



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

**DESARROLLO DE UNA ESTACIÓN
ELECTRÓNICA PARA LA TRANSMISIÓN
INALÁMBRICA DE POTENCIA
ELÉCTRICA Y OBTENCIÓN DE DATOS,
APLICADA A DISPOSITIVOS BIOMEMS
IMPLANTABLES**

TESIS

que para obtener el título de

INGENIERO EN ELÉCTRICA ELECTRÓNICA

P R E S E N T A

GARCÍA LEAL JOSÉ LUIS

DIRECTOR DE TESIS:

DR. OSCAR PILLONI CHOREÑO



Ciudad Universitaria, Cd. Mx., 2024

A mi familia ... Elena, Angélica y Saúl.

Agradecimientos

Agradecimientos personales

Desarrollar este proyecto ha sido una de las experiencias más enriquecedoras y hermosas de mi vida (aunque estuve cerca de no poder concluirlo). Pienso que no son suficientes las siguientes palabras como un gesto de agradecimiento, pero me gustaría expresar mi gratitud a todas las personas que me apoyaron y que me alentaron para dar lo mejor de mí y culminar este trabajo de tesis.

Agradezco a mi asesor de tesis, el Dr. Oscar Pilloni Choreño, por permitirme formar parte de su equipo de trabajo e investigación. Reconozco y le doy las gracias por el esfuerzo, la orientación y el tiempo que me ha dedicado a lo largo de estos meses que he trabajado con él. Con su apoyo invaluable y siendo un excelente mentor, he aprendido una cantidad enorme de cosas dentro del ámbito académico, intelectual y social; como proponer proyectos y desarrollarlos al margen de los objetivos, trabajar en equipo, desarrollar una forma de pensar y razonar con un pensamiento crítico, entre muchas otras cosas.

Agradezco a mi amada familia por su apoyo incondicional: A mi madre Elena, a quien amo y admiro. Le agradezco profundamente por su amor incondicional, su sacrificio y la educación que me ha dado (aunque a veces parezca que no lo he aprovechado). A mi hermana Angélica, que ha sido un soporte y una fuente de inspiración y me ha expresado su bondad y cariño.

AGRADECIMIENTOS

A mi hermano Saúl, que entre risas, hemos compartido magníficos momentos y me ha enseñado el valor de la perseverancia. Y a nuestra gatita Silvie, que aunque ya no pueda escucharla, me regaló una cálida compañía en esta y otras etapas de mi vida.

Al equipo multidisciplinario de investigación y desarrollo de BioMEMS del Instituto de Ingeniería y de la Facultad de Ingeniería, en especial a mi amiga Frida por su convivencia y por su apoyo en los momentos que lo he necesitado.

A mi amigo Luis Donaldo que siempre se acuerda de mí y continúa alentándome en seguir con mi profesión.

Y finalmente, a todas las personas que han colaborado para que siga formándome como profesional, dándome un empujón de motivación, escuchándome, asistiéndome en mis necesidades y favoreciendo mi desarrollo personal y profesional.

Agradecimientos institucionales

Atentamente agradezco a:

Al programa UNAM-PAPIIT número TA100321 cuyo financiamiento me ha permitido realizar esta tesis.

Al laboratorio de Micro-Sistemas BioMEMS y Lab on a Chip BioMEMS de la Facultad de Ingeniería de la UNAM, al laboratorio de BioMEMS y al cubículo de Electrónica del Instituto de Ingeniería de la UNAM, por haber proporcionado las instalaciones y equipo necesario para llevar a cabo este trabajo.

A la sala de juntas de electromecánica por brindar el espacio para la preparación de la defensa oral de esta tesis.

Al Dr. Oscar Pilloni Choreño del Instituto de Ingeniería de la UNAM, por facilitar las instalaciones para el diseño y fabricación de tarjetas electrónicas, y por habilitar del cubículo de Electrónica para realizar las pruebas y obtener los resultados del proyecto.

A la Dra. Laura Adriana Oropeza Ramos de la Facultad de Ingeniería de la UNAM, por facilitar el acceso al laboratorio de Micro-Sistemas BioMEMS y Lab on a Chip BioMEMS.

Resumen

Los dispositivos implantables ofrecen alternativas favorables en el campo de la medicina para mejorar el cuidado de la salud, debido a las funcionalidades que pueden llevar a cabo en beneficio de un paciente, como el monitoreo de signos vitales y la rehabilitación de órganos, dando como resultado una forma de coadyuvar en la prevención y tratamiento de enfermedades y patologías.

A menudo, estos dispositivos de implantación médica necesitan energía eléctrica para permanecer activos y puedan realizar acciones, según su aplicación, como la dosificación de fármacos, la rehabilitación y monitoreo de la actividad fisiológica de órganos. Por lo que muchos de estos dispositivos también necesitan una interacción con sistemas externos para recibir instrucciones para su funcionamiento o para compartir información relevante procedentes de su actividad.

La implementación de una Estación Electrónica como un sistema externo para brindar un suministro de potencia eléctrica y al mismo tiempo habilitar una comunicación para el intercambio de datos biológicos e instrucciones pueden ser una alternativa atractiva para el desarrollo, implementación y avance en la tecnología de dispositivos médicos implantables de monitoreo fisiológico.

En este trabajo se exploró el desarrollo de una Estación Electrónica como un complemento para prototipos de dispositivos BioMEMS que se han estado desarrollado en el Instituto de Ingeniería de la UNAM.

Esta Estación electrónica está enfocada en integrar en un sólo sistema, dos funcionalidades inalámbricas principales: la obtención de datos fisiológicos procedentes de los prototipos de dispositivos BioMEMS y el suministro inalámbrico de potencia eléctrica hacia éstos. En el diseño, desarrollo y pruebas de este sistema se hizo énfasis en las condiciones de una posible implantación futura de estos dispositivos BioMEMS, con el fin de implementar una Estación Electrónica con miras de una aplicación médica.

Índice general

Agradecimientos	I
Resumen	IV
Índice de figuras	x
Índice de cuadros	XIII
Introducción	14
Motivación	14
Hipótesis	15
Justificación	15
Objetivo	16
1. Marco Teórico	17
1.1. Dispositivos Médicos Implantables (IMDs)	17
1.1.1. Descripción general	17
1.1.2. Perspectivas generales de los BioMEMS aplicados a tecnologías de im- plantación	19
1.1.3. Clasificación de los IMDs por su principio de funcionamiento	20

1.2. Suministros de potencia eléctrica a Dispositivos Médicos Implantables Activos (AIMDs)	22
1.2.1. Transferencia Percutánea	22
1.2.2. Cosecha de energía	23
1.2.3. Baterías y Capacitores	24
1.2.4. Transferencia inalámbrica	28
1.3. Estrategias de transferencia de potencia eléctrica inalámbrica	30
1.3.1. Sistemas de acoplos no electromagnéticos	31
1.3.2. Sistemas de acoplos por ondas electromagnéticas	35
1.3.3. Amplificadores de potencia para sistemas WPT	46
1.4. Comunicación para AIMDs	50
1.4.1. Tipos de comunicación	50
1.4.2. Comunicación inalámbrica	51
1.4.3. Transferencia de datos a alta frecuencia (HF)	52
1.4.4. Transmisión de datos por muy alta frecuencia (VHF) y ultra alta frecuencia (UHF)	54
1.4.5. Comunicación para un AIMD mediante Bluetooth	55
2. Diseño y construcción de la Estación Electrónica (EEPICI)	57
2.1. Requerimientos y especificaciones	58
2.2. Diseño del sistema de Transferencia de Potencia Inalámbrica	61
2.2.1. Diseño del acoplo inductivo	64
2.2.2. Diseño del circuito resonante	68
2.2.3. Diseño del rectificador-regulador de voltaje para el prototipo	69
2.2.4. Diseño de la etapa de potencia	70
2.2.5. Límites de potencia del amplificador debido a la absorción de energía por los tejidos de la piel	73
2.3. Diseño del sistema de Comunicación y de Control	74

ÍNDICE GENERAL

2.3.1. Diseño de la Comunicación Inalámbrica	74
2.3.2. Propuesta del sistema de Control	77
2.4. Diseño de alimentación eléctrica para la Estación	78
2.5. Fabricación del prototipo de EEPICI	80
2.5.1. Fabricación de los inductores L_1 y L_2 y las tarjetas electrónicas	80
2.5.2. Desarrollo de software de control	91
2.5.3. Desarrollo del firmware del prototipo de dispositivo BioMEMS	92
3. Prueba de concepto de EEPICI	95
3.1. Diseño experimental	95
3.1.1. Transferencia de potencia inalámbrica	95
3.1.2. Comunicación inalámbrica para la obtención de datos	97
3.2. Resultados	98
3.2.1. Transferencia de potencia inalámbrica	98
3.2.2. Comunicación inalámbrica	102
3.3. Discusión	104
3.4. Prueba integral de la EEPICI	106
3.4.1. Resultado experimental	107
4. Conclusiones	109
Referencias	112
Anexos	121
A. Código de Matlab	121
A.1. Código para la obtención de parámetros teóricos de L_1 y L_2 y potencia máxima permitida	121
A.2. Código para la implementación de la aplicación del Software Control de EEPICI124	

B. Cálculos matemáticos	131
B.1. Potencia adicional debido a la eficiencia del acoplo inductivo	131
C. Códigos de C	133
C.1. Desarrollo del Monitoreo y Almacenamiento en Sensor Controller Studio . . .	133
C.2. Código del archivo ejecutable main.c	134
C.3. Código del archivo ejecutable simple_peripheral.c	137
C.4. Programación del microcontrolador CC2640R2F (5x5mm)	154
C.4.1. Revisión de tarjeta electrónica para BioMEMS y depurador XDS110 Debug Probe	154
C.4.2. Conexiones para la depuración del microcontrolador	155
C.4.3. Configuración de depuración para XDS110 Debug Probe	157

