el resultado obtenido del momento de inercia se obtiene que un perfil cuadrado tubular de 1in por 1in en calibre 20 (véase Anexo D) es suficiente para satisfacer que la deflexión no sea mayor a la propuesta.

Para comprobar que el cálculo del momento de inercia, se realizó un análisis estático de la viga modelada en el software de SolidWorks®. Las condiciones del estudio fueron restricciones de tipo geometría fija en las caras de los extremos y una malla basado en curvatura.

La fuerza que se aplicó fue la de reacción de la chumacera, puesto que es la que actúa sobre la viga A , fuerza tiene un valor de 42.123 N, generando una un esfuerzo de von Mises máximo de 0.939 MPa y una deflexión máxima de 0.00027 mm mostrada en la figura 61, siendo

prácticamente igual a la propuesta de 0.0002 mm , por lo que se concluye que el sistema es capaz de resistir y se valida el diseño de la viga *A*.

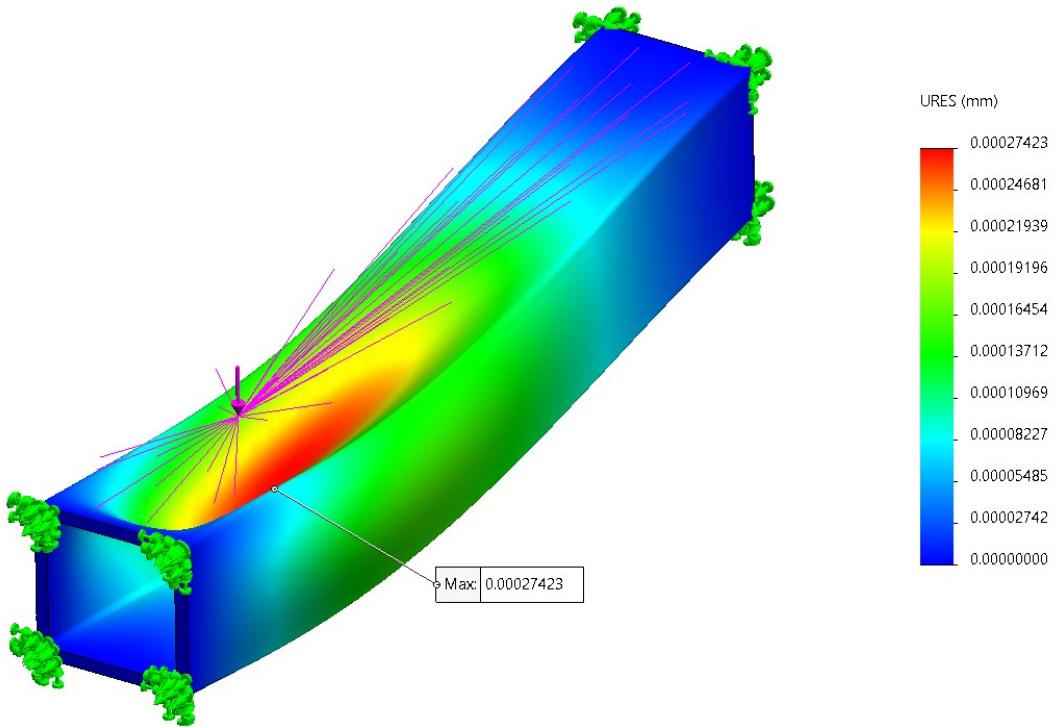


Figura 61: Deflexión máxima de la viga *B*.

Cálculo de la viga B

La viga *B* al ser un elemento muy largo que está sometido a mucha carga, puesto que soporta todo el peso de los 9 paneles que se encuentran en el mecanismo, es importante que esta viga brinde seguridad estructural para que el sistema sea durable y funcione de forma adecuada.

En la figura 62 se observa la distribución de las cargas puntuales, la magnitud de las fuerzas que actúan sobre la viga *B* son la reacciones en el punto *E* de la viga *A*.

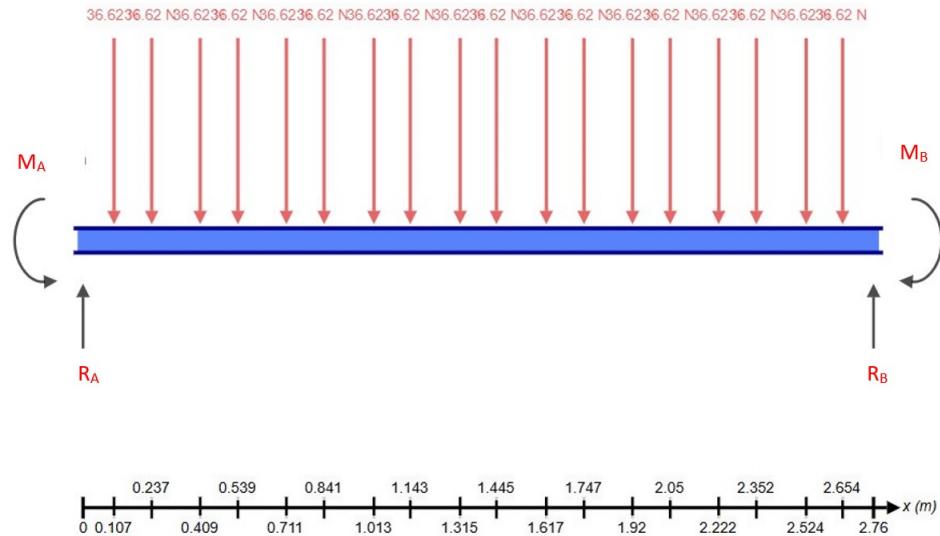


Figura 62: Diagrama de cuerpo libre de la viga B .

El procedimiento es el mismo que para la viga A, primero se obtiene la ecuación del momento y se continua con el método de la doble integración. Las reacciones en la viga fueron determinadas por un algoritmo realizado en el software de Matlab debido a la extensión de la fórmula, dando como resultado.

$$R_A = 329,66222 N$$

$$R_B = 329,66222 N$$

$$M_A = 154,25805 Nm$$

$$M_B = 154,25805 Nm$$

La fórmula del momento se tiene como

$$EI \frac{d^2y}{dx^2} = -M_A + R_A x - 36,62(x - 0,10702) - 36,62(x - 0,23693) - 36,62(x - 0,40911) \quad (95)$$

$$-36,62(x - 0,53402) - 36,62(x - 0,7112) - 36,62(x - 0,89111) - 36,62(x - 1,01329)$$

$$-36,62(x - 1,1432) - 36,62(x - 1,31538) - 36,62(x - 1,44529) - 36,62(x - 1,61747)$$

$$-36,62(x - 1,74738) - 36,62(x - 1,91956) - 36,62(x - 2,04947) - 36,62(x - 2,22165)$$

$$-36,62(x - 3,35156) - 36,62(x - 2,52374) - 36,62(x - 2,65365)$$

Al integrar una vez la ecuación anterior se obtiene la ecuación para encontrar la pendiente

$$EI \frac{dy}{dx^2} = \left(\frac{RA}{2} - \frac{16479}{50} \right) x^2 + \left(\frac{80032274345423366092067}{87960930222080000000} - M_A \right) x$$

Y al integrar nuevamente se obtiene la ecuación para el desplazamiento

$$EIy = \left(\frac{RA}{6} - \frac{5493}{50} \right) x^3 + \left(\frac{80032274345423366092067}{17592186044416} - \frac{M_A}{2} \right) x^2$$

De igual forma, las constantes de integración son cero debido a las condiciones de frontera. Para encontrar el valor del momento de inercia se considera el valor de x como 1.38 m debido a que ahí se encuentra el momento de flexión maximo, ya que se trata de una viga simétrica. Las condiciones que se establecen para la viga son un acero ASTM A36 que cuenta con un módulo de elasticidad $E = 200000\text{ MPa}$ y se propone una deflexión máxima en la viga de 1 mm .

Al despejar I de la ecuación del desplazamiento se obtiene

$$I = \frac{5,751617 \times 10^{11} \text{ Nmm}^3}{(200000 \frac{\text{N}}{\text{mm}^2})(1\text{ mm})} = 143790,425\text{ mm}^4 = 14,38\text{ cm}^4$$

Y de igual forma, revisar en los catálogos de los fabricantes de aceros [73]. Se deberá elegir el momento de inercia inmediato superior, para este caso, se encontró que el perfil adecuado es un cuadrado tubular de 2 in por 2 in en calibre catorce (véase Anexo D). Posteriormente, se realizó un análisis estático de la viga modelada en el software de SolidWorks®. Las condiciones del estudio fueron restricciones de tipo geometría fija en las caras de los extremos y una mallado basado en curvatura.

Se localizó cada fuerza en la posición exacta mediante cargas remotas, cada una con un valor de 36.62 N , generando una un esfuerzo de von Mises máximo de 22.474 MPa y una deflexión máxima de 0.972 mm mostrada en la figura 63, siendo ligeramente menor que la propuesta de 1 mm , por lo que se concluye que el sistema es capaz de resistir las cargas de operación y validando el diseño de la viga B.

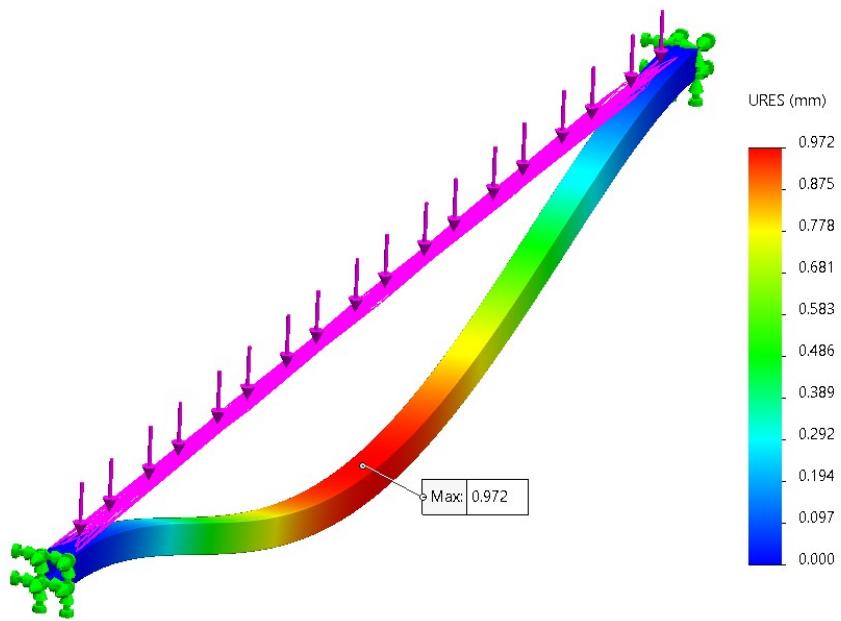


Figura 63: Deflexión máxima de la viga *B*.

2.2.4. MF4. Módulo de interfaz de usuario

Experiencia de usuario y flujo de trabajo La interfaz de usuario de nuestra aplicación cuenta con un estilo de diseño neumorfista, respondiendo a las tendencias de diseño actuales, lo que genera una seguridad de uso para una aplicación moderna de carácter profesional, con una estética atractiva, consistencia visual, pero sobre todo interactividad y enfoque en el funcionamiento, permitiendo a los usuarios interactuar con el sistema de manera eficiente [74]. A continuación, se describe el flujo de trabajo de la interfaz de usuario 64.

Al iniciar el programa, el sistema verifica si hay datos de estado anteriores disponibles para mostrar:

Inicio del Programa: El sistema comienza verificando si existen datos de estado previos.

Datos de Estado Anteriores:

- **Sí:** Si hay datos de estado anteriores, se despliegan las condiciones actuales del cuarto, incluyendo la fecha del último acondicionamiento (MF4.1.).
- **No:** Si no hay datos de estado previos, se inicia el proceso de lectura de datos actuales (MF4.2.).

Una vez desplegadas las condiciones actuales del cuarto, el sistema se encuentra en estado de reposo, el usuario puede elegir entre las acciones:

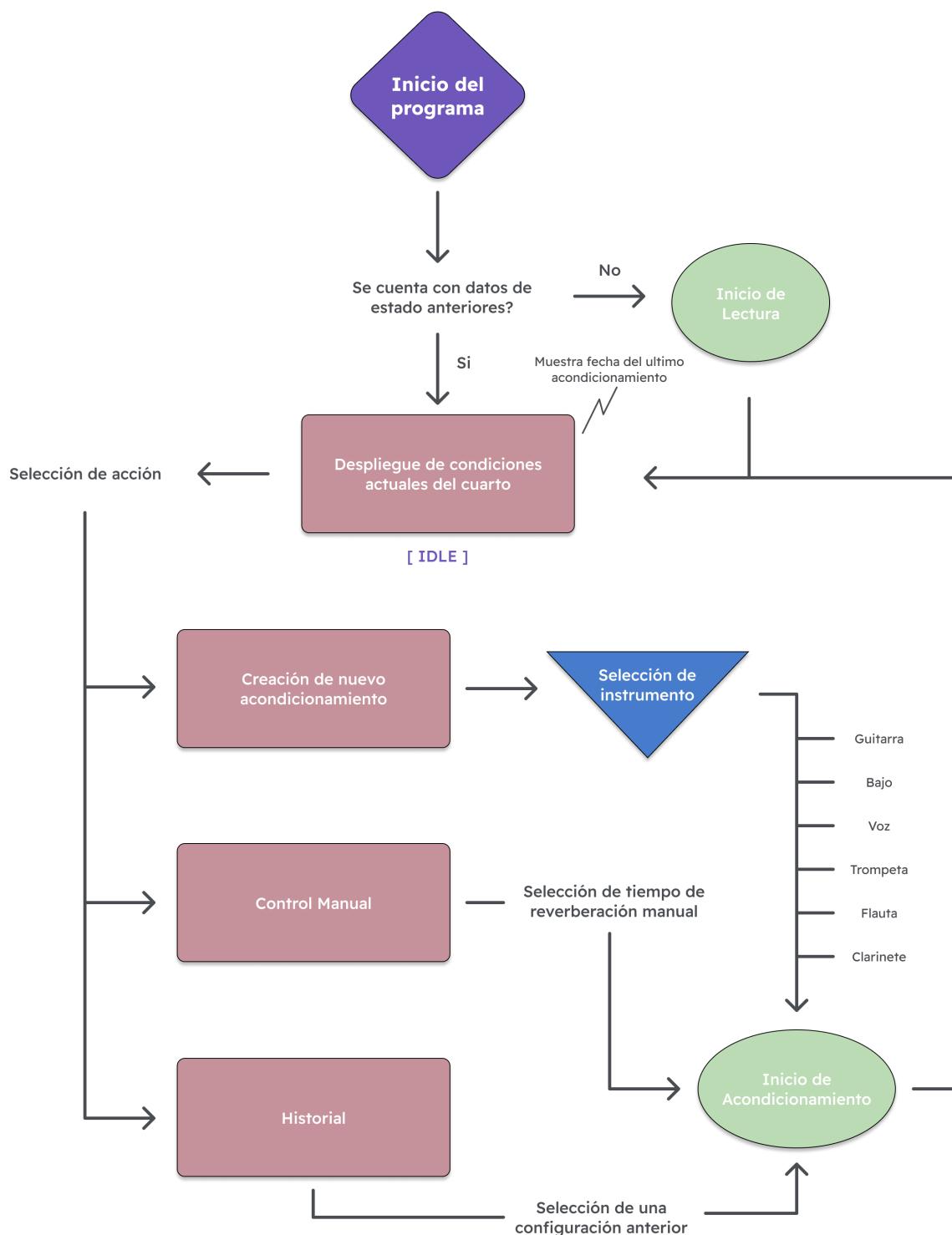


Figura 64: Diagrama de Flujo de la aplicación

Creación de Nuevo Acondicionamiento:

- El usuario selecciona esta opción para iniciar un nuevo acondicionamiento.
- **Selección de Instrumento:** El usuario elige el tipo de instrumento (Guitarra, Bajo, Voz, Trompeta, Flauta, Clarinete).
- **Inicio de Acondicionamiento:** Una vez seleccionado el instrumento, se inicia el acondicionamiento del cuarto.

Control Manual:

- El usuario puede ajustar manualmente el tiempo de reverberación.
- **Inicio de Acondicionamiento:** Despues de ajustar manualmente, se procede con el acondicionamiento.

Historial:

- El usuario puede revisar el historial de acondicionamientos previos.
- **Selección de Configuración Anterior:** El usuario puede seleccionar una configuración anterior para replicarla.
- **Inicio de Acondicionamiento:** Una vez seleccionada, se inicia el acondicionamiento.

La interfaz está compuesta por elementos que facilitan la interacción del usuario con el sistema, estos son:

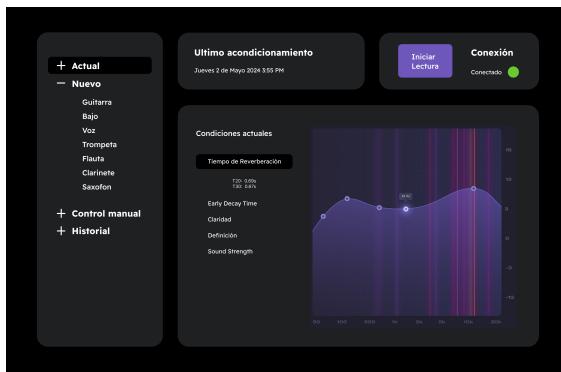
Panel de Navegación: Situado en el lado izquierdo de la pantalla, este panel permite al usuario navegar entre las diferentes secciones de la aplicación, como la visualización de condiciones actuales, la creación de nuevos acondicionamientos, el control manual y el historial.

Botón de Iniciar Lectura: Este botón permite al usuario iniciar la lectura de las condiciones actuales del cuarto en cualquier momento.

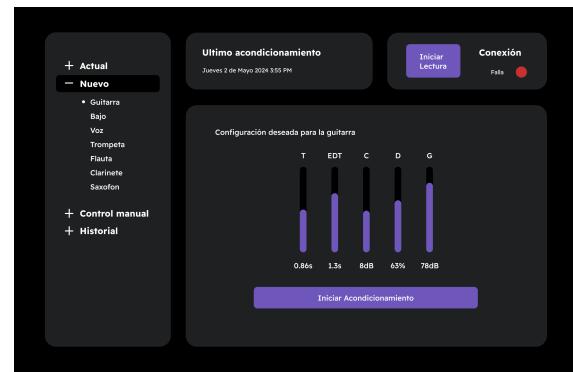
Estado de Conexión: Un indicador de conexión muestra si el sistema está conectado correctamente o si ha ocurrido algún fallo.

Principal: Donde se encuentra la información de las diferentes acciones accesibles por el panel de navegación. Es la sección principal de la interfaz y varía su contenido según la página en la que se esté ubicado.

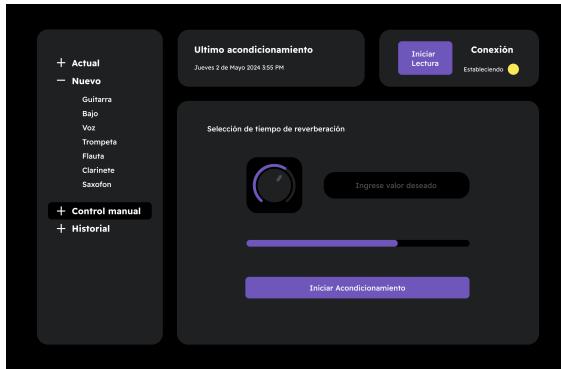
Finalmente, podemos observar las imágenes en la Figura 65 de las diferentes vistas de la interfaz.



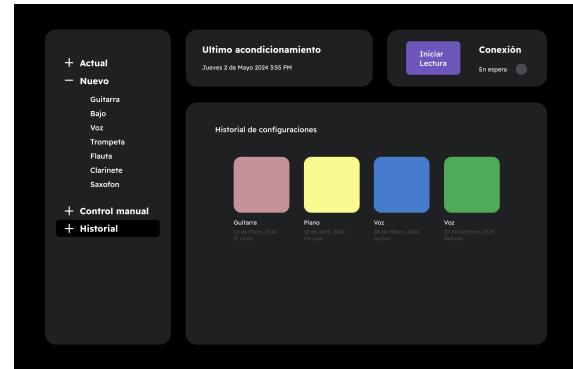
(a) Página de la acústica actual



(b) Página generación de una nueva acústica



(c) Página de control manual



(d) Página del historial de configuraciones

Figura 65: Vistas de la interfaz de usuario

2.2.4.1 Tiempos de reverberación deseados

Preguntarse cuál es el tiempo de reverberación óptimo para un cuarto es una pregunta complicada, y la respuesta corta sería *"depende"*.

El tiempo de reverberación óptimo en un cuarto está dado por el uso que se le dará al cuarto. Incluso dentro de los recintos provistos para la música, el tiempo de reverberación óptimo cambia en función del tipo de música e instrumento que se busque tocar. [75]

Por ejemplo, en iglesias grandes, donde el instrumento principal es el órgano, el tiempo de reverberación puede ir desde los 8 hasta los 11 segundos, mientras que en los salones se busca que este sea menor a 1 segundo. Las salas de grabación para la radio suelen tener tiempo de reverberación no mucho mayores a 0.2 segundos, mientras que algunas bandas prefieren tiempos de 3.5 segundos para que sus instrumentos suenen más "fuertes". [75]

Podemos hacer referencia a la figura 66 [76], en la que se puede observar el tiempo de reverberación recomendado en función del tipo de música y del volumen del cuarto. En nuestro caso, el volumen es de $88m^2$, por lo que tenemos un rango que va desde los 0.35 (Para música amplificada) hasta los 0.85 segundos (Para música silenciosa) para el tiempo

de reverberación.

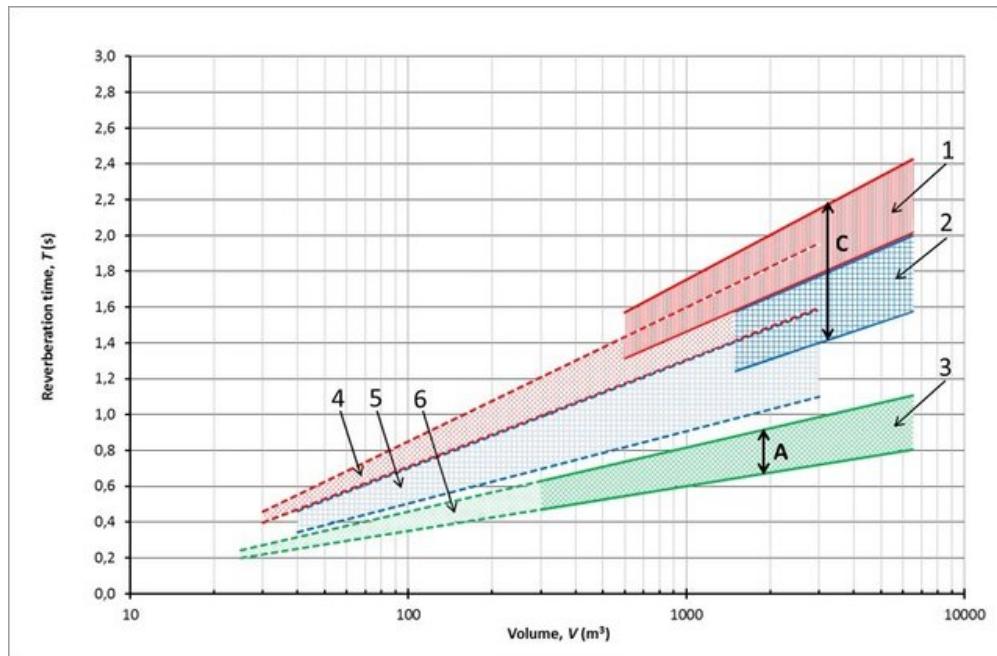


Figura 66: Coeficientes de absorción de los paneles recubiertos

Cabe resaltar que estos valores son para un recinto que se graba lo menos reverberante que se pueda, y se aplican posteriormente los filtros necesarios de manera digital para hacer la música más "potente." como si se hubiera grabado en un recinto mucho más espacioso. Ya que la justificación del proyecto está relacionada con permitir reverberaciones más altas para disminuir el uso de filtros digitales posteriores; los tiempos de reverberación que se podrán medir en el cuarto deben tener un rango más amplio.

De la misma figura 66 podemos ver que para los recintos más grandes, el rango del tiempo de reverberación, va desde los 0.8 segundos (Para música amplificada) hasta los 2.4 segundos (Para música silenciosa).

Si se hacen los cálculos mediante el programa de *PanelPositions*, llevar el tiempo de reverberación de 2.4 segundos hasta los 0.8 segundos, requiere de 31 paneles. De acuerdo a nuestro diseño de 9 paneles por pared, esto requeriría implementar nuestro sistema en 4 paredes, lo cual involucra un elevado costo. Se decidió entonces probar con el límite superior de la música amplificada, el cual es de 1.1 segundos. El programa nos dice que son necesarios 18 paneles para llevar el tiempo de reverberación de 2.4 segundos hasta los 1.1 segundos (esto debido a la naturaleza de la absorción, en la que para reducir a la mitad el tiempo de reverberación es necesario duplicar la absorción). Estos 18 paneles se ajustan muy bien a nuestro diseño, ya que nos permite implementar el sistema en únicamente dos paredes, reduciendo el costo a la mitad y estando aún dentro del límite aceptable para la música amplificada.

Por último, la modificación del tiempo de reverberación para cada instrumento dependerá del tipo de música a la que pertenezca cada instrumento, por ejemplo, una guitarra eléctrica caerá en los 1.1 segundos, mientras que una guitarra podrá ir desde los 2 hasta los 2.4 segundos de acuerdo con la figura 66.

Composiciones tecnológicas de la interfaz Para el desarrollo ya de una manera técnica, hemos seleccionado tecnologías que nos permiten crear una aplicación robusta, eficiente y de fácil mantenimiento [77]. La tecnología principal que utilizaremos es Electron, complementada con otras herramientas y marcos de trabajo.

Electron

Descripción: Electron es un marco de trabajo para construir aplicaciones de escritorio utilizando tecnologías web HTML, CSS y JavaScript. Permite crear aplicaciones multiplataforma que pueden ejecutarse en Windows, macOS y Linux desde una única base de código [78].

Justificación:

Ventaja	Descripción
Multiplataforma	Electron permite desarrollar una sola vez y desplegar en múltiples sistemas operativos, lo cual reduce el tiempo y los costos de desarrollo.
Tecnologías Web	Al utilizar HTML, CSS y JavaScript, podemos aprovechar la vasta cantidad de recursos, bibliotecas y herramientas disponibles en el ecosistema web.
Facilidad de Desarrollo	Electron simplifica la creación de interfaces de usuario atractivas y funcionales, lo que nos permite centrarnos en la lógica de negocio y la experiencia del usuario.
Comunidad y Soporte	Electron cuenta con una amplia comunidad de desarrolladores y una extensa documentación, facilitando la resolución de problemas y la implementación de características avanzadas.

Tabla 98: Selección de marco de trabajo: Electron

React

Descripción: React es una biblioteca de JavaScript para construir interfaces de usuario. Permite crear componentes reutilizables que gestionan su propio estado, lo que facilita el desarrollo de aplicaciones dinámicas y responsivas [79].

Justificación:

Ventaja	Descripción
---------	-------------

Componentización	React permite dividir la interfaz de usuario en componentes reutilizables, lo que mejora la mantenibilidad y la escalabilidad del código.
Virtual DOM	React utiliza un DOM virtual para minimizar las actualizaciones en el DOM real, lo que mejora significativamente el rendimiento de la aplicación.
Comunidad y Ecosistema	React tiene una gran comunidad y un ecosistema robusto, con numerosas bibliotecas y herramientas que facilitan el desarrollo.
Aprendizaje y Uso	React es relativamente fácil de aprender y usar, con una sintaxis declarativa que simplifica la creación de interfaces de usuario complejas.

Tabla 99: Selección de librería de desarrollo principal: React

Redux

Descripción: Redux es una biblioteca de JavaScript para gestionar el estado de la aplicación de manera predecible. Comúnmente acompañada de React [80].

Justificación:

Ventaja	Descripción
Manejabilidad del Estado	Redux proporciona una manera clara y predecible de gestionar el estado de la aplicación, facilitando la depuración y el mantenimiento del código.
Escalabilidad	Redux es adecuado para aplicaciones grandes y complejas donde el manejo del estado puede volverse complicado.
Ecosistema	Redux tiene un amplio ecosistema de herramientas y middleware que facilitan tareas comunes como la asincronía y la persistencia del estado.

Tabla 100: Selección de gestor del estado: Redux

Styled Components

Descripción: Styled Components es una biblioteca de JavaScript que permite escribir estilos CSS dentro del código JavaScript utilizando una técnica llamada CSS-in-JS [81].

Justificación:

Ventaja	Descripción

Encapsulación	Styled Components permite encapsular estilos dentro de los componentes, evitando conflictos de nombres y facilitando la mantenibilidad.
Dinamismo	Permite aplicar estilos dinámicos basados en props y estado, lo cual es ideal para interfaces de usuario interactivas.
Integración	Styled Components se integra perfectamente con React, permitiendo una experiencia de desarrollo fluida y consistente.

Tabla 101: Selección de herramienta de estilizado: Styled Components

D3.js

Descripción: D3.js es una biblioteca de JavaScript para la visualización de datos que permite manipular documentos basados en datos [82].

Justificación:

Ventaja	Descripción
Flexibilidad	D3.js permite crear visualizaciones de datos altamente personalizadas y interactivas.
Actualización Dinámica	Facilita la actualización dinámica de las visualizaciones en respuesta a cambios en los datos.
Comunidad y Recursos	D3.js cuenta con una amplia comunidad y una gran cantidad de recursos y ejemplos disponibles para aprender y resolver problemas.

Tabla 102: Selección de herramienta de visualización: D3.js

La combinación de Electron con React, Redux, Styled Components y D3.js funciona en sinergia de la siguiente manera: Electron para desarrollar la aplicación multiplataforma, React y Redux facilitan la construcción de la interfaz de usuario dinámica y gestionada de manera eficiente. Styled Components asegura que nuestros estilos sean modulares y mantenibles, y por último D3.js nos permite ofrecer visualizaciones de datos ricas e interactivas [83]. Esta selección de tecnologías está cuidadosamente diseñada para satisfacer las necesidades de nuestro proyecto, garantizando tanto la experiencia de desarrollo, como el resultado para el usuario [84].

2.2.5. MF5. Módulo gestor de energía

2.2.5.1 MF5.1 Submódulo de conversión y regulación de la tensión de entrada

Para el diseño del módulo gestor de energía primero se debió de tener en consideración todos los componentes eléctricos y electrónicos que serán utilizados para poner en funcionamiento todo el sistema, además de los valores de tensión y corriente a los que trabajan. Se

tuvo en consideración que se tienen componentes que trabajan con corriente alterna y otros componentes que trabajan con corriente directa, por lo que se debió de regular y distribuir diferentes valores de tensión a los diferentes módulos del sistema, como son el módulo de procesamiento, el módulo de generación y medición de la acústica y sobre todo el módulo modificador de la acústica, en el cual se concentra la mayor parte de los componentes electrónicos, los cuales nos permiten controlar tanto el movimiento del motor como el acople de los prismas a la corona para poder girar dichos prismas tanto como sea necesario.

Elección del solenoide Para poder acoplar y desacoplar los paneles de forma correcta y cuando el sistema lo indique, se optó por usar un mecanismo, el cual utiliza solenoides lineales. Para realizar la selección correcta de dichos componentes, primero se debió tener la fuerza que se necesita para poder mover el mecanismo de acople y la medida que debe de tener el émbolo móvil del solenoide, las cuales, de acuerdo con los cálculos, son de 6.97N y 10mm respectivamente.

Una vez realizados los cálculos, se procedió a realizar una búsqueda en internet para poder encontrar un solenoide lineal que cumpliera con las características deseadas y que además tuviera un bajo costo. Existieron muchas opciones, las cuales, se podían encontrar en las páginas de Amazon y Mercado Libre, pero dichos solenoides no tenían la fuerza requerida, y aquellos que si contaban con la fuerza requerida resultaban tener precios elevados, lo cual no es conveniente para los costos totales, por lo que se procedió a realizar la búsqueda de los componentes en AliExpress, en donde encontramos una gran cantidad de estos componentes a precios accesibles, de entre los cuales destacan los siguientes:

Nombre	Imagen	Características	Ventajas y desventajas
Electroimán de marco tipo push & pull, TAU0630T[85]		<ul style="list-style-type: none"> ■ Voltaje: 12Vdc-24Vdc ■ Corriente: 12Vdc-2A; 24Vdc/1A ■ Potencia: 24W ■ Distancia de viaje: 10mm ■ Fuerza: 10Kg = 9.81N ■ Precio: \$25 	<ul style="list-style-type: none"> ■ Ventajas <ul style="list-style-type: none"> • Precio bajo • Fuerza mayor a la necesaria ■ Desventajas <ul style="list-style-type: none"> • Corriente elevada • Potencia elevada

Solenoide lineal tipo Push Pull, 0730B[86]		<ul style="list-style-type: none"> ■ Voltaje: 12Vdc - c/24Vdc ■ Corriente: 12Vdc - c/600mA; 24Vdc - c/300mA ■ Potencia: 7.2W ■ Distancia de viaje: 10mm ■ Fuerza: 8N ■ Precio: \$96 	<p>■ Ventajas</p> <ul style="list-style-type: none"> • Potencia baja • Aparente mayor calidad en los materiales de fabricación • Mayor número de valoraciones positivas <p>■ Desventajas</p> <ul style="list-style-type: none"> • Costo elevado • Fuerza justa para realizar el trabajo requerido
--	---	---	--

Tabla 103: Ventajas y desventajas de las opciones para solenoides lineales

El primer solenoide tiene la fuerza de 1kg, lo que convertido a Newtons son 9.81N, por lo que resulta ser suficiente para poder realizar la tarea que se necesita, además, puede funcionar a 12Vdc o 24Vdc, a una potencia de 24W, por lo que dependiendo de cuál voltaje se elija, el solenoide trabajará con 2A o 1A respectivamente, en nuestro caso se decidió trabajar un voltaje de 12Vdc debido a que se planeó desde el principio utilizar una fuente de computadora, las cuales son mas eficientes y entregan voltajes y corrientes controlados, estas fuentes tienen salidas de voltaje ya determinadas, como son 3.3V, 5V y 12V, por lo que utilizar solenoides que trabajen con uno de estos voltajes es la elección más conveniente., sin embargo, el único inconveniente es que se estará trabajando con corrientes más elevadas, como lo son 2A. El costo del primer solenoide es de alrededor de 25 pesos mexicanos, el cual es un precio accesible debido a que se necesitan 18 solenoides en total, dando un costo total de 450 pesos mexicanos. Sin embargo, si se llegaran a girar los prismas al mismo tiempo se tendría una potencia consumida de 432W, lo que provocaría que la potencia de la fuente que seleccionemos sea más grande, incrementando su valor.