Índice de figuras

1.1. Circuito de carga de un capacitor.	26
1.2. Curvas de la carga de un capacitor y la corriente suministrada.	27
1.3. Esquema general de transferencia de potencia óptica.	32
1.4. Sistema general de acoplo acústico para WPT.	33
1.5. Esquema general de un enlace capacitivo para WPT	36
1.6. Esquema general de un acoplo inductivo de WPT.	39
1.7. Modelo de Inductor y su representación eléctrica.	40
1.8. Circuitos resonantes.	42
1.9. Modelo eléctrico de un enlace inductivo resonante LC paralelo.	42
1.10. Circuito amplificador de potencia clase E.	47
1.11. Circuito amplificador clase E para un acoplo inductivo.	48
1.12. Circuito amplificador de potencia clase E.	48
1.13. Esquema de comunicación de datos y transferencia de potencia de un enlace inductivo.	53
1.14. Esquema de comunicación de datos por radiofrecuencia y transferencia de potencia con enlace inductivo.	54
2.1. Descripción gráfica del prototipo de Dispositivo BioMEMS.	59
2.2. Descripción de la estructura del microcontrolador.	60

2.3. Diagrama de flujo del diseño de Inductores L_1 y L_2	65
2.4. Gráfica de la potencia y eficiencia del acoplo inductivo.	67
2.5. Circuito rectificador-regulador de voltaje para alimentar al prototipo.	70
2.6. Diagrama de flujo del proceso de la Estación Electrónica para una comunicación Bluetooth con el prototipo.	75
2.7. Diagrama de flujo del programa para el microcontrolador del prototipo del dispositivo BioMEMS.	76
2.8. Circuito rectificador-regulador de voltaje de DC.	79
2.9. Diseño de la tarjeta electrónica receptora.	81
2.10. Modelo PCB del inductor L_1	82
2.11. Fabricación de PCBs: Inductor L_1 , L_2 y circuito rectificador-regulador.	82
2.12. Esquema de circuito eléctrico para encontrar L_1 y L_2 real.	83
2.13. Circuito de amplificador de potencia con acoplo inductivo.	85
2.14. Gráfica en el tiempo de la respuesta del amplificador de potencia para diferentes cargas eléctricas resistivas.	86
2.15. Desarrollo de la tarjeta electrónica de la etapa de potencia.	87
2.16. Resultado final de las tarjetas electrónicas.	88
2.17. Encapsulado del elemento receptor.	88
2.18. Versiones anteriores del inductor L_2 . Destacando las características R_{sL_2} y N_2 de cada versión: a) $L = 2.048\mu H$, $R_{sL} = 6.3\Omega$ $N = 15$; b) $L = 2.4\mu H$, $R_{sL} = 5.2\Omega$ $N = 15$; c) y d) $L = 2.048\mu H$, $R_{sL} = 3.5\Omega$ $N = 20$	89
2.19. Versiones anteriores de L_1 con $N = 9$, $L = 2.9\mu H$, y $R = 1\Omega$; y del amplificador clase C con circuito elevador de voltaje.	90
2.20. Interfaz gráfica del Software de Control para la EEPICI.	92
3.1. Base para los elementos del acoplo inductivo.	96
3.2. Esquema general para la prueba de obtención de datos.	97
3.3. Prueba de transmisión de potencia del primer experimento.	99

ÍNDICE DE FIGURAS

3.4. Resultado de la transmisión de potencia con el prototipo como carga.	99
3.5. Nueva prueba de transmisión de potencia con el prototipo como carga R_L . . .	100
3.6. Prueba adicional del prototipo como carga R_L	101
3.7. Comparación del comportamiento del V_R ante variaciones de la carga R_L . . .	102
3.8. Prueba de comunicación inalámbrica para la obtención de datos de la variable física.	103
3.9. Recepción de datos de prueba por correo electrónico.	103
3.10. Montaje de EEPICI.	106
3.11. Prueba de la EEPICI energizando al prototipo y obteniendo los datos de prueba.	107
3.12. Prueba de la EEPICI energizando al prototipo y obteniendo los datos de prueba variables.	108
C.1. Depurador XDS110.	155
C.2. Terminales de protocolo JTAG y de energización del microcontrolador. . . .	156
C.3. Configuración de conexiones JTAG.	156

Índice de cuadros

1.1. Cuadro comparativo de diferentes desarrollos de WPT basado en acoplos acústicos.	34
1.2. Aplicaciones desarrolladas con las características de las bobinas del acoplo inductivo, la potencia entregada, su eficiencia y el medio de trabajo para la experimentación.	38
1.3. Aplicaciones de AIMD con diferentes niveles comunicaciones y velocidad de transferencia de datos.	51
2.1. Comparación de los parámetros teóricos y obtenidos en la fabricación de los inductores.	84
2.2. Nuevos valores de capacitores para la etapa de potencia.	84

Introducción

Motivación

En el Instituto de Ingeniería de la UNAM se han desarrollado prototipos de dispositivos Bio-MEMS para implementar, en un futuro, dispositivos biomédicos implantables en un cuerpo biológico. Sin embargo, estos prototipos necesitan ser energizados de forma inalámbrica para poder monitorear una o más variables físicas (como la temperatura corporal, la presión arterial, entre otras variables) y enviar esta información hacia una plataforma electrónica donde puedan leerse estos datos biológicos.

Ante la falta de un sistema para habilitar el suministro de potencia eléctrica y el envío de datos, se ha planteado el desarrollo de una Estación Electrónica de bajo costo, como una herramienta que proporcione potencia eléctrica hacia los prototipos y genere una comunicación con éstos para la obtención de los datos que han capturado. Además, se ha planteado que estos datos sean enviados hacia una plataforma digital mediante una red informática.

Hipótesis

Es posible desarrollar una Estación Electrónica que suministre potencia eléctrica, de manera inalámbrica, a un prototipo de dispositivo BioMEMS implantable. Además, es viable incorporar un sistema que habilite una comunicación inalámbrica para obtener los datos capturados por el prototipo.

Para lograr lo anterior, se plantea el uso de una estrategia de transmisión de potencia eléctrica basada en técnicas de acoplo electromagnético, que permita la implementación de esta operación de manera efectiva. Por otro lado, se propone una comunicación inalámbrica (como Bluetooth, por ejemplo) para habilitar un intercambio de información entre la Estación Electrónica y los prototipos de dispositivos BioMEMS. Por consiguiente, los prototipos podrán ser energizados, para que operen como dispositivos de monitoreo, y posteriormente, puedan enviar la información recolectada hacia la Estación Electrónica.

Así mismo, al implementar una Estación Electrónica de carácter inalámbrico es factible que en un futuro se consiga energizar a un dispositivo BioMEMS implantable y crear una comunicación con él.

Justificación

Pertinente al área de desarrollo de dispositivos médicos implantables, resulta importante el diseño e implementación de una Estación Electrónica que permita la comunicación inalámbrica con dispositivos electrónicos implantados, habilitados para ser suministrados con potencia eléctrica inalámbrica, y de los cuales se puede acceder a su información por una comunicación de la misma forma.

Para desarrollar un sistema de suministro de potencia eléctrica y obtención de datos fisiológicos hacia y desde un dispositivo médico implantable, es imprescindible evaluar las condiciones de implantación, como las dimensiones físicas de los dispositivos, la potencia

eléctrica que demandan y las causas de interferencia electromagnética generadas por los organismos que albergan el dispositivo implantado. Lo anterior, vuelve compleja la implementación de un sistema adecuado y funcional.

Como resultado, no existe una alternativa comercial de bajo costo que cumpla con las tareas antes mencionadas, lo que limita el alcance de la implementación de dispositivos médicos de implantación.

No obstante, el desarrollo de una Estación Electrónica para la transferencia de potencia de forma inalámbrica para prototipos de dispositivos BioMEMS puede dar una alternativa para la energización y obtención de datos de dispositivos médicos que, en un futuro, sean implantables.

Objetivo

Diseñar, construir y probar una Estación Electrónica, con funcionalidad inalámbrica, que permita un suministro de potencia eléctrica y comunicación hacia un prototipo de dispositivo BioMEMS implantable.

Objetivos particulares

- Diseñar e implementar una etapa de potencia inalámbrica para suministrar energía a prototipos de dispositivos BioMEMS implantables.
- Desarrollar una metodología de recepción inalámbrica de datos procedentes del prototipo de dispositivo BioMEMS.
- Desarrollar un método de envío de datos hacia un servicio de correo electrónico mediante un protocolo de comunicación de redes.
- Desarrollar una interfaz gráfica para el control de la Estación Electrónica.
- Probar la Estación, por medio de su funcionalidad inalámbrica, transmitiendo potencia hacia y recibiendo datos desde un prototipo de dispositivo implantable BioMEMS.

1. Marco Teórico

1.1. Dispositivos Médicos Implantables (IMDs)

1.1.1. Descripción general

Los dispositivos médicos implantables (Implantable Medical Devices, IMDs por sus siglas en inglés) son sistemas que involucran disciplinas como la mecánica, la electrónica, la bioquímica y la biomedicina, entre otras, para construir dispositivos miniatura destinados a ser introducidos en un cuerpo biológico, permitiendo su implantación en tejidos u órganos [1,2].

Entre sus posibles aplicaciones, se usan en la rehabilitación de órganos, en la prevención, tratamiento, control y monitoreo de enfermedades, en el control de disfunciones fisiológicas, en el suministro dosificado de fármacos, entre otros [3].

Los IMDs han sido objeto de investigación y desarrollo debido a que pueden ser utilizados para mejorar significativamente el cuidado de la salud y la calidad de vida del paciente, mientras a la vez, se logran reducir los costos asociados a la atención médica [4]. Como ejemplos de lo anterior, estos dispositivos pueden reducir las intervenciones invasivas para el tratamiento y control de enfermedades, optimizar la administración de fármacos, mejorar la fiabilidad en el monitoreo de señales fisiológicas relevantes y facilitar la rehabilitación de órganos de manera directa [1].

Estos dispositivos tienen un campo muy amplio de aplicaciones, por ejemplo, personas con problemas de audición pueden usar implantes cocleares para recuperar en parte el sentido del oído [5], enfermedades como el Parkinson puede aminorarse con la implantación de estimuladores cerebrales [6], algunos trastornos del sistema neuronal pueden aminorarse con estimuladores incrustados en la médula espinal [7], y se puede lograr una optimización en la administración de fármacos mediante la inserción de microsistemas dosificadores automáticos [4].

Otra ventaja de los IMD ante practicas médicas tradicionales implica una mejora al mantener la integridad del paciente, pues sin el uso constante de cables percutáneos o jeringas que perforan la piel, si así se requiere (como en las muestras para análisis sanguíneo), se disminuye el riesgo de posibles infecciones o situaciones dolorosas para la toma de cada muestra. También brindan mayor movilidad al paciente, pues no hay limitaciones motrices, comparado con equipos de medición clásicos que puede resultar engorrosos, como brazaletes, electrodos o termómetros [8].

La implementación de dispositivos médicos implantables es compleja, debido a las condiciones y requerimientos que deben cumplir estos dispositivos; por ejemplo, se debe lograr la miniaturización de los dispositivos para minimizar su impacto invasivo en el cuerpo humano, se deben construir usando materiales biocompatibles para garantizar la seguridad del paciente, el diseño de los sistemas debe incorporar medidas para reducir la respuesta de cuerpo extraño del organismo y se debe incluir un suministro de potencia eléctrica para los dispositivos que necesiten una fuente de alimentación [9].

Debido a lo anterior, los IMDs se pueden beneficiar de tecnologías emergentes, como los BioMEMS, que involucren el diseño y fabricación de dispositivos miniatura para resolver problemáticas específicas.

1.1.2. Perspectivas generales de los BioMEMS aplicados a tecnologías de implantación

La inclusión de sistemas micro electromecánicos enfocados al campo biológico (BioMEMS, por sus siglas en inglés), promete resolver gran parte de los retos de la implementación de IMDs, dando un enfoque de desarrollo de dispositivos implantados construidos con materiales biocompatibles [4], o en algunos casos biodegradables [10].

Implementando la tecnología BioMEMS para los procesos de fabricación de IMDs es posible desarrollar sistemas miniatura de dimensiones milimétricas y micrométricas, facilitar la incorporación de nuevos sistemas complementarios, así como minimizar el consumo energético. Con estos procesos también es realizable una reproducción a gran escala de estos dispositivos con una alta fiabilidad y bajos niveles de costos de fabricación [11].

Por otro lado, algunos retos en el desarrollo de dispositivos BioMEMS están asociados principalmente a la degradación de los materiales con que son construidos y a la complejidad de su diseño. Un ejemplo de esto último es la recuperación de la información de los dispositivos, que vistas como señales eléctricas o mecánicas, pueden verse disminuidas conforme el dispositivo es de menor tamaño [12].

A pesar de los avances prometedores en el diseño y el desarrollo de IMDs con esta tecnología, sigue habiendo problemáticas que no pueden resolverse, por ejemplo, hasta el momento no es posible integrar una fuente de suministro de potencia BioMEMS, por lo que se usan suministros de potencia convencionales, como baterías implantables. Sin embargo, esto agranda el tamaño del dispositivo, además, su tiempo de vida activa dentro del organismo es limitado.

1.1.3. Clasificación de los IMDs por su principio de funcionamiento

En general, las aplicaciones de dispositivos implantables en el campo de la medicina es muy amplia, es por ello que existen diferentes clasificaciones para los IMDs. La primera clasificación agrupa a los IMDs por su tipo de aplicación, que de manera general, abarca las funciones de monitoreo, de rehabilitación, de suministro de fármacos y de regulación de órganos. Una segunda clasificación se centra en el tiempo que el implante está dentro del organismo, por ejemplo, para periodos de varios años, pocos años o varios meses. Una tercera asocia los IMDs por los tipos de materiales con los que se construyen, resultando en sistemas biodegradable, bioabsorbible o desechables [13].

Una cuarta clasificación identifica los dispositivos implantables que integran elementos activos, como microcontroladores, sensores activos, dosificadores químicos, transeptores, etc. Al incorporar estos elementos característicos a un IMD, convierten a este último en un sistema activo con mayor capacidad funcional.

El desarrollo de un IMD activo (AIMD por sus siglas en inglés), crea la necesidad de tener una fuente de energía eléctrica, pues de ella depende que el dispositivo implantado se mantenga operativo. Como se mencionó con anterioridad, la inclusión de una fuente de alimentación puede complicar el diseño de un dispositivo médico; sin embargo, los AIMD promueven mayor variedad de aplicaciones, debido a que permiten mayor personalización y mejor rendimiento de operación en las múltiples aplicaciones que pueden ejecutar dentro de un cuerpo biológico. Además, es factible un desarrollo escalable del sistema, y esto a su vez, hace posible que el dispositivo ejecute funciones complejas de manera simultánea [14].

Como ejemplo de las capacidades de un microsistema activo, muchos AIMDs incorporan sistemas de comunicación inalámbrica para enviar datos biológicos hacia una unidad de trabajo externo, que comúnmente se encuentra fuera del paciente, al mismo tiempo que ejecutan las actividades fisiológicas principales [15].

1.1 DISPOSITIVOS MÉDICOS IMPLANTABLES (IMDs)

En las siguientes secciones se hará énfasis en el suministro de potencia eléctrica para AIMDs, se hablará de las estrategias principales relevantes para el presente trabajo y finalizaremos hablando de las estrategias de comunicación de datos para esta clase de dispositivos.

1.2. Suministros de potencia eléctrica a Dispositivos Médicos Implantables Activos (AIMDs)

1.2.1. Transferencia Percutánea

Debido a la necesidad de tener una fuente de alimentación para AIMDs, se han implementado técnicas convencionales para suministrar potencia eléctrica a dispositivos médicos habilitados para ser implantados, buscando que estas técnicas en la medicina sean seguras, confiables, eficientes y energéticamente suficientes.

La técnica de transferencia percutánea consiste en insertar electrodos médicos en el cuerpo humano (conocidos como cables percutáneos), atravesando algunos tejidos de órganos, como la piel, hasta llegar al dispositivo implantado. Este método se ha usado ampliamente para transferir potencia eléctrica a AIMDs desde fuentes energéticas, como baterías situadas fuera del organismo o bien, colocadas internamente pero a la periferia del implante [16].

En la actualidad se busca evitar usar cables percutáneos para estas prácticas, debido a los riesgos potenciales para la salud de un paciente. El uso de cables percutáneos representa un sistema significativamente invasivo para el organismo, dando pauta a posibles consecuencias como la degradación del aislamiento de los propios cables, provocando infecciones cutáneas, hematomas o erosiones en el organismo [17].

Añadido a estos inconvenientes, para la colocación de un AIMD a un nivel profundo, en comparación con implantaciones más superficiales, como las subdérmicas, esta técnica resulta médicamente inaceptable debido a los posibles riesgos antes mencionados y las cicatrizaciones en tejidos u órganos periféricos [18]. En concreto, esta técnica no es adecuada para procurar la seguridad del paciente, de modo que se han hechos esfuerzos en aplicar otras tecnologías de suministro de potencia hacia un implante sin sacrificar la seguridad del paciente.

1.2.2. Cosecha de energía

Para aplicaciones de AIMDs que necesitan ocupar un mínimo espacio dentro del organismo y tener una actividad de larga duración, es posible implementar un suministro de alimentación a partir de la recolección de energía del propio entorno en el que es implantado el dispositivo; esta técnica es mejor conocida en inglés como Energy Harvesting. A pesar de que la recolección de energía desde el entorno resulta ser un proceso lento, el uso de Energy Harvesting ofrece una fuente energética consistente y es casi imposible agotar las fuentes primarias de donde se obtiene la energía [14].

Los métodos de obtención de energía se han basado en la captación de diferentes tipos de energías: a partir de la energía solar mediante celdas fotovoltaicas, de la energía del movimiento del cuerpo mediante transductores piezoeléctricos y triboeléctricos, y de la energía térmica con recolectores térmicos [18, 19].

Algunos sistemas basados en transductores piezoeléctricos ha logrado obtener una potencia considerablemente suficiente para aplicaciones implantables de ultrabajo consumo energético a partir de los movimiento de órganos del cuerpo, como del corazón y de los pulmones, alcanzando pocos microwatts de potencia entregada [20]; del diafragma, liberando unos cuantos microwatts de potencia eléctrica [21]; y de la sangre que circula en arterias principales, entregando potencias no mayores a 40nW [22].

La principal restricción para el uso de la recolección de energía desde el entorno del implante es la cantidad de potencia entregada, pues suelen ser niveles muy bajos que sólo aplicaciones de muy bajo consumo pueden aprovechar. Por lo tanto, muchas aplicaciones de AIMDs centran la atención en otras fuentes de energía que puedan entregar la potencia suficiente que ellos demandan.

1.2.3. Baterías y Capacitores

Para satisfacer una mayor demanda de potencia eléctrica, una estrategia ampliamente usada es incorporar una fuente de alimentación local dentro del AIMD, como capacitores, baterías recargables o no recargables. Con esta técnica es posible prescindir de cables percutáneos y procurar la integridad del paciente, haciendo énfasis en implementar un sistema remoto que esté activo por un periodo prolongado de forma autónoma.

El uso de baterías miniatura no recargables ha sido un recurso muy recurrente en AIMD a largo plazo que permite un funcionamiento constante para dispositivos con una actividad críticamente continua. Por exponer algunos ejemplos, los desfibriladores cardioversores de implante subdérmico llegan a tener una actividad constante de hasta casi 8 años con baterías implantadas [17]; de la misma manera, un marcapasos opera de forma continua desde 3 hasta 10 años [19]; por último, el desarrollo de un micro marcapasos estima una duración de hasta 15 años activo antes de necesitar un reemplazo de batería [17].

En contra parte, al integrar este tipo de baterías de no recarga en un dispositivo implantable, es muy probable que el tamaño del dispositivo incremente significativamente, pues las dimensiones de las baterías actuales no logran reducirse al grado de ser comparables con los volúmenes de los AIMD; por tanto, resulta perjudicial incorporar una fuente de alimentación de este tipo en aplicaciones que demandan un suministro prolongado de energía, y a la vez, una huella física mínima [23].

Otro inconveniente del uso de baterías no recargables se refleja cuando el suministro de energía se agota y hay una necesidad de intervención quirúrgica para reemplazar la batería o todo el implante. Para evitar dichas intervenciones, se ha apostado por incluir supercapacitores y baterías recargables a dispositivos implantables; esto a su vez permite extender la vida útil del AIMD. Con un aspecto de recarga, estos almacenadores de energía pueden ser de menor capacidad, lo que reduce su tamaño y por tanto el dispositivo implantable.

En cuanto a la implementación de AIMDs con almacenadores de energía, como baterías recargables o super-Capacitores, el enfoque se inclina hacia aplicaciones que tienen un comportamiento intermitente, es decir, con tiempos de actividad de manera periódica y no necesariamente de forma continua todo el tiempo. A pesar de las cantidades menores de capacidad de almacenamiento energético, la aplicación de esta tecnología puede aprovecharse en el monitoreo y la rehabilitación de órganos, en el suministro de fármacos, entre otros.