El segundo solenoide tiene una potencia menor debido a que según las especificaciones del vendedor, trabajan a 12V, pero a 0.6A, dando una potencia de 7.2W por cada uno de los solenoides, por lo que daría un total 129.6W de potencia consumida si se llegara a utilizar todos los solenoides al mismo tiempo, lo cual reduciría la potencia suministrada por la fuente, al igual que el costo en esta, sin embargo, el costo de estos solenoides es mayor, llegando a costar alrededor de 94 pesos mexicanos, dando un costo total de 1692 pesos mexicanos por los 18 solenoides.

Por lo antes descrito, se decidió elegir el primer solenoide, ya que al seleccionar el primer solenoide el costo total de estos sería tres veces menor en comparación al costo total si se

selecciona el segundo solenoide, además, el inconveniente de la alta potencia se podría solucionar al no acoplar todos los prismas al mismo tiempo, reduciendo así el costo de la fuente. En todas las especificaciones acerca del funcionamiento de los solenoides se recomienda que los ciclos de trabajo de estos no sean prolongados, incluso en las especificaciones del segundo solenoide se puede observar la recomendación de usarlo menos de 30 segundos, sin embargo, dice que lo óptimo sería mantenerlos 1 segundo encendidos y 1 segundo apagados si se necesita un uso continuo de este dispositivo, esto debido a que la resistencia interna de estos componentes es demasiado baja, lo que hace que se calienten al pasar una alta corriente por ellos.

Elección del MOSFET Ahora bien, estos solenoides deben de ser activados en el momento en el que el sistema lo decida, por lo que se deben de utilizar componentes electrónicos como interruptores, los cuales al recibir una señal eléctrica se activen, en este caso podemos utilizar los transistores de unión bipolar (BJT por sus siglas en inglés “bipolar junction transistor”) o los transistores de efecto de campo metal-oxido-semiconductor (MOSFET por sus siglas en inglés “metal-oxide-semiconductor field-effect transistor”), los cuales, en el caso de los BJT, controlan la corriente de salida en función de una corriente de entrada menor, y, en caso de los MOSFETs, controlan la corriente de salida en función de un voltaje de entrada, lo cual lo hace un control más sencillo, además de que pueden resistir voltajes y corrientes demasiado elevadas, por lo que son ideales para una tarea donde necesitamos que se manejen corrientes de 2A y tensiones de 12V. Los MOSFETs pueden ser polarizados de diferentes formas, aunque la forma más común es utilizar un divisor de voltaje conectado a la compuerta (G debido a su nombre en inglés, gate), y dependiendo del voltaje que hay entre la compuerta y la fuente (S debido a su nombre en inglés, source), y de que MOSFET se esté utilizando, este dejará pasar una corriente entre las terminales drenador (D debido a su nombre en inglés, drain) y fuente, la cual es mayor si aumentamos el voltaje entre compuerta y fuente.

Los MOSFETs son componentes muy variados, los cuales pueden soportar tanto voltajes y corrientes pequeñas, del orden de milí amperes y unidades de volts, hasta centenas de amperes y volts, por lo que se podrían haber elegido un MOSFET con características que sobrepasan las requeridas en demasía, como lo es el modelo IRL2203, el cual es un MOSFET de enriquecimiento de canal N que permite tener un voltaje de drenador a fuente de hasta 30V y una corriente de drenador de hasta 116A. Uno de los requerimientos indispensables para la elección del MOSFET fue el valor de su voltaje compuerta-fuente de umbral, el cual es el voltaje mínimo que debe de tener entre compuerta y fuente para que exista una corriente de drenador, este voltaje no debía de ser mayor a 3.3V, y la razón de esto es que el encargado de accionar estos solenoides se propuso que fuera la Raspberry, utilizando los puertos GPIO (General Purpose Input/Output, que en español sería Entrada/Salida de Propósito General), los cuales solo tienen el voltaje de salida antes mencionado, y una corriente de salida de 16mA según las especificaciones, véase en el anexo F.

Fueron considerados muchos MOSFETs como opciones, pero se determinó que el mejor

componente sería el modelo IRL2703, el cual puede soportar voltajes de drenador a fuente de 30V y una corriente de drenador de 24A, por lo que lo hace apto para soportar los valores de corriente y voltaje que se manejarán en el trabajo terminal, aunado a lo anterior, tenemos un voltaje de compuerta a fuente de umbral mínimo de 1V, por lo que se puede controlar este componente con el voltaje de salida de los GPIO de la Raspberry, además, uno de los factores que también se consideró fue la accesibilidad a la compra del producto, debido a que en algunas ocasiones la compra de algunos MOSFETs resulta una tarea difícil.

Diseño de la PCB que controla el accionamiento de los solenoides El siguiente paso fue determinar el punto óptimo de funcionamiento para el IRL2703, para lo cual se debió de caracterizar el MOSFET, encontrando el valor del factor de transconductancia (k), para ello existe la siguiente fórmula.[87]

$$k = \frac{I_{D_{encendido}}}{(V_{GS_{encendido}} - V_{GS_{(th)}})^2} \quad (96)$$

En donde $V_{GS_{(th)}}$ es el voltaje de compuerta a fuente de umbral, aunque en inglés se le denomina “Gate Threshold Voltage” y por eso tiene ese símbolo. En ocasiones, y sobre todo en los MOSFET IRL o IRF, el valor de $I_{D_{encendido}}$ no se encuentra en la hoja de datos del componente, pero si se puede encontrar una gráfica en donde comparan el valor de V_{GS} respecto a I_D y lo que se puede hacer es buscar un punto en donde puedas reconocer de forma clara ambos valores, el cual debe de encontrarse en la región lineal (**región óhmica**) de la gráfica, para posteriormente utilizar la fórmula anterior y obtener el valor de k , aunque en esta ocasión se optó por elegir dos puntos, obtener el valor de k en ambos puntos y posteriormente promediarlo, como se muestra a continuación (véase la figura 3 del anexo E, donde se puede ver la gráfica antes mencionada).

Punto A

$$I_D = 3,5AV_{GS} = 3V$$

$$k = \frac{3,5A}{(3V - 1V)^2}$$

$$k = 0,875 \frac{A}{V^2}$$

Punto B

$$I_D = 0,7AV_{GS} = 2,5V$$

$$k = \frac{2,5A}{(0,7V - 1V)^2}$$

$$k = 0,311 \frac{A}{V^2}$$

Teniendo estos valores se procedió a realizar el promedio

$$k = \frac{0,875 \frac{A}{V^2} + 0,311 \frac{A}{V^2}}{2}$$

$$k = 0,593 \frac{A}{V^2}$$

Se procede a obtener el valor de V_{GS} si queremos que I_D sea de 2A utilizando la fórmula 96 y despejando el valor deseado.

$$V_{GS} = \sqrt{\frac{I_D}{k} + V_{GS_{(th)}}} \quad (97)$$

$$V_{GS} = \sqrt{\frac{2A}{0,593 \frac{A}{V^2}}} + 1V$$

$$V_{GS} = 2,836V$$

Con el valor de V_{GS} proporcionado por el cálculo anterior se puede obtener las resistencias del divisor de voltaje, aunque para ello se debe de proponer una de estas resistencias y calcular el valor de la otra dependiendo del valor de esta, para ello se propone que la resistencia R2 sea de $10M\Omega$, esto para poder mantener la corriente solicitada a la Raspberry lo más baja posible.

$$V_{OUT} = \frac{R_2}{R_1 + R_2} V_{IN} \quad (98)$$

Despejamos R1

$$R_1 = \frac{R_2 * V_{IN}}{V_{OUT}} - R_2 \quad (99)$$

$$R_1 = \frac{10M\Omega * 3,3V}{2,836V} - 10M\Omega$$

$$R_1 = 1,636M\Omega$$

A pesar de todos estos cálculos, al utilizar el simulador Proteus 8, el valor de $V_{GS_{(th)}}$ cambia, dando como resultado que los valores antes calculados cambien, por lo que se volvió a realizar el cálculo de la constante k obteniendo los valores de I_D , V_{GS} y $V_{GS_{(th)}}$ del simulador, llegando a los siguientes resultados.

$$V_{GS} = 3,5V I_D = 6,21A V_{GS(th)} = 2,4V$$

Se obtiene el valor de k

$$k = \frac{6,21A}{(3,5V - 2,4V)^2}$$

$$k = 5,38 \frac{A}{V^2}$$

Se obtiene el valor de V_{GS} si queremos que I_D sea de 2A

$$V_{GS} = \sqrt{\frac{2A}{5,38 \frac{A}{V^2}}} + 2,4V$$

$$V_{GS} = 3,01V$$

Como se puede observar el valor de V_{GS} necesario para obtener una $I_D=2A$ es de 3.01V, si se comparan estos resultados con los que se obtuvieron anteriormente se obtienen valores similares, por lo que el cálculo es correcto, y si se prueban estos valores en el simulador se puede comprobar la veracidad de estos cálculos. Ahora se procede a obtener el valor de las resistencias del divisor de voltaje si queremos un $V_{GS} = 3V$.

$$R_1 = \frac{10M\Omega * 3,3V}{3V} - 10M\Omega$$

$$R_1 = 1M\Omega$$

Una vez que se conocen los componentes que serán utilizados para el accionamiento de los solenoides, corresponde realizar un esquemático de todo el circuito, el cual debe de estar detallado, para posteriormente realizar el diseño de la PCB.

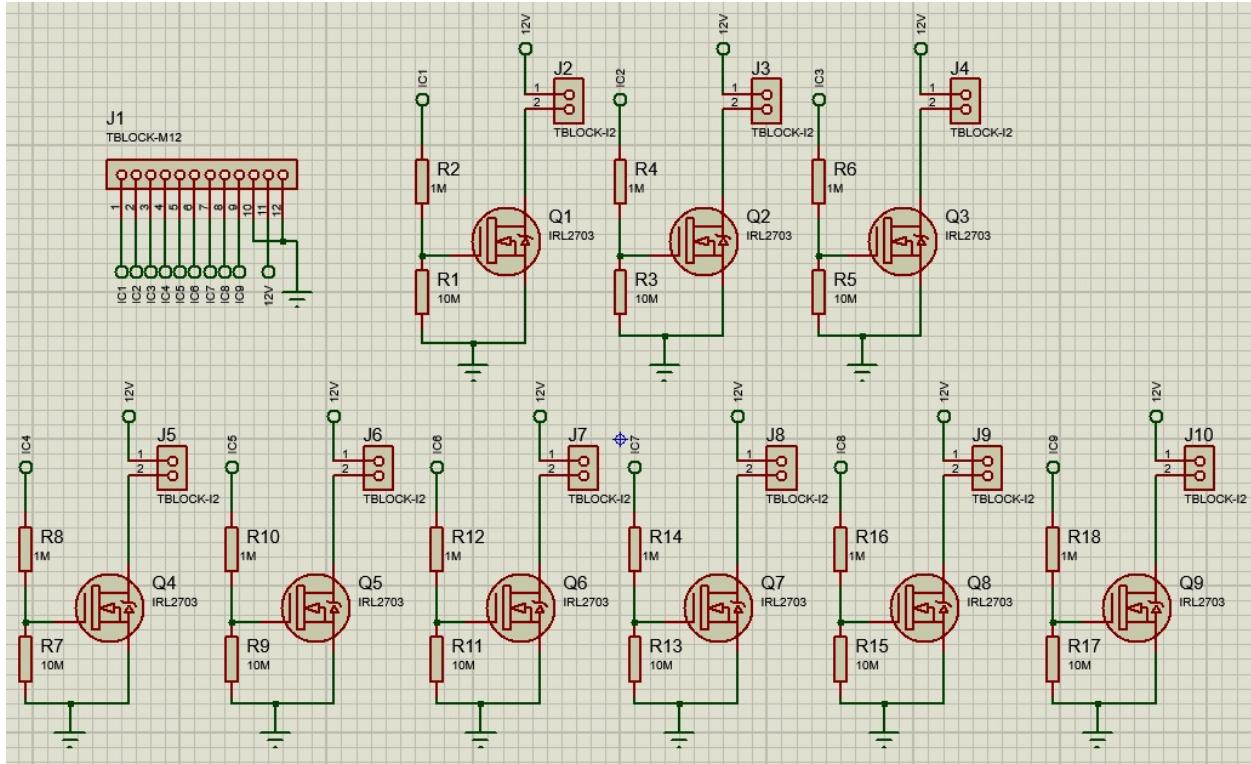


Figura 67: Esquemático del circuito que acciona los solenoides lineales

Como se puede ver en la figura 67, el uso de etiquetas es conveniente para mantener el esquemático claro y conciso, sobre todo cuando se tiene una gran cantidad de componentes electrónicos, los cuales tienen varias conexiones. El esquemático ayuda a visualizar cada uno de los componentes del circuito, así como sus conexiones entre sí y los valores que tienen las resistencias, capacitores e inductores, en caso de que existan en el diseño, pero, si se realizan los esquemáticos en un software que permita realizar diseños de PCBs, el esquemático ayuda en demasía a señalar como se conectan los pads de las huellas de los componentes al momento de colocar estos en el diseño de la PCB.

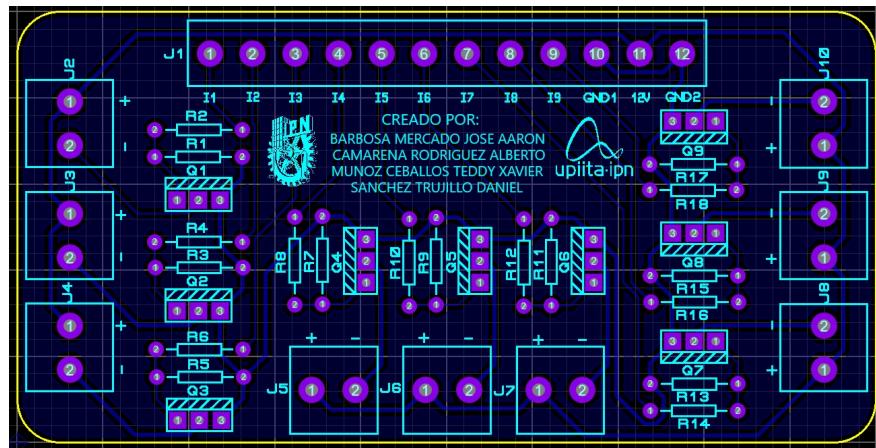
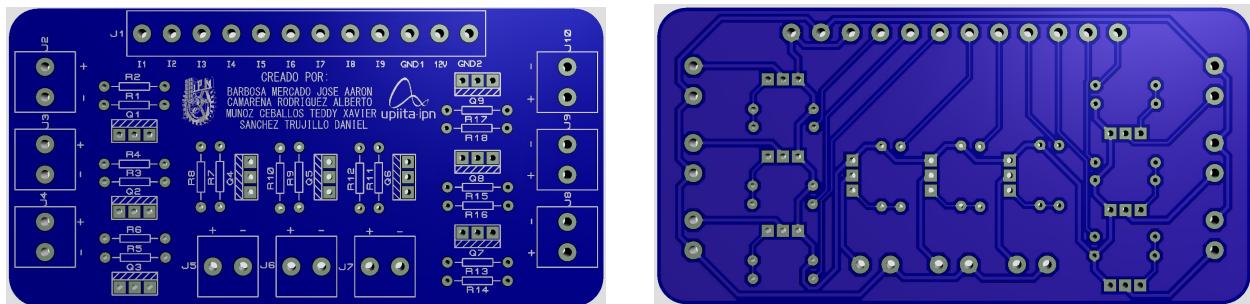


Figura 68: PCB del circuito que acciona los solenoides lineales

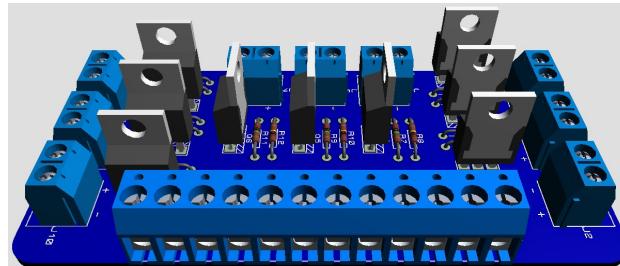
Para realizar el diseño se tuvo en consideración la norma más usada para el diseño de circuitos impresos, la **IPC-2221**, la cual define los requerimientos para el diseño de PCBs, establece unos principios de diseño y recomendaciones, por lo que el diseño resulta ser apropiado para su futura fabricación. Los componentes en la PCB se encuentran distribuidos de forma que las pistas sean lo más cortas posible, además fueron utilizadas borneras para las conexiones de salida y de entrada, teniendo una de 12 conexiones para las entradas de los GPIO y la fuente de 12V, y borneras de 2 entradas para las conexiones de cada uno de los solenoides.[88]

Cada uno de los componentes se encuentra señalado con una letra y un número, al igual que en el esquemático, además, cada una de las conexiones de entrada y salida están claramente señaladas para facilitar la conexión física. Los anchos de pista y la separación entre pistas y entre componentes fue establecida con base en la norma IPC-2221. A continuación se muestra de manera más detallada el diseño de la PCB.**ipc2221**



(a) Parte frontal de la PCB

(b) Parte posterior de la PCB



(c) PCB con los componentes sobreuestos

Figura 69: Modelo 3D de la PCB

Además del diseño en 3D también se tienen una imagen de las pistas, la cual sirve para poder fabricar la PCB por el método de planchado, esta solo es ilustrativa, ya que aún no se decide el método de fabricación que se utilizará.

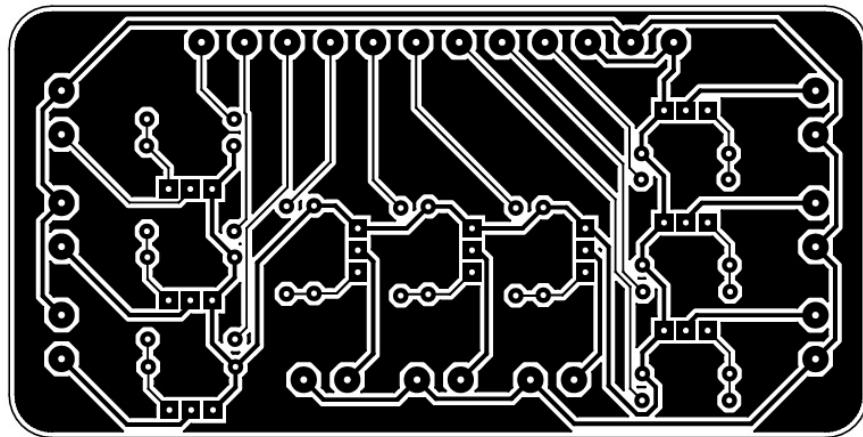


Figura 70: PCB del circuito que acciona los solenoides lineales

Selección del motor Lo siguiente en el diseño del módulo de gestión de energía es la selección del motor, aunque esta selección depende de cálculos realizados al momento de diseñar del módulo modificador de la acústica. Los valores que se requieren son la potencia necesaria para mover el sistema, la cual fue de 282.89W, el torque necesario, siendo de 1.539Nm, y la velocidad a la que debe girar el motor para poder mover los prismas a la velocidad deseada, siendo de 640 revoluciones por minuto (rpm), por lo que el motor que se busca debe de contar con estas características. La búsqueda de este componente resultó tener cierta dificultad debido a que no se encontraron motores que cumplieran con el torque requerido, por lo que se decidió buscar un motor con la velocidad elevada y el torque lo más alto posible para posteriormente reducir la velocidad con una relación de transmisión, lo que permitirá aumentar el torque que proporciona el motor. La mejor opción encontrada fue la que se muestra a continuación.



Figura 71: Motor seleccionado[70]

Este motor tiene dos configuraciones de velocidades, la primera de 1800 rpm y la segunda de 3000 rpm, en este caso se eligió la opción que cuenta con 1800 rpm, debido a que el torque que proporciona el motor con esta velocidad y funcionando a 24V es de 8.5 kgf.cm,

que convertido a newton metro es de 0.8336Nm, sin embargo, el torque sigue siendo insuficiente para los requerimientos del sistema, por lo que se optó por utilizar una relación de transmisión, disminuyendo la velocidad para poder aumentar el torque. Si se divide el torque que se necesita (1.539Nm) entre el torque proporcionado por el motor (0.8336Nm) da una relación de 1.846 y si se divide la velocidad del motor entre la relación de transmisión da como resultados la velocidad máxima que llegaría al tornillo sin fin, aunque las velocidades del motor antes mencionadas solo son alcanzadas cuando no se tiene cargas, por lo que las velocidades al estar cargado el motor disminuyen, siendo precisos, el motor tiene una velocidad cuando se encuentra cargado de 1680 rpm, por lo que se necesitaría una transmisión de 1:2 o un poco mayor para aumentar el torque hasta los niveles requeridos y la velocidad máxima que tendría el tornillo sin fin sería de 840rpm, suficiente para los requerimientos del sistema.

Model	Motor characteristics table								
	Voltage DC(V)	Power (W)	no-load speed		Load parameters			Brush life (H)	Motor weight (kg)
			speed (RPM)	current (A)	speed (RPM)	moment (kg.cm)	current (A)		
5D300CN-G-12-18S	12	300	1800	2.75	1650	7.5	30	3000	3.2
5D300CN-G-12-30S			3000	3.10	2600	8.1	30		
5D300CN-G-24-18S			1800	1.67	1680	8.5	15		
5D300CN-G-24-30S			3000	1.89	2800	9.2	18		

Figura 72: Información de las características del motor proporcionadas por el vendedor[89]

De acuerdo con la figura 72, el motor trabaja con una corriente de 15A y ,ya que el motor funciona a un voltaje de 24V, se puede obtener la potencia de este, la cual es igual a 360W, el cual es un número mayor al establecido en la información del vendedor, sin embargo, consideramos que el valor de corriente proporcionado es el máximo, por lo que la corriente suministrada a este competente debe de ser mayor o igual a 18A.

Debido a que la fuente de alimentación solo proporciona valores de hasta 12V se necesita usar un convertidor de 12Vdc a 24Vdc, debido a esto se realizó una búsqueda en internet de este dispositivo, encontrando la siguiente opción.



Figura 73: Convertidor de 12Vdc a 24Vdc[90]

El convertidor da 12V de salida y puede dar hasta 20A de corriente, por lo que es ade-

cuado para poder accionar el motor. Hay que tener en consideración que se construirán dos paneles, por lo que se requiere dos motores y dos convertidores de voltaje en total a menos que se decida realizar el movimiento de los paneles de forma asíncrona, moviendo un panel y posteriormente el otro, en cuyo caso solo se necesitaría un convertidor.

Para poder controlar la velocidad del motor se consideraron varios circuitos, sin embargo, la corriente a la que funciona el motor es demasiado elevada para poder utilizar una PCB, por lo que se optó por buscar un dispositivo que nos ayudara con esta tarea, y después de una larga búsqueda se encontró lo siguiente.

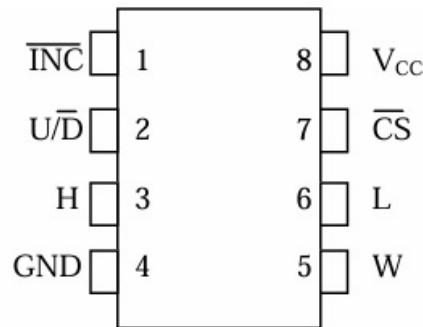


Figura 74: Driver para controlar la velocidad del motor[91]

El dispositivo de la figura 74 es un driver que puede controlar la velocidad del motor modulando el ancho de pulsos que llega a este, lo cual se consigue moviendo el potenciómetro que se muestra en la figura, sin embargo, para poder conseguir que el sistema sea automatizado se tendrían que realizar modificaciones al dispositivo, tales como remplazar el potenciómetro físico por uno digital, como el que se muestra a continuación.



(a) Potenciómetro digital real



(b) Diagrama del potenciómetro digital

Figura 75: Potenciómetro digital modelo DS1804-100[92]

Este dispositivo trabaja de forma incremental, a diferencia de algunos que trabajan por medio de comunicación SDI. Para variar la resistencia del potenciómetro se debe de conectar una señal cuadrada al pin INC, el potenciómetro detectará los flancos de bajada y disminuirá o aumentará la resistencia dependiendo de si el pin U/D se encuentra con una señal en alto o en bajo, si la señal está en bajo la resistencia baja, y si la señal se encuentra en alto, la resistencia aumenta. Las terminales H y L son high y low respectivamente, lo que quiere decir que la fuente va conectada a H a la tierra a L, siendo W la salida del potenciómetro. El pin CS debe de mantenerse conectado a tierra todo el tiempo, el pin Vcc debe de estar alimentado con 5V y el pin GND debe de estar conectado a tierra.

Selección del encoder Ahora bien, para tener un control preciso del sistema se debe saber la posición de los prismas, lo cual es posible de hacer con el uso de encoders, sin embargo, al tener una gran cantidad de prismas resulta poco conveniente utilizar un encoder para cada uno, por lo que se ideó la manera de medir la posición de los paneles solo con medir la posición del tornillo sin fin, reduciendo el número de encoders a uno por panel, lo que es benéfico para el costo final del proyecto, sin embargo, la elección del encoder tuvo que realizarse de forma correcta, ya que existen varios tipos de encoders los cuales trabajan de diferentes formas.

Tipo	Funcionamiento	Características
Óptico	Utilizan una fuente de luz y un disco codificado para generar señales eléctricas.	Son conocidos por su alta resolución y precisión, y se utilizan en aplicaciones que requieren un control muy preciso, como en máquinas, herramienta y sistemas de posicionamiento de alta precisión.

Magnético	Utilizan un sensor magnético y un disco magnético para detectar el movimiento del eje.	Son más robustos y menos susceptibles a las condiciones ambientales adversas, como la suciedad y la humedad, lo que los hace adecuados para entornos industriales exigentes.
Capacitivos	Emplean un sensor capacitivo para medir el cambio en la capacitancia causado por el movimiento del eje.	Son adecuados para aplicaciones en las que se requiere una alta precisión y una respuesta rápida, como en sistemas de control de velocidad de motores.
Inductivos	Usan un sensor inductivo para detectar el movimiento del eje mediante cambios en el campo magnético.	Son resistentes a las condiciones ambientales adversas y se utilizan en aplicaciones donde se requiere una alta fiabilidad y durabilidad, como en sistemas de control de elevadores y grúas.

Tabla 104: Comparativa entre los diferentes tipos de encoders[93]

Los encoders más comunes que se pueden encontrar en internet son los encoders ópticos, por lo que son los que fueron seleccionados para el sistema, si embargo, existen dos tipos de encoders ópticos, los incrementales y los absolutos, los cuales miden de forma diferente la posición y la velocidad, teniendo ventajas y desventajas uno con respecto del otro, no obstante, el elevado costo y la poca accesibilidad a los encoders absolutos hace que la decisión final sean encoders incrementales, de los cuales hay una gran variedad de opciones en el mercado.

Para realizar la elección del encoder se tuvieron en consideración la velocidad máxima a la que gira el tornillo sin fin (640rpm) y la resolución que se buscaba, la cual es el movimiento más pequeño detectado por el encoder, y se mide en pulsos por revolución (ppr), por consiguiente, si el mínimo movimiento que se desea medir es el de un grado, la resolución debe de ser de 360 ppr, sin embargo, si se tienen dos señales cuadradas como salida del encoder, las cuales están desfasadas 90 grados, es posible medir los flancos de subida y de bajada de ambas señales para obtener una resolución de 1440 pulsos, lo que permitiría medir cambios de hasta un cuarto de grado. El encoder que se escogió fue un encoder rotativo fotoeléctrico de 360 ppr como el que se muestra a continuación.



Figura 76: Encoder seleccionado[94]

Según las especificaciones del vendedor, el encoder trabaja con una tensión de entrada de 8V a 24V, consume corrientes de 60mA hasta 150mA y puede llegar alcanzar velocidades de 5000 rpm, por lo tanto, este encoder es adecuado para cubrir las necesidades que tiene el sistema. La relación de transmisión que se tiene entre el tornillo sin fin y la corona es de **1:32**, por lo que, haciendo los cálculos, se llega a la conclusión que para girar 1 grado la corona, se debe de girar 32 grados el tornillo sin fin, los cuales son más fáciles de medir si se tiene un encoder de 360 ppr. Para realizar el acople del encoder y el tornillo sin fin se deberá de realizar una modificación en el tornillo sin fin en donde pueda acoplarse el eje del encoder.

Elección de la fuente Una vez que se eligieron los demás componentes del submódulo de conversión y regulación de la tensión de entrada se procedió a realizar la elección de la fuente de poder, la cual debía de ser una fuente de poder de las que se utilizan en computadoras, debido a que estás están diseñadas para soportar sobrecorriente, sobretensión y cortocircuito, además de que algunas de las fuentes de poder cuentan con la certificación 80 PLUS, la cual es una certificación que consiste en hacer pruebas de las fuentes de alimentación de una PC para comprobar su eficiencia al 20%, 50% y 100% de las cargas, en cada una de las pruebas la eficiencia de la fuente debe de ser mayor a 80%. La certificación 80 PLUS fue iniciado en 2004 por Ecova Plug Load Solutions y al principio las fuentes de alimentación solo necesitaban un 80% de eficiencia para obtener la certificación. Con el paso de los años, a medida que las fuentes de alimentación se fueron haciendo cada vez más eficientes, se crearon nuevos estándares.

La diferencia entre las seis certificaciones 80 Plus disponibles se resume en la figura 77. Los números expresados representan la eficiencia mínima que una fuente de alimentación debe ofrecer para cada carga con el objeto de obtener una certificación dada.[95]

Certificación	NIVEL DE CARGA			
	10%	20%	50%	100%
80 Plus White*		82%	85%	82%
80 Plus Bronze		85%	88%	85%
80 Plus Silver		87%	90%	87%
80 Plus Gold		90%	92%	89%
80 Plus Platinum		92%	94%	90%
80 Plus Titanium		90%	94%	96%
EFICIENCIA REQUERIDA				

Figura 77: Diferentes certificaciones 80 PLUS[95]

Teniendo en consideración lo anterior, primero se procedió a calcular la potencia total consumida por todos los componentes electrónicos, empezando por los solenoides lineales, los cuales consumen cada uno 24W, siendo un total de 18 solenoides lo que da como resultado una potencia consumida de 432W si son utilizados todos los solenoides al mismo tiempo. Los motores tienen una potencia consumida de 300W según sus especificaciones, siendo necesarios

dos motores, daría una potencia total consumida de 600w al utilizarlos al mismo tiempo. Los encoders cuentan con un consumo de corriente máximo de 1.5W, siendo dos en total, da una potencia total consumida de 3W. Si se realiza la suma de todas las potencias consumidas se obtiene un total de 1035W, lo cual es un valor elevado para una fuente de PC. A pesar de la existencia de fuentes de poder que pueden suministrar la potencia antes mencionada, su costo resulta ser demasiado elevado, por lo se optó por cambiar el funcionamiento del sistema, moviendo los paneles de forma asíncrona, primero moviendo los prismas de un panel y posteriormente los prismas del otro, lo que reduciría la potencia total consumida a la mitad, dando un total de 517.5W, aunque para poder realizar una correcta elección de la fuente se debe de multiplicar la potencia total consumida del sistema por un factor de diseño debido a que las fuentes a seleccionar tienen una eficiencia de alrededor de 80 % por lo que se decidió que el factor de diseño fuera de 1.2, dando una potencia total de 621W, siendo las fuentes de 650W aquellas con una potencia superior más próxima a la calculada. Haciendo una búsqueda en internet, se pudo encontrar la siguiente fuente de poder (véase toda la información de la fuente en el anexo G).



Figura 78: Fuente de Poder MSI MAG A650BN, 650W[96]

La fuente anterior tiene una potencia de 650W y una certificación 80 PLUS Bronze por lo que tiene una eficiencia de del 85 %, siendo suficiente para cubrir las necesidades del sistema, además de ser de una marca reconocida como lo es MSI y tener buenas reseñas de los compradores. Por todas las razones antes mencionadas se llegó a la conclusión que la elección de esta fuente fue correcta.

2.2.5.2 MF5.2 Submódulo de medición del consumo energético

El consumo eléctrico o energético es aquella cantidad de energía consumida en un determinado periodo de tiempo y se mide en kilovatios, hora (kWh)[97]. Una forma de tomar control de este consumo y reducir la factura de electricidad es a través del uso de medidores de consumo eléctrico, el cual es un dispositivo que se utiliza para medir la cantidad de energía eléctrica real que se consume en un hogar, negocio o sistema. Estos dispositivos se conectan a la red eléctrica y miden la cantidad de energía eléctrica que fluye a través del cableado eléctrico.[98]

Las ventajas del monitoreo de la energía eléctrica van desde controlar en tiempo real los datos de consumo, potencia y tensión, identificar y solucionar problemas, hasta disminuir y controlar la cantidad de energía utilizada por los componentes.[99]

El uso de un medidor del consumo eléctrico beneficia a nuestro sistema, ya que con él se puede observar la corriente y el voltaje consumido, así como la potencia consumida. Con la visualización de los parámetros eléctricos antes mencionados se puede tener una idea del consumo del sistema en tiempo real, y al momento de probar diferentes formas de como mover los primas de los paneles, ya sea de uno por uno o varios al mismo tiempo, y los tiempos de operación del motor y los solenoides, se podrá visualizar cuál es la manera más eficiente de trabajo.

Para poder realizar el monitoreo del consumo eléctrico se debe de elegir un medidor, el cual sea capaz de soportar la medición de los valores máximos de los parámetros eléctricos de todo el módulo energético. Se realizó una búsqueda exhausta en internet para poder encontrar un dispositivo capaz de cumplir con las necesidades del sistema, y después de un tiempo se encontró el siguiente medidor de consumo eléctrico.



Figura 79: Medidor de consumo eléctrico[100]

El medidor mostrado en la figura 79 es capaz de medir y mostrar 6 parámetros, siendo voltaje, corriente, potencia activa, energía, frecuencia y factor de potencia, además de tener una función de alarma de sobrecarga, la cual se activa si se sobrepasa el umbral de potencia, cuyo valor puede ser reconfigurado. El medidor puede trabajar a voltajes de 110Vac o 220Vac, medir valores de corriente 100A máximo y valores de potencia máximos de 22000W, los cuales son valores límite adecuados para la medición del consumo eléctrico de todo el sistema. La forma en que se planea conectarse el medidor es la siguiente.