Un claro ejemplo de desarrollo con esta tecnología es un implante de Electroencefalógrafo (EEG) con Estimulación Cerebral Profunda (DBS, por sus siglas en inglés). Las funciones principales del dispositivo son el monitoreo de la actividad eléctrica del encéfalo como una herramienta de investigación y, la estimulación profunda como terapia cerebral. Este dispositivo tiene incorporado una batería que puede ser recargada desde una plataforma electrónica externa para que el AIMD se mantenga operativo por periodos cortos pero suficientes en el monitoreo y estimulación cerebral [24].

Para establecer o proponer el tiempo que el almacenador puede suministrar potencia por sí mismo hacia un AIMD, se debe considerar la cantidad de energía eléctrica que puede contener dicho almacenador electrónico. En la expresión (1.1) se describe la cantidad de carga eléctrica, por unidad de tiempo, que puede ser almacenada (con unidades Amper-hora, Ah). El Almacenamiento de Carga Eléctrica ACE está en función de la potencia suministrada al dispositivo P , de el voltaje V que entrega el almacenador y de un intervalo de tiempo t :

$$ACE = \frac{P}{V} \times t[Ah] \quad (1.1)$$

En caso de usar un capacitor o un super-Capacitor, es necesario calcular la capacitancia del mismo a partir de la cantidad de energía que se desee almacenar (representada en mAh) [25].

CAPÍTULO 1. MARCO TEÓRICO

Con la expresión (1.2) se puede obtener la carga eléctrica de un super-Capacitor o un capacitor simple a partir del parámetro anterior, y finalmente, con la ecuación del capacitor (1.3) se obtiene su capacitancia.

$$q_c = \frac{ACE}{0.270Ah} [Coulomb] \quad (1.2)$$

$$C = \frac{q_c}{V_c} [Farad] \quad (1.3)$$

Para recargar una batería, es necesario suministrar una corriente y un voltaje específico para no dañarla. Mientras que, para cargar eléctricamente un capacitor, sólo se necesita definir la cantidad de carga eléctrica y la rapidez de carga del mismo.

El siguiente circuito describe un arreglo eléctrico para cargar un capacitor. A través de un voltaje $V_{aplicado}$ es posible cargar un capacitor con una corriente máxima propuesta. Esto se logra colocando una resistencia en serie R_{carga} para limitar el flujo de corriente eléctrica, obteniendo así un comportamiento de la corriente y que es descrito por la ecuación (1.7).

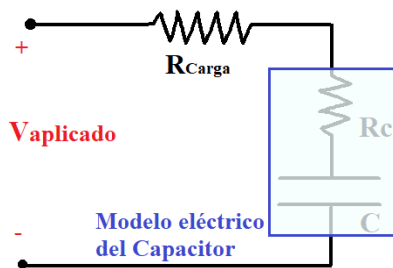


Figura 1.1: Circuito de carga de un capacitor.

En la figura 1.2 se puede observar la curva de almacenamiento de carga eléctrica de un capacitor a través del tiempo, representada de color azul, donde se puede notar que para un mínimo de carga eléctrica en el capacitor, existe una corriente máxima que fluye hacia él para almacenar cargas eléctricas (representada por la curva roja).

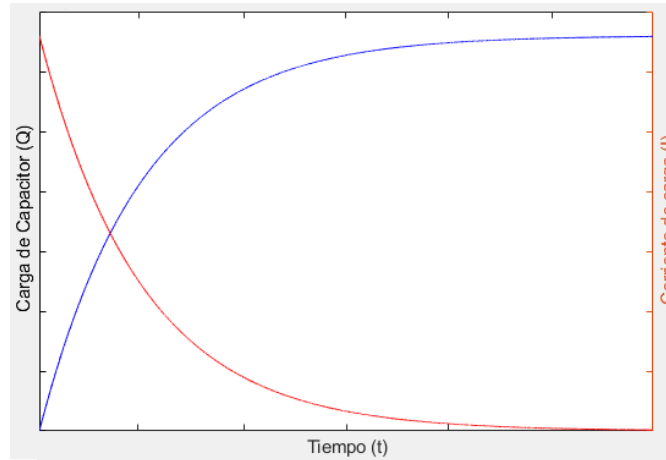


Figura 1.2: Curvas de la carga de un capacitor y la corriente suministrada.

El comportamiento transitorio (o a través del tiempo) del almacenamiento de carga eléctrica del capacitor está descrito por:

$$Q_{super-C} = C \times V_{aplicado} \times [1 - e^{-\frac{t}{\tau}}] \quad (1.4)$$

donde C es la capacitancia, $V_{aplicado}$ es el voltaje aplicado al circuito para cargar el capacitor (ver la figura 1.1), t es el tiempo y τ es la constante de tiempo que vale $R \times C$; donde R es la resistencia en serie equivalente del circuito de carga.

Por otro lado, despejando la constante de tiempo τ de la ecuación (1.4), y para una carga eléctrica deseada Q_d , obtenemos:

$$\tau = -\frac{t}{\ln\left[1 - \frac{Q_d}{C \times V_{aplicado}}\right]} \quad (1.5)$$

siendo τ la constante de tiempo que depende de los componentes eléctricos:

$$\tau = RC \quad (1.6)$$

Por último, el comportamiento de la corriente en función del tiempo se puede visualizar en la curva roja de la figura 1.2 y se puede apreciar que es cada vez menor conforme transcurre

el tiempo, por lo que para un tiempo inicial $t = 0$ (donde el capacitor está completamente descargado), en la ecuación (1.7) tenemos una corriente máxima en este instante, definida como:

$$i_{carga} = \frac{V_{aplicado}}{R} \cdot e^{\frac{-t}{RC}} \Big|_{t=0} \rightarrow I_{máx} = \frac{V_{aplicado}}{R} \quad (1.7)$$

Usando almacenadores con la cualidad de recarga, es posible prolongar la vida útil del dispositivo implantable y conservar una huella física mínima, sin embargo, esto lleva consigo la necesidad de tener un suministro de potencia para recargar ya sea, capacitores o baterías recargables en lapsos cortos de tiempo, por ejemplo, de días, semanas o meses.

Ante esta problemática, muchas aplicaciones optan por el uso de fuentes de alimentación externas que puedan proporcionar potencia eléctrica o energía suficiente a los almacenadores antes mencionados. La mayoría de estas aplicaciones usan metodologías de suministro de potencia de forma inalámbrica, pues con ellas es posible recargar estos almacenadores sin el uso de cables percutáneos.

1.2.4. Transferencia inalámbrica

Energizar un AIMD de manera inalámbrica consiste en transmitir energía de un punto a otro sin la necesidad de cables que transporten la potencia hacia el implante. Para poder energizar a los AIMD desde el exterior, normalmente se usan estrategias de acoplos (como se verán más adelante) para transmitir la energía de un punto a otro mediante energía lumínica, sonora o por campos eléctricos y/o magnéticos.

En general, con estas técnicas es posible satisfacer las necesidades energéticas de los implantes y respetar las condiciones de su implantación, como lograr una mayor miniaturización de implante, prescindiendo de baterías de tamaños considerablemente grandes; disminuir la complejidad de diseño de AIMD, simplificando las condiciones de implantación; reducir los

costos de implementación, en comparación con otros dispositivos remotos de tamaños comparables; obtener un rango extenso de potencia entregada, que va desde pocos mW hasta algunas centenas de ellos, [18, 26].

En las siguientes secciones se hablará a profundidad sobre las diferentes estrategias de suministro inalámbrico más relevantes, dando una introducción del principio de funcionamiento, las dificultades que presenta cada una y algunas aplicaciones destacadas empleando dichas estrategias.

1.3. Estrategias de transferencia de potencia eléctrica inalámbrica

La transferencia de potencia inalámbrica (WPT por sus siglas en inglés), no ha sido una metodología reciente, pues desde la década de los 90's se han propuesto sistemas inalámbricos basados en inducir fuerzas electromotrices para transferir potencia eléctrica de forma remota. Este fenómeno se descubrió en 1888 por una serie de experimentos hechos por H. R. Hertz; confirmando la teoría de Maxwell sobre la existencia de radiación electromagnética, Hertz logró generar un acoplo inductivo y así transmitir electricidad entre dos bobinas deparadas a poca distancia entre ellas [27].

El gran potencial que promete la transmisión inalámbrica de potencia ha inclinado el interés en su investigación para desarrollar sistemas cada vez más eficientes, confiables y de bajo costo, no solo en el campo automotriz y en dispositivos inteligentes, sino también para aplicaciones en la industria, la milicia, el hogar, la medicina, el entretenimiento, etc.

Para el desarrollo de AIMDs con suministro de WPT, se deben de tomar en cuenta algunas condiciones para su implementación; por ejemplo, se debe diseñar una etapa de potencia que cubra la demanda energética del dispositivo implantable y contemplar los tamaños miniatura de los dispositivos implantados para garantizar la fiabilidad de la transmisión en diferentes medios como la piel y órganos periféricos, mientras a su vez, no debe de comprometer la seguridad del paciente (esto para evitar quemaduras y daños a los tejidos celulares y órganos) [14, 18].

Por lo anterior, en cuanto al desarrollo de AIMDs con sistemas WPT se ha buscado implementar nuevas estrategias para transmitir potencia inalámbricamente usando disciplinas como la acústica, la óptica, la termodinámica y el electromagnetismo [18, 28]. Con estas nuevas técnicas se ha podido ampliar la aplicación de transmisiones remotas en aplicaciones particulares en el campo de la medicina.

1.3. ESTRATEGIAS DE TRANSFERENCIA DE POTENCIA ELÉCTRICA INALÁMBRICA

De manera general, todas las estrategias se basan en dos componentes principales: un elemento transmisor (o más de uno), que expulsa, irradia o genera un tipo de energía, ya sea sonora, lumínica, electromagnética o térmica (y pueda ser transferida a través de uno o más medios, como el aire, los tejidos de la piel, las fibras musculares, etc). Por otro lado, se tiene un elemento receptor, incorporado al dispositivo implantado, que capta la energía y la transforma en corriente eléctrica y es acondicionada para almacenarse o usarse inmediatamente [29].

Las estrategias de WPT se pueden clasificar según la forma de irradiar la energía; la clasificación general consiste en si el sistema de WPT trabaja o no con ondas electromagnéticas (EM). En las siguientes secciones se abordarán las estrategias con y sin el uso de EM enfocadas al suministro eléctrico de AIMDs, explicando el principio de funcionamiento, resaltando los requerimientos de su diseño y desarrollo, exponiendo los retos de su implementación y mencionando las aplicaciones destacadas en el ámbito médico.