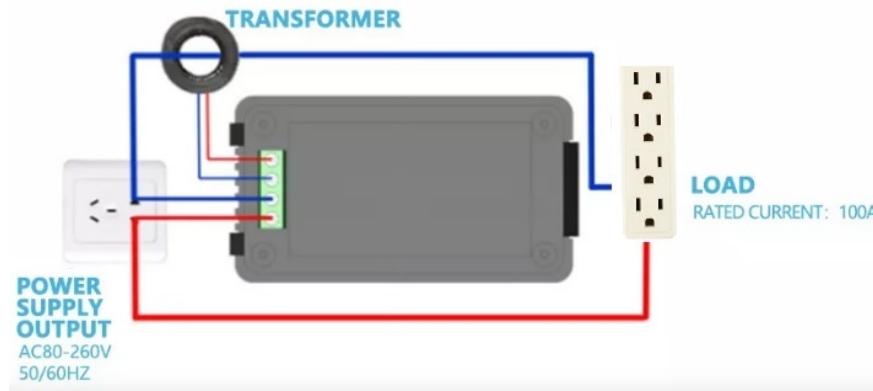


Figura 80: Diagrama de la conexión del medidor de consumo eléctrico

Como se puede ver en la figura 80 el medidor se conecta directamente a la alimentación de 127Vac, las otras dos conexiones que tiene el medidor son para conectar el transformador por el cual pasará uno de los cables de alimentación, el cual irá conectado a un multicontacto en donde se planea conectar los demás componentes eléctricos y electrónicos, como lo es la bocina, la fuente y el eliminador que va conectado a la Raspberry. El diagrama del módulo gestor de energía, en donde se visualiza como se distribuye la energía eléctrica en el sistema, se muestra a continuación.

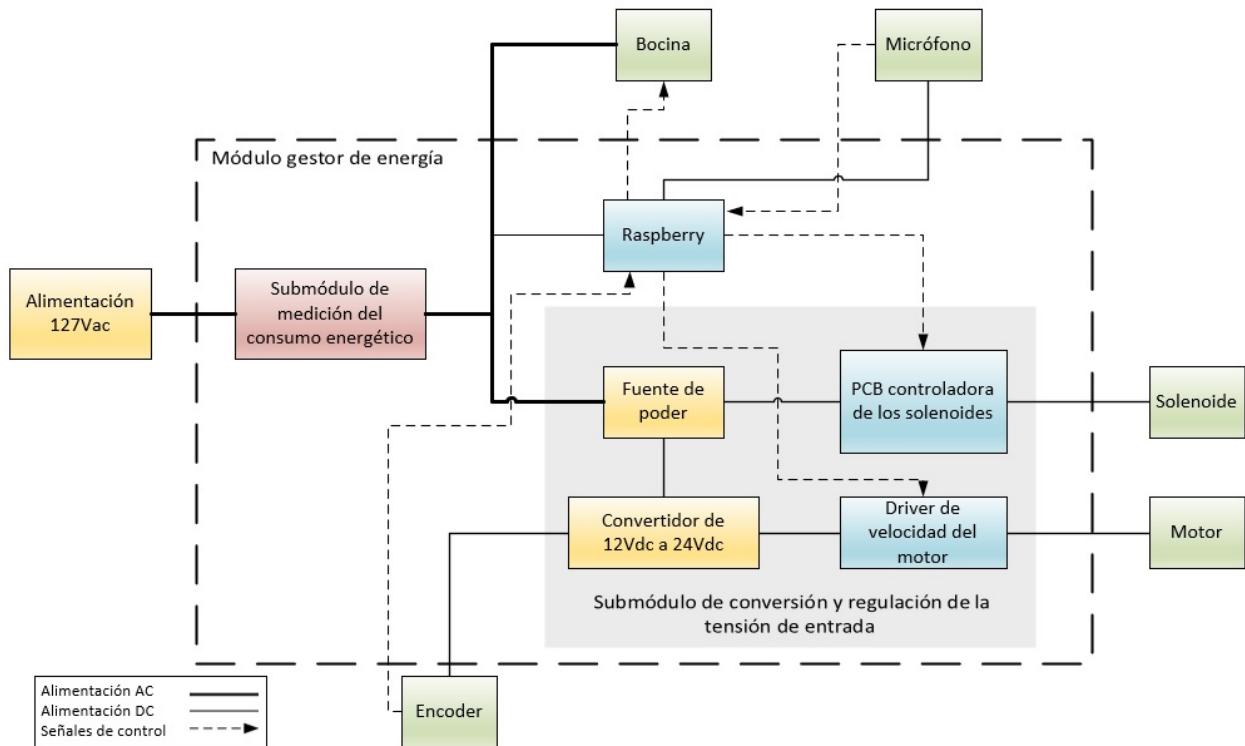


Figura 81: Diagrama del módulo gestor de energía

En la figura 81 se puede observar como el sistema de gestión de energía se conecta directamente a la alimentación de 127Vac para posteriormente pasar por el submódulo de

medición del consumo energético, el cual realiza la medición del consumo de todo el sistema, y después alimenta los componentes que necesiten corriente alterna, como es la fuente de poder, la bocina (la cual no pertenece al módulo gestor de energía pero si se alimenta con 127Vac) y la Raspberry, la cual no funciona a 127Vac pero es alimentada por medio de un eliminador, el cual proporciona el voltaje y la corriente necesaria para su funcionamiento.

La fuente de poder transforma el voltaje de corriente directa a continua, para posteriormente distribuirla a los diferentes componentes que lo requieran como la PCB controladora de los solenoides, la cual posteriormente transmite la energía a los solenoides cuando la Raspberry envíe las señales de control requeridas. Debido a que el motor requiere de 24Vdc para poder funcionar y la fuente de poder entrega máximo 12Vdc, la fuente necesita enviar dicho voltaje al convertidor de 12Vdc a 24Vdc para que este pase ese voltaje al driver de velocidad del motor y dependiendo de las señales de control de la Raspberry, el driver enviará el voltaje necesario al motor para que este gire a la velocidad deseada.

Los encoders trabajan de 8 a 24V por lo que pueden ser conectados desde el convertidor o desde la fuente de poder, y transmiten señales a la Raspberry para que esta decida que señales de control enviar al driver de velocidad del motor. Por último, el micrófono es alimentado desde la Raspberry y este a su vez envía las señales que recibe de vuelta a la Raspberry para que sean procesadas.

La Raspberry al ser el dispositivo que controla todo el sistema se encuentra dentro de varios módulos, como el módulo de procesamiento, el módulo modificador de la acústica y en este caso el módulo gestor de energía, ya que envía las señales necesarias para que todo el sistema funcione como fue planeado. El módulo gestor de energía tiene un costo elevado debido a que requiere de muchos componentes electrónicos de valor elevado, no obstante, se diseñó considerando los costos más bajos en los componentes sin comprometer la calidad de estos, además de las menores potencias consumidas y sobre todo se planea que las rutinas de movimiento de los paneles y los tiempos de funcionamiento del motor sean lo más corto que se pueda, para poder bajar el consumo energético del sistema tanto como sea posible.

2.3. Integración del sistema

Por ultimo, se integro todo el diseño en el archivo CAD, haciendo el redimensionado de acuerdo con los parámetros calculados durante el diseño detallado. Se muestra el sistema integrado en las figuras 82, 83 y 84, además de un explosionado del sistema en la figura 85.

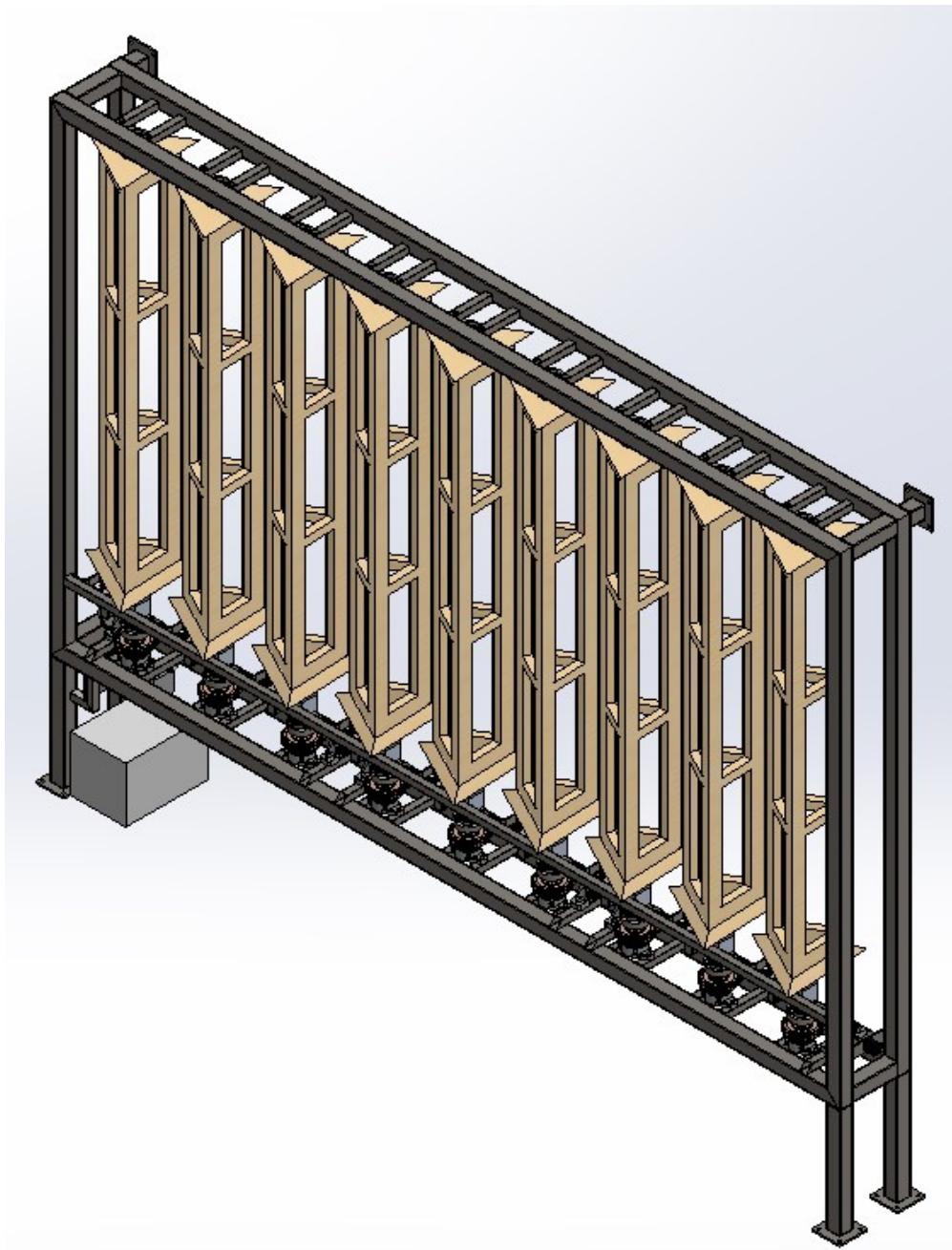


Figura 82: Vista isométrica del sistema integrado

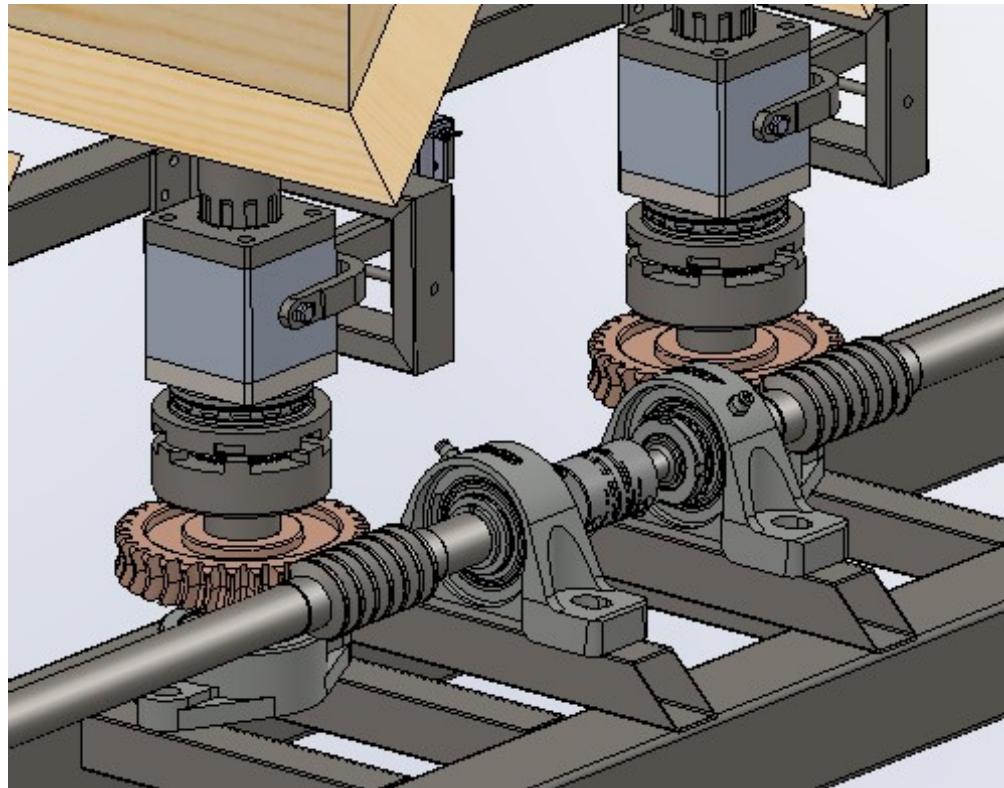


Figura 83: Vista isométrica del mecanismo de movimiento y acoplamiento

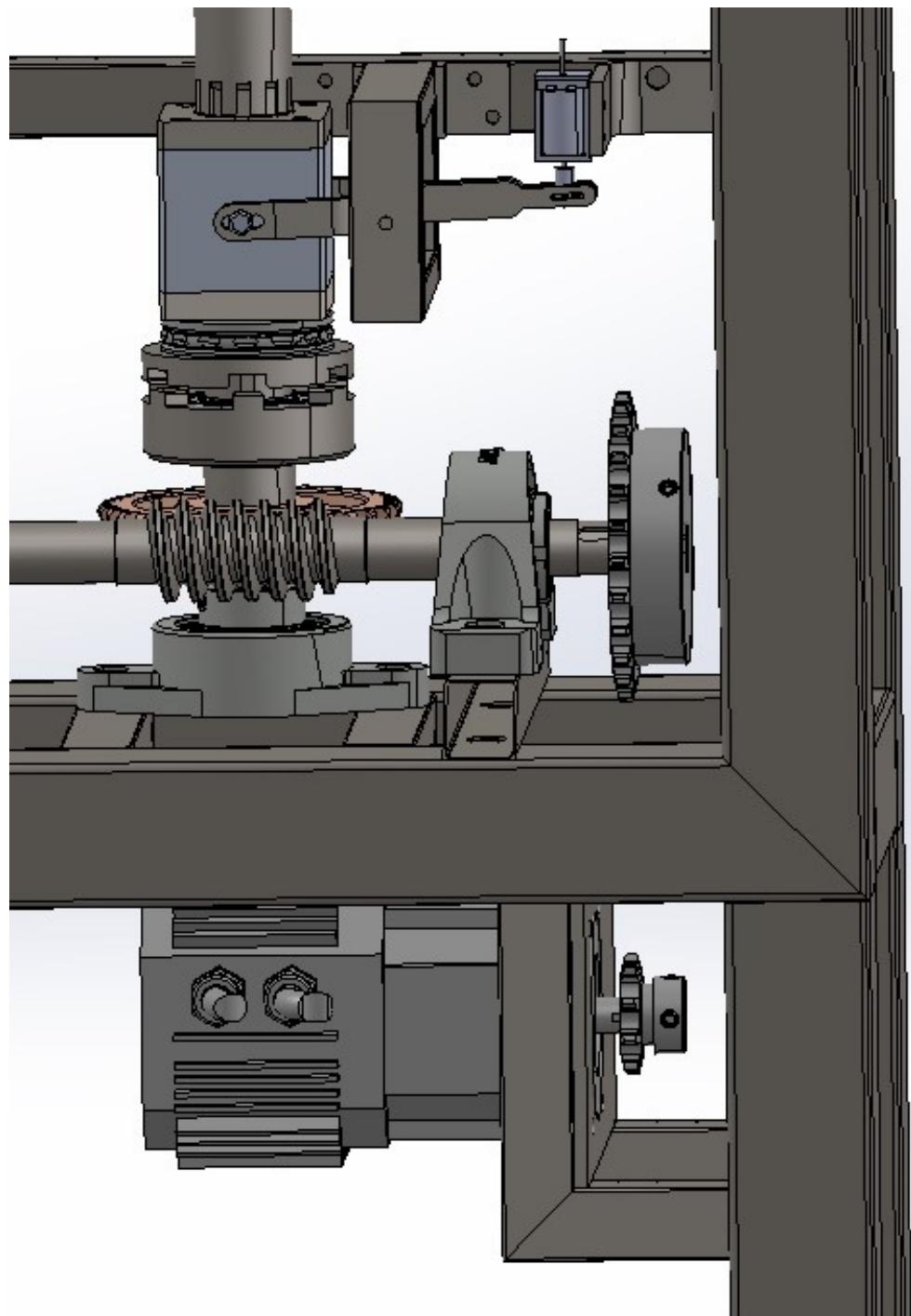


Figura 84: Vista isométrica del motor y del mecanismo de movimiento y de acoplamiento

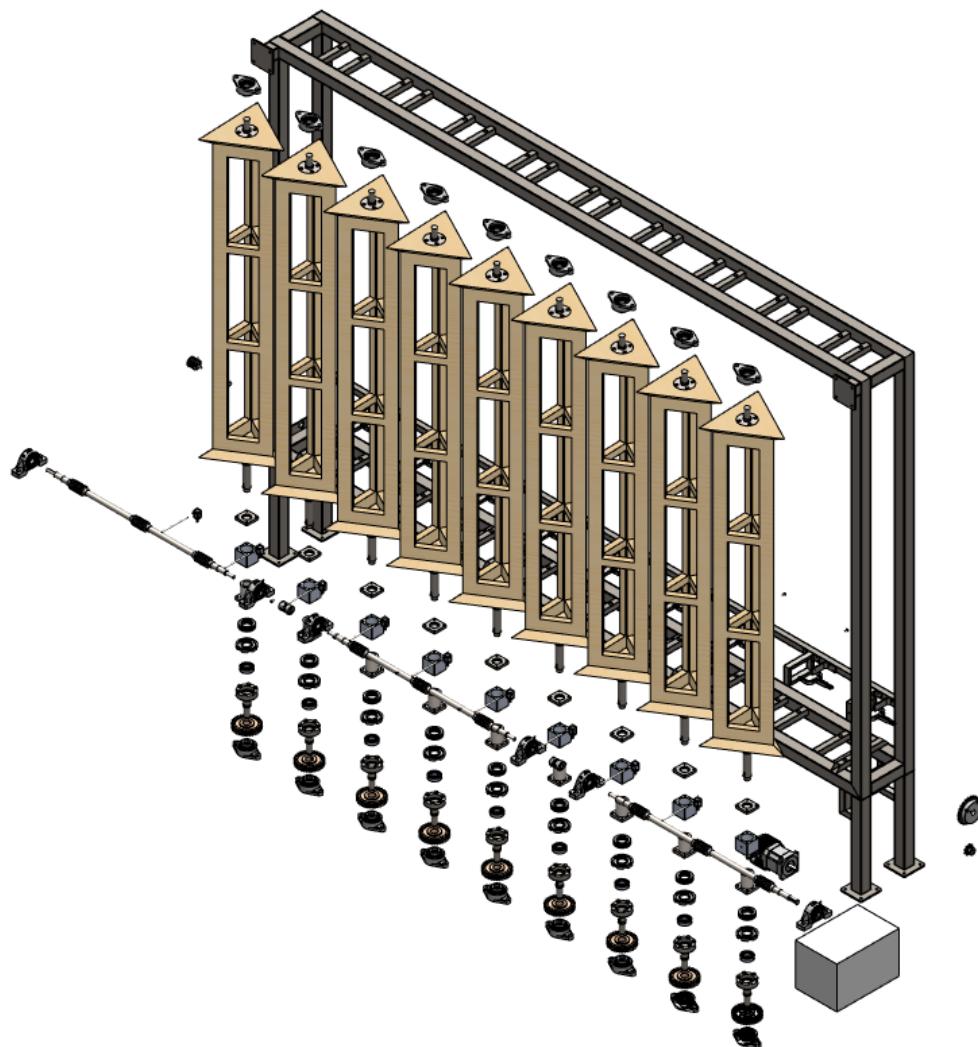


Figura 85: Vista explosionada del sistema integrado

2.4. Administración

2.4.1. Costos estimados

Basándonos en la implementación y elementos elegidos, se hizo una recopilación de los costos estimados de todos los componentes del proyecto, clasificando dichos componentes en aquellos que se adquieren y los que se fabrican,

No.	Nombre	Cantidad	Precio unitario (MXN)	Subtotal (MXN)
1	PTR de 2 pulgadas calibre 14	4	520.00	2080.00
2	PTR de 1 pulgadas calibre 20	1.5	145.00	217.50
3	Tabla de madera <i>triply...</i>	3	615.00	1,845.00

4	Barra de 1.5" de acero AISI 1020	3	130.00	390.00
5	Electrodo	2	80.00	160.00
6	Madera dimensionada	12	559.00	6,708.00
Total			11,400.50	

Tabla 105: Lista de materias primas

No.	Nombre	Material	Cantidad	Precio unitario (MXN)	Subtotal (MXN)
1	Miembro estructural	PTR	2	4,000.00	8,000.00
2	Prisma	Madera	18	400.00	7,200.00
3	Tornillo sin fin	AISI 1020	6	1500.00	9,000.00
4	Corona	Latón	18	300	5,400.00
5	Mecanismo de acople	-	18	300.00	5,400.00
6	PCB	Placa fenólica	2	10.00	20.00
7		-	-	-	-
Total					35,020.00

Tabla 106: Lista de componentes de fabricación

No.	Nombre	Cantidad	Precio unitario (MXN)	Subtotal (MXN)
1	Solenoide lineal	18	25.00	450.00
2	IRL2703	18	6.20	111.60
3	Resistencias de 1MΩ	18	0.40	7.20
4	Resistencias de 10MΩ	18	0.40	7.20
5	Bornera de 2 terminales	18	1.85	33.30
6	Bornara de 12 terminales	2	12.60	25.20
7	Placa fenólica de 10 cm x 15 cm	2	29.00	58.00
8	Motor de alta velocidad de 300W	2	1510.00	3,020.00
9	Convertidor de 12Vdc a 24Vdc	1	695.00	695.00
10	Driver para controlar la velocidad del motor	2	482.00	964.00
11	Potenciómetro digital	2	5.59	11.18
12	Encoder	2	182.00	364.00
13	Fuente de Poder MSI	1	900.00	900.00
14	Medidor de consumo el eléctrico	1	280.00	280.00
15	Multicontacto	1	80.00	80.00
16	Raspberry Pi 4 modelo B	1	No aplica	No aplica
17	Bocina...	1	No aplica	No aplica
18	Micrófono...	1	No aplica	No aplica
19	Tornillos de 1/2 pulgada	48	6.50	312.00
20	Cople de 1/2"	4	50.00	200.00
21	Chumacera de pie de 20mm	12	110.00	1,320.00

22	Chumacera de pared de 20mm	36	110.00	3,960.00
23	Rodamiento de 40mm	18	70.00	1,260.00
24	Catarina 1	2	120.00	240.00
25	Catarina 2	2	600.00	1,200.00
26	Paneles absensor	18	700	12,600
Total				28,098.68

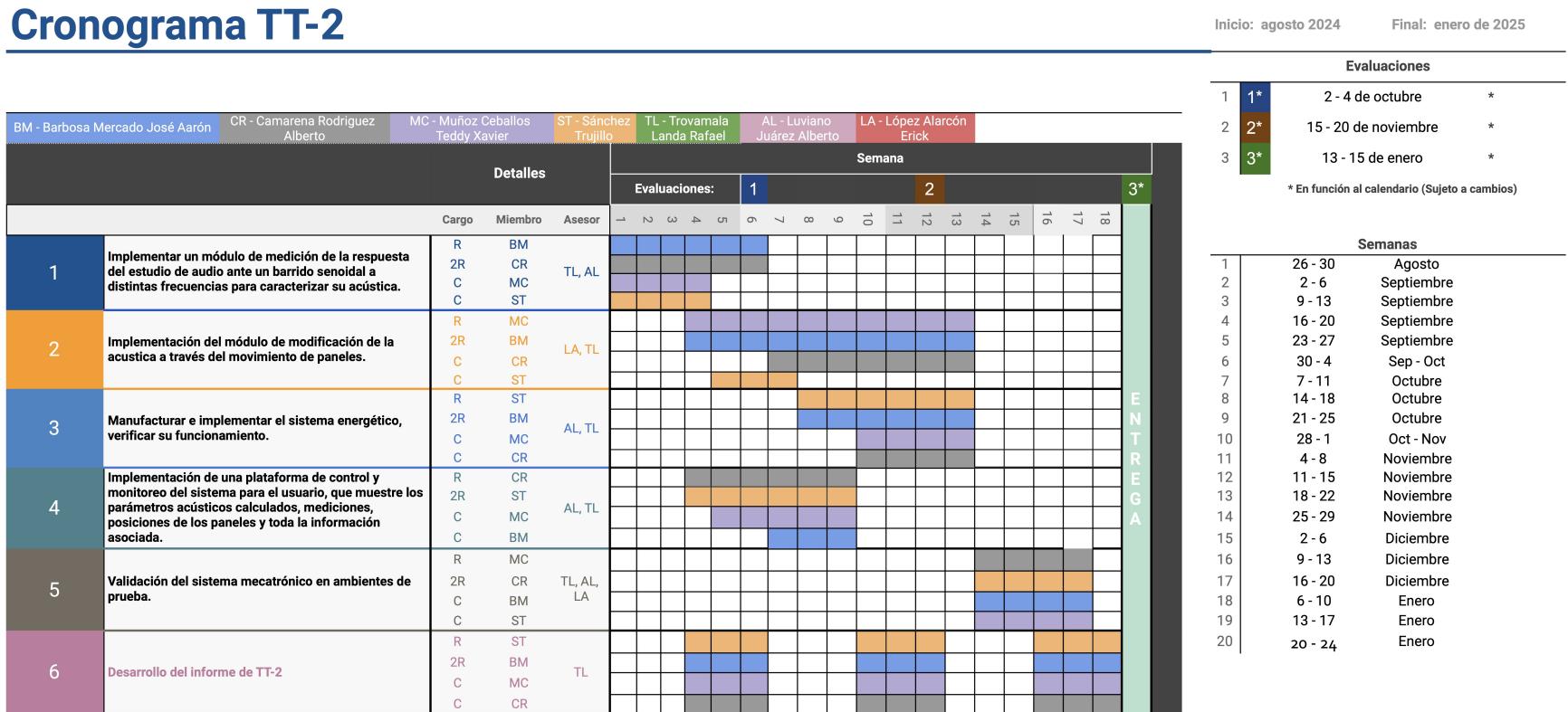
Tabla 107: Lista de componentes de adquisición

Sumando las cantidades totales del costo de la materia prima, los componentes de fabricación y los componentes de adquisición se obtiene un costo total aproximado del proyecto de 75,519.18 pesos mexicanos.

2.4.2. Planeación de la implementación

Por último, y con base en los objetivos propuestos para Trabajo Terminal 2, se hizo una planeación de las actividades a realizar en la figura 86, así como de sus responsables y periodos de realización.

Cronograma TT-2



Apéndices

A. Códigos y scripts

A.1. Función para calcular los modos normales de vibración

```

1 function [ modes_length ,modes_width ,modes_depth ,SchroederFreq ] = CalcRoomModes( 
2   room_dim , RT60 ,num_modes)
3   c = 343; % m/s
4   f_length = c/room_dim(1);
5   f_width = c/room_dim(2);
6   f_depth = c/room_dim(3);
7   modes_length = zeros (1 ,num_modes);
8   modes_width = zeros (1 ,num_modes);
9   modes_depth = zeros (1 ,num_modes);
10
11  for i = 1:num_modes
12    modes_length(i) = f_length*(0.5*i);
13    modes_width(i) = f_width*(0.5*i);
14    modes_depth(i) = f_depth*(0.5*i);
15  end
16
17 SchroederFreq = 2000*sqrt (RT60/prod (room_dim));
18 end

```

A.2. Función para calcular la pared y posición de impacto

```

1 function [ surfaceofimpact ,displacement ] = getImpactWall( ray_xyz ,ray_dxxyz ,
2   roomDims)
3 %GETIMPACTWALL Determine which wall the ray encounters
4 surfaceofimpact = -1;
5 displacement = 1000;
6 % Compute time to intersection with x-surfaces
7 if (ray_dxxyz(1) < 0)
8   displacement = -ray_xyz(1) / ray_dxxyz(1);
9   if displacement==0
10     displacement=1000;
11   end
12   surfaceofimpact = 1; %"SmallWall"
13 elseif (ray_dxxyz(1) > 0)
14   displacement = (roomDims(1) - ray_xyz(1)) / ray_dxxyz(1);
15   if displacement==0
16     displacement=1000;c
17   end
18   surfaceofimpact = 2; %"OpSmallWall"
19 end
20 % Compute time to intersection with y-surfaces
21 if ray_dxxyz(2)<0
22   t = -ray_xyz(2) / ray_dxxyz(2);
23   if (t<displacement) && t>0

```

```

23     surfaceofimpact = 3; %> LargeWall";
24     displacement = t;
25 end
26 elseif ray_dxyz(2)>0
27     t = (roomDims(2) - ray_xyz(2)) / ray_dxyz(2);
28     if (t<displacement) && t>0
29         surfaceofimpact = 4; %> OpLargeWall";
30         displacement = t;
31     end
32 end
33 % Compute time to intersection with z-surfaces
34 if ray_dxyz(3)<0
35     t = -ray_xyz(3) / ray_dxyz(3);
36     if (t<displacement) && t>0
37         surfaceofimpact = 5; %> Floor";
38         displacement = t;
39     end
40 elseif ray_dxyz(3)>0
41     t = (roomDims(3) - ray_xyz(3)) / ray_dxyz(3);
42     if (t<displacement) && t>0
43         surfaceofimpact = 6; %> Ceiling";
44         displacement = t;
45     end
46 end
47
48 displacement = displacement * ray_dxyz;
49
50 end

```

A.3. Función para hacer los ajustes lineales a la curva de decaimiento

```

1 function [LF_EDT,LF_T20,LF_T30,LF_T60] = GetLinearFits(L,vectorTiempo)
2
3 [~, Minus5DbIdx] = min( abs(L+5));
4 [~, Minus15DbIdx] = min( abs(L+15));
5 [~, Minus25DbIdx] = min( abs(L+25));
6 [~, Minus35DbIdx] = min( abs(L+35));
7 [~, Minus65DbIdx] = min( abs(L+65));
8
9 X_EDT = vectorTiempo(Minus5DbIdx:Minus15DbIdx);
10 X_T20 = vectorTiempo(Minus5DbIdx:Minus25DbIdx);
11 X_T30 = vectorTiempo(Minus5DbIdx:Minus35DbIdx);
12 X_T60 = vectorTiempo(Minus5DbIdx:Minus65DbIdx);
13
14 L_EDT = L(Minus5DbIdx:Minus15DbIdx);
15 L_T20 = L(Minus5DbIdx:Minus25DbIdx);
16 L_T30 = L(Minus5DbIdx:Minus35DbIdx);
17 L_T60 = L(Minus5DbIdx:Minus65DbIdx);
18
19 P_EDT = polyfit(X_EDT,L_EDT,1);
20 P_T20 = polyfit(X_T20,L_T20,1);

```

```

21 P_T30 = polyfit(X_T30,L_T30,1);
22 P_T60 = polyfit(X_T60,L_T60,1);
23
24 LF_EDT = P_EDT(1)*vectorTiempo+P_EDT(2);
25 LF_T20 = P_T20(1)*vectorTiempo+P_T20(2);
26 LF_T30 = P_T30(1)*vectorTiempo+P_T30(2);
27 LF_T60 = P_T60(1)*vectorTiempo+P_T60(2);
28 end

```

A.4. Función para obtener el vector normal a la pared

```

1 function N = getWallNormalVector(surfaceofimpact)
2 %GETWALLNORMALVECTOR Get the normal vector of a surface
3 switch surfaceofimpact
4 case 1 %>"SmallWall"
5     N = [1 0 0];
6 case 2 %>"OpSmallWall"
7     N = [-1 0 0];
8 case 3 %>"LargeWall"
9     N = [0 1 0];
10 case 4 %>"OpLargeWall"
11    N = [0 -1 0];
12 case 5 %>"Floor"
13    N = [0 0 1];
14 case 6 %>"Ceiling"
15    N = [0 0 -1];
16 end
17
18 end

```

A.5. Función para mostrar el cuarto y las posiciones del receptor y transmisor

```

1 function plotRoom(roomDimensions, receiverCoord, sourceCoord, figHandle)
2 %PLOTROOM Helper function to plot 3D room with receiver/transmitter points
3 figure(figHandle)
4 X = [0; roomDimensions(1); roomDimensions(1); 0; 0];
5 Y = [0; 0; roomDimensions(2); roomDimensions(2); 0];
6 Z = [0; 0; 0; 0; 0];
7 figure;
8 hold on;
9 plot3(X,Y,Z,"k",LineWidth=1.5);
10 plot3(X,Y,Z+roomDimensions(3),"k",LineWidth=1.5);
11 set(gca,"View",[-28,35]);
12 for k=1:length(X)-1
13     plot3([X(k);X(k)], [Y(k);Y(k)], [0; roomDimensions(3)], "k",LineWidth=1.5);
14 end
15 grid on
16 xlabel("X (m)")
17 ylabel("Y (m)")

```

```

18 zlabel("Z (m)")
19 plot3(sourceCoord(1),sourceCoord(2),sourceCoord(3),"bx",LineWidth=2)
20 plot3(receiverCoord(1),receiverCoord(2),receiverCoord(3),"ro",LineWidth=2)
21 end

```

A.6. Función obtener vectores esféricos uniformemente distribuidos

```

1
2 function X=RandSampleSphere(N)
3 % RANDSAMPLESPHERE Return random ray directions
4
5 % Sample the unfolded right cylinder
6 z = 2*rand(N,1)-1;
7 lon = 2*pi*rand(N,1);
8
9 % Convert z to latitude
10 z(z<-1) = -1;
11 z(z>1) = 1;
12 lat = acos(z);
13
14 % Convert spherical to rectangular co-ords
15 s = sin(lat);
16 x = cos(lon).*s;
17 y = sin(lon).*s;
18
19 X = [x y z];
20 end

```

A.7. Código para obtener la respuesta al impulso del recinto por medio de Ray Tracing

```

1 clear; close all; clc;
2
3 %% SetUp
4 room_dim = [6, 4, 3.4];
5 source_pos = [0.3, 0.3, 0.5];
6 mic_pos = [3.3, 2.3, 1.3];
7 mic_radius = 0.0875;
8 impResTime = 10;
9
10 plotRoom(room_dim, mic_pos, source_pos, 1)
11
12 %% Generate Rays
13 N = 5000;
14 rng(0)
15 rays = RandSampleSphere(N);
16
17 %% Reflections and Scattering Coefficients

```

```

18| FVect = [125 250 500 1000 2000 4000 8000];
19|
20| abs_coeffs = [];
21| abs_coeffs(:,1) = [0.02,0.02,0.03,0.03,0.04,0.05,0.05]; %Concrete
22| abs_coeffs(:,2) = [0.70,0.45,0.65,0.60,0.75,0.65,0.65]; %AbsPanels
23| abs_coeffs(:,3) = [0.14,0.10,0.06,0.08,0.10,0.10,0.10]; %Door
24|
25| scatt_coeffs = [];
26| scatt_coeffs(:,1) = [0.30,0.50,0.60,0.60,0.70,0.70,0.70];
27| scatt_coeffs(:,2) = [0.20,0.40,0.50,0.50,0.60,0.60,0.60];
28| scatt_coeffs(:,3) = [0.30,0.50,0.60,0.60,0.70,0.70,0.70];
29|
30| num_FBands = length(FVect);
31|
32| abs_map = {};
33| abs_map{1} = zeros(round(room_dim(2)*100),round(room_dim(3)*100),num_FBands);
34| abs_map{2} = zeros(round(room_dim(2)*100),round(room_dim(3)*100),num_FBands);
35| abs_map{3} = zeros(round(room_dim(1)*100),round(room_dim(3)*100),num_FBands);
36| abs_map{4} = zeros(round(room_dim(1)*100),round(room_dim(3)*100),num_FBands);
37| abs_map{5} = zeros(round(room_dim(1)*100),round(room_dim(2)*100),num_FBands);
38| abs_map{6} = zeros(round(room_dim(1)*100),round(room_dim(2)*100),num_FBands);
39|
40| scatt_map = {};
41| scatt_map{1} = zeros(round(room_dim(2)*100),round(room_dim(3)*100),num_FBands)
42| ;
42| scatt_map{2} = zeros(round(room_dim(2)*100),round(room_dim(3)*100),num_FBands)
43| ;
43| scatt_map{3} = zeros(round(room_dim(1)*100),round(room_dim(3)*100),num_FBands)
44| ;
44| scatt_map{4} = zeros(round(room_dim(1)*100),round(room_dim(3)*100),num_FBands)
45| ;
45| scatt_map{5} = zeros(round(room_dim(1)*100),round(room_dim(2)*100),num_FBands)
46| ;
46| scatt_map{6} = zeros(round(room_dim(1)*100),round(room_dim(2)*100),num_FBands)
47| ;
47|
48| %% Define wall materials
49| [abs_map, scatt_map] = updateAbsScattCoeffs(abs_map, scatt_map, 1, abs_coeffs
50| (:,1), ...
50|     scatt_coeffs(:,2), [0,0], [room_dim(2), room_dim(3)]); %SmallWall
51| [abs_map, scatt_map] = updateAbsScattCoeffs(abs_map, scatt_map, 2, abs_coeffs
51| (:,1), ...
51|     scatt_coeffs(:,2), [0,0], [room_dim(2), room_dim(3)]); %OpSmallWall
52| [abs_map, scatt_map] = updateAbsScattCoeffs(abs_map, scatt_map, 3, abs_coeffs
52| (:,1), ...
52|     scatt_coeffs(:,2), [0,0], [room_dim(1), room_dim(3)]); %LargeWall
53| [abs_map, scatt_map] = updateAbsScattCoeffs(abs_map, scatt_map, 4, abs_coeffs
53| (:,1), ...
53|     scatt_coeffs(:,2), [0,0], [room_dim(1), room_dim(3)]); %OpLargeWall
54| [abs_map, scatt_map] = updateAbsScattCoeffs(abs_map, scatt_map, 5, abs_coeffs
54| (:,1), ...
54|     scatt_coeffs(:,2), [0,0], [room_dim(1), room_dim(2)]); %Floor
55| [abs_map, scatt_map] = updateAbsScattCoeffs(abs_map, scatt_map, 6, abs_coeffs
55| (:,1), ...

```

```

60 scatt_coeffs (:,2), [0,0], [room_dim(1), room_dim(2)]); %Ceiling
61
62 %% Reflections map
63 ref_map = cell(1,num_FBands);
64 for i = 1:numel(abs_map)
65     ref_map{i} = sqrt(1-abs_map{i});
66 end
67
68 %% Energy Histogram
69 histTimeStep = 0.0010;
70 nTBins = round(impResTime/histTimeStep);
71 nFBins = length(FVect);
72 TFHist = zeros(nTBins,nFBins);
73
74 %% Ray Tracing
75 for iBand = 1:nFBins
76     fprintf(" Calculating rays for band %d\n",iBand)
77     % Perform ray tracing independently for each frequency band.
78     for iRay = 1:size(rays,1)
79         % Select ray direction
80         ray = rays(iRay,:);
81         % All rays start at the source/transmitter
82         ray_xyz = source_pos;
83         % Set initial ray direction. This direction changes as the ray is
84         % reflected off surfaces.
85         ray_dxyz = ray;
86         % Initialize ray travel time. Ray tracing is terminated when the
87         % travel time exceeds the impulse response length.
88         ray_time = 0;
89         % Initialize the ray energy to a normalized value of 1.      Energy
90         % decreases when the ray hits a surface.
91         ray_energy = 1;
92
93         while (ray_time <= impResTime)
94
95             % Determine the surface that the ray encounters
96             [surfaceofimpact,displacement] = getImpactWall(ray_xyz, ...
97                                         ray_dxyz,room_dim);
98
99             % Determine the distance traveled by the ray
100            distance = sqrt(sum(displacement.^2));
101
102            % Determine the coordinates of the impact point
103            impactCoord = ray_xyz+displacement;
104
105            if surfaceofimpact > 4
106                pointOfImpact = [impactCoord(1),impactCoord(2)];
107            elseif surfaceofimpact < 3
108                pointOfImpact = [impactCoord(2),impactCoord(3)];
109            else
110                pointOfImpact = [impactCoord(1),impactCoord(3)];
111            end
112
113            % Update ray location/source

```

```

114 ray_xyz = impactCoord;
115
116 % Update cumulative ray travel time
117 c = 343; % speed of light (m/s)
118 ray_time = ray_time+distance/c;
119
120 % Apply surface reflection to ray's energy
121 % This is the amount of energy that is not lost through
122 % absorption.
123
124 ReflectionAtPoint = ref_map{surfaceofimpact}(ceil(100*
pointOfImpact(1)),ceil(100*pointOfImpact(2)),iBand);
125 ray_energy = ray_energy*ReflectionAtPoint;
126
127 % Apply diffuse reflection to ray energy
128 % This is the fraction of energy used to determine what is
129 % detected at the receiver
130 DifussionAtPoint = scatt_map{surfaceofimpact}(ceil(100*
pointOfImpact(1)),ceil(100*pointOfImpact(2)),iBand);
131 rayrecv_energy = ray_energy*DifussionAtPoint;
132
133 % Determine impact point-to-receiver direction.
134 rayrecvvector = mic_pos-impactCoord;
135
136 % Determine the ray's time of arrival at receiver.
137 distance = sqrt(sum(rayrecvvector.*rayrecvvector));
138 recv_timeofarrival = ray_time+distance/c;
139
140 if recv_timeofarrival>impResTime
141     break
142 end
143
144 if ray_energy < 0.000001
145     break
146 end
147
148 % Determine amount of diffuse energy that reaches the receiver.
149 % See (5.20) in [2].
150
151 % Compute received energy
152 N = getWallNormalVector(surfaceofimpact);
153 cosTheta = sum(rayrecvvector.*N)/(sqrt(sum(rayrecvvector.^2)));
154 cosAlpha = sqrt(sum(rayrecvvector.^2)-mic_radius.^2)/sum(
155 rayrecvvector.^2);
156 E = (1-cosAlpha)*2*cosTheta*rayrecv_energy;
157
158 % Update energy histogram
159 tbin = floor(recv_timeofarrival/histTimeStep + 0.5);
160 TFHist(tbin,iBand) = TFHist(tbin,iBand) + E;
161
162 % Compute a new direction for the ray.
163 % Pick a random direction that is in the hemisphere of the
164 % normal to the impact surface.
165 d = rand(1,3);

```

```

165      d = d/norm(d);
166      if sum(d.*N)<0
167          d = -d;
168      end
169
170      % Derive the specular reflection with respect to the incident
171      % wall
172      ref = ray_dxyz-2*(sum(ray_dxyz.*N))*N;
173
174      % Combine the specular and random components
175      d = d/norm(d);
176      ref = ref/norm(ref);
177      ray_dxyz = DifussionAtPoint*d+(1-DifussionAtPoint)*ref;
178      ray_dxyz = ray_dxyz/norm(ray_dxyz);
179      end
180  end
181 end
182
183 %% View
184 figure(1)
185 bar(histTimeStep*(0:size(TFHist)-1),TFHist)
186 grid on
187 xlabel("Time (s)")
188 legend(["125 Hz","250 Hz","500 Hz","1000 Hz","2000 Hz","4000 Hz"])
189
190 %% Generate Impulse Response
191 fs = 44100;
192 V = prod(room_dim);
193 t0 = ((2*V*log(2))/(4*pi*c^3))^(1/3); % eq 5.45 in [2]
194 poissonProcess = [];
195 timeValues = [];
196 t = t0;
197 while (t<impResTime)
198     timeValues = [timeValues t]; %#ok
199     % Determine polarity.
200     if (round(t*fs)-t*fs) < 0
201         poissonProcess = [poissonProcess 1]; %#ok
202     else
203         poissonProcess = [poissonProcess -1]; %#ok
204     end
205     % Determine the mean event occurrence (eq 5.44 in [2])
206     mu = min(1e4,4*pi*c^3*t^2/V);
207     % Determine the interval size (eq. 5.44 in [2])
208     deltaTA = (1/mu)*log(1/rand); % eq. 5.43 in [2]
209     t = t+deltaTA;
210 end
211 randSeq = zeros(ceil(impResTime*fs),1);
212 for index=1:length(timeValues)
213     randSeq(round(timeValues(index)*fs)) = poissonProcess(index);
214 end
215 flow = [115 225 450 900 1800 3600 7200];
216 fhigh = [135 275 550 1100 2200 4400 8800];
217 NFFT = 8192;
218 win = hann(882,"symmetric");

```

```

219 sfft = dsp.STFT(Window = win ,OverlapLength=441,FFTLength=NFFT, FrequencyRange="onesided");
220 isfft = dsp.ISTFT(Window=win ,OverlapLength=441, FrequencyRange="onesided");
221 F = sfft .getFrequencyVector( fs );
222 RCF = zeros( length( FVect ) ,length( F ) );
223 for index0 = 1:length( FVect )
224     for index=1:length( F )
225         f = F(index);
226         if f<FVect(index0) && f>=flow(index0)
227             RCF(index0 ,index) = .5*(1+cos(2*pi*f/FVect(index0)));
228         end
229         if f<fhigh(index0) && f>=FVect(index0)
230             RCF(index0 ,index) = .5*(1-cos(2*pi*f/(FVect(index0)+1)));
231         end
232     end
233 end
234 frameLength = 441;
235 numFrames = length( randSeq)/frameLength ;
236 y = zeros( length( randSeq) ,numel(FVect));
237 for index=1:numFrames
238     x = randSeq(( index -1)*frameLength+1:index *frameLength );
239     X = sfft (x);
240     X = X.*RCF. ';
241     y(( index -1)*frameLength+1:index *frameLength ,:) = isfft (X);
242 end
243 impTimes = (1/ fs)*(0: size(y ,1)-1);
244 hisTimes = histTimeStep/2 + histTimeStep *(0:nTBins) ;
245 W = zeros( size(impTimes ,2) ,numel(FVect));
246 BW = fhigh-flow ;
247 for k=1:size(TFHist ,1)
248     gk0 = floor ((k-1)*fs*histTimeStep)+1;
249     gk1 = floor (k*fs*histTimeStep);
250     yy = y(gk0:gk1 ,:) .^ 2;
251     val = sqrt(TFHist(k ,:))./sum(yy ,1) .*sqrt(BW/( fs /2));
252     for iRay=gk0:gk1
253         W(iRay ,:)= val;
254     end
255 end
256 %% Create Impulse Response
257 y_2 = y .*W;
258 ip = sum(y_2 ,2) ;
259 ip = ip ./max( abs(ip));
260 ip = ip + rand(length(ip ),1)/1000;
261 vectorTiempo = (1/ fs)*(0: numel(ip )-1);
262 figure
263 plot(vectorTiempo ,ip .^ 2)
264 grid on
265 xlabel("Time (s)")
266 ylabel("Impulse Response")
267 schroeder_cumsum = cumsum( flipud(ip .^ 2));
268 schroeder_normalized = schroeder_cumsum / max(schroeder_cumsum);
269 L = flipud(10*log10(schroeder_normalized));
270 [ SignalSize , ~] = size(ip );

```

```

272
273 vectorTiempo = (1:SignalSize)/fs ;
274 [LF_EDT,LF_T20,LF_T30,LF_T60] = GetLinearFits(L,vectorTiempo) ;
275
276 [~, EDT_Idx] = min( abs(LF_EDT+60)) ;
277 T60delEDT = vectorTiempo(EDT_Idx)
278 [~, T20_Idx] = min( abs(LF_T20+60)) ;
279 T60delT20 = vectorTiempo(T20_Idx)
280 [~, T30_Idx] = min( abs(LF_T30+60)) ;
281 T60delT30 = vectorTiempo(T30_Idx)
282 [~, T60_Idx] = min( abs(LF_T60+60)) ;
283 T60delT60 = vectorTiempo(T60_Idx)
284
285 figure
286 plot(vectorTiempo,L, 'LineWidth' ,2)
287 title('Schroeder Integration')
288 xlabel('Time (s)')
289 ylabel('Decay (Db)')
290 hold on
291 plot(vectorTiempo,LF_EDT)
292 plot(vectorTiempo,LF_T20)
293 plot(vectorTiempo,LF_T30)
294 plot(vectorTiempo,LF_T60)
295 legend('Energy Decay Curve','Linear fit for EDT','Linear fit for T20','Linear
         fit for T30',...
         'Linear fit for T60','Location','southwest')
296 title('Linear fits')
297 xlabel('Time (s)')
298 ylabel('Decay (Db)')
299 ylim([-80 0])
300 hold off

```

A.8. Función para calcular los parámetros acústicos de acuerdo a ISO3382

```

1 function [RIRsq,DC,t,AcousticParams] = RIR_Analisys(method, input,
   room_dimensions, printFlag)
2 %RIR_Analisys Calculates acoustic parameters as in ISO3382. use method 1
3 %for wav file (input is name of wavfile), use method 2 for direct input,
4 %(structure like with .y and .fs)
5 if method == 1
6     [y, fs] = audioread(input);
7 else
8     y = input.y;
9     fs = input.fs;
10 end
11
12 if size(y,2) > 1
13     y = mean(y,2);
14 end
15 sound_duration = length(y)/fs;
16 t = ((0:length(y)-1)/fs)';

```

```

17 y = bandpass(y,[88.39 176.8],fs);
18 RIRsq = y.^2;
19 schroeder_cumsum = cumsum(flipud(RIRsq));
20 schroeder_normalized = schroeder_cumsum / max(schroeder_cumsum);
21 DC = flipud(10*log10(schroeder_normalized));
22
23 [LF_EDT,LF_T20,LF_T30,LF_T60] = GetLinearFits(DC,t);
24
25 [~, EDT_Idx1] = min(abs(LF_EDT+10));
26 [~, EDT_Idx2] = min(abs(LF_EDT));
27 T60delEDT = 6*(t(EDT.Idx1)-t(EDT.Idx2));
28
29 [~, EDT_Idx1] = min(abs(LF_T20+20));
30 [~, EDT_Idx2] = min(abs(LF_T20));
31 T60delT20 = 3*(t(EDT.Idx1)-t(EDT.Idx2));
32
33 [~, EDT_Idx1] = min(abs(LF_T30+30));
34 [~, EDT_Idx2] = min(abs(LF_T30));
35 T60delT30 = 2*(t(EDT.Idx1)-t(EDT.Idx2));
36
37 [~,maxIdx] = max(RIRsq);
38 RIRsq_trunc = RIRsq(maxIdx:end);
39 Energy0_50 = trapz(RIRsq_trunc(1:round(0.05*fs)));
40 Energy0_80 = trapz(RIRsq_trunc(1:round(0.08*fs)));
41 Energy0_end = trapz(RIRsq_trunc(1:end));
42
43 D50 = Energy0_50/Energy0_end;
44 D80 = Energy0_80/Energy0_end;
45 C50 = 10*log(D50/(1-D50));
46 C80 = 10*log(D80/(1-D80));
47 G = 10*log10(T60delT20/prod(room_dimensions))+45;
48
49 AcousticParams = struct('T60delEDT',T60delEDT,'T60delT20',T60delT20,'
50 T60delT30',T60delT30 ...
51 , 'D50',D50, 'D80',D80, 'C50',C50, 'C80',C80, 'G',G);
52
53 if printFlag == true
54 figure
55 plot(t,y)
56 title('Impulse Response')
57 xlabel('Time (s)')
58 ylabel('Amplitude')
59 xlim([0 sound_duration])
60
61 figure
62 plot(t,DC,"LineWidth",2)
63 title('Schroeder Integration')
64 xlabel('Time (s)')
65 ylabel('Decay (Db)')
66 ylim([-80 0.1])
67 xlim([0 sound_duration])
68
69 hold on

```

```

70 plot(t,LF_EDT)
71 plot(t,LF_T20)
72 plot(t,LF_T30)
73 legend('Decay Curve','Linear fit for EDT','Linear fit for T20','Linear
74 fit for T30','Location','southwest')
75 hold off
76
77 fprintf("\nT60 from EDT: %f",T60delEDT)
78 fprintf("\nT60 from T20: %f",T60delT20)
79 fprintf("\nT60 from T30: %f",T60delT30)
80
81 fprintf("\nD50: %f",D50)
82 fprintf("\nC50: %f",C50)
83 fprintf("\nC80: %f",C80)
84 fprintf("\nG: %f",G)
85 fprintf("\n")
86 end
end

```

A.9. Código obtener la respuesta al impulso de un recinto por medio de mediciones

```

1 clc; clear; close all;
2 %
3 fs = 44100;
4 room_dim = [6, 5, 3.5];
5 device = "ASIO4ALL v2";
6 aPR = audioPlayerRecorder("SampleRate",fs,"Device",device);
7 %
8 sweep_dur = 5;
9 duration_per_run = 8;
10 start_silence = 1;
11 silence_dur = duration_per_run - sweep_dur - start_silence;
12 sweep_range = [10 8000];
13 percentage = 100;
14 outputLevel = 20*log10(percentage/100);
15
16 sweepsine = sweeptone(sweep_dur,silence_dur,fs,"SweepFrequencyRange", ...
17 sweep_range,"ExcitationLevel",outputLevel);
18 %
19 excitation = [zeros(start_silence*fs,1); sweepsine];
20 sampling = 1024;
21 excitation_length = length(excitation);
22 bufOut = dsp.AsyncBuffer(excitation_length);
23 bufIn = dsp.AsyncBuffer(excitation_length);
24 write(bufOut, excitation);
25 %
26 while bufOut.NumUnreadSamples > 0
27     exci = read(bufOut, sampling);
28     [rec, num_under, num_over] = aPR(exci);
29     write(bufIn, rec);
30     if num_under>0 || num_over>0

```

```

31     fprintf("Underrun by %d frames , overrun by %d frames.\n",num_under,
32         num_over)
33 end
34 release(aPR);
35 read(bufIn,start_silence*fs);
36 audioFromDevice = read(bufIn);
37 %
38 time = (1:length(audioFromDevice))/fs;
39 RIR = impzest(sweepsine,audioFromDevice);
40 [max_RIR,Idx] = max(abs(RIR));
41 n_RIR = RIR(Idx-round(fs/100):end)/max_RIR;
42 time_RIR = (1:length(n_RIR))/fs;
43 ImpulseResponse.fs = fs;
44 ImpulseResponse.y = n_RIR;
45
46 plot(time,audioFromDevice)
47 title('Audio recorded')
48 xlabel('Time(s)')
49 ylabel('Amplitude')
50
51 plot(time_RIR, n_RIR)
52 title('Estimated impulse response')
53 xlabel('Time(s)')
54 ylabel('Amplitude')
55 %
56 [temp_y,ImpulseResponse.fs] = audioread(['C:\Users\User\Desktop\RIR_Database\' ...
57     '382908_uzbazur_little-basement-ir-impulse-response.wav']);
58 if size(temp_y,2) > 1
59     ImpulseResponse.y = mean(temp_y,2);
60 end
61 %
62 RIR = struct('General',ImpulseResponse);
63 RIR.f125.y = bandpass(ImpulseResponse.y,[88.39 176.8],ImpulseResponse.fs);
64 RIR.f125.fs = ImpulseResponse.fs;
65 RIR.f250.y = bandpass(ImpulseResponse.y,[176.8 353.6],ImpulseResponse.fs);
66 RIR.f250.fs = ImpulseResponse.fs;
67 RIR.f500.y = bandpass(ImpulseResponse.y,[353.6 707.1],ImpulseResponse.fs);
68 RIR.f500.fs = ImpulseResponse.fs;
69 RIR.f1k.y = bandpass(ImpulseResponse.y,[707.1 1414],ImpulseResponse.fs);
70 RIR.f1k.fs = ImpulseResponse.fs;
71 RIR.f2k.y = bandpass(ImpulseResponse.y,[1414 2828],ImpulseResponse.fs);
72 RIR.f2k.fs = ImpulseResponse.fs;
73 RIR.f4k.y = bandpass(ImpulseResponse.y,[2828 5657],ImpulseResponse.fs);
74 RIR.f4k.fs = ImpulseResponse.fs;
75 %
76 [~,DC,vectorTiempo, AcousticParams.General] = RIR_Analisys(2,RIR.General,
77     room_dim,true);
77 [~,~,~, AcousticParams.f125] = RIR_Analisys(2,RIR.f125,room_dim,false);
78 [~,~,~, AcousticParams.f250] = RIR_Analisys(2,RIR.f250,room_dim,false);
79 [~,~,~, AcousticParams.f500] = RIR_Analisys(2,RIR.f500,room_dim,false);
80 [~,~,~, AcousticParams.f1k] = RIR_Analisys(2,RIR.f1k,room_dim,false);
81 [~,~,~, AcousticParams.f2k] = RIR_Analisys(2,RIR.f2k,room_dim,false);

```

```

82 [~,~,~, AcousticParams.f4k] = RIR_Analisys(2,RIR.f4k,room_dim, false);
83 %
84 [~,~,~, SchroederFreq] = CalcRoomModes(room_dim,AcousticParams.General.
85 T60delT20,20)
86 %
86 V = prod(room_dim);
87 Current_abs = 0.161*V/AcousticParams.General.T60delT20;
88 DesiredT60 = 0.27
89 Needed_abs = 0.161*V/DesiredT60;
90 Offset_abs = Needed_abs-Current_abs;
91 MaterialAbs.Abs_Panels = 0.8;
92 MaterialAbs.Wall = 0.02;
93 Surface = Offset_abs/(MaterialAbs.Abs_Panels-MaterialAbs.Wall)

```

A.10. Código para simular las activaciones de los solenoides

```

1 clc; clear; close all;
2 %
3 step = 0.01;
4 time_of_motion = 3;
5 time_of_hold = 1;
6 time_total = 3*(time_of_hold + time_of_motion)+1;
7 t = 0:step:time_total;
8 %
9 arrange = [1,2,2,3,3,1,2,1,3]';
10 num_panels = length(arrange);
11 M = repmat(arrange,1,length(t));
12 %
13 Signals = zeros(size(M));
14 offset1 = 0;
15 offset2 = 0;
16 offset3 = 1;
17 for i = 1:num_panels
18     if arrange(i) == 2
19         Signals(i,offset1+((time_of_motion+time_of_hold)/step):end-1/step) =
20             1;
21         offset1 = offset1 + 10;
22     elseif arrange(i) == 3
23         Signals(i,offset2+(2*time_of_motion+2*time_of_hold)/step:end-1/step) =
24             1;
25         offset2 = offset2 + 10;
26     elseif arrange(i) == 1
27         Signals(i,offset3:end-1/step) = 1;
28         offset3 = offset3 + 10;
29     end
30 end
31 %
32 plot(t,Signals(:,:,1), 'LineWidth',1.2)
33 legends = cell(num_panels,1);
34 for i = 1:num_panels
35     legends{i} = append('Panel ', int2str(i));
36 end

```

```

35 legend(legends, 'Location ', 'east ')
36 ylim([-0.1,1.1])
37 xlim([0,13])
38 grid on

```

A.11. Función para cambiar los mapas de absorción y refracción

```

1 function [abs_map,scatt_map] = updateAbsScattCoeffs(abs_map,scatt_map,key,
2 abs_coeffs,scatt_coefs,Vertex1,Vertex2)
3 norm = Vertex2-Vertex1;
4 if any(norm<0)
5     disp('Not valid vertex positions')
6     return
7 else
8     Idx_X1 = round(100*Vertex1(1))+1;
9     Idx_Y1 = round(100*Vertex1(2))+1;
10    Idx_X2 = round(100*Vertex2(1));
11    Idx_Y2 = round(100*Vertex2(2));
12
13    try
14        abs_map{key}(Idx_X1:Idx_X2,Idx_Y1:Idx_Y2,:)= reshape(abs_coeffs,
15        1,1,length(abs_coeffs)).*ones(1+Idx_X2-Idx_X1,1+Idx_Y2-Idx_Y1,length(
16        abs_coeffs));
17        scatt_map{key}(Idx_X1:Idx_X2,Idx_Y1:Idx_Y2,:)= reshape(
18        scatt_coefs,1,1,length(scatt_coefs)).*ones(1+Idx_X2-Idx_X1,1+Idx_Y2-Idx_Y1
19        ,length(scatt_coefs));
20    catch
21        disp('The vertex positions exceeds the dimensions of that surface')
22    end
23    return
24 end
25 end

```

A.12. Código con librería para obtener la respuesta al impulso del recinto por medio de Ray Tracing

```

1 # -*- coding: utf-8 -*-
2 """RIR Simulation.ipynb
3
4 Automatically generated by Colab.
5
6 Original file is located at
7     https://colab.research.google.com/drive/1UHPLk6gX1fATqFftVJzKPByaXz00jumA
8 """
9
10 !pip install pyroomacoustics
11 !pip install gdown
12 !pip install playsound
13 !pip install pydub

```

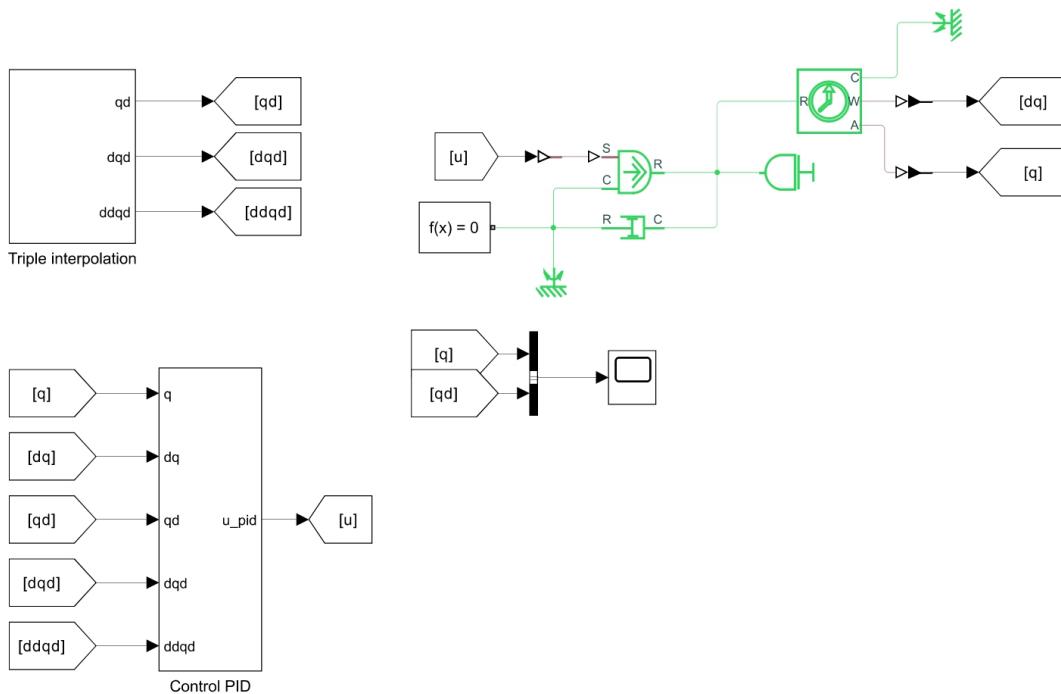
```
14 ! pip install wavfile
15 ! pip install IPython
16 ! pip install librosa
17
18 import pyroomacoustics as pra
19 import numpy as np
20 import matplotlib.pyplot as plt
21 import gdown
22 import IPython.display as ipd
23 from scipy.io import wavfile
24 import librosa
25 from pyroomacoustics.directivities import (
26     DirectivityPattern,
27     DirectionVector,
28     CardioidFamily,
29 )
30
31 room_dim = [6, 4, 3.4] # length, width, height in meters
32 source_pos = [0.3, 0.3, 0.5]
33 mic_locs = np.array([
34     [3.0, 2.0, 1.0], # mic 1
35     [3.3, 2.3, 1.3], # mic 2
36     [2.7, 1.7, 0.7] # mic 3
37 ])
38
39 mats = pra.make_materials(
40     ceiling="hard_surface",
41     floor="hard_surface",
42     east="hard_surface",
43     west="hard_surface",
44     north="hard_surface",
45     south="hard_surface",
46 )
47
48 file_id = '1CKobbsHBa2Nfz0slXoYR634-Vi2e7fpg'
49 output_name = 'guitar_16k.wav'
50 gdown.download(id=file_id, output=output_name)
51
52 fs, audio = wavfile.read(output_name)
53 ipd.Audio('guitar_16k.wav')
54
55 room = pra.ShoeBox(
56     room_dim, fs=fs, materials=mats, max_order=3,
57     use_rand_ism=True, max_rand_disp=0.05, ray_tracing=True,
58     air_absorption=False
59 )
60 room.set_ray_tracing()
61 room.add_source(source_pos, signal=audio, delay=0.5)
62 room.add_microphone_array(mic_locs)
63 fig, ax = room.plot()
64 ax.set_xlim([-0.1, room.dim[0]])
65 ax.set_ylim([-0.1, room.dim[1]])
66 ax.set_zlim([-0.1, room.dim[2]])
```

```

67
68 room.compute_rir()
69
70 sq_RIR = room.rir[0][0]**2
71 time_vector = np.arange(0, len(sq_RIR)/fs, 1/fs)
72 sq_RIR = librosa.util.normalize(sq_RIR)
73 sq_RIR_audio = sq_RIR*32767
74
75 wavfile.write("sq_RIR.wav", int(fs), sq_RIR_audio.astype(np.int16))
76 ipd.Audio("sq_RIR.wav")
77
78 plt.plot(time_vector, sq_RIR)
79 plt.show()
80 room.plot_rir()
81
82 rt60_mea = room.measure_rt60()
83 T60 = pra.experimental.measure_rt60(room.rir[0][0], fs=room.fs, plot=True)
84
85 for m in range(room.n_mics):
86     for s in range(room.n_sources):
87         print(
88             "RT60 between the microphone", m, "and the source", s, "is", rt60_mea[m,
89             s]
90         )
91 Avg_rt60 = np.mean(rt60_mea[:, 0])
92 print("The measured RT60 is", Avg_rt60)
93
94 room.simulate()
95 time_vector2 = np.arange(0, len(room.mic_array.signals[1, :])/fs, 1/fs)
96 song = librosa.util.normalize(room.mic_array.signals[1, :])
97 plt.plot(time_vector2, song)
98 song_audio = song*32767
99
100
101 wavfile.write("filtered_song.wav", int(fs), song_audio.astype(np.int16))
102 ipd.Audio("filtered_song.wav")

```

A.13. Programa de Simulink para el control inteligente de posición y velocidad



A.14. Programa de *Python* para la generación de las posiciones de los paneles

```

1 import numpy as np
2
3 NameToKey = { 'Reflie': 1, 'Abs': 2, 'Scatt': 3}
4 AbsCoeffs = { 'Concrete': [0.02, 0.02, 0.03, 0.03, 0.04, 0.05, 0.05],
5               'AbsPanels': [0.15, 0.7, 1, 0.6, 0.85, 0.9, 0.9],
6               'Door': [0.14, 0.10, 0.06, 0.08, 0.10, 0.10, 0.10]}
7 ScattCoeffs = { 'Concrete': [0.30, 0.50, 0.60, 0.60, 0.70, 0.70, 0.70],
8                  'AbsPanels': [0.20, 0.40, 0.50, 0.50, 0.60, 0.60, 0.60],
9                  'Door': [0.20, 0.40, 0.50, 0.50, 0.60, 0.60, 0.60]}
10 room_dim = [4.0, 3.4, 6.0]
11 PanelSize = [0.3, 1.33]
12
13 def PanelsToChange( Current_RT60 , Needed_RT60 , room_dim):
14     V = np.prod(room_dim)
15     Panel_Surface = np.prod(PanelSize)
16     Current_Abs = 0.161*V/Current_RT60
17     Needed_Abs = 0.161*V/Needed_RT60