1.3.1. Sistemas de acoplos no electromagnéticos

Acoplo Óptico

El uso de la óptica para transferir potencia eléctrica ha sido una de las técnicas más recientes en cuestión de investigación y desarrollo. El principio de funcionamiento consta de un acoplo óptico entre un transmisor de luz fuera del organismo y un receptor óptico incorporado al AIMD, mismos que se encuentran separados entre sí por los tejidos de la piel.

El sistema de transmisión cuenta con un circuito que modula una potencia eléctrica y es dirigida a un emisor de luz para ser irradiada en energía lumínica. Por otro lado, un sistema receptor incorporado al implante, puede captar la energía lumínica a través de fotosensores. El esquema general del principio de funcionamiento de un sistema de acoplo óptico se ilustra en la imagen 1.3.

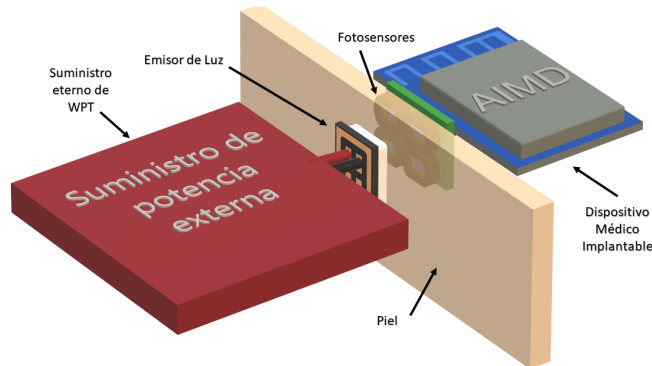


Figura 1.3: Esquema general de transferencia de potencia óptica.

Una de las principales ventajas de este método es la miniaturización lograda de los sistemas WPT con óptica, que al ser incorporados al AIMD, afectan en menor medida el tamaño del dispositivo. Por ejemplo, el trabajo desarrollado en 2001 por Goto *et al.* [30] lograron entregar una potencia lumínica de $22mW/cm^2$ usando un haz infrarrojo.

En 2015 se logró construir un sistema óptico micrométrico que puede transferir una potencia de $168\mu W$ en un área de $500 * 500\mu m^2$. Para una posible aplicación médica de implante subcutáneo, se simuló un espesor de tejidos subcutáneos de 3mm de grosor, para realizar experimentos in vitro, sin embargo, no se contempló la pérdida de energía lumínica por la absorción en los tejidos simulados [31].

A pesar de los avances que se han logrado con esta técnica, existe un importante decremento en la potencia lumínica entregada al receptor a medida que incrementa el espesor de la piel debido a que gran parte de la energía lumínica es absorbida o reflejada por los tejidos de la piel. Por otro lado, el mayor problema del acoplo óptico es la necesidad de un estricto emparejamiento entre el láser y los receptores de luz para entregar la potencia deseada.

En concreto, la cantidad de potencia obtenida en la recepción lumínica es escasa; para rangos comprendidos entre varios μW para tamaños milimétricos, esta estrategia puede implementarse en IMDs de ultra baja potencia.

1.3. ESTRATEGIAS DE TRANSFERENCIA DE POTENCIA ELÉCTRICA INALÁMBRICA

Debido a que el organismo puede absorber la energía irradiada por esta técnica, y las demás que se mencionarán más adelante, es necesario que estos desarrollos contengan el análisis de los niveles de absorción específica de energía (SAR por sus siglas en inglés), sin embargo, para esta metodología hay una escasez de dicho análisis [18].

Acoplo Acústico

La transmisión de energía con esta metodología se implementa mediante elementos acústicos para transmitir ondas ultrasónicas entre un dispositivo transmisor (como emisor de ondas acústicas) y un dispositivo receptor implantado (como una membrana acústica). Las ondas ultrasónicas, emitidas desde fuera del organismo, tienen que atravesar varios medios a su camino, como los tejidos de piel y órganos, para llegar al receptor incorporado al AIMD [32]. El arreglo del enlace o acoplo acústico se ilustra en la figura 1.4.

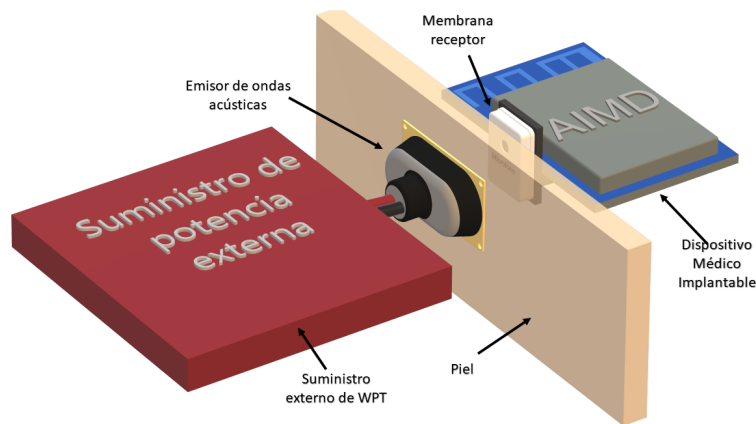


Figura 1.4: Sistema general de acoplo acústico para WPT.

Los sistemas WPT basados en acoplos acústicos, ofrecen un campo de aplicación mayor en comparación con los sistemas de transmisión óptica. Algunas propiedades destacadas de esta metodología son: la libertad de trabajar con un rango amplio de frecuencias de operación sin sufrir interferencias con ondas electromagnéticas, conseguir menores cantidades de SAR en

CAPÍTULO 1. MARCO TEÓRICO

tejidos y órganos periféricos al implante, obtener diseños de receptores de tamaños reducidos para implantaciones profundas, y tener una PTE comparable con otras estrategias eficientes de WPT, como la transmisión por acoplo inductivo [33].

Uno de los desarrollos sobresalientes empleando transductores piezoeléctricos fue elaborado por los investigadores S. Arra, *et al.*, quienes consiguieron una transmisión de potencia a una distancia de 10cm (ver Cuadro 2.1). El experimento alcanzó una PTE del 35 % con agua desgacificada como medio de propagación. No obstante, los autores constatan un decremento en la PTE para implantaciones médicas [34].

En la siguiente tabla se han expuesto algunos desarrollos que pueden ser comparados entre sí. Por ejemplo, se puede observar que las potencias entregadas pueden alcanzar algunos mW . Sin embargo, es probable que la potencia entregada disminuya considerablemente debido a que parte de la energía transmitida es absorbida por los tejidos biológicos del organismo.

Autor	Eficiencia	Potencia entregada	Densidad de potencia	Profundidad	Medio de transmisión
Patrick L. [35]	0.022 %	0.5mW	30mW/cm ²	120 mm	Implantación en ratas (in vivo) y fantoma de tejidos
A. Sanni [33]	1 %	8mW	10.2mW/cm ²	70 mm	Agua
Mazzilli [36]	1.6 %	28mW	93.3mW/cm ²	105mm	Agua

Cuadro 1.1: Cuadro comparativo de diferentes desarrollos de WPT basado en acoplos acústicos.

1.3. ESTRATEGIAS DE TRANSFERENCIA DE POTENCIA ELÉCTRICA INALÁMBRICA

Esta forma de WPT tiene un gran potencial para brindar un suministro de potencia a AIMDs con características de implantación profunda y de una demanda de potencia baja (pocas decenas de mW).

Sin embargo, aún hay limitaciones que dificultan la implementación de estos sistemas en dispositivos implantables, pues aún faltan pruebas que afirmen que, usando esta metodología, no se compromete la integridad de los tejidos de la piel y los órganos periféricos. También falta la optimización de este tipo de acoplos para minimizar las pérdidas de energía cuando los transductores del acoplo se desemparejan entre sí.

A continuación se presentarán las metodologías de WPT que usan ondas electromagnéticas para transferir la energía.

1.3.2. Sistemas de acoplos por ondas electromagnéticas

Para sistemas inalámbricos con acoplos que funcionan transmitiendo la potencia mediante electromagnetismo se clasifican por campo cercano, medio y lejano. Para considerar un sistema de campo cercano, se cumplen dos condiciones: la distancia que existe entre el elemento transmisor (Tx) y receptor (Rx) es menor que la longitud de onda¹ de la frecuencia de trabajo del sistema², mientras a la vez, las dimensiones de Tx son, como mínimo, la mitad de la longitud de onda [37].

En cambio, para decir que un sistema de WPT sea de campo medio, la distancia entre los elementos Tx y Rx es mayor que las dimensiones del elemento de radiación Tx. En comparación con los sistemas de campo cercano, estos últimos pueden transferir potencia eléctrica a distancias mayores, manteniendo considerablemente, la eficiencia de transmisión [38].

¹La longitud de una onda es la distancia, en metros, en la que se repite la forma de la onda.

²La frecuencia de trabajo hace referencia al número de repeticiones de la energía emitida por el transmisor en un segundo, y este parámetro es medido en Hertz (Hz).

Las principales metodologías para la transferencia de potencia inalámbricamente mediante acoplos EM son 2, y usan campos únicamente magnéticos, eléctricos; o bien, una combinación de ambos campos como un tercer método. estos se verán a continuación.

Acoplo Capacitivo

La implementación de un suministro de potencia para AIMDs usando un acoplo capacitivo, se logra haciendo dos enlaces sin contacto directo entre dos pares de electrodos en la cercanía de la piel. En la figura 1.5 se ilustra cómo se generan estos dos enlaces capacitivos. Los electrodos 1 y 2 al emparejarse con los electrodos 3 y 4, respectivamente, crean dos capacitores de placas paralelas. En cuanto al dieléctrico de cada capacitor, este es formado por los tejidos de la piel.

Al aplicar una diferencia de potencial a los electrodos exteriores, se genera una capacitancia de acoplo en cada enlace capacitivo, lo que origina una corriente eléctrica en los electrodos 4 y 3, que se encuentran implantados dentro del organismo, y que dan origen a la fuente de energía para el AIMD. De esta manera, es posible transmitir potencia eléctrica hacia un AIMD mediante la generación de campos eléctricos [39, 40].