```

```
18 AbsFromSinglePanel = Panel_Surface*(AbsCoeffs[ 'AbsPanels '][2] - AbsCoeffs[ 'Concrete '][2])
19 PanelsToChange = (Needed_Abs-Current_Abs)//AbsFromSinglePanel
20 return PanelsToChange
21
22 class Node:
23     def __init__(self , data , position):
24         self.Type = data
25         self.Position = position
26         self.next = None
27
28 class LinkedList:
29     def __init__(self):
30         self.head = None
31
32     def insertAtBegin(self , data , position):
33         new_node = Node(data , position)
34         if self.head is None:
35             self.head = new_node
36             return
37         else:
38             new_node.next = self.head
39             self.head = new_node
40
41     def insertAtEnd(self , data , position):
42         new_node = Node(data , position)
43         if self.head is None:
44             self.head = new_node
45             return
46
47         current_node = self.head
48         while(current_node.next):
49             current_node = current_node.next
50
51         current_node.next = new_node
52
53     def updateNode(self , val , index):
54         current_node = self.head
55         position = 0
56         if position == index:
57             current_node.Type = val
58         else:
59             while(current_node != None and position != index):
60                 position = position+1
61                 current_node = current_node.next
62
63             if current_node != None:
64                 current_node.Type = val
65             else:
66                 print("Index not present")
67
68     def returnAtIndex(self ,index):
69         if self.head == None:
70             return None
```

```

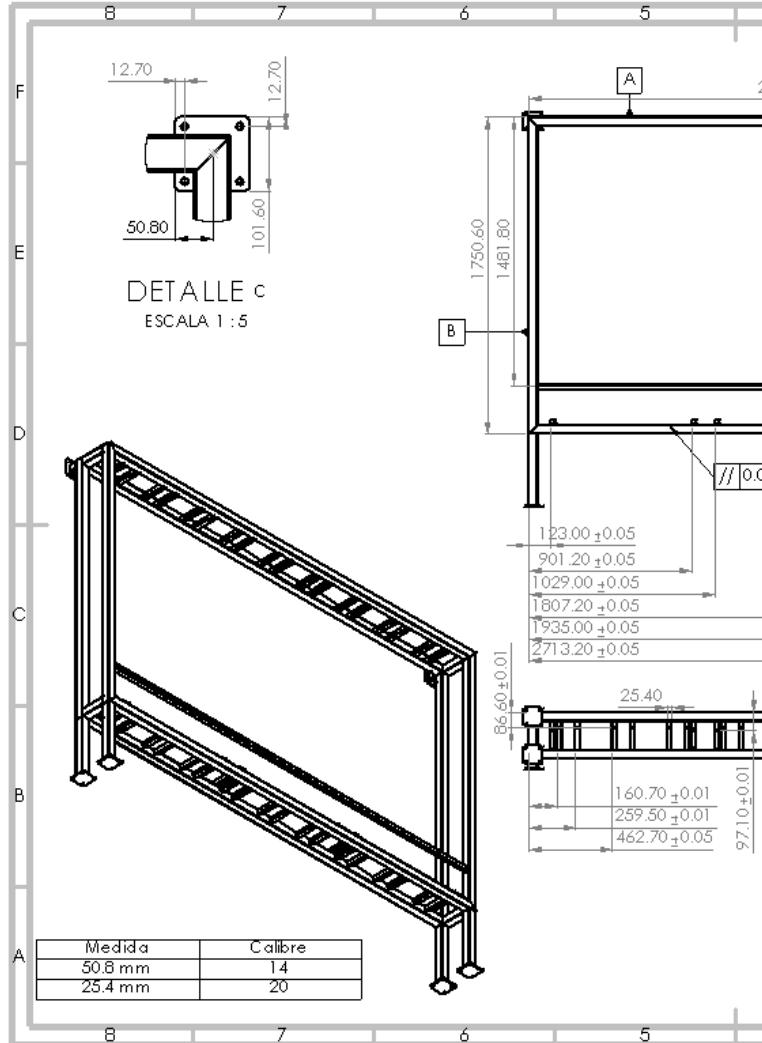
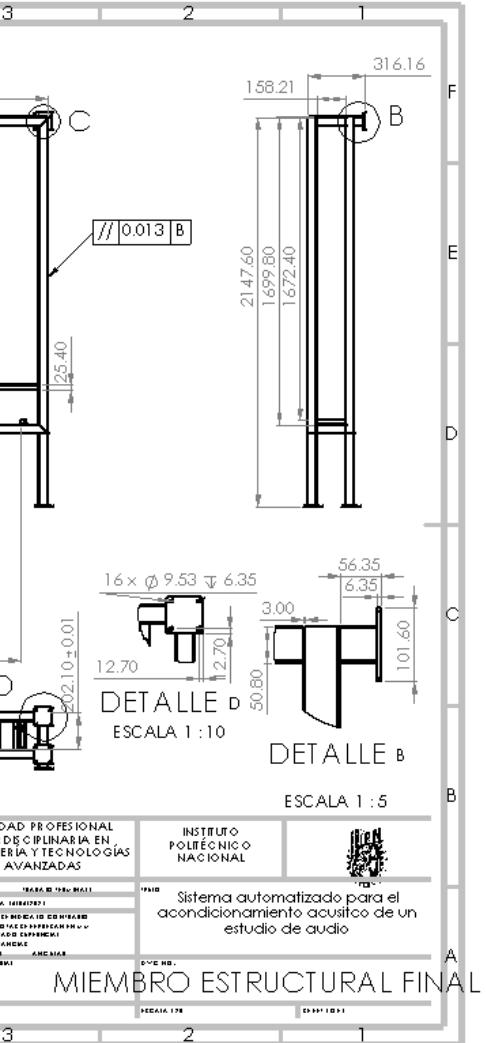
71     current_node = self.head
72     position = 0
73     while(current_node != None and position != index):
74         current_node = current_node.next
75         position += 1
76     if current_node != None:
77         return current_node.Type, current_node.Position
78     else:
79         print("Index not present")
80         return None
81
82
83 PanelOrder = LinkedList()
84 PanelOrder.insertAtBegin(NameToKey['Refle'], 14)
85 PanelOrder.insertAtEnd(NameToKey['Refle'], 5)
86 PanelOrder.insertAtEnd(NameToKey['Refle'], 12)
87 PanelOrder.insertAtEnd(NameToKey['Refle'], 16)
88 PanelOrder.insertAtEnd(NameToKey['Refle'], 3)
89 PanelOrder.insertAtEnd(NameToKey['Refle'], 7)
90 PanelOrder.insertAtEnd(NameToKey['Refle'], 10)
91 PanelOrder.insertAtEnd(NameToKey['Refle'], 18)
92 PanelOrder.insertAtEnd(NameToKey['Refle'], 1)
93 PanelOrder.insertAtEnd(NameToKey['Refle'], 9)
94 PanelOrder.insertAtEnd(NameToKey['Refle'], 13)
95 PanelOrder.insertAtEnd(NameToKey['Refle'], 15)
96 PanelOrder.insertAtEnd(NameToKey['Refle'], 4)
97 PanelOrder.insertAtEnd(NameToKey['Refle'], 6)
98 PanelOrder.insertAtEnd(NameToKey['Refle'], 11)
99 PanelOrder.insertAtEnd(NameToKey['Refle'], 17)
100 PanelOrder.insertAtEnd(NameToKey['Refle'], 2)
101 PanelOrder.insertAtEnd(NameToKey['Refle'], 8)
102
103 x = PanelsToChange(1, 0.7, room.dim)
104 print(f'Changing {int(x)} panels to Absorption')
105
106 if x > 18:
107     print('Only 18 panels available')
108     x = 18
109
110 DPanels = int(input(f"Enter the number of diffusion panels to put ({18-x} Available): "))
111 if DPanels > 18-x :
112     print("Not a valid number, using 0")
113     DPanels = 0
114
115 for i in range(18):
116     if i < x:
117         PanelOrder.updateNode(NameToKey['Abs'], i)
118     elif i < x + DPanels:
119         PanelOrder.updateNode(NameToKey['Scatt'], i)
120     else:
121         PanelOrder.updateNode(NameToKey['Refle'], i)
122
123

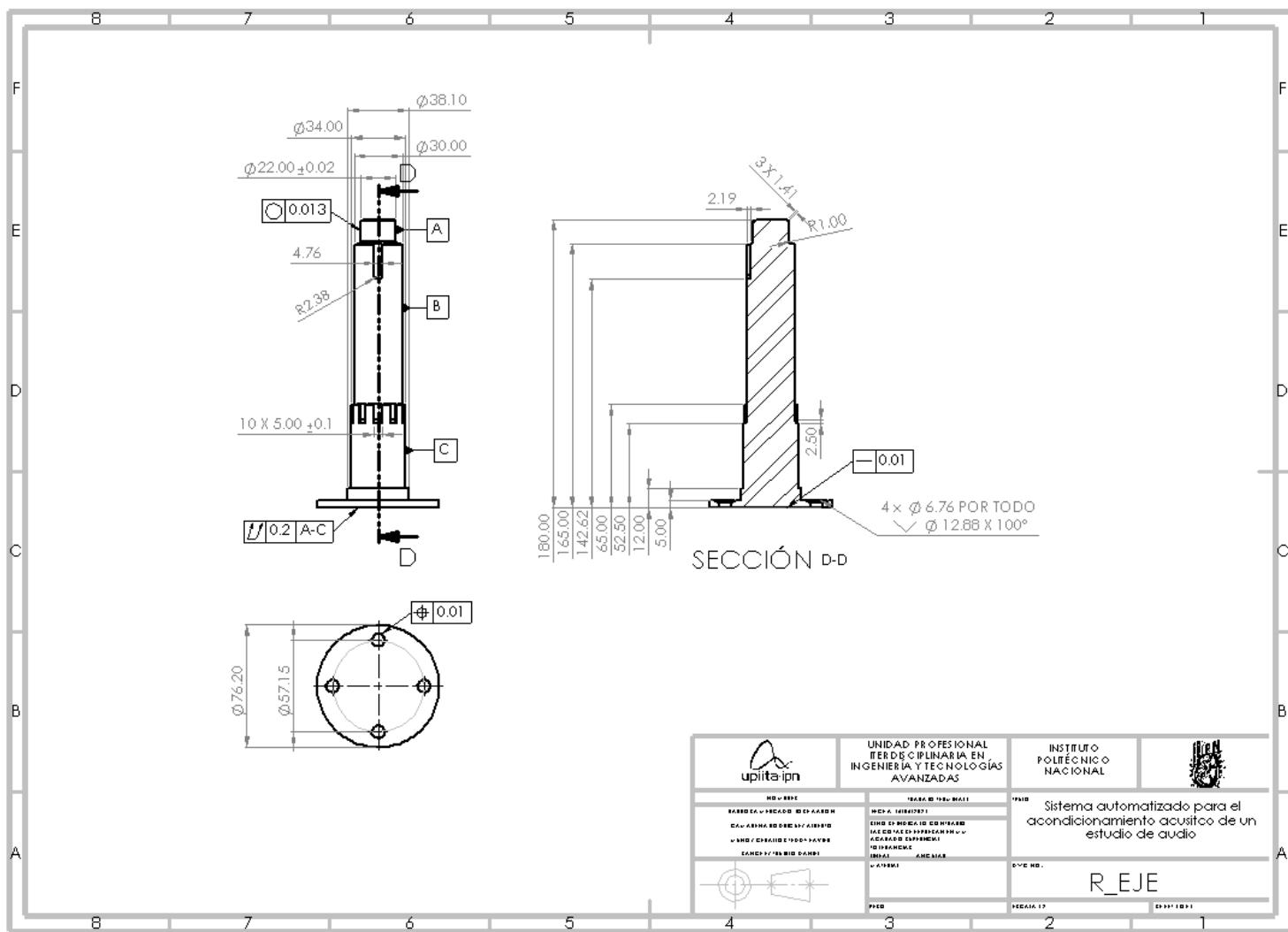
```

```
124| def GetPanels(PanelOrder):  
125|     PanelArray = {}  
126|     for i in range(18):  
127|         currentPanel_Type, currentPanel_Pos = PanelOrder.returnAtIndex(i)  
128|         PanelArray[currentPanel_Pos] = currentPanel_Type  
129|     return PanelArray  
130|  
131| PanelArray = GetPanels(PanelOrder)  
132| for i in range(1,19):  
133|     print(PanelArray[i])
```

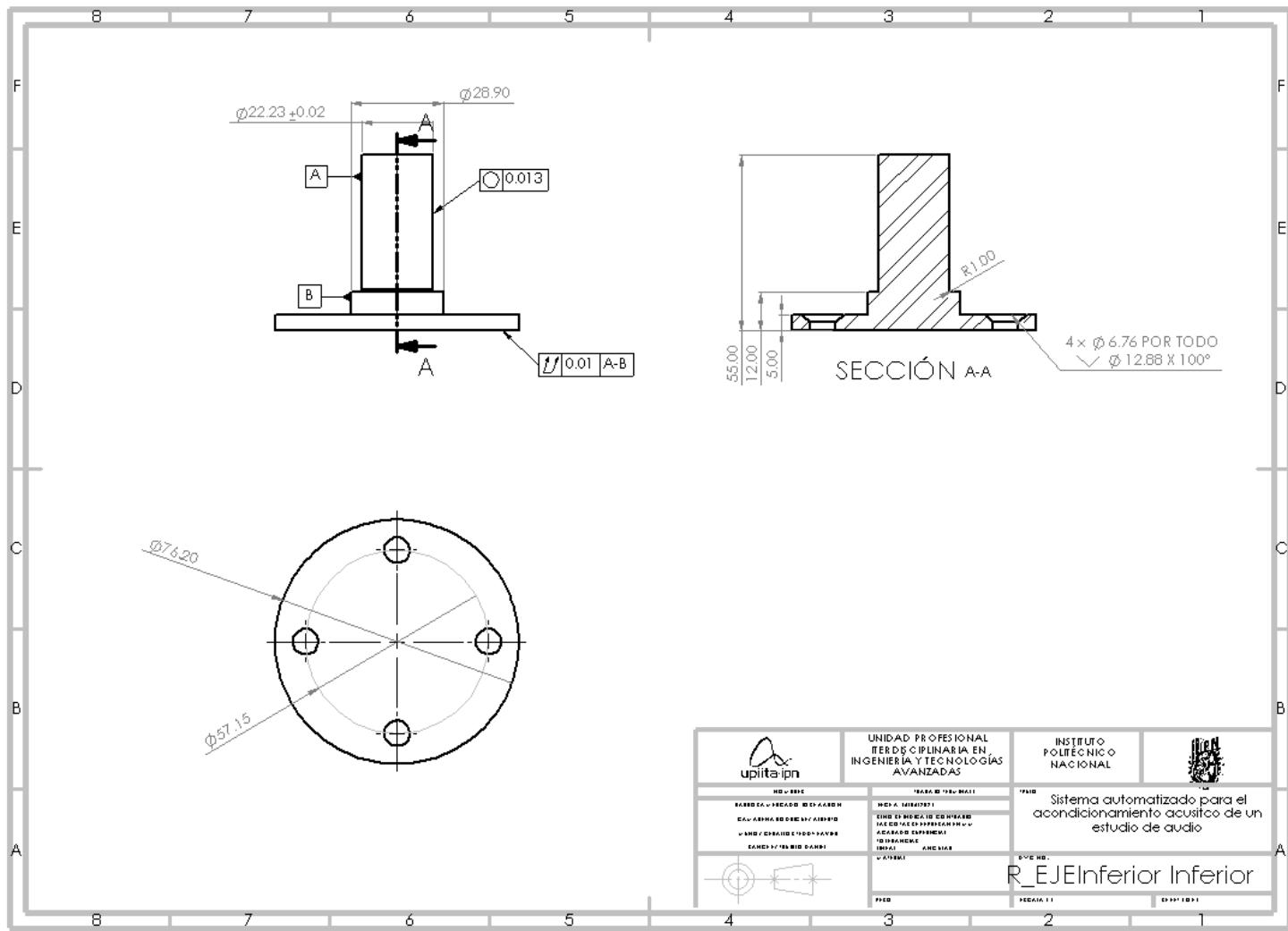
Anexos

A. Planos de manufactura

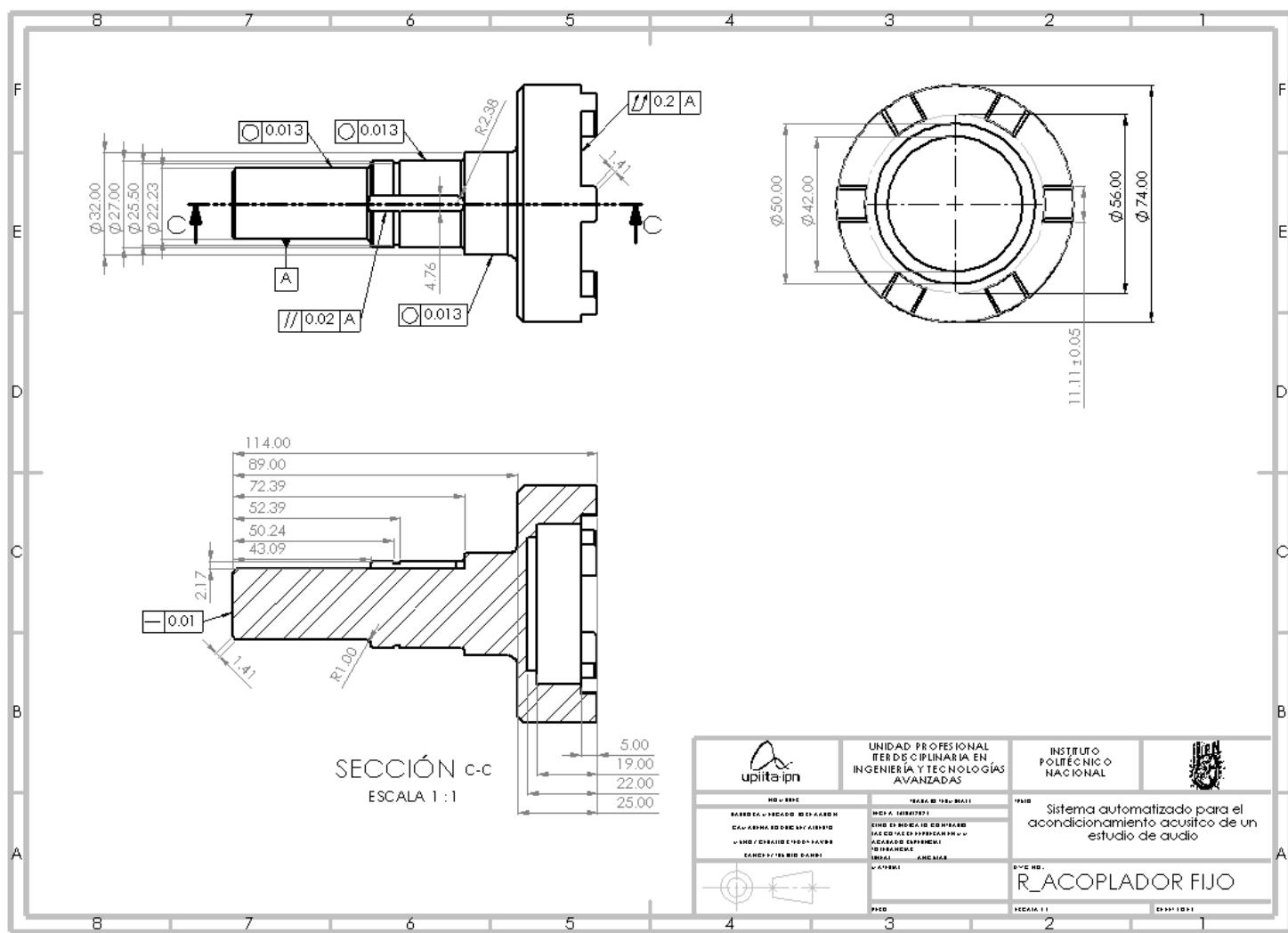


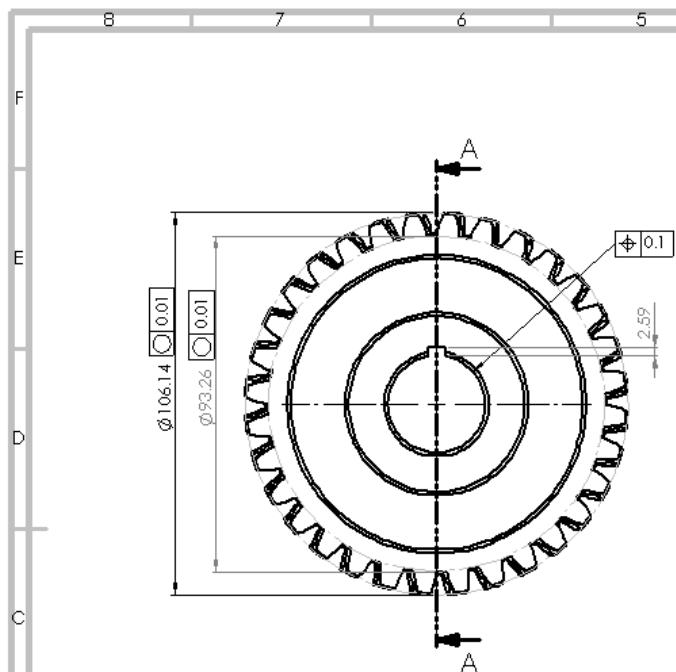
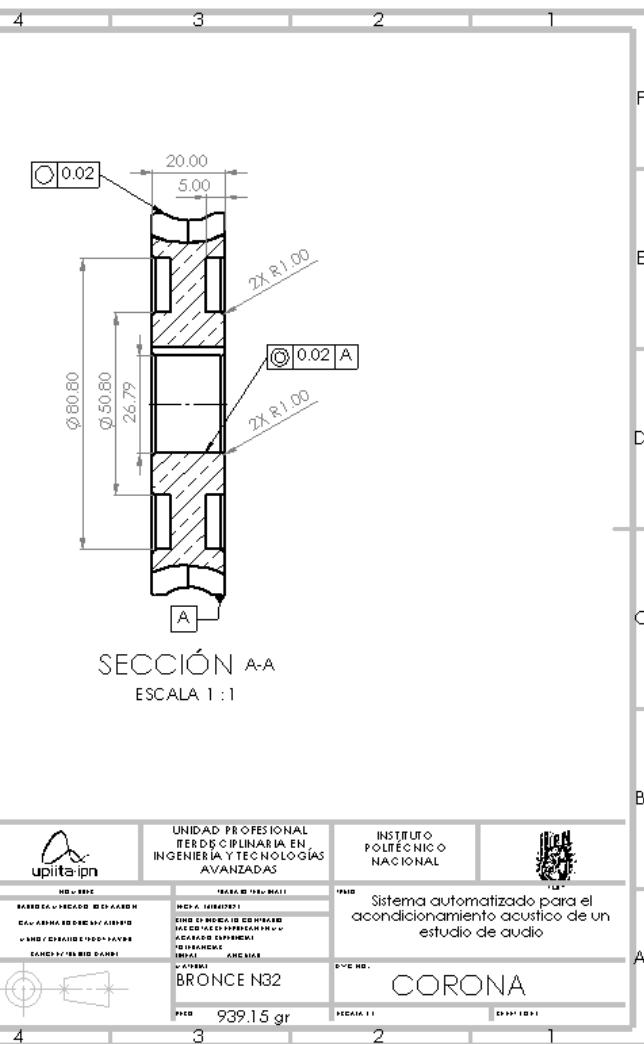


	UNIDAD PROFESIONAL INTERDISCIPLINARIA EN INGENIERÍA Y TECNOLOGÍAS AVANZADAS	
PROFESIÓN: SISTEMAS AUTOMATIZADOS PARA EL ACOJEDIMENTO ACÚSTICO DE UN ESTUDIO DE AUDÍO	TRABAJO TÍPICO: DISEÑO DE SISTEMAS AUTOMATIZADOS PARA EL ACONDICIONAMIENTO ACÚSTICO DE ESTUDIOS DE AUDÍO	FECHA: 17/01/2017
ESTRUCTURA: SISTEMA AUTOMATIZADO PARA EL ACONDICIONAMIENTO ACÚSTICO DE UN ESTUDIO DE AUDÍO	TIPO: SISTEMA AUTOMATIZADO PARA EL ACONDICIONAMIENTO ACÚSTICO DE UN ESTUDIO DE AUDÍO	FECHA: 17/01/2017



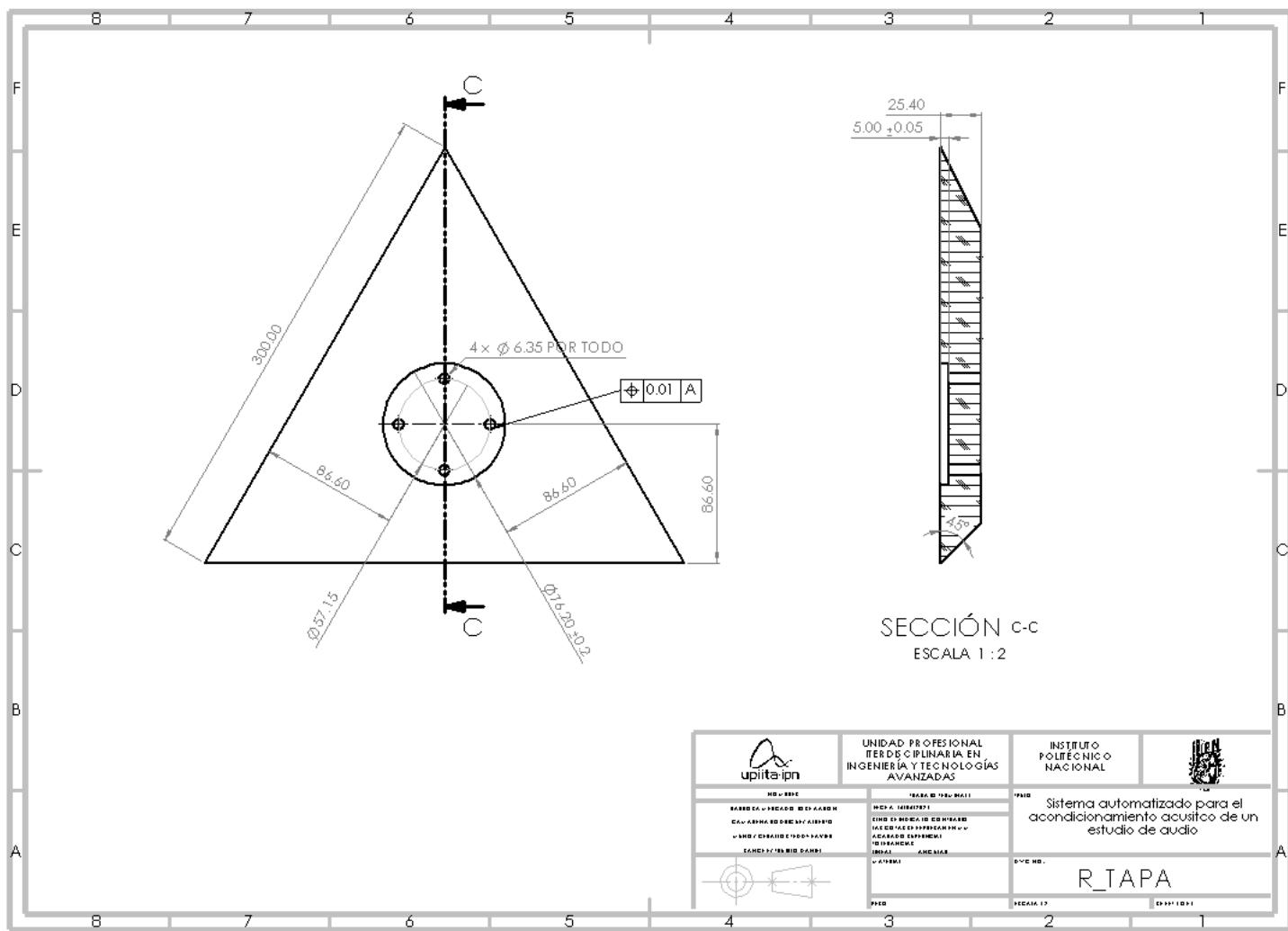
	UNIDAD PROFESIONAL TERCER CICLO EN INGENIERIA Y TECNOLOGIAS AVANZADAS		Sistema automatizado para el acondicionamiento acústico de un estudio de audio
ESTUDIOS DE INVESTIGACIONES DE AVANZADA EN INGENIERIA Y SISTEMAS COMPUTACIONALES DAMONTE RIBOLINI	PROF. J. M. GARCIA PROF. L. G. VILLENA PROF. J. M. GARCIA PROF. J. M. GARCIA	INSTITUTO POLITECNICO NACIONAL	R_EJEInferior Inferior

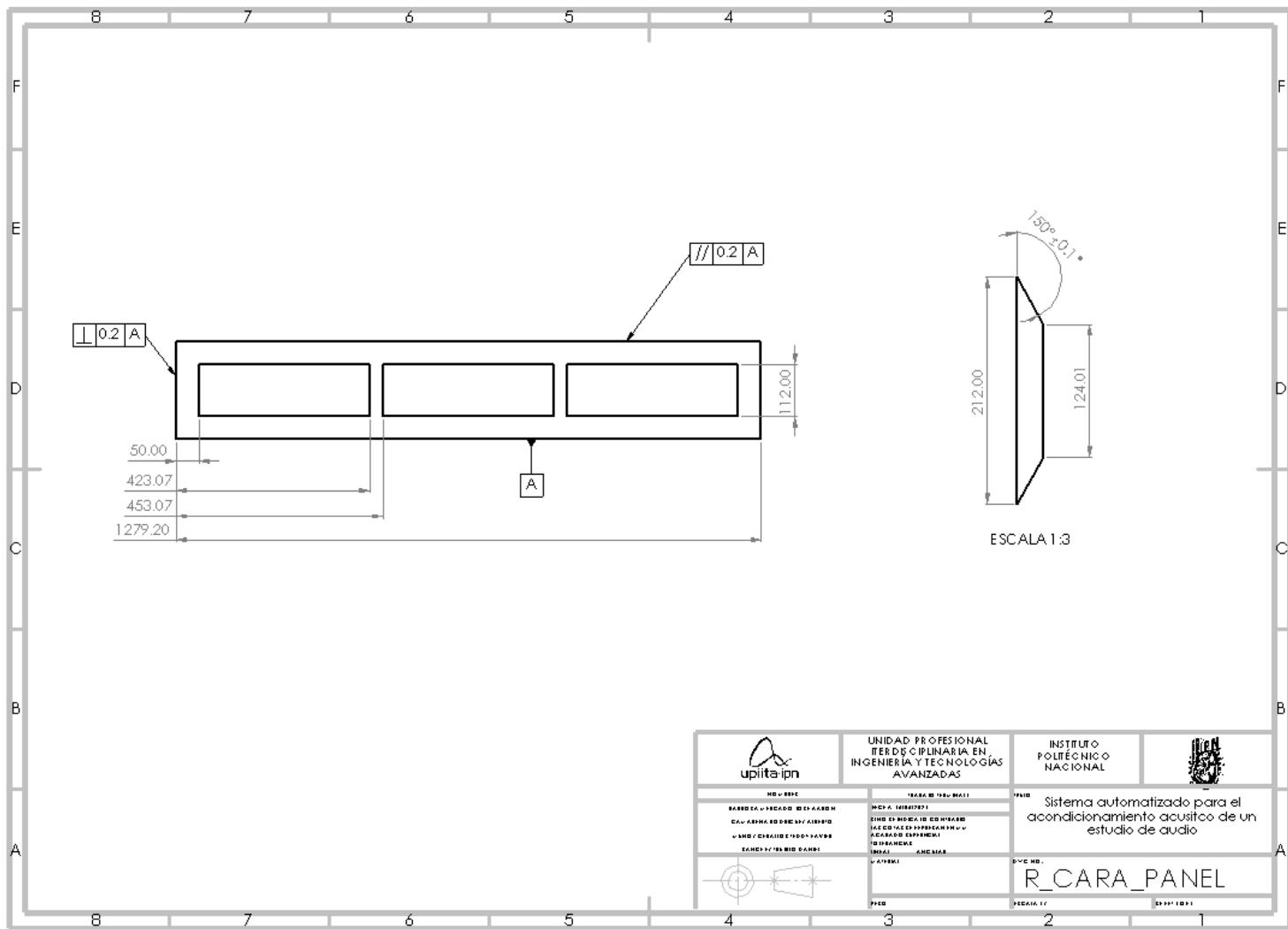


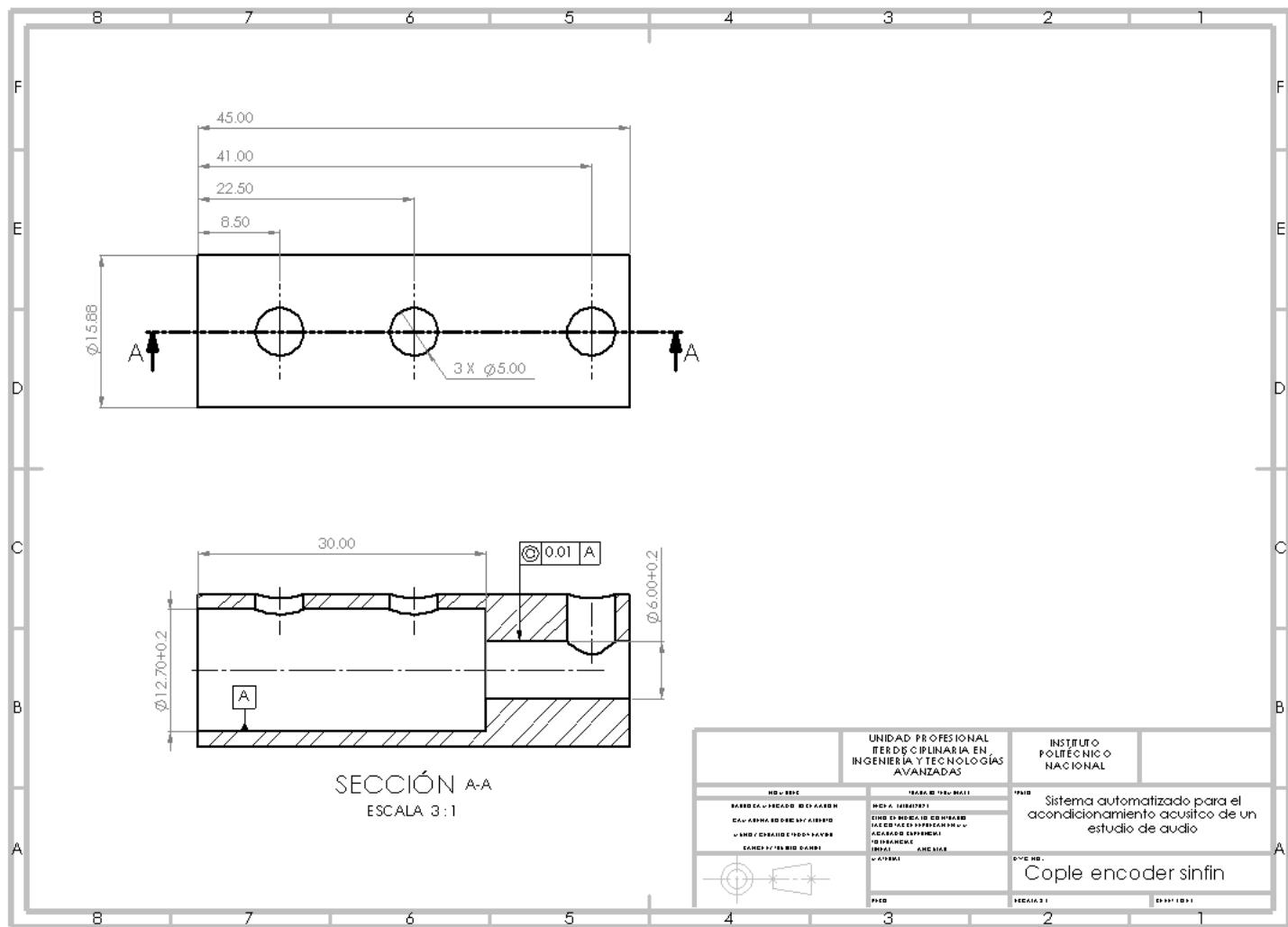


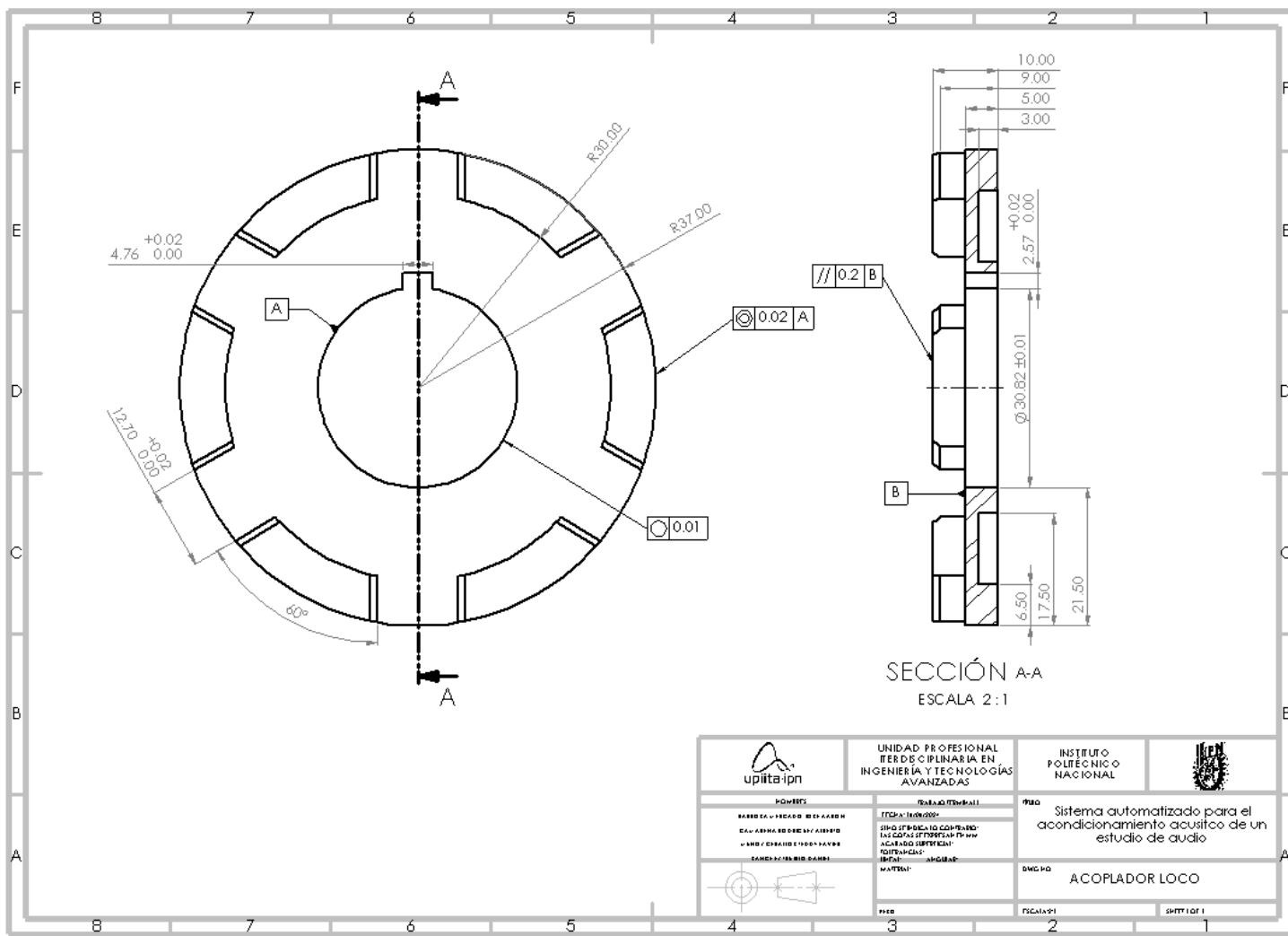
A	Número de dientes	N_g	32
	Modulo	M	3
	Angulo de presion	ϕ_n	20 °
	Diametro de paso	d _g	96 mm