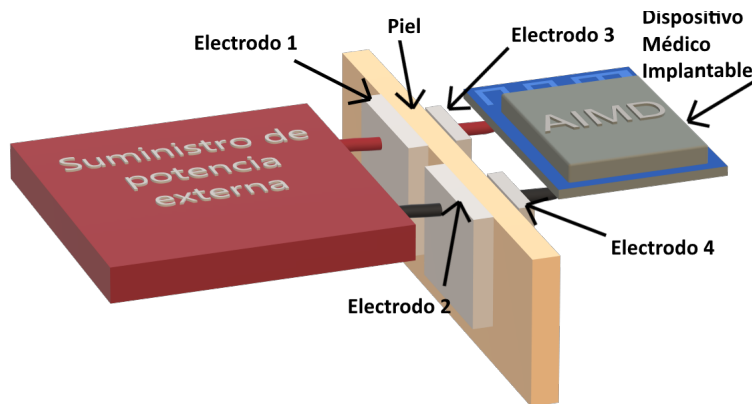


Figura 1.5: Esquema general de un enlace capacitivo para WPT

1.3. ESTRATEGIAS DE TRANSFERENCIA DE POTENCIA ELÉCTRICA INALÁMBRICA

El primer experimento para un AIMD con acoplo capacitivo se realizó en el trabajo de Sodagar y Amiri en 2009 [41]; demostraron que es posible reflejar un voltaje mediante este tipo de acoplo a distancias de separación menores a un milímetro, así como la transferencia de datos a través del mismo enlace. Las dimensiones de los electrodos transmisores son de $25\text{mm} \cdot 25\text{mm}$ y los electrodos de recepción de 12mm por lado, mientras que para las pruebas el dieléctrico fue aire, considerando que para las dimensiones del enlace capacitivo, asemejan las características del dieléctrico de la piel, por lo que la capacitancia específica es equivalente para un experimento in vivo.

Los modelos de acoplos capacitivo simple, resonante y resonante con placa intermedia, propuestos por Narayanamoorthi en 2019 [42], han aportado un avance significativo para esta metodología, logrando eficiencias de hasta 42 % para distancias de separación entre placas de 25mm en pruebas in vitro, y de 14mm para pruebas in vivo. Los experimentos fueron realizados en disoluciones acuosas y se simuló un conjunto de capas de la cabeza humana (la piel, la grasa, el cráneo y el cerebro) para obtener resultados de pruebas in vitro. Por otro lado, también se realizaron pruebas in vivo con piel de cerdo.

De acuerdo a Narayanamoorthi, uno de los retos para acoplos capacitivos es mantener la cantidad de potencia en la transmisión. Al ser un sistema de acoplo capacitivo, esta cantidad está determinada en gran medida por la capacitancia específica generada por el acoplo, y esta última está en función de la separación entre los electrodos que forman cada enlace.

Los sistemas con acoplos capacitivos se han investigado a profundidad para la transferencia de datos de dispositivos implantables. Sin embargo, para tener desarrollos de sistemas WPT con este método, es necesario optimizar estos sistemas para incrementar la potencia que puede entregarse en el lado receptor, así como realizar pruebas de SAR para consolidar método de potencia como seguro en aplicaciones médicas. En concreto, este método es prometedor para la transferencia de potencia, como se dio a conocer en [42], pero aún falta más investigación que complemente y optimice este sistema de transferencia de potencia.

Acoplo Inductivo resonante

La metodología más estudiada e implementada para la transmisión inalámbrica de potencia en aplicaciones médicas usan inductores como antenas para transferir energía entre ellos [18]. Estos sistemas pueden mejorarse y adecuarse fácilmente a las necesidades de cada aplicación. A continuación se muestra un cuadro comparativo de algunas aplicaciones ya desarrolladas.

Autor	Aplicación	Características de Bobina TX	Características de Bobina Rx	Potencia Entregada y Nivel de SAR	Eficiencia y Frecuencia	Distancia y medio de transmisión
Manoufali M. Bialkowski K.S. Mohammed B.J. Mills P.C. Abbosh A. [43]	Implante cerebral	Impresa Circular $d_{ext}=6\text{mm}$ $d_{int}=3\text{mm}$	Impresa Tridimensional $L_{Lado} = 1\text{mm}$ $V = 0.9\text{mm}^3$	82mW 1.9W/kg	$\eta = 0.7\%$ 402MHz	10mm Aire
					$\eta = 0.02\%$ 402MHz	10mm in vivo: Piel de puerco
Feng P., Yeon P. Cheng Y. Ghovanloo M. Constandinou [44]		No impresa Circular $d_{ext}=17\text{mm}$	Impresa Exagonal $L = 4\text{mm}$	10mW 0.155W/kg	$\eta = 3.05\%$ 318MHz	11mm en vitro: Costillas de cordero
Pan G. Xin W. Yan G. Chen J. [45]	Neuro-Estimulador	Convencional Circular $d_{ext}=30\text{mm}$	Convencional Circular $d_{ext}=20\text{mm}$	180mW 0.1W/kg	$\eta = 65.8\%$ 1MHz	5mm in vivo
Mabrouki A. Latrach M. Lorrain V. [46]	Implante Ocular con	Convencional Circular $d_{ext}=40\text{mm}$	Convencional Circular $d_{ext}=20\text{mm}$	- 0.66W/Kg	$\eta = 5\%$ 2MHz	40mm in vivo
Sun T. Xie X. Li G. Gu Y. Deng Y. Wang, Z [47]		Convencional Circular $d_{ext}=45\text{mm}$	Convencional Circular $d_{ext}=10\text{mm}$	2W 0.02w/Kg	$\eta = 17.5\%$ 13.56MHz	16-20mm 2mm de piel 2mm de grasa 7mm de hueso
Jow U. M. Ghovanloo M. [48]	Sin aplicación	Impresa Cuadrada $d_{ext}=24\text{mm}$ $d_{int} = 9.4\text{mm}$	Impresa Cuadrada $d_{ext}=10\text{mm}$ $d_{int} = 7.2\text{mm}$	—	$\eta = 30.8\%$ 13.56MHz	10mm en vitro: con tejido muscular

Cuadro 1.2: Aplicaciones desarrolladas con las características de las bobinas del acoplo inductivo, la potencia entregada, su eficiencia y el medio de trabajo para la experimentación.

1.3. ESTRATEGIAS DE TRANSFERENCIA DE POTENCIA ELÉCTRICA INALÁMBRICA

En la figura 1.6 se ilustra el esquema general de un acoplo inductivo, que consta del emparejamiento de dos inductores, incluso si se encuentran separados por una barrera orgánica como la piel. Al haber una cercanía entre el inductor receptor Rx y el inductor transmisor Tx, se puede crear un enlace o acoplo inductivo, con lo que es posible transferir energía de un inductor hacia el otro.

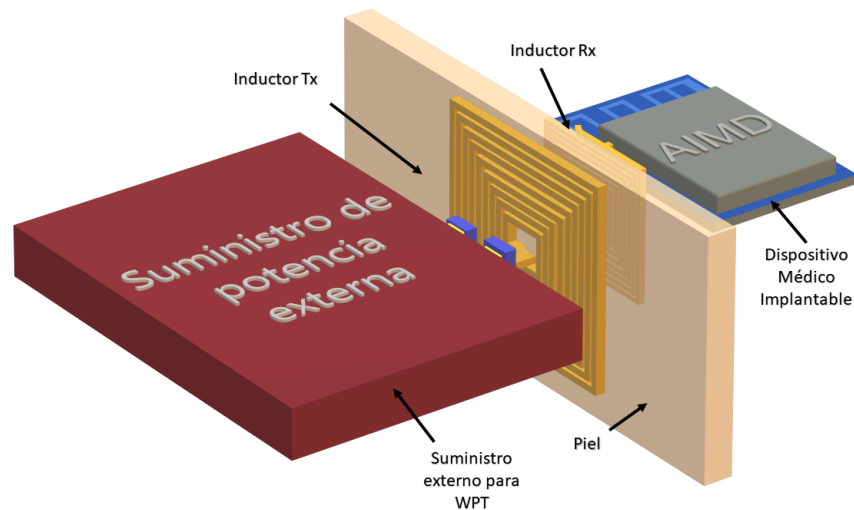


Figura 1.6: Esquema general de un acoplo inductivo de WPT.

Para el desarrollo de sistemas WPT para AIMDs con acoplo inductivo, se deben considerar parámetros de diseño a partir de las condiciones de implantación, pues estas determinan la profundidad a la que se encuentra el implante y por tanto, la distancia mínima de separación entre bobinas, estas condiciones también delimitan el tamaño de la antena receptora y sobre todo la potencia que puede entregarse de un extremo al otro.

Para lograr tener una sistema WPT bajo las condiciones de implantación, se busca conseguir el mejor enlace inductivo, lo cual se logra implementando uno y/o más de tres métodos generales, que son: fabricando inductores de alta calidad (esto medido por el Factor de calidad, Q), creando transmisores y receptores resonantes, y/o manteniendo una alineación apropiada entre las bobinas Rx y Tx [18, 49, 50].

CAPÍTULO 1. MARCO TEÓRICO

En cuanto al primer método, en la expresión del Factor de Calidad de un inductor, descrito en (1.8), se puede notar que la inductancia L y frecuencia de trabajo f pueden aumentar el factor Q si estas incrementan. En cambio, el material con el que está hecho el inductor crea una pequeña oposición al flujo de corriente que pasa a través de él, caracterizada como la resistencia propia R del inductor, y disminuye el factor de calidad.

$$Q_L = \frac{(2\pi f L)}{R} \quad (1.8)$$

Para incrementar la inductancia en un inductor se han aplicado diferentes métodos de fabricación, siendo estos por impresión 3D permitiendo dimensiones menores (como se puede comparar en el cuadro 1.2) o por embobinados convencionales que permiten mayor versatilidad de aplicación; también se han diseñado diferentes geometrías para los inductores como cuadradas, hexagonales, espirales o circulares, y teniendo formas planares o tridimensionales.

Los inductores de fabricación más rápida y muy conocidos son aquellos construidos por pistas de circuitos PCBs, que tiene la forma planar y en espiras, como se muestra en figura 1.7. La inductancia propia de estos elementos puede alcanzar algunas unidades y decenas de μH dependiendo del área A que ocupa, es decir, para las dimensiones L_m y L_n , de su espesor t y del número de vueltas N , ver ecuación (1.9).