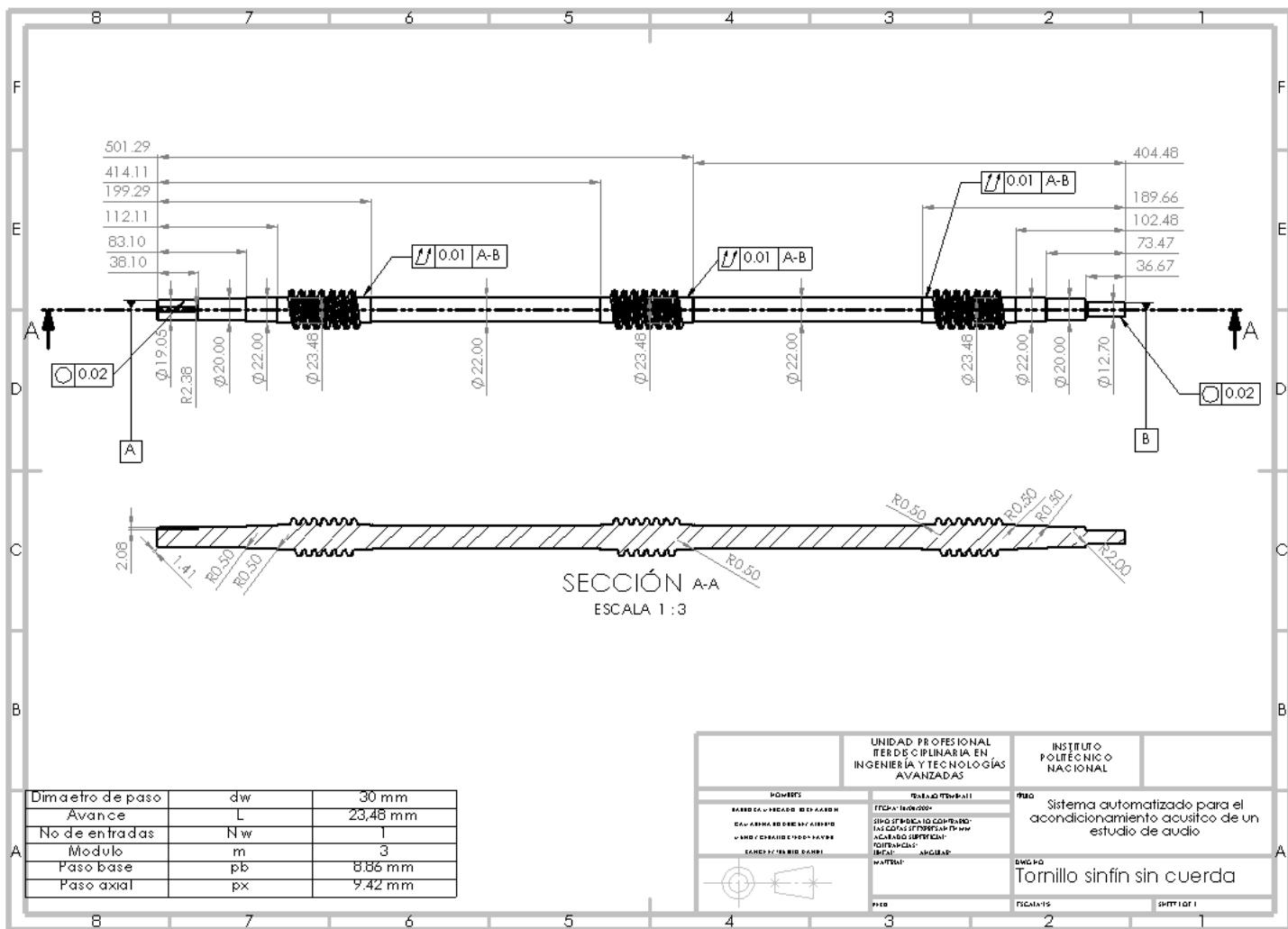


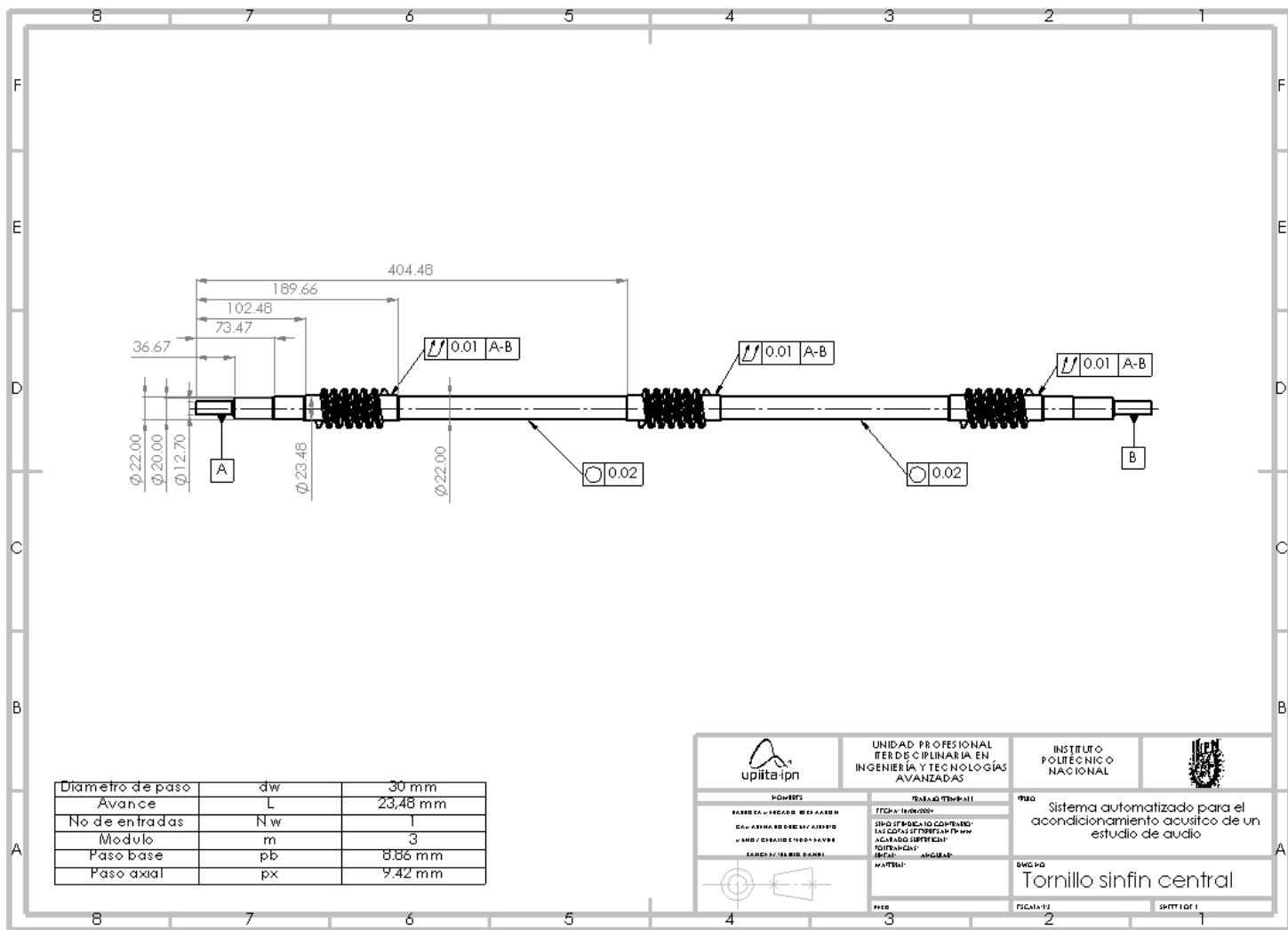


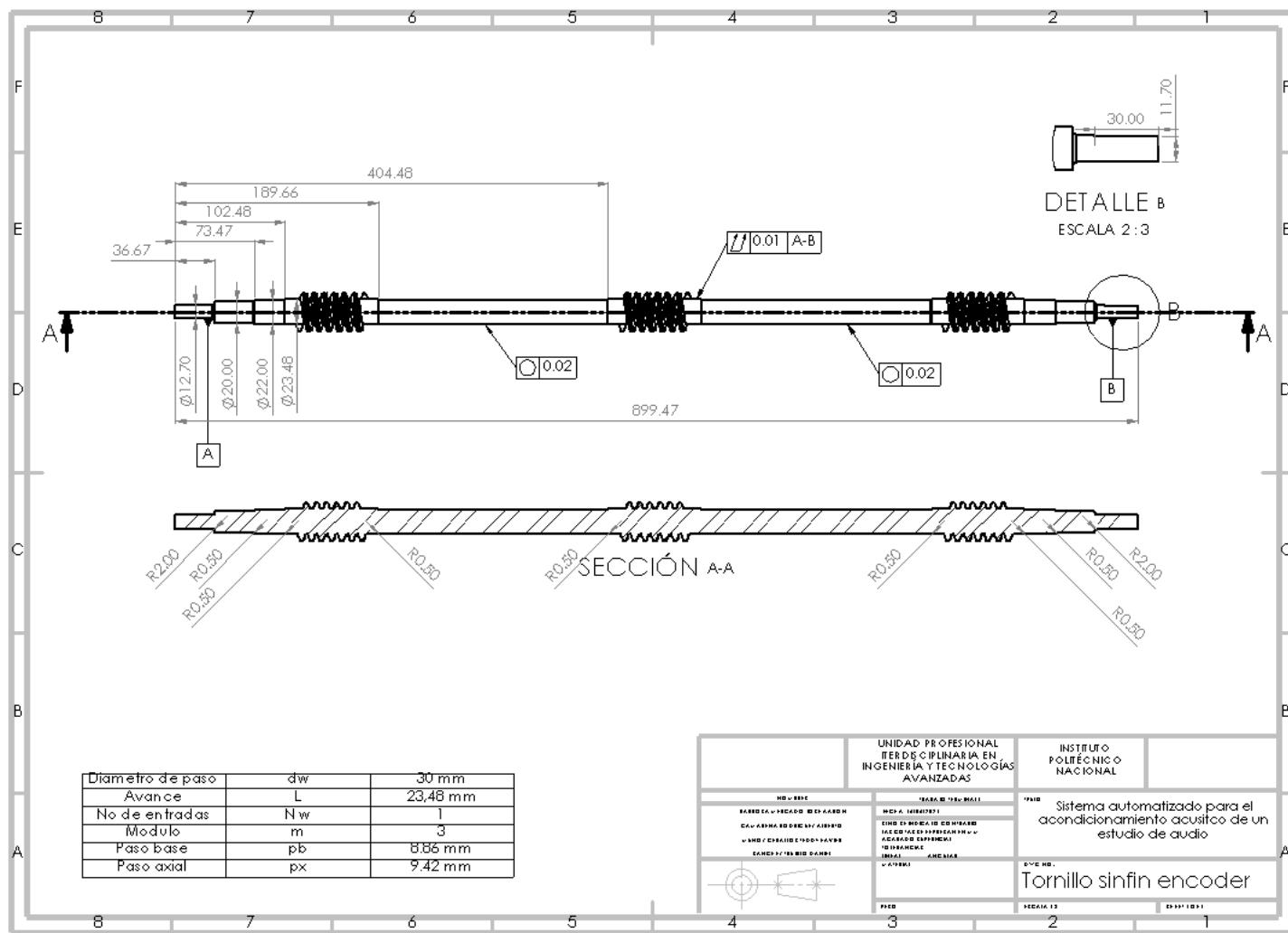


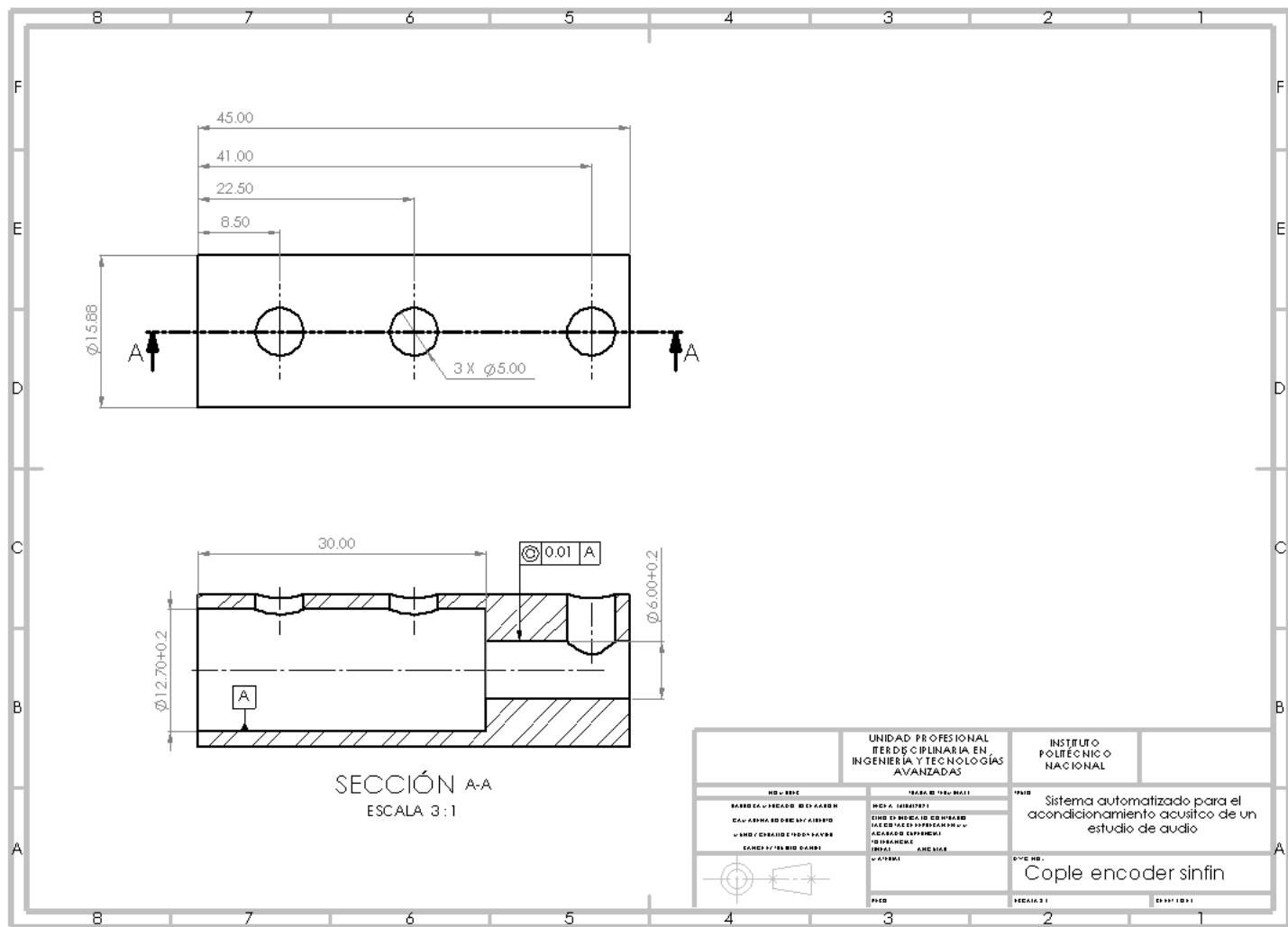


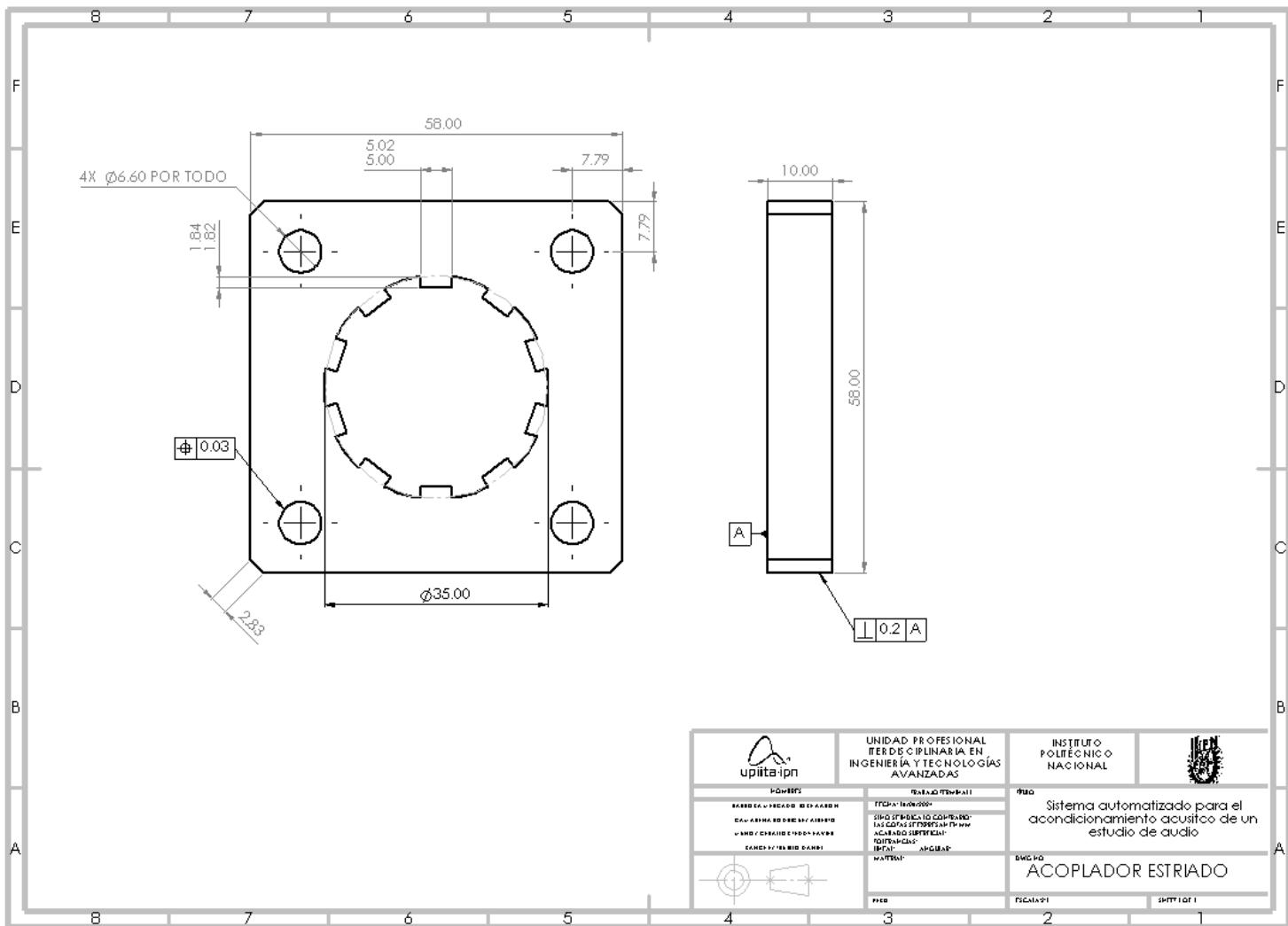
SECCIÓN A-A
ESCALA 1 : 3

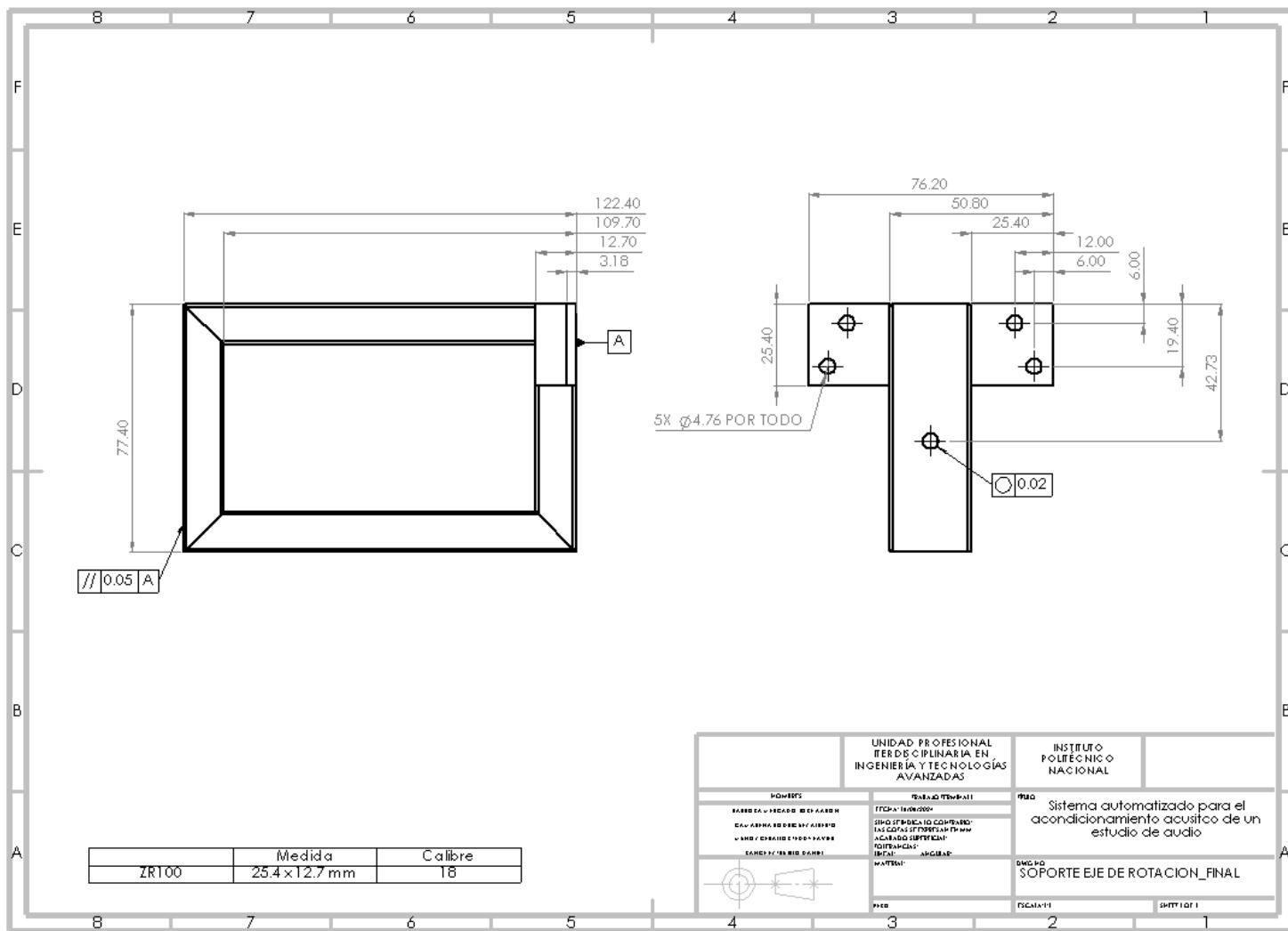


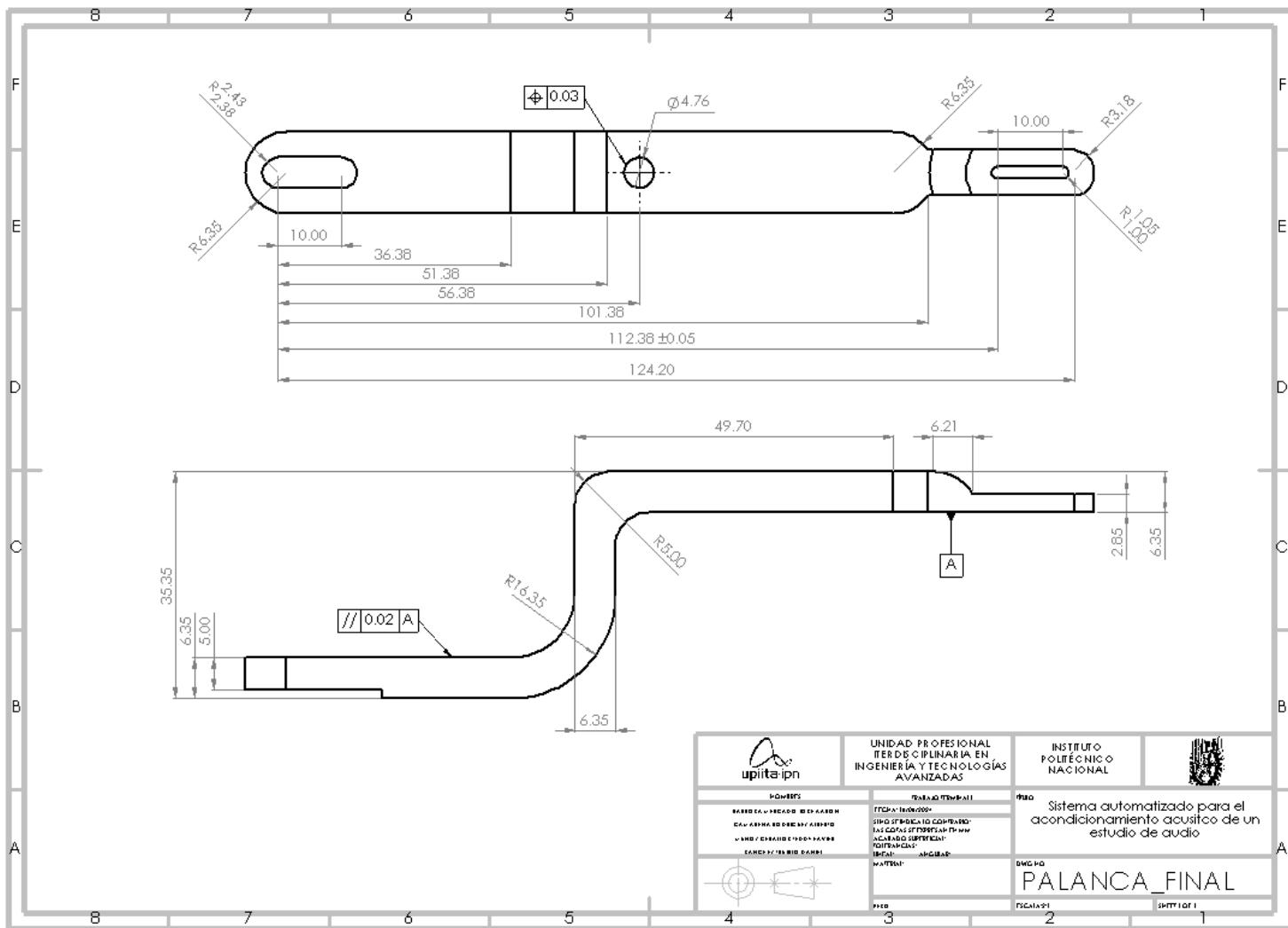


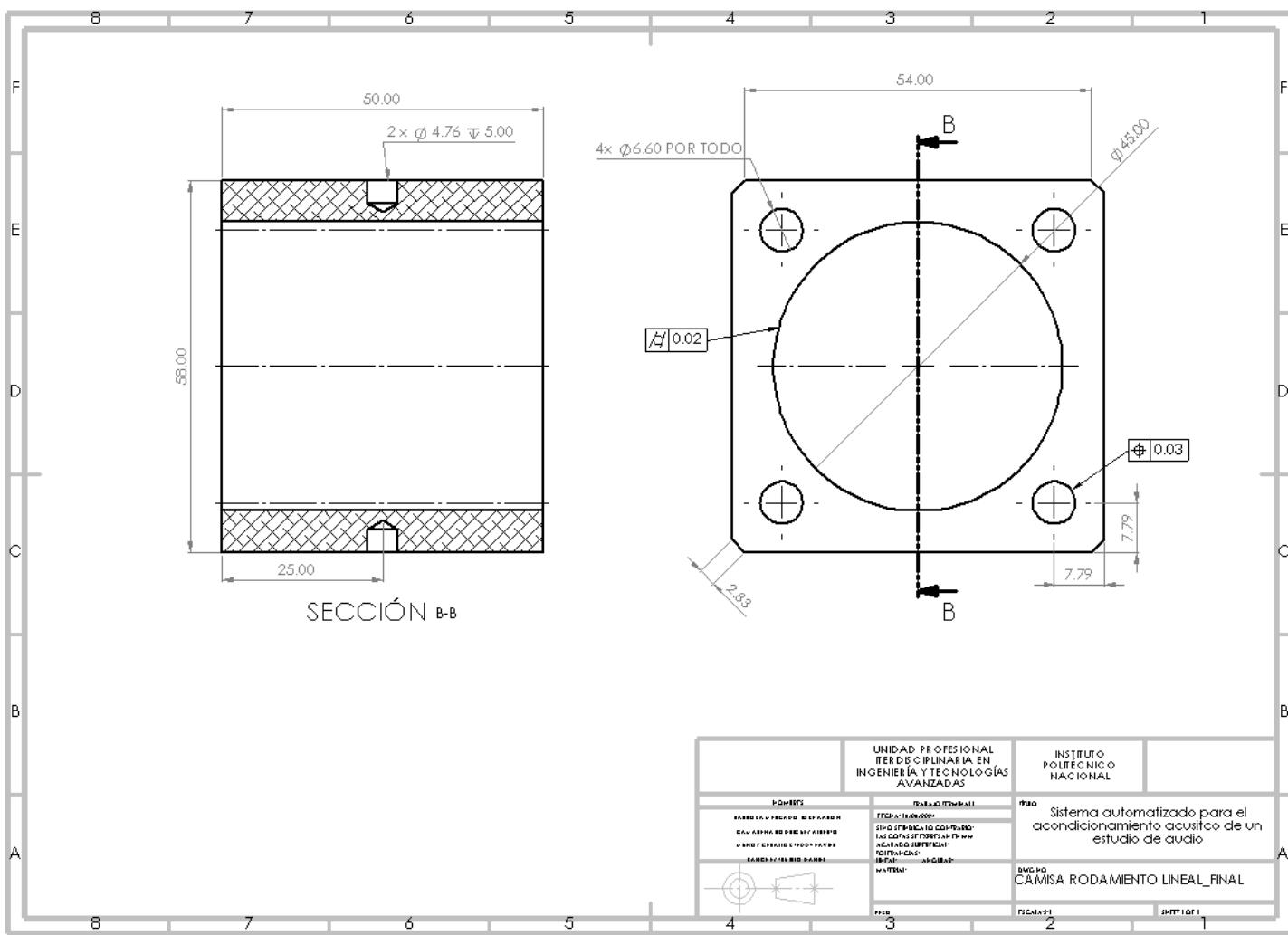












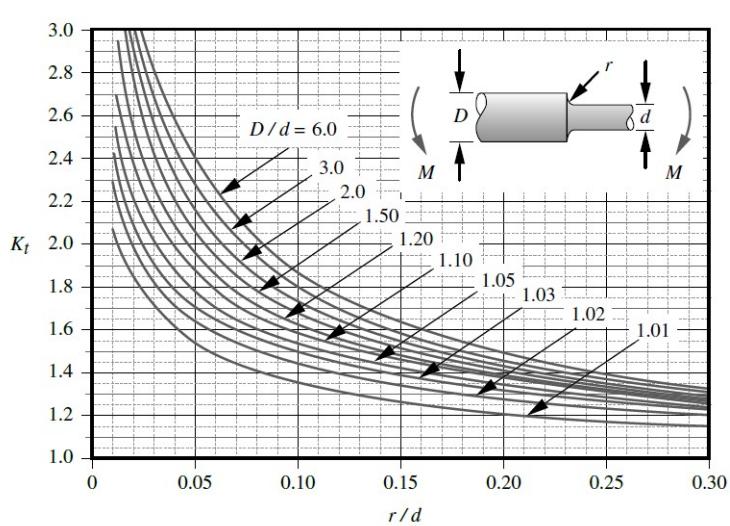
B. Propiedades mecánicas de aceros al carbón