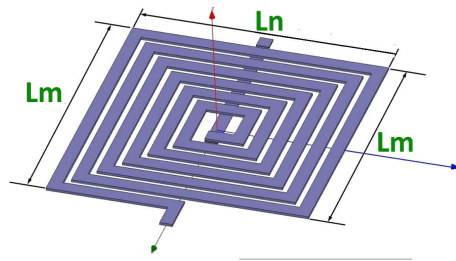


Figura 1.7: Modelo de Inductor y su representación eléctrica.

Para un inductor planar de geometría cuadrada, la inductancia propia L viene dada por la ecuación (1.9), descrita por Mohan *et al.* [51], donde N es el número de vueltas del inductor, d_{prom} es el diámetro promedio de cada espira expresado como $d_{ext} \times (1 + (1 - \phi)/(1 + \phi))$,

1.3. ESTRATEGIAS DE TRANSFERENCIA DE POTENCIA ELÉCTRICA INALÁMBRICA

μ es la permeabilidad del aire, $C_1 = 1.27$, $C_2 = 2.07$, $C_3 = 0.18$ y $C_4 = 0.13$, son los coeficientes establecidos descritos para la geometría cuadrada del inductor, y por último, $\phi = (d_{ext} - d_{int}) / (d_{ext} + d_{int})$ es la relación de los diámetros del inductor.

$$L = \frac{C_1 \mu N^2 d_{prom}}{2} \cdot \left[l_n \left(\frac{C_2}{\phi} \right) + C_3 \phi + C_4 \phi^2 \right] \quad (1.9)$$

En la primera expresión de (1.10) se muestra la resistencia por unidad R_{uL} del inductor, es decir, la resistencia intrínseca promedio por cada vuelta del inductor. De modo que, la resistencia propia del inductor es proporcional al número de vueltas del inductor, expresada como R_{sL} .

$$R_{uL} = \frac{4\rho}{\phi t} \quad \rightarrow \quad R_{sL} = R_{uL} N^2 \quad (1.10)$$

donde ρ es la resistividad del material conductor del que está hecho el inductor, y t su espesor.

El segundo método es implementar circuitos resonantes (conocidos como circuitos tanque) en el lado transmisor Tx y en el lado receptor Rx, ya que ayudan a crear un fenómeno de resonancia en cada lado de la transmisión. De esta manera, se maximiza la transferencia de energía cuando los circuitos de transmisión y recepción están operando a una frecuencia conocida como frecuencia de resonancia, o en otras palabras, se encuentran sintonizados [49].

Un acoplo inductivo resonante puede ser sintonizado al añadir un elemento eléctrico en paralelo o serie al inductor de transmisión de potencia. Este elemento puede ser un capacitor como se muestra en las figura a) y b) de 1.8). Este arreglo forma un circuito LC, donde el fenómeno de resonancia para estos dos arreglos se da a la frecuencia resonante f_o , descrita por la ecuación (1.17).

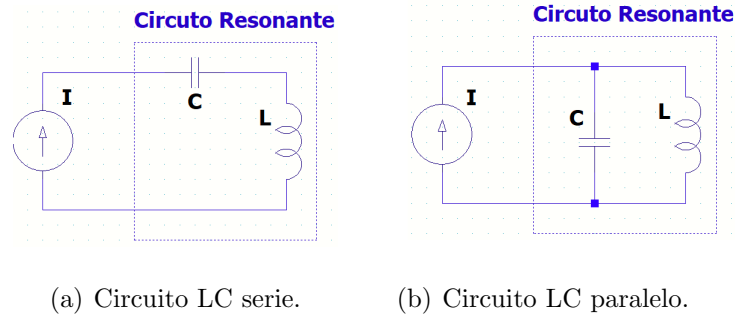


Figura 1.8: Circuitos resonantes.

$$f_o = \frac{1}{2\pi\sqrt{LC}} \quad (1.11)$$

De esta manera, un modelo eléctrico de un enlace o acoplo inductivo resonante puede describirse gráficamente en la figura 1.9, teniendo en cuenta la configuración del circuito resonante paralelo en el lado transmisor y en el lado receptor. Las resistencias R_{s1} y R_{s2} son la resistencia propias de cada inductor, R_S la resistencia interna de la fuente de voltaje V_1 , y por último R_L la carga eléctrica del dispositivo implantable.

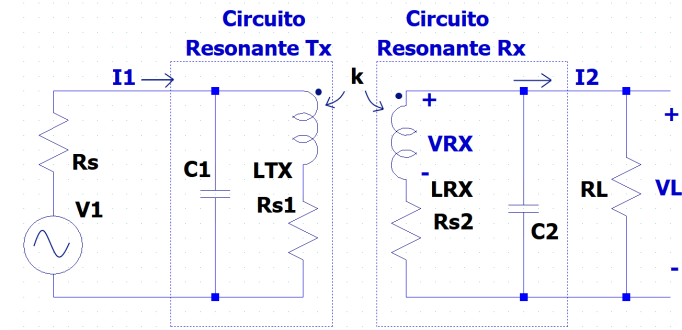


Figura 1.9: Modelo eléctrico de un enlace inductivo resonante LC paralelo.

La transmisión de potencia parte de aplicar una corriente variable i_1 a la frecuencia f_o al arreglo LC de transmisión, lo que generará un campo magnético de densidad B_{TX} , y debido

1.3. ESTRATEGIAS DE TRANSFERENCIA DE POTENCIA ELÉCTRICA INALÁMBRICA

a la cercanía que hay entre el inductor L_{TX} y L_{RX} , se genera un campo magnético en el segundo inductor, de densidad $B_{RX} = kB_{TX}$, lo que produce una corriente eléctrica variable i_2 y un voltaje inducido V_{RX} [18].

El voltaje inducido en el lado receptor es calculado en el análisis de Nima Soltani *et al.* [52] como la siguiente expresión, donde N_2 es el número de vueltas del inductor L_{RX} y A_2 es su área:

$$V_{RX} = 2\pi f_o N_2 B_{RX} A_2 \quad (1.12)$$

Por otro lado, la potencia que se entrega al sistema está descrita en la siguiente ecuación:

$$P_L = \frac{V_{RL}^2}{R_L} = I_{RL}^2 \cdot R_L \quad (1.13)$$

o bien, expresada por el voltaje inducido V_{RX} :

$$P_L = \left(\frac{V_{RX}}{R_{s2} + R_L} \right)^2 \cdot R_L \quad \rightarrow \quad P_L = \frac{(2\pi f_o N_2 B_{RX} A_2)^2}{(R_{s2} + R_L)^2} \cdot R_L \quad (1.14)$$

El tercer método es aumentar la frecuencia fundamental o de resonancia lo más posible sin infringir las condiciones de implantación. Al aplicar estos tres métodos a un sistema de acoplo inductivo, se puede obtener un sistema de transferencia de potencia mejorado que sea capaz de entregar la potencia necesaria para un AIMD.

Continuando con las condiciones de implantación, para el diseño de un enlace inductivo es necesario incluir la restricción de los niveles de SAR para la piel y otros tejidos que pueden ser comprometidos en la implementación de esta técnica. La expresión (1.15) describe el SAR en función de la intensidad del campo magnético total H que es generado por el enlace inductivo; siendo ρ y σ los parámetros de la conductividad y la densidad de la piel u otros tejidos del organismo [14, 39].

$$SAR = \frac{\sigma |H|^2}{\rho} \quad (1.15)$$

En cuanto a los niveles de SAR, el Instituto de Ingenieros Eléctricos Electrónicos (IEEE por sus siglas en inglés) estandarizó los niveles seguros de H presente en un organismo para asegurar la integridad de distintos tipos de tejidos, de fibras y de células.

CAPÍTULO 1. MARCO TEÓRICO

Pertinente a un sistema WPT de acoplo inductivo, en dicho estándar se encuentran dos límites relevantes: H_1 , de $\frac{16.3}{10^{-6}f_o} [\frac{A/m}{Hz}]$, refiriéndose a una intensidad H producida por un campo magnético proveniente del exterior (del inductor transmisor que se encuentra fuera del organismo) y que envuelve completamente un cuerpo humano; y H_2 de $\frac{36.4}{10^{-6}f_o} [\frac{A/m}{Hz}]$ para una intensidad de campo magnético inducida dentro del organismo pero que envuelve sólo un área parcial de cuerpo [53].

En el trabajo de Nima Soltani *et al.* se realizó el análisis matemático para calcular la cantidad de potencia máxima que puede transmitirse en un sistema WTP con acoplo inductivo resonante partiendo H_1 y H_2 . Esta potencia máxima está expresada en 1.16:

$$P_{m\acute{a}x.WPT} = \frac{(1.63A_2N_2)^2(2\pi)^4}{R_{s2}} \Big|_{f_o < f_c} \quad (1.16)$$

donde este análisis se enfoca en obtener la máxima cantidad que puede transmitirse y al mismo tiempo mantener los niveles seguros de SAR. Se puede concluir, que los factores que determinan en mayor medida la cantidad de potencia que puede transmitirse están relacionadas con las propiedades físicas del inductor receptor; su número de vueltas, su resistencia propia y su área.

Esta condición se cumple al tener una frecuencia f_o menor a f_c , descrita por la siguiente expresión:

$$f_c = \frac{\sqrt{3} \cdot R_{s2}}{\pi \cdot L_2} \quad (1.17)$$

A partir de frecuencias mayores a f_c , el factor determinante de la máxima potencia entregable es la frecuencia de trabajo [52].

Radiación de microondas

La radiación de microondas para la energización de IMDs es un método que se encuentra clasificado como una técnica de campo lejano, pues las distancias entre los extremos de la transmisión de potencia pueden ser grandes, desde algunos centímetros hasta algunos metros [50].

Un sistema WPT de radiación de microondas se basa en la irradiación de ondas electromagnéticas en un rango comprendido entre 400MHz y 300GHz. Una propiedad sobresaliente de este método son los tamaños reducidos de las antenas de transmisión y recepción, pues pueden ser del orden milimétrico y micrométrico [18]; en comparación con todas las demás técnicas ya descritas. Estas antenas no afectan en mayor medida el tamaño del implante, permitiendo implantaciones en organismos con espacios extremadamente reducidos.

Los problemas del uso del método de radiación de ondas de súper alta frecuencia (SHF) se presentan en la alta direccionalidad de transmisión que hace propenso que el transmisor y receptor se desalineen y la PTE decrezca brutalmente. Además, existen dos limitaciones que hacen poco factible la implementación de esta técnica para alimentar IMDs, por un lado, la potencia recibida suele ser de pocos mW [18], y por el otro lado, hay una escasez en el análisis de la absorción de energía por los órganos y tejidos, pues al transmitir energía a altas frecuencias, esta es altamente absorbible por el entorno biológico [28].

1.3.3. Amplificadores de potencia para sistemas WPT

Indistintamente de la forma de transferir la energía, sea por energía lumínica, sonora, por campos magnéticos o eléctricos o por radiación, la mayoría de estas metodologías necesitan una etapa de amplificación para suministrar una potencia al elemento que irradia la energía y así habilitar la transmisión desde el exterior del organismo.

Los amplificadores de potencia (PA, por sus siglas en inglés) más usados en las etapas de potencia para sistemas de WPT enfocados a IMDs y AIMDs son los clase C, E, D y F, pues son amplificadores eficientes que pueden entregar una potencia trabajando a frecuencias comprendidas en el rango de las unidades o las decenas de MHz [18,37].

La principal función de un clase C es amplificar una fracción³ de una onda eléctrica que es vista como la señal de entrada al amplificador y a menudo tienen una forma sinusoidal. Una vez que se amplifica esa porción de la señal, a la salida es reconstruida la forma sinusoidal mediante un arreglo resonante LC y de esta forma se obtiene una amplificación de potencia de la onda eléctrica [54].

No obstante, en la práctica es complicado diseñar un PA clase C, debido a que el transistor puede llegar fácilmente a regiones de operación no deseadas, como la saturación para un TBJ o la región óhmica para un FET, sobrecargándolo y por tanto tener un PA deficiente. Además, ante las mínimas variaciones que pudieran presentar la señal de entrada, el suministro de energía eléctrica al sistema y las reactancias⁴ capacitivas e inductivas de los elementos que conforman al PA, hacen que el ángulo de conducción θ cambie drásticamente, lo que hace susceptible un decremento en el rendimiento de esta etapa [25].

Por otro lado, existen otros métodos de amplificación conocidos como Amplificadores de Potencia de Conmutación (APC) y comprenden las clases E, D y F [55]. Estos consisten en

³La fracción de la señal que se amplifica se mide por un ángulo de conducción θ , siendo este menor a π radianes de la señal periódica de entrada.

⁴La reactancia es una definición en eléctrica como la oposición de flujo de corrientes eléctricas alternas a través de capacitores o inductores.

1.3. ESTRATEGIAS DE TRANSFERENCIA DE POTENCIA ELÉCTRICA INALÁMBRICA

hacer trabajar un transistor como un interruptor cerrado ante un valor alto de una señal de entrada de forma cuadrada V_e , y como un interruptor abierto ante un valor bajo de la misma. En términos generales, a la salida del transistor se obtiene un voltaje v_c y una corriente i_c como se ilustra en la figura 1.10. Nuevamente, con un circuito resonante, es posible obtener una señal V_o a la salida con una forma sinusoidales.

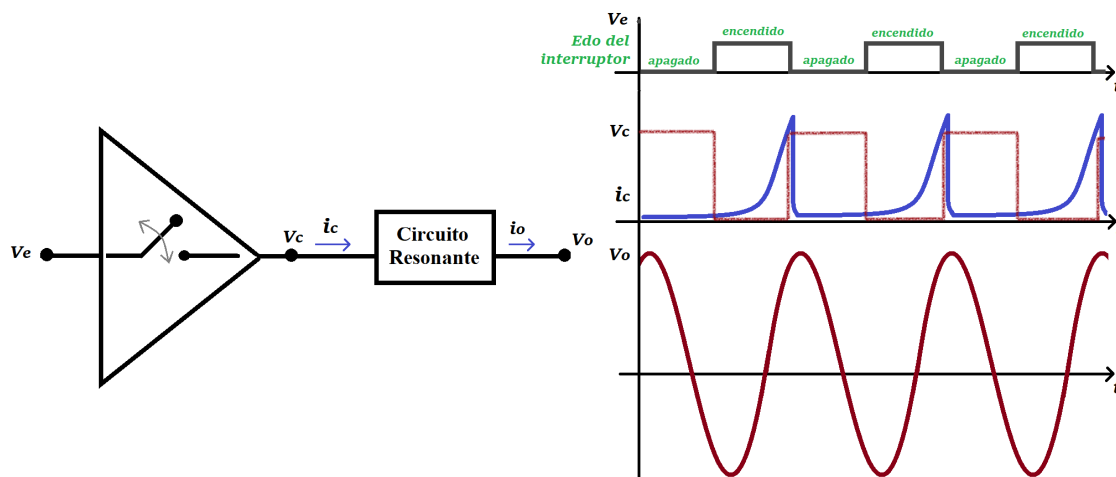


Figura 1.10: Circuito amplificador de potencia clase E.

Un sistema WPT para acoplos inductivos puede implementarse con un amplificador de conmutación clase E, como se ilustrarse en la figura 1.11. Este amplificador tiene un transistor MOSFET canal-N, descrito por $M1$, y es energizado colocando un inductor RFC (del inglés Radio Frequency Choke), entre la fuente de alimentación V_{DD} y la conexión a *Drain* del transistor, esto con el fin de permitir el flujo de la corriente de polarización i_{DD} y de amortiguar las corrientes de alterna generadas por la conmutación.

Se tiene un capacitor C_p para crear un efecto resonante junto con RFC ; el arreglo compuesto por C_1 y L_1 se usa para sintonizar la salida a la frecuencia f_o ; y la señal V_e , acoplada a la compuerta *Gate* a través de C_e , es la entrada para generar la conmutación. También se ilustra el parámetro R_L , que representa el modelo eléctrico del AIMD al que se le entrega la potencia del amplificador mediante el acoplamiento entre un inductor L_1 y L_2 .

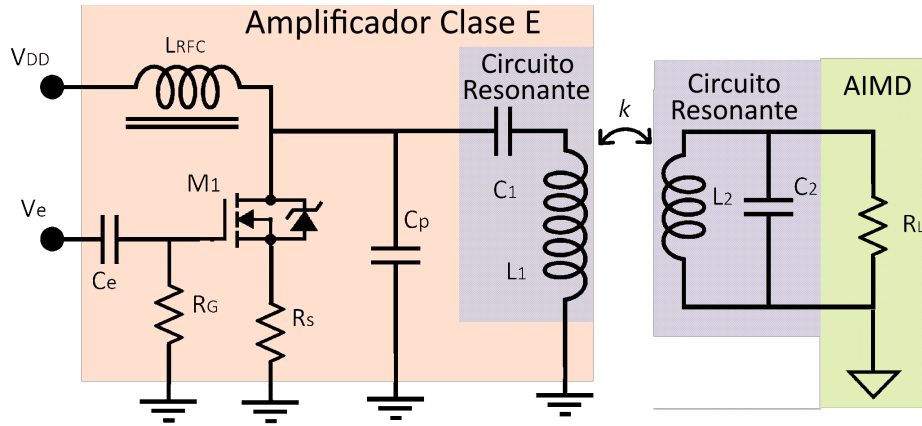


Figura 1.11: Circuito amplificador clase E para un acoplo inductivo.

Para simplificar el diseño del amplificador de clase E para WPT de acoplo inductivo, un modelo equivalente del circuito anterior puede ilustrarse en la figura 1.12, donde la carga R_{Lref} representa el modelo eléctrico del AIMD o IMD vista desde el lado del transmisor.

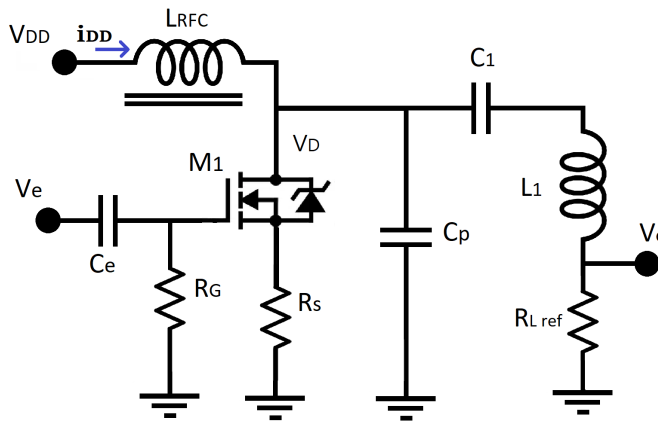


Figura 1.12: Circuito amplificador de potencia clase E.

El valor de esta carga reflejada se expresa en términos de la frecuencia de operación f_o , el factor de acoplamiento k , las inductancias L_1 y L_2 de cada inductor, según [56], esta expresión resulta ser:

1.3. ESTRATEGIAS DE TRANSFERENCIA DE POTENCIA ELÉCTRICA INALÁMBRICA

$$R_{Lref} = \frac{(2\pi f_o)2 \times (k\sqrt{L_1 L_2})2}{R_L} \quad (1.18)$$

En el desarrollo de Narayanamoorthi, *el al.* [57] se expresa la potencia entregada a la carga R_L , siendo para este circuito R_{Lref} , mediante la siguiente ecuación:

$$P_o = \frac{V_z^2}{2R_{Lref}} = \frac{2V_{DC}^2 \text{sen}^2(2\pi\delta) \cdot \text{sen}(2\pi\delta + \varphi)}{\pi^2(1 - \delta)^2 R_{Lref}} \quad (1.19)$$

donde δ es el ciclo de trabajo de la señal de entrada, y comprende un rango de (0,1); φ es el ángulo de desfase y V_z es el voltaje máximo en el inductor L_1 .

Por otro lado, en [58] y en el libro de Krauss [25] se obtuvo la potencia de salida, para un ciclo de trabajo $\delta = 0.5$, de modo que la potencia entregada depende principalmente del voltaje de alimentación al amplificador y la resistencia de carga:

$$P_o = \frac{2}{1 + \pi^2/4} \cdot \frac{V_{DC}^2}{R_{Lref}} \quad (1.20)$$

Por otro lado, el valor de los componentes necesitados en el diseño del amplificador clase E, C_p y C_1 , por facilidad pueden calcularse por métodos numéricos, estas expresiones están descritas en el trabajo de Nathan O. Sokal [58], respectivamente:

$$C_p = \left(\frac{1}{(2\pi f_o) \cdot (1 + \frac{\pi^2}{4}) \cdot (\frac{\pi}{2})} \right) \