A.2 Mechanical Properties of Selected Carbon Steels

AISI Number	Condition ^a	Yield Strength, MPa	Ultimate Tensile Strength, MPa	Elongation in 50 mm, %	Reduction in Area, %	Brinell Hardness, HB
1006	Hot Rolled	170	300	30	55	85
	Cold Drawn	280	330	20	45	95
1010	Cold Drawn	305	365	20	40	105
	Hot Rolled	180	325	28	50	95
1015	Cold Drawn	325	385	18	40	111
	Hot Rolled	190	340	28	50	100
1018	Cold Drawn	370	440	15	40	126
	Hot Rolled	220	400	25	50	116
1020	Q&T 870° C	295	395	37	60	100
	Cold Drawn	350	420	15	40	121
1030	Hot Rolled	205	380	25	50	110
	Annealed	317	430	30	60	130
1030	N 925° C	345	520	32	61	150
	Q&T 205° C	648	848	17	47	495
	Q&T 315° C	621	800	19	53	400
	Q&T 425° C	579	731	23	60	300
	Q&T 540° C	517	669	28	65	250
	Q&T 650° C	441	586	32	70	210
	Cold Drawn	440	525	12	35	149
1040	Hot Rolled	260	70	20	40	130
	Annealed	350	520	30	57	150
	N 900° C	374	590	28	55	170
	Q&T 205° C	593	779	19	48	262
	Q&T 425° C	552	758	21	54	240
	Q&T 650° C	434	634	29	65	192
	Cold Drawn	490	585	12	35	170
1050	Hot Rolled	290	525	18	40	149
	Annealed	365	636	24	40	190
	N 900° C	427	748	20	39	220
	Q&T 205° C	807	1120	9	27	514
	Q&T 425° C	793	1090	13	36	444
	Q&T 650° C	538	717	28	65	235
	Cold Drawn	580	690	10	30	197
1060	Hot Rolled	340	620	15	35	179
	Annealed	372	626	22	38	179
	N 900° C	421	776	18	37	230
	Q&T 425° C	765	1080	14	40	310
	Q&T 540° C	669	965	17	45	280
	Q&T 650° C	524	800	23	40	230
	Hot Rolled	370	680	12	30	201
1080	Q&T 800° C	380	615	25	30	255
	Hot Rolled	420	770	10	25	229
1095	Annealed	380	658	13	21	190
	N 900° C	500	1010	9	13	293
	Q&T 315° C	813	1260	10	30	375
	Q&T 425° C	772	1210	12	32	360
	Q&T 540° C	676	1090	15	37	320
	Q&T 650° C	552	896	21	30	269
	Hot Rolled	455	825	10	25	248
	Cold Drawn	525	680	10	20	197

Note: ^a N, normalized; Q&T, quenched and tempered

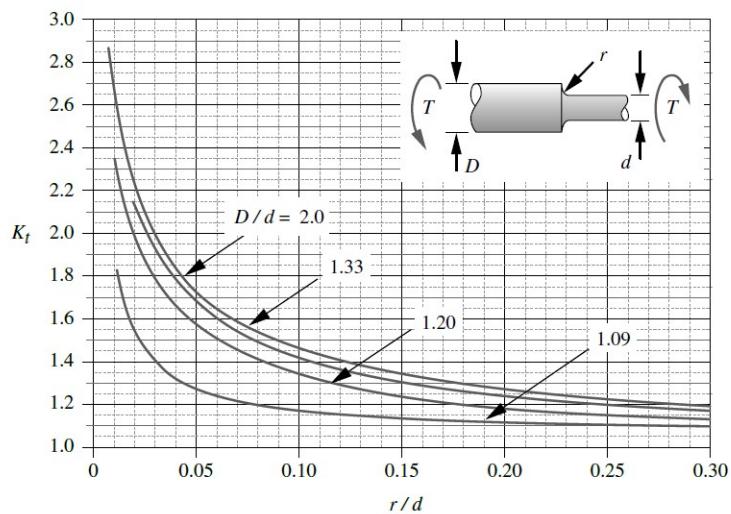
C. Factores de concentración de esfuerzos



$$K_t \equiv A \left(\frac{r}{d} \right)^b$$

donde:

D/d	A	b
6.00	0.878 68	-0.332 43
3.00	0.893 34	-0.308 60
2.00	0.908 79	-0.285 98
1.50	0.938 36	-0.257 59
1.20	0.970 98	-0.217 96
1.10	0.951 20	-0.237 57
1.07	0.975 27	-0.209 58
1.05	0.981 37	-0.196 53
1.03	0.980 61	-0.183 81
1.02	0.960 48	-0.177 11
1.01	0.919 38	-0.170 32



$$K_t \equiv A \left(\frac{r}{d} \right)^b$$

donde:

D/d	A	b
2.00	0.863 31	-0.238 65
1.33	0.848 97	-0.231 61
1.20	0.834 25	-0.216 49
1.09	0.903 37	-0.126 92

D. Dimensiones y propiedades del tubo cuadrado PHERT

H TUBO CUADRADO PHERT DIMENSIONES Y PROPIEDADES														PERFIL		
PERFIL	PERFILES ESTRUCTURALES RECTANGULARES												PERFIL			
	DIMENSIONES Y PESOS TEÓRICOS				PROPIEDADES ESTRUCTURALES											
	pulgadas	millímetros	CAL	COLOR	ESPESOR plgs	LONG mm	PESO M	AREA Kg/m	I_x cm ⁴	S_x cm ³	r_x cm	I_y cm ⁴	S_y cm ³	r_y cm		
H0505	1/2 x 1/2	12.7 x 12.7	20		0.036	0.912	6	0.34	0.43	0.10	0.16	0.48	0.10	0.16	0.48	H0505
H0505	1/2 x 1/2	12.7 x 12.7	18		0.048	1.214	6	0.44	0.56	0.12	0.20	0.47	0.12	0.20	0.47	H0505
H0505	1/2 x 1/2	12.7 x 12.7	16		0.060	1.519	6	0.53	0.68	0.14	0.23	0.46	0.14	0.23	0.46	H0505
H0606	5/8 x 5/8	15.9 x 15.9	20		0.036	0.912	6	0.43	0.55	0.20	0.26	0.61	0.20	0.26	0.61	H0606
H0606	5/8 x 5/8	15.9 x 15.9	18		0.048	1.214	6	0.56	0.71	0.26	0.32	0.60	0.26	0.32	0.60	H0606
H0606	5/8 x 5/8	15.9 x 15.9	16		0.060	1.519	6	0.69	0.87	0.30	0.38	0.59	0.30	0.38	0.59	H0606
H0707	3/4 x 3/4	19.0 x 19.0	20		0.036	0.912	6	0.52	0.66	0.36	0.38	0.74	0.36	0.38	0.74	H0707
H0707	3/4 x 3/4	19.0 x 19.0	18		0.048	1.214	6	0.68	0.87	0.46	0.48	0.73	0.46	0.48	0.73	H0707
H0707	3/4 x 3/4	19.0 x 19.0	16		0.060	1.519	6	0.84	1.07	0.55	0.58	0.72	0.55	0.58	0.72	H0707
H0707	3/4 x 3/4	19.0 x 19.0	14		0.075	1.897	6	1.02	1.30	0.65	0.68	0.70	0.65	0.68	0.70	H0707
H0808	13/16 x 13/16	20.6 x 20.6	20		0.036	0.912	6	0.57	0.72	0.47	0.45	0.81	0.47	0.45	0.81	H0808
H0808	13/16 x 13/16	20.6 x 20.6	18		0.048	1.214	6	0.74	0.94	0.60	0.58	0.79	0.60	0.58	0.79	H0808
H0808	13/16 x 13/16	20.6 x 20.6	16		0.060	1.519	6	0.91	1.16	0.71	0.69	0.78	0.71	0.69	0.78	H0808
H0909	7/8 x 7/8	22.2 x 22.2	20		0.036	0.912	6	0.61	0.78	0.59	0.53	0.87	0.59	0.53	0.87	H0909
H0909	7/8 x 7/8	22.2 x 22.2	18		0.048	1.214	6	0.80	1.02	0.75	0.68	0.86	0.75	0.68	0.86	H0909
H0909	7/8 x 7/8	22.2 x 22.2	16		0.060	1.519	6	0.99	1.26	0.90	0.81	0.95	0.90	0.81	0.95	H0909
H1010	1 x 1	25.4 x 25.4	20		0.036	0.912	6	0.70	0.89	0.89	0.70	1.00	0.89	0.70	1.00	H1010
H1010	1 x 1	25.4 x 25.4	18		0.048	1.214	6	0.92	1.17	1.15	0.90	0.99	1.15	0.90	0.99	H1010
H1010	1 x 1	25.4 x 25.4	16		0.060	1.519	6	1.14	1.45	1.38	1.09	0.98	1.38	1.09	0.98	H1010
H1010	1 x 1	25.4 x 25.4	14	AZUL	0.075	1.897	6	1.40	1.78	1.65	1.30	0.96	1.65	1.30	0.96	H1010
H1010	1 x 1	25.4 x 25.4	13	VERDE	0.090	2.278	6	1.66	2.11	1.90	1.49	0.95	1.90	1.49	0.95	H1010
H1010	1 x 1	25.4 x 25.4	12		0.105	2.657	6	1.90	2.42	2.11	1.66	0.93	2.11	1.66	0.93	H1010
H1212	1 1/4 x 1 1/4	31.8 x 31.8	20		0.036	0.912	6	0.89	1.12	1.78	1.12	1.26	1.78	1.12	1.26	H1212
H1212	1 1/4 x 1 1/4	31.8 x 31.8	18		0.048	1.214	6	1.17	1.48	2.31	1.45	1.25	2.31	1.45	1.25	H1212
H1212	1 1/4 x 1 1/4	31.8 x 31.8	16		0.060	1.519	6	1.45	1.84	2.80	1.77	1.24	2.80	1.77	1.24	H1212
H1212	1 1/4 x 1 1/4	31.8 x 31.8	14	AZUL	0.075	1.897	6	1.78	2.27	3.38	2.13	1.22	3.38	2.13	1.22	H1212
H1212	1 1/4 x 1 1/4	31.8 x 31.8	13		0.090	2.278	6	2.11	2.69	3.91	2.46	1.21	3.91	2.46	1.21	H1212
H1212	1 1/4 x 1 1/4	31.8 x 31.8	12	BLANCO	0.105	2.657	6	2.43	3.09	4.40	2.77	1.19	4.40	2.77	1.19	H1212
H1212	1 1/4 x 1 1/4	31.8 x 31.8	11		0.120	3.038	6	2.75	3.49	4.85	3.05	1.18	4.85	3.05	1.18	H1212

E. Hoja de datos del MOSFET IRL2703

International

IR Rectifier

PRELIMINARY

PD - 9.1359

IRL2703

HEXFET® Power MOSFET

- Logic-Level Gate Drive
- Advanced Process Technology
- Dynamic dv/dt Rating
- 175°C Operating Temperature
- Fast Switching
- Fully Avalanche Rated

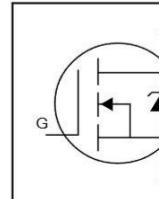
Description

Fifth Generation HEXFETs from International Rectifier utilize advanced processing techniques to achieve the lowest possible on-resistance per silicon area. This benefit, combined with the fast switching speed and ruggedized device design that HEXFET Power MOSFETs are well known for, provides the designer with an extremely efficient device for use in a wide variety of applications.

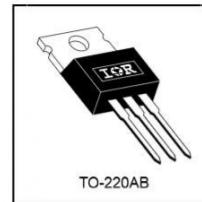
The TO-220 package is universally preferred for all commercial-industrial applications at power dissipation levels to approximately 50 watts. The low thermal resistance and low package cost of the TO-220 contribute to its wide acceptance throughout the industry.

Absolute Maximum Ratings

	Parameter	Max.	Units
$I_D @ T_C = 25^\circ C$	Continuous Drain Current, $V_{GS} @ 10V$	24	
$I_D @ T_C = 100^\circ C$	Continuous Drain Current, $V_{GS} @ 10V$	17	A
I_{DM}	Pulsed Drain Current \textcircled{D}	96	
$P_D @ T_C = 25^\circ C$	Power Dissipation	45	W
	Linear Derating Factor	0.30	$W/\text{^\circ C}$
V_{GS}	Gate-to-Source Voltage	± 16	V
E_{AS}	Single Pulse Avalanche Energy \textcircled{D}	77	mJ
I_{AR}	Avalanche Current \textcircled{D}	14	A
E_{AR}	Repetitive Avalanche Energy \textcircled{D}	4.5	mJ
dv/dt	Peak Diode Recovery $dv/dt \textcircled{D}$	3.5	V/ns
T_J	Operating Junction and	-55 to +175	
T_{STG}	Storage Temperature Range		$^\circ C$
	Soldering Temperature, for 10 seconds	300 (1.6mm from case)	
	Mounting torque, 6-32 or M3 screw.	10 lbf·in (1.1N·m)	



$V_{DSS} = 30V$
 $R_{DS(on)} = 0.04\Omega$
 $I_D = 24A$



Thermal Resistance

	Parameter	Min.	Typ.	Max.	Units
R_{JC}	Junction-to-Case	—	—	3.3	
R_{CS}	Case-to-Sink, Flat, Greased Surface	—	0.50	—	$^\circ C/W$
R_{JA}	Junction-to-Ambient	—	—	62	

11/18/96

IRL2703

International
Rectifier

Electrical Characteristics @ $T_J = 25^\circ\text{C}$ (unless otherwise specified)

	Parameter	Min.	Typ.	Max.	Units	Conditions
$V_{(\text{BR})\text{DSS}}$	Drain-to-Source Breakdown Voltage	30	—	—	V	$V_{GS} = 0V, I_D = 250\mu\text{A}$
$\Delta V_{(\text{BR})\text{DSS}/\Delta T_J}$	Breakdown Voltage Temp. Coefficient	—	0.030	—	V/ $^\circ\text{C}$	Reference to $25^\circ\text{C}, I_D = 1\text{mA}$
$R_{DS(\text{on})}$	Static Drain-to-Source On-Resistance	—	—	0.040	Ω	$V_{GS} = 10V, I_D = 14\text{A}$ ④
		—	—	0.060		$V_{GS} = 4.5V, I_D = 12\text{A}$ ④
$V_{GS(\text{th})}$	Gate Threshold Voltage	1.0	—	—	V	$V_{DS} = V_{GS}, I_D = 250\mu\text{A}$
g_f	Forward Transconductance	6.4	—	—	S	$V_{DS} = 25V, I_D = 14\text{A}$
I_{DSS}	Drain-to-Source Leakage Current	—	—	25	μA	$V_{DS} = 30V, V_{GS} = 0V$
		—	—	250		$V_{DS} = 24V, V_{GS} = 0V, T_J = 150^\circ\text{C}$
I_{GSS}	Gate-to-Source Forward Leakage	—	—	100	nA	$V_{GS} = 16V$
	Gate-to-Source Reverse Leakage	—	—	-100		$V_{GS} = -16V$
Q_g	Total Gate Charge	—	—	15	nC	$I_D = 14\text{A}$
Q_{gs}	Gate-to-Source Charge	—	—	4.6		$V_{DS} = 24V$
Q_{gd}	Gate-to-Drain ("Miller") Charge	—	—	9.3		$V_{GS} = 4.5V$, See Fig. 6 and 13 ④
$t_{d(on)}$	Turn-On Delay Time	—	8.5	—	ns	$V_{DD} = 15V$
t_r	Rise Time	—	140	—		$I_D = 14\text{A}$
$t_{d(off)}$	Turn-Off Delay Time	—	12	—		$R_G = 12\Omega, V_{GS} = 4.5V$
t_f	Fall Time	—	20	—		$R_D = 1.0\Omega$, See Fig. 10 ④
L_D	Internal Drain Inductance	—	4.5	—	nH	Between lead, 6mm (0.25in.) from package and center of die contact
L_S	Internal Source Inductance	—	7.5	—		
C_{iss}	Input Capacitance	—	450	—	pF	$V_{GS} = 0V$
C_{oss}	Output Capacitance	—	210	—		$V_{DS} = 25V$
C_{rss}	Reverse Transfer Capacitance	—	110	—		$f = 1.0\text{MHz}$, See Fig. 5

Source-Drain Ratings and Characteristics

	Parameter	Min.	Typ.	Max.	Units	Conditions
I_S	Continuous Source Current (Body Diode)	—	—	24	A	MOSFET symbol showing the integral reverse p-n junction diode.
I_{SM}	Pulsed Source Current (Body Diode) ①	—	—	96		
V_{SD}	Diode Forward Voltage	—	—	1.3	V	$T_J = 25^\circ\text{C}, I_S = 14\text{A}, V_{GS} = 0V$ ④
t_{rr}	Reverse Recovery Time	—	65	97	ns	$T_J = 25^\circ\text{C}, I_F = 14\text{A}$
Q_{rr}	Reverse Recovery Charge	—	140	210	nC	$dI/dt = 100\text{A}/\mu\text{s}$ ④
t_{on}	Forward Turn-On Time	Intrinsic turn-on time is negligible (turn-on is dominated by $L_S + L_D$)				

Notes:

- ① Repetitive rating; pulse width limited by max. junction temperature. (See fig. 11)
- ② $V_{DD} = 15V$, starting $T_J = 25^\circ\text{C}$, $L = 570\mu\text{H}$
 $R_G = 25\Omega, I_{AS} = 14\text{A}$. (See Figure 12)
- ③ $I_{SD} \leq 14\text{A}$, $dI/dt \leq 140\text{A}/\mu\text{s}$, $V_{DD} \leq V_{(\text{BR})\text{DSS}}$,
 $T_J \leq 175^\circ\text{C}$
- ④ Pulse width $\leq 300\mu\text{s}$; duty cycle $\leq 2\%$.

International
IGR Rectifier

IRL2703

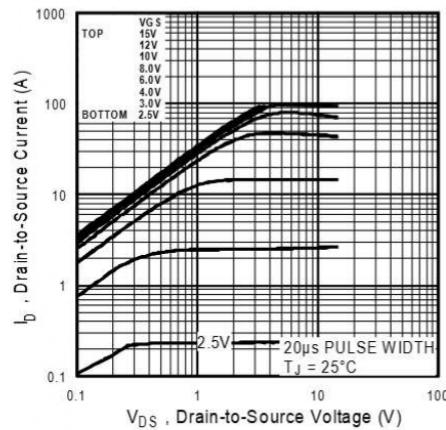


Fig 1. Typical Output Characteristics

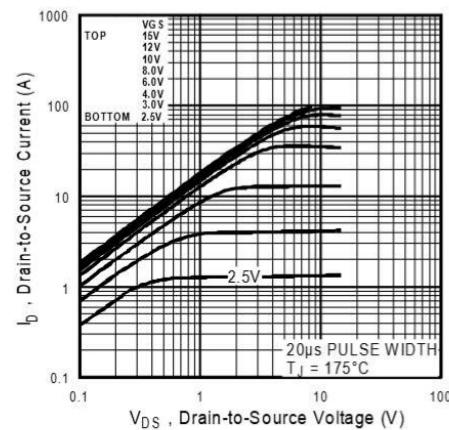


Fig 2. Typical Output Characteristics

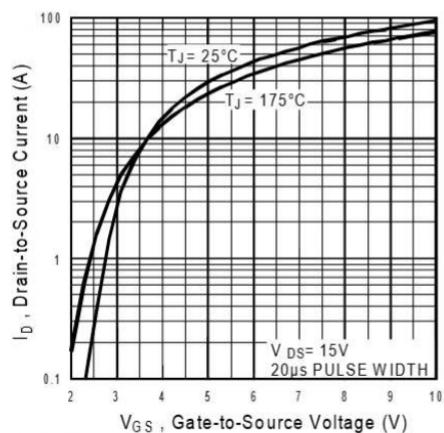


Fig 3. Typical Transfer Characteristics

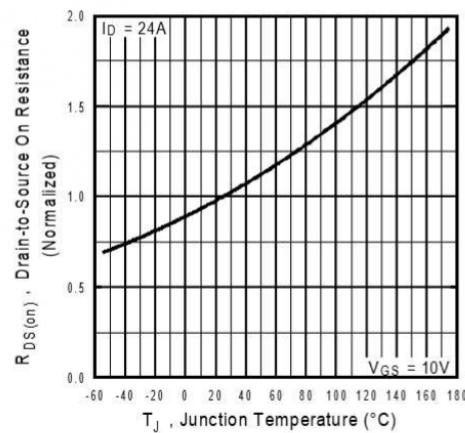


Fig 4. Normalized On-Resistance Vs. Temperature

IRL2703

International
IR Rectifier

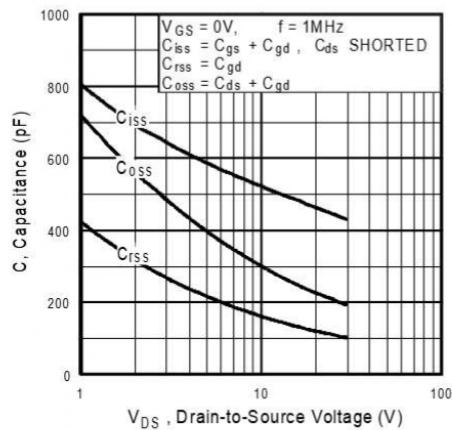


Fig 5. Typical Capacitance Vs.
Drain-to-Source Voltage

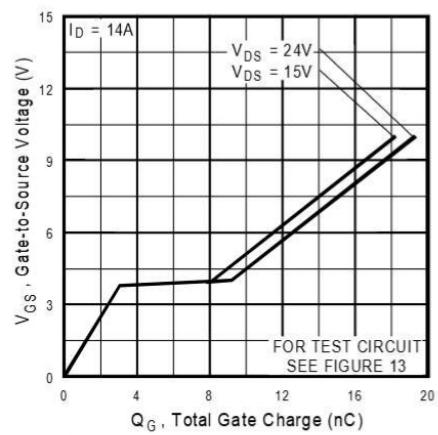


Fig 6. Typical Gate Charge Vs.
Gate-to-Source Voltage

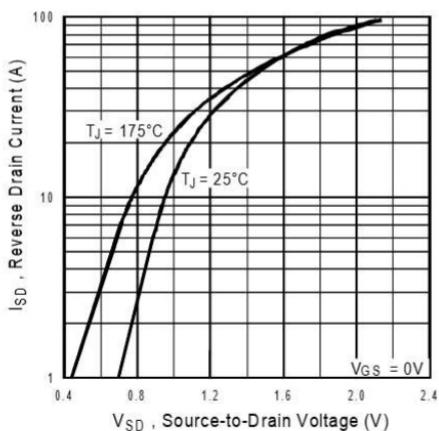


Fig 7. Typical Source-Drain Diode
Forward Voltage

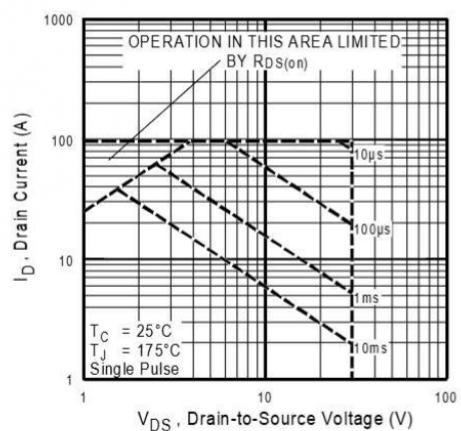


Fig 8. Maximum Safe Operating Area

IRL2703

International
IR Rectifier

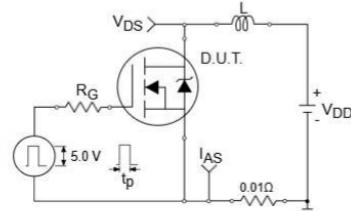


Fig 12a. Unclamped Inductive Test Circuit

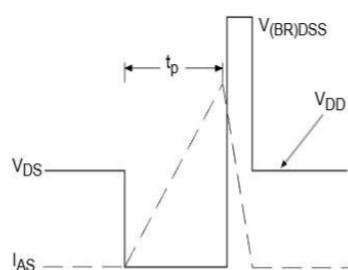


Fig 12b. Unclamped Inductive Waveforms

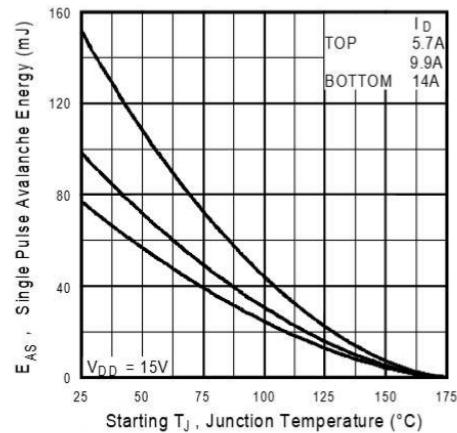


Fig 12c. Maximum Avalanche Energy Vs. Drain Current

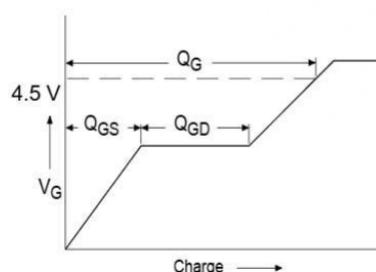


Fig 13a. Basic Gate Charge Waveform

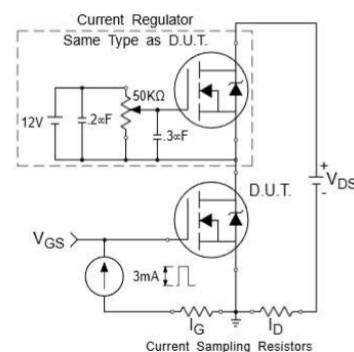
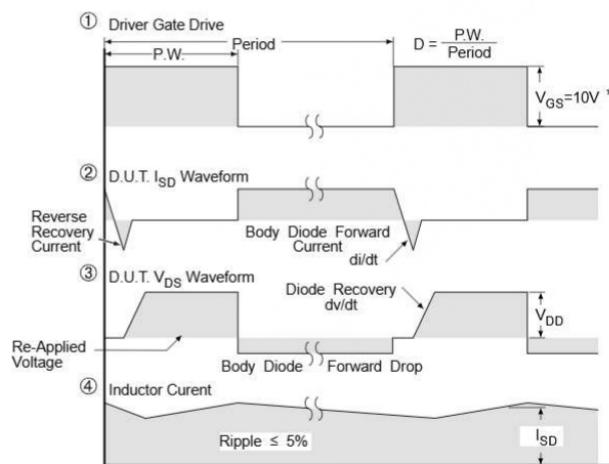
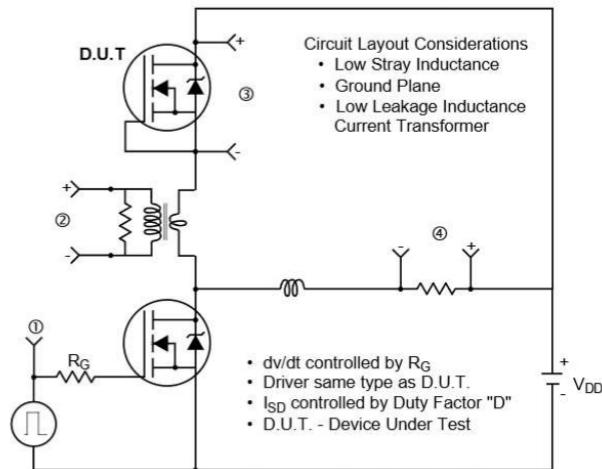


Fig 13b. Gate Charge Test Circuit

International
IR Rectifier

IRL2703

Peak Diode Recovery dv/dt Test Circuit



* $V_{GS} = 5V$ for Logic Level Devices

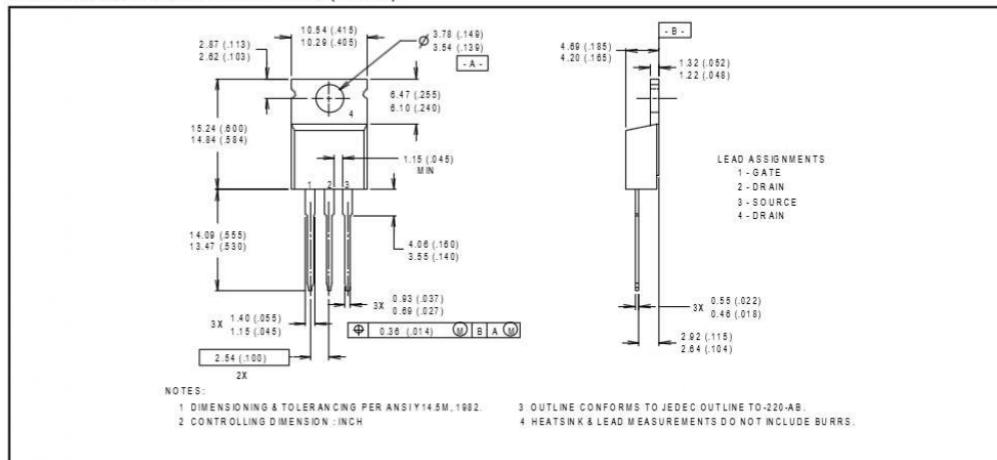
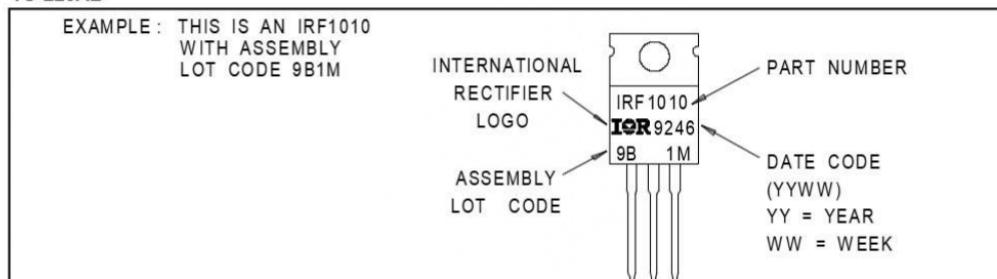
Fig 14. For N-Channel HEXFETs

IRL2703

International
IR Rectifier

Package Outline**TO-220AB Outline**

Dimensions are shown in millimeters (inches)

**Part Marking Information****TO-220AB**

International
IR Rectifier

WORLD HEADQUARTERS: 233 Kansas St., El Segundo, California 90245, Tel: (310) 322 3331

EUROPEAN HEADQUARTERS: Hurst Green, Oxted, Surrey RH8 9BB, UK Tel: ++ 44 1883 732020

IR CANADA: 7321 Victoria Park Ave., Suite 201, Markham, Ontario L3R 2Z8, Tel: (905) 475 1897

IR GERMANY: Saalburgstrasse 157, 61350 Bad Homburg Tel: ++ 49 6172 96590

IR ITALY: Via Liguria 49, 10071 Borgaro, Torino Tel: ++ 39 11 451 0111

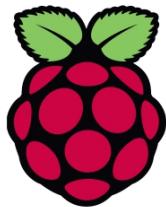
IR FAR EAST: K&H Bldg., 2F, 3-30-4 Nishi-Ikeburo 3-Chome, Toshima-Ki, Tokyo Japan 171 Tel: 81 3 3983 0086

IR SOUTHEAST ASIA: 315 Outram Road, #10-02 Tan Boon Liat Building, Singapore 0316 Tel: 65 221 8371

<http://www.irf.com/> Data and specifications subject to change without notice. 11/96

F. Hoja de datos de la Raspberry Pi 4 Model B

DATA SHEET



Raspberry Pi 4 Model B

Release 1.1

March 2024

Copyright 2024 Raspberry Pi (Trading) Ltd. All rights reserved.



2 Features

2.1 Hardware

- Quad core 64-bit ARM-Cortex A72 running at 1.5GHz
- 1, 2 and 4 Gigabyte LPDDR4 RAM options
- H.265 (HEVC) hardware decode (up to 4Kp60)
- H.264 hardware decode (up to 1080p60)
- VideoCore VI 3D Graphics
- Supports dual HDMI display output up to 4Kp60

2.2 Interfaces

- 802.11 b/g/n/ac Wireless LAN
- Bluetooth 5.0 with BLE
- 1x SD Card
- 2x micro-HDMI ports supporting dual displays up to 4Kp60 resolution
- 2x USB2 ports
- 2x USB3 ports
- 1x Gigabit Ethernet port (supports PoE with add-on PoE HAT)
- 1x Raspberry Pi camera port (2-lane MIPI CSI)
- 1x Raspberry Pi display port (2-lane MIPI DSI)
- 28x user GPIO supporting various interface options:
 - Up to 6x UART
 - Up to 6x I2C
 - Up to 5x SPI
 - 1x SDIO interface
 - 1x DPI (Parallel RGB Display)
 - 1x PCM
 - Up to 2x PWM channels
 - Up to 3x GPCLK outputs

2.3 Software

- ARMv8 Instruction Set
- Mature Linux software stack
- Actively developed and maintained



- Recent Linux kernel support
- Many drivers upstreamed
- Stable and well supported userland
- Availability of GPU functions using standard APIs

3 Mechanical Specification

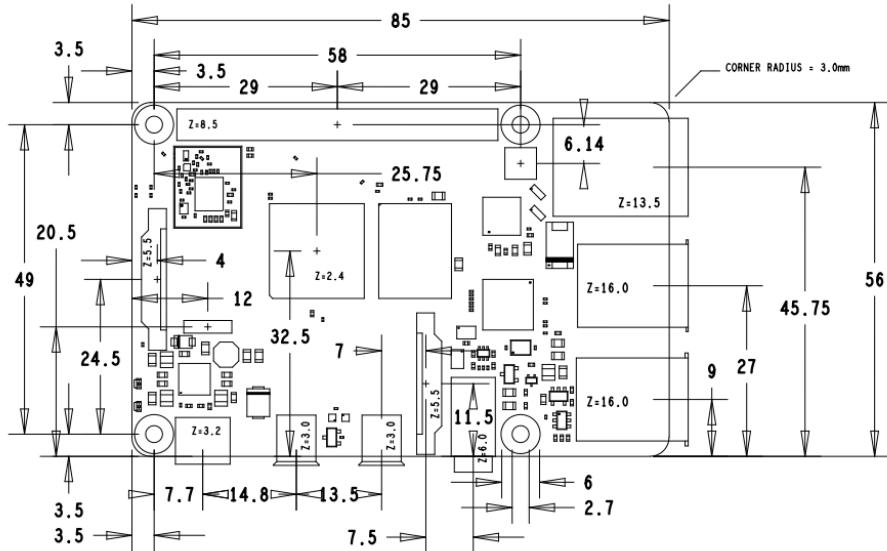


Figure 1: Mechanical Dimensions

4 Electrical Specification

Caution! Stresses above those listed in Table 2 may cause permanent damage to the device. This is a stress rating only; functional operation of the device under these or any other conditions above those listed in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Symbol	Parameter	Minimum	Maximum	Unit
VIN	5V Input Voltage	-0.5	6.0	V

Table 2: Absolute Maximum Ratings

Please note that VDD_IO is the GPIO bank voltage which is tied to the on-board 3.3V supply rail.



Symbol	Parameter	Conditions	Minimum	Typical	Maximum	Unit
V_{IL}	Input low voltage ^a	$VDD_IO = 3.3V$	0	-	0.8	V
V_{IH}	Input high voltage ^a	$VDD_IO = 3.3V$	2.0	-	VDD_IO	V
I_{IL}	Input leakage current	$TA = +85^\circ C$	-	-	10	μA
C_{IN}	Input capacitance	-	-	5	-	pF
V_{OL}	Output low voltage ^b	$VDD_IO = 3.3V, IOL = -2mA$	-	-	0.4	V
V_{OH}	Output high voltage ^b	$VDD_IO = 3.3V, IOH = 2mA$	$VDD_IO - 0.4$	-	-	V
I_{OL}	Output low current ^c	$VDD_IO = 3.3V, VO = 0.4V$	7	-	-	mA
I_{OH}	Output high current ^c	$VDD_IO = 3.3V, VO = 2.3V$	7	-	-	mA
R_{PU}	Pullup resistor	-	18	47	73	k Ω
R_{PD}	Pulldown resistor	-	18	47	73	k Ω

^a Hysteresis enabled^b Default drive strength (8mA)^c Maximum drive strength (16mA)

Table 3: DC Characteristics

Pin Name	Symbol	Parameter	Minimum	Typical	Maximum	Unit
Digital outputs	t_{rise}	10-90% rise time ^a	-	TBD	-	ns
Digital outputs	t_{fall}	90-10% fall time ^a	-	TBD	-	ns

^a Default drive strength, $CL = 5pF, VDD_IO = 3.3V$

Table 4: Digital I/O Pin AC Characteristics



Figure 2: Digital IO Characteristics

4.1 Power Requirements

The Pi4B requires a good quality USB-C power supply capable of delivering 5V at 3A. If attached downstream USB devices consume less than 500mA, a 5V, 2.5A supply may be used.



5 Peripherals

5.1 GPIO Interface

The Pi4B makes 28 BCM2711 GPIOs available via a standard Raspberry Pi 40-pin header. This header is backwards compatible with all previous Raspberry Pi boards with a 40-way header.

5.1.1 GPIO Pin Assignments

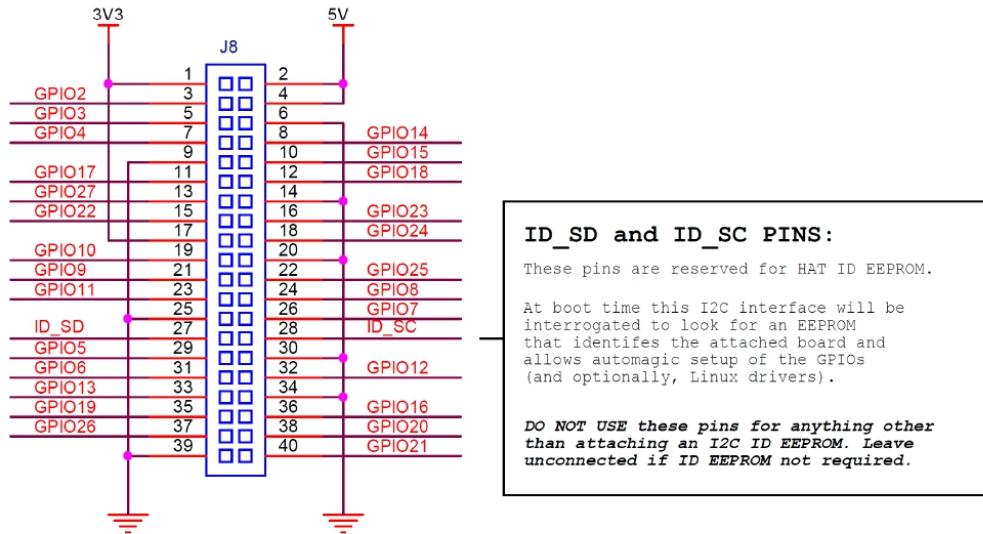


Figure 3: GPIO Connector Pinout

As well as being able to be used as straightforward software controlled input and output (with programmable pulls), GPIO pins can be switched (multiplexed) into various other modes backed by dedicated peripheral blocks such as I2C, UART and SPI.

In addition to the standard peripheral options found on legacy Pis, extra I2C, UART and SPI peripherals have been added to the BCM2711 chip and are available as further mux options on the Pi 4. This gives users much more flexibility when attaching add-on hardware as compared to older models.



5.1.2 GPIO Alternate Functions

GPIO	Default	Default					
		Pull	ALT0	ALT1	ALT2	ALT3	ALT4
0	High	SDA0	SA5	PCLK	SPI3_CE0.N	TXD2	SDA6
1	High	SCL0	SA4	DE	SPI3_MISO	RXD2	SCL6
2	High	SDA1	SA3	LCD_VSYNC	SPI3_MOSI	CTS2	SDA3
3	High	SCL1	SA2	LCD_HSYNC	SPI3_SCLK	RTS2	SCL3
4	High	GPCLK0	SA1	DPLD0	SPI4_CE0.N	TXD3	SDA3
5	High	GPCLK1	SA0	DPLD1	SPI4_MISO	RXD3	SCL3
6	High	GPCLK2	SOE_N	DPLD2	SPI4_MOSI	CTS3	SDA4
7	High	SPI0_CE1_N	SWE_N	DPLD3	SPI4_SCLK	RTS3	SCL4
8	High	SPI0_CE0_N	SD0	DPLD4	-	TXD4	SDA4
9	Low	SPI0_MISO	SD1	DPLD5	-	RXD4	SCL4
10	Low	SPI0_MOSI	SD2	DPLD6	-	CTS4	SDA5
11	Low	SPI0_SCLK	SD3	DPLD7	-	RTS4	SCL5
12	Low	PWM0	SD4	DPLD8	SPI5_CE0.N	TXD5	SDA5
13	Low	PWM1	SD5	DPLD9	SPI5_MISO	RXD5	SCL5
14	Low	TXD0	SD6	DPLD10	SPI5_MOSI	CTS5	TXD1
15	Low	RXD0	SD7	DPLD11	SPI5_SCLK	RTS5	RXD1
16	Low	FL0	SD8	DPLD12	CTS0	SPI1_CE2_N	CTS1
17	Low	FL1	SD9	DPLD13	RTS0	SPI1_CE1_N	RTS1
18	Low	PCM_CLK	SD10	DPLD14	SPI6_CE0.N	SPI1_CE0_N	PWM0
19	Low	PCM_FS	SD11	DPLD15	SPI6_MISO	SPI1_MISO	PWM1
20	Low	PCM_DIN	SD12	DPLD16	SPI6_MOSI	SPI1_MOSI	GPCLK0
21	Low	PCM_DOUT	SD13	DPLD17	SPI6_SCLK	SPI1_SCLK	GPCLK1
22	Low	SD0_CLK	SD14	DPLD18	SD1_CLK	ARM_TRST	SDA6
23	Low	SD0_CMD	SD15	DPLD19	SD1_CMD	ARM_RTCK	SCL6
24	Low	SD0_DAT0	SD16	DPLD20	SD1_DAT0	ARM_TDO	SPI3_CE1_N
25	Low	SD0_DAT1	SD17	DPLD21	SD1_DAT1	ARM_TCK	SPI4_CE1_N
26	Low	SD0_DAT2	TE0	DPLD22	SD1_DAT2	ARM_TDI	SPI5_CE1_N
27	Low	SD0_DAT3	TE1	DPLD23	SD1_DAT3	ARM_TMS	SPI6_CE1_N

Table 5: Raspberry Pi 4 GPIO Alternate Functions

Table 5 details the default pin pull state and available alternate GPIO functions. Most of these alternate peripheral functions are described in detail in the BCM2711 Peripherals Specification document which can be downloaded from the [hardware documentation](#) section of the website.



5.1.3 Display Parallel Interface (DPI)

A standard parallel RGB (DPI) interface is available the GPIOs. This up-to-24-bit parallel interface can support a secondary display.

5.1.4 SD/SDIO Interface

The Pi4B has a dedicated SD card socket which supports 1.8V, DDR50 mode (at a peak bandwidth of 50 Megabytes / sec). In addition, a legacy SDIO interface is available on the GPIO pins.

5.2 Camera and Display Interfaces

The Pi4B has 1x Raspberry Pi 2-lane MIPI CSI Camera and 1x Raspberry Pi 2-lane MIPI DSI Display connector. These connectors are backwards compatible with legacy Raspberry Pi boards, and support all of the available Raspberry Pi camera and display peripherals.

5.3 USB

The Pi4B has 2x USB2 and 2x USB3 type-A sockets. Downstream USB current is limited to approximately 1.1A in aggregate over the four sockets.

5.4 HDMI

The Pi4B has 2x micro-HDMI ports, both of which support CEC and HDMI 2.0 with resolutions up to 4Kp60.

5.5 Audio and Composite (TV Out)

The Pi4B supports near-CD-quality analogue audio output and composite TV-output via a 4-ring TRS 'A/V' jack.

The analog audio output can drive 32 Ohm headphones directly.

5.6 Temperature Range and Thermals

The recommended ambient operating temperature range is 0 to 50 degrees Celsius.

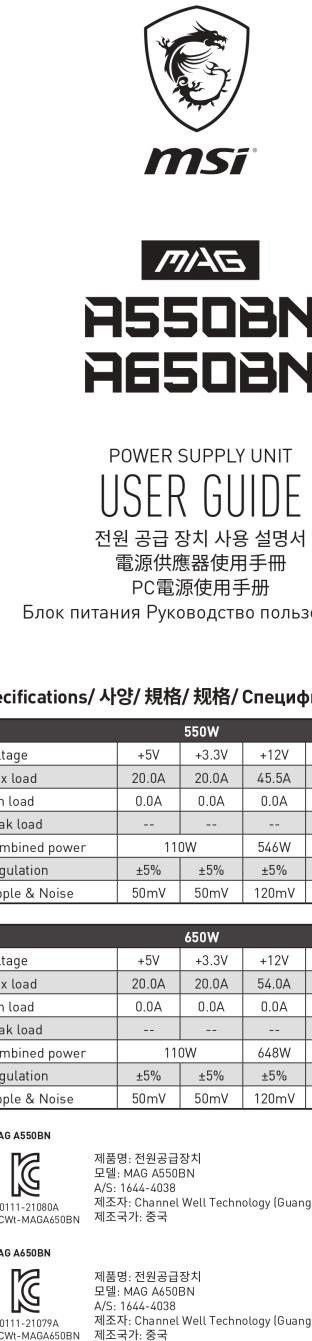
To reduce thermal output when idling or under light load, the Pi4B reduces the CPU clock speed and voltage. During heavier load the speed and voltage (and hence thermal output) are increased. The internal governor will throttle back both the CPU speed and voltage to make sure the CPU temperature never exceeds 85 degrees C.

The Pi4B will operate perfectly well without any extra cooling and is designed for sprint performance - expecting a light use case on average and ramping up the CPU speed when needed (e.g. when loading a webpage). If a user wishes to load the system continually or operate it at a high temperature at full performance, further cooling may be needed.

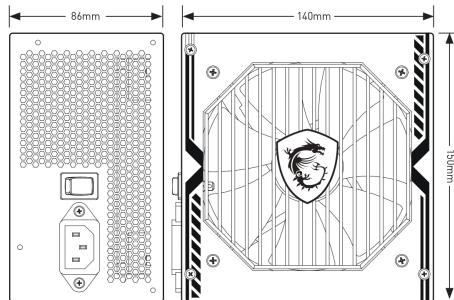
6 Availability

Raspberry Pi guarantees availability of the Pi4B until at least January 2031.

G. Hoja de datos de la Fuente de Poder MSI MAG A650BN



Dimensions/ 치수/ 外型尺寸/ Размеры



Number of Power Connectors/ 전원 커넥터 수량/

電源接頭數量/ 电源接头数量/

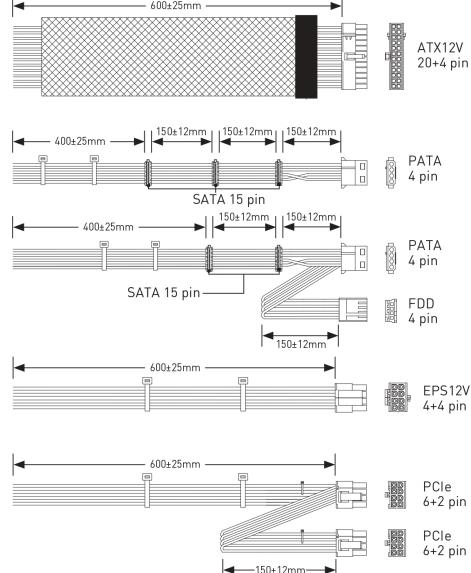
Кол-во силовых соединителей

Power Connectors	A550BN	A650BN
ATX12V (Motherboard) 20+4 pin	1	1
EPS12V (CPU) 4+4 pin	1	1
PCIe [VGA] 6+2 pin	2	2
SATA 15 pin	5	5
Peripheral (PATA) 4 pin	2	2
FDD 4 pin	1	1

Connectivity and Cable Lengths/ 연결 및 케이블 길이/

連接頭和電纜長度/ 连接性和电缆长度/

Подключение и длина кабелей



Referencias

- [1] Library of Congress. «Library of Congress.» (Agosto de 2016), dirección: <https://www.loc.gov/collections/edison-company-motion-pictures-and-sound-recordings/articles-and-essays/history-of-edison-sound-recordings/history-of-the-cylinder-phonograph/>. [Accessed: 2023-05-11].
- [2] Sheffielddav, *Sheffielddav*. dirección: <https://www.sheffielddav.com/education/which-type-sound-engineer-are-you-destined-be>, [Accessed: 2023-05-11].
- [3] E-Home Recording Studio, *Acoustic treatment 101: The Ultimate Setup/Planning Guide*. dirección: <https://ehomerecordingstudio.com/acoustic-treatment/>, [Accessed: 2023-09-14].
- [4] J. Albano, *Studio Acoustic Part 2: Standing Waves*. ASK.AUDIO.
- [5] M. Ervine, *Acoustic treatment - A complete guide*. dirección: https://www.acoustiblok.co.uk/acoustic-treatment-a-complete-guide/#How_to_Acoustically_Treat_Your_Room, [Accessed: 2023-09-14].
- [6] M. Irwin, *Where to install acoustic panels: Resonics acoustics*. dirección: <https://resonics.co.uk/where-to-install-acoustic-panels/>, [Accessed: 2023-09-14].
- [7] B. Pack, *Tips for repairing Bad Recordings*. dirección: <https://www.masteringthemix.com/blogs/learn/tips-for-repairing-bad-recordings>, [Accessed: 2023-09-14].
- [8] E. Winer, *The audio expert*, 2nd. Routledge, 2017, ISBN: 9781351840064.
- [9] W. Weir. «How Humans Conquered Echo.» (2012), dirección: <https://www.theatlantic.com/entertainment/archive/2012/06/how-humans-conquered-echo/258557/>. [Accessed: 2023-09-21].
- [10] W. Weir. «Studio Workflow & Why it Matters.» (2022), dirección: <https://www.studioassistant.io/blog/studio-workflow-why-it-matters>. [Accessed: 2023-09-21].
- [11] García, O. J. Mantilla, «Diseño y construcción de un prototipo de panel acústico variable rotatorio de 360º con control automático.,» P. A. V. R., Colombia, inf. téc., 2014.
- [12] L. Sen y M. Peng, *Acoustic Enclosure with Intelligent Controllable Noise*. dirección: <https://nvc.sjtu.edu.cn/EN/10.3969/j.issn.1006-1355-2011.04.042>, [Accessed: 2023-09].
- [13] Y. Lei, J. Pan y M. P. Sheng, «Investigation of structural response and noise reduction of an acoustical enclosure using SEA method,» *Applied Acoustics*, vol. 73, n.º 4, 2012.
- [14] G. Thün, K. Velikov, C. Ripley, L. Sauvé y W. McGee, «Soundsphere: resonant chamber,» *ResearchGate*, vol. 45, n.º 4, págs. 348-357, 2012.
- [15] A. C. Isbert, *Diseño acústico de espacios arquitectónicos*, 1st. Edicions UPC, 1998.
- [16] F. Miyara, *Acústica y sistemas de sonido*, 3rd. Rosario: UNR Editora, 2004.

- [17] T. D. Rossing, *Handbook of Acoustics*, 1st. New York: Springer, 2007.
- [18] J. Geusemeir y S. Moehringer, «VDI 2206- A New Guideline for the Design of Mechatronic Systems,» *IFAC Proceedings Volumes*, vol. 35, n.^o 2, págs. 785-790, 2002.
- [19] G. Pahl y W. Beitz, *Engineering design*. London: Springer, 1996.
- [20] I. S. Institution, «ISO3382,» *Acoustics - Measurement of Room Acoustic Parameters*, págs. 1-26, 2009.
- [21] A. N. Standard, «ISO3382,» *Specifications for octave-band and fractional-octave-band analog and digital filters*, págs. 1-31, 2004.
- [22] R. E. Wizard. «Filtered IR Graph.» (2024), dirección: https://www.roomeqwizard.com/help/help_en-GB/html/graph_filteredir.html. [Accessed: 2023-09-21].
- [23] Dirección: <https://svantek.com/academy/rt60-reverberation-time/#:~:text=What%20is%20a%20decay%20curve, the%20sound%20source%20has%20stopped.>, [Accessed: 2024-06-06].
- [24] T. Cox, P. D'Antonio y M. Avis, «Room sizing and optimization at low frequencies,» *Journal of the Audio Engineering Society*, n.^o 52, págs. 640-651, 2004.
- [25] M. Schoroeder, «The Schoroeder Frequency Revisited,» *Journal of the Acoustical Society of America*, n.^o 99, págs. 3240-3241, 1996.
- [26] M. Long, *Architectural Acoustics*. Academic Press, 2014.
- [27] D. Schroder, *Physically Based Real-Time Auralization of Interactive Virtual Environments*. Reason, 2011.
- [28] MathWorks. «Room Impulse Response Simulation with Stochastic Ray Tracing.» (2024), dirección: <https://www.mathworks.com/help/audio/ug/room-impulse-response-simulation-with-stochastic-ray-tracing.html#RoomImpulseWithRayTracingExam13>. [Accessed: 2023-09-21].
- [29] J. Holger, «Scattering in Room Acoustics and Related Activities in ISO and AES,» 2001.
- [30] J. Kingman, *Poisson Processes*. Clarendon Press, 1992.
- [31] T. M. Audio. «Audio Spectrum Explained.» (2020), dirección: <https://www.teachmeaudio.com/mixing/techniques/audio-spectrum#:~:text=Midrange%3A%20500%20Hz%20to%202%20kHz&text=The%20midrange%20determines%20how%20prominent,instruments%20a%20horn%2Dlike%20quality..> [Accessed: 2023-09-21].
- [32] B. Curless, «Parametric Curves,» CSE 557, inf. téc., 2017. dirección: <https://courses.cs.washington.edu/courses/cse557/17au/assets/lectures/param-curves-1pp.pdf>, [Accessed: 2023-09-21].
- [33] A. Luviano, «Curvas de Bezier para interpolacion de trayectorias,» IPN - UPIITA, inf. téc., 2022.
- [34] A. Luviano, «Leyes de control - Control de Sistemas Roboticos,» IPN - UPIITA, inf. téc., 2022.

- [35] J. Llata, E. Gonzalez, D. Fernandez y et al., «Capitulo 4. Respuesta del regimen transitorio,» Universidad de Cantabria, inf. téc., 2022. dirección: https://ocw.unican.es/pluginfile.php/1829/course/section/1439/ejercicios_4.pdf.
- [36] MQTT. «MQTT: The Standard for IoT Messaging.» (2024), dirección: <https://mqtt.org/> (visitado 11-06-2024).
- [37] CoAP. «Constrained Application Protocol (CoAP).» (2024), dirección: <https://coap.space/> (visitado 11-06-2024).
- [38] WebSockets. «The WebSockets API.» (2024), dirección: https://developer.mozilla.org/en-US/docs/Web/API/WebSockets_API (visitado 11-06-2024).
- [39] gRPC. «A high performance, open source universal RPC framework.» (2024), dirección: <https://grpc.io/> (visitado 11-06-2024).
- [40] M. D. Network. «An Overview of HTTP.» (2024), dirección: <https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview> (visitado 11-06-2024).
- [41] M. Fazio, R. Ranjan, M. Girolami, J. Taheri, S. Dustdar y M. Villari, «A Note on the Convergence of IoT, Edge, and Cloud Computing in Smart Cities,» *IEEE*, vol. 5, n.º 5, págs. 22-24, 2018.
- [42] A. Rahim y R. Giaffreda, «IoT and cloud convergence: Opportunities and challenges,» *IEEE*, vol. 1, 2014.
- [43] A. W. Services. «Amazon Web Services (AWS).» (2024), dirección: <https://aws.amazon.com/> (visitado 11-06-2024).
- [44] G. Cloud. «Google Cloud Platform.» (2024), dirección: <https://cloud.google.com/> (visitado 11-06-2024).
- [45] M. Azure. «Microsoft Azure.» (2024), dirección: <https://azure.microsoft.com/> (visitado 11-06-2024).
- [46] I. Cloud. «IBM Cloud.» (2024), dirección: <https://www.ibm.com/cloud> (visitado 11-06-2024).
- [47] O. Cloud. «Oracle Cloud.» (2024), dirección: <https://www.oracle.com/cloud/> (visitado 11-06-2024).
- [48] DigitalOcean. «DigitalOcean – The developer cloud.» (2024), dirección: <https://www.digitalocean.com/> (visitado 11-06-2024).
- [49] CompareCloud. «Compare Cloud Services.» (2024), dirección: <https://comparecloud.in/> (visitado 11-06-2024).
- [50] T. Capris, P. Melo, I. M. Pires y N. Garcia, «Comparison of SQL and NoSQL databases with different workloads: MongoDB vs MySQL evaluation,» *ICDABI*, 2023.
- [51] G. Cloud. «Cloud Bigtable.» (2024), dirección: <https://cloud.google.com/bigtable> (visitado 11-06-2024).
- [52] G. Cloud. «Cloud Firestore.» (2024), dirección: <https://cloud.google.com/firestore> (visitado 11-06-2024).

- [53] G. Cloud. «Cloud Memorystore.» (2024), dirección: <https://cloud.google.com/memorystore> (visitado 11-06-2024).
- [54] MongoDB. «MongoDB.» (2024), dirección: <https://www.mongodb.com/> (visitado 11-06-2024).
- [55] S. Vishesh, «Back-End Web-Application Development and the Role of an Admin,» *IJARCCE*, 2017.
- [56] Django. «Django: The web framework for perfectionists with deadlines.» (2024), dirección: <https://www.djangoproject.com/> (visitado 11-06-2024).
- [57] P. Projects. «Flask: The Python micro framework for building web applications.» (2024), dirección: <https://flask.palletsprojects.com/> (visitado 11-06-2024).
- [58] FastAPI. «FastAPI: A modern, fast (high-performance), web framework for building APIs with Python 3.6+.» (2024), dirección: <https://fastapi.tiangolo.com/> (visitado 11-06-2024).
- [59] Spring. «Spring Boot: Makes it easy to create stand-alone, production-grade Spring based Applications.» (2024), dirección: <https://spring.io/projects/spring-boot> (visitado 11-06-2024).
- [60] MathWorks. «Audio Toolbox - MATLAB.» (2024), dirección: <https://www.mathworks.com/products/audio.html>. [Accessed: 2023-12-15].
- [61] ASIO4ALL. «Audio Toolbox - MATLAB.» (2023), dirección: <https://asio4all.org/>. [Accessed: 2023-12-15].
- [62] S. Guy-Bart, J. Embrechets y D. Archembau, «Comparison of Different Impulse Response Measurement Techniques,» University of Liege, Institut Montefiore, inf. téc., 2002.
- [63] Dirección: <https://dsp.stackexchange.com/questions/22111/what-does-it-mean-to-deconvolve-the-impulse-response>, [Accessed: 2024-06-06].
- [64] Acustico. «Sound Absorbing Panels.» (2024), dirección: <https://www.acustico.com/en/acoustics-insights/sound-absorbing-panels.html>. [Accessed: 2024-06-10].
- [65] pyroomacoustics. «Materials Database - pyroomacoustics.» (2016), dirección: <https://pyroomacoustics.readthedocs.io/en/pypi-release/pyroomacoustics.materials.database.html>. [Accessed: 2024-06-10].
- [66] R. Gate. «Why do porous materials only absorb high frequency noises and not low frequency noises?» (2013), dirección: https://www.researchgate.net/post/Why_do_porous_materials_only_absorb_high_frequency_noises_and_not_low_frequency_noises. [Accessed: 2024-06-10].
- [67] A. Pilch y T. Kamisinski, «The effect of Geometrical and Material Modification of Sound Diffusers on Their Acoustic Parameters,» AGH University of Science and Technology Departament of Mechannics and Vibroacoustics, inf. téc., 2011.
- [68] Dirección: <https://acousticgeometry.com/types-of-sound-diffusers/>, [Accessed: 2024-06-06].

- [69] R. Budynas y J. Keith, *Diseño en ingeniería mecánica de Shigley*, 9th. España: McGraw.Hill, 2012.
- [70] Motor de cc de 300W, 12V, 24V, regulador de velocidad de alta velocidad, 1800rpm-3000rpm, motor de gran par delantero y inverso. dirección: https://es.aliexpress.com/item/32805073241.html?srcSns=sns_WhatsApp&spreadType=socialShare&bizType=ProductDetail&social_params=21509958417&aff_fcid=45687d1d7ede4236afeb112e31718152919883-08617-_msThjPo&tt=MG&aff_fsk=_msThjPo&aff_platform=default&sk=_msThjPo&aff_trace_key=45687d1d7ede4236afeb112e341c0bc8-1718152919883-08617-_msThjPo&shareId=21509958417&businessType=ProductDetail&platform=AE&terminal_id=5b77b12be99b421c8e41b9f1703208ef&afSmartRedirect=y, [Accessed: 2024-06-10].
- [71] S. Schmid, B. Hamrock y B. Jacobson, *Fundamentals of Machine Elements*, 3rd. USA: CRC Press, 2014.
- [72] R. Norton, *Diseño de máquinas un enfoque integrado*, 4th. México: PEARSON, 2011.
- [73] Desarrollo de Tablas de las características estructurales de perfiles LM. dirección: https://perfileslm.com/assets/tablas_it esm.pdf, [Accessed: 2024-06-13].
- [74] Justinmind. «Neumorphism: why it's all the hype in UI design.» (2024), dirección: <https://www.justinmind.com/ui-design/neumorphism> (visitado 11-06-2024).
- [75] L. Davis. «Reverberation time measurements.» (2024), dirección: <https://www.larsondavis.com/learn/building-acoustics/Reverberation-Time-in-Room-Acoustics>. [Accessed: 2024-06-10].
- [76] «ACOUSTICS IN THE MULTIPURPOSE HALLS OF THE NEW MAIN LIBRARY AND THE NEW MUNCH MUSEUM IN OSLO,» 2015.
- [77] T. Berners-Lee y M. Fischetti, «Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web,» *MIT Press*, vol. 1, n.º 1, págs. 23-45, 1989.
- [78] Electron. «Electron — Build cross-platform desktop apps with JavaScript, HTML, and CSS.» (), dirección: <https://www.electronjs.org> (visitado 11-06-2024).
- [79] React. «React – A JavaScript library for building user interfaces.» (2024), dirección: <https://reactjs.org> (visitado 11-06-2024).
- [80] Redux. «Redux – A Predictable State Container for JS Apps.» (2024), dirección: <https://redux.js.org> (visitado 11-06-2024).
- [81] S. Components. «Styled Components – Visual primitives for the component age.» (2024), dirección: <https://styled-components.com> (visitado 11-06-2024).
- [82] D3.js. «D3.js – Data-Driven Documents.» (2024), dirección: <https://d3js.org> (visitado 11-06-2024).
- [83] I. Jacobs y N. Walsh, «Architecture of the World Wide Web, Volume One,» World Wide Web Consortium (W3C), W3C Recommendation, 2004. dirección: <https://www.w3.org/TR/wearch/>.
- [84] B. A. Myers, «Challenges of HCI design and implementation,» *interactions*, vol. 2, n.º 1, págs. 73-83, 1995.

- [85] AliExpress, *Marco de empuje y tracción, electroimán de 24V, carrera de TAU-0630T, 0-10mm, fuerza de Keeping 0-10N, 12V, electroimán solenoide.* dirección: [\[87\] R. Boylestad y L. Nashelsky, *Electrónica: Teoría de Circuitos y Dispositivos Electrónicos*, 10th. Pearson educación, 2009.

\[88\] A. C. E. Industries, «IPC-2221B Generic Standard on Printed Board Design,» 2012.

\[89\] Dirección: \[\\[91\\] *Driver de velocidad del motor de CC, módulo de control del driver del motor de cepillo para polvo, DC 9V-60V, 12V, 24V, 36V, 48V, 60V, regulador modulador de ancho de pulso del motor, 20A, 1200W, PWM.* dirección: \\[Ing. Mecatrónica\\]\\(https://www.amazon.com/dp/B07P6NWJJ3/?coliid=IAH9N7VRJOHQ1&colid=1P79KIOZTQ9X9&psc=1&ref_=list_c_wl_lv_ov_lig_dp_it_im, \\[Accessed: 2024-06-10\\].</p></div><div data-bbox=\\)\]\(https://es.aliexpress.com/item/1005005268257994.html?spm=a2g0o.productlist.main.37.4a437cb0IGbEkN&algo_pvid=bdde2cfb-9297-4e3b-a597-f7835e13d955&aem_p4p_detail=2024061017122330625399467400011289859&algo_exp_id=bdde2cfb-9297-4e3b-a597-f7835e13d955-18&pdp_npi=4%40dis%21MXN%21680.01%21591.66%21%21%2136.25%2131.54%21%40210318c317180647432526469e3070%2112000038553207163%21sea%21MX%210%21AB&curPageLogUid=Nktilyxy1AWbb&utparam-url=scene%3Asearch%7Cquery_from%3A&search_p4p_id=20240610171223306253994677, \[Accessed: 2024-06-10\].</p><p>\[90\] <i>Convertidor De Voltaje Dc-dc De 12v A 24v 20a 480w.</i> dirección: <a href=\)](https://es.aliexpress.com/item/32733032074.html?spm=a2g0o.productlist.main.1.53d01396J543RP&algo_pvid=347225d3-4aa9-4db5-a26f-f7a94a05a17e&algo_exp_id=347225d3-4aa9-4db5-a26f-f7a94a05a17e-0&pdp_npi=4%40dis%21MXN%2135.31%2121.22%21%212.03%211.22%21%40210330dd17172872105598305ec7b6%2112000015918050529%21sea%21MX%210%21AB&curPageLogUid=YSuIUCJTkfYA&utparam-url=scene%3Asearch%7Cquery_from%3A, [Accessed: 2024-06-10].</p><p>[86] AliExpress, <i>Solenoide lineal tipo Push Pull, interruptor de 12V, marco abierto, electroimán, fuerza de Keeping 5N-65N, imanes de viaje de 5mm - 20mm.</i> dirección: <a href=)

- [92] DS1804-100+. dirección: <https://www.digikey.com/es/products/detail/analog-devices-inc-maxim-integrated/DS1804-100/1087309>, [Accessed: 2024-06-10].
- [93] A. Technologies. «¿Qué es un encoder y qué tipos existen?» (2024), dirección: <https://www.tca-automation.com/que-es-un-encoder-y-que-tipos-existen/>. [Accessed: 2024-06-10].
- [94] *Codificador rotativo Incremental fotoeléctrico 38S6G5-B-G24N 50/100/200/360/400/500/600PPR AB, codificador óptico de 2 fases DC8-24V NPN PNP.* dirección: [https://www.profesionalreview.com/2017/11/06/certificacion-80-plus-que-es-como-funciona/#Como_se_obtiene_la_Certificacion_80_Plus](https://es.aliexpress.com/item/1005005779039462.html?src=google&src=google&albch=shopping&acnt=631-313-3945&slnk=&plac=&mtctp=&albtt=Google_7_shopping&gcls=aw.ds&albagn=888888&isSmbAutoCall=false&needSmbHouyi=false&src=google&albch=shopping&acnt=631-313-3945&slnk=&plac=&mtctp=&albtt=Google_7_shopping&gcls=aw.ds&albagn=888888&ds_e_adid=&ds_e_matchtype=&ds_e_device=c&ds_e_network=x&ds_e_product_group_id=&ds_e_product_id=es1005005779039462&ds_e_product_merchant_id=105642917&ds_e_product_country=MX&ds_e_product_language=es&ds_e_product_channel=online&ds_e_product_store_id=&ds_url_v=2&albcn=19979986703&albag=&isSmbAutoCall=false&needSmbHouyi=false&gad_source=1&gclid=CjwKCAjw65-zBhBkEiwAjrqRMGKL6WMalU-NP9Atc_N_3yMk9jbHG9rYCRn1FpL-ii90j4BD9AzObhoC-ZwQAvD_BwE&aff_fcid=65ecdeba421449939ebbd66befa855-1718093670222-03912-UneMJZVf&aff_fsk=UneMJZVf&aff_platform=aaf&sk=UneMJZVf&aff_trace_key=65ecdeba421449939ebbd66befa855-1718093670222-03912-UneMJZVf&terminal_id=ffa13731159b436f8c27ed2e8269c93c&afSmartRedirect=y, [Accessed: 2024-06-10].</p><p>[95] M. Á. Navas. «Certificación 80 PLUS ¿Qué es? ¿Cómo funciona?» (2017), dirección: <a href=). [Accessed: 2023-06-13].
- [96] CyberPuerta, *Fuente de Poder MSI MAG A650BN 80 PLUS Bronze, 20+4 pin ATX, 120mm, 650W.* dirección: <https://www.cyberpuerta.mx/Computo-Hardware/Componentes/Fuentes-de-Poder-para-PC-s/Fuente-de-Poder-MSI-MAG-A650BN-80-PLUS-Bronze-20-4-pin-ATX-120mm-650W.html>, [Accessed: 2024-06-13].
- [97] pepeenergy, *¿QUÉ ES EL CONSUMO ELÉCTRICO?* Dirección: <https://www.pepeenergy.com/blog/glosario/definicion-consumo-electrico/>, [Accessed: 2024-06-13].
- [98] Melfosur Soluciones Energéticas, *¿Qué son y para qué sirven los medidores de consumo eléctrico?* 2023. dirección: <https://www.melfosur.es/para-que-sirven-los-medidores-de-consumo-electrico/>, [Accessed: 2024-06-13].
- [99] K. Viteri, *Monitoreo de energía eléctrica: ¿por qué es importante?* Dirección: <https://tractian.com/es/blog/monitoreo-de-energia-electrica-por-que-es-importante>, [Accessed: 2024-06-13].

- [100] M. Libre, *Medidor Consumo Electrico 6 En 1 Fp Vac Iac Watts Kwh Frecue.* dirección: https://articulo.mercadolibre.com.mx/MLM-2463817406-medidor-consumo-electrico-6-en-1-fp-vac-iac-watts-kwh-frecue-_JM, [Accessed: 2024-06-13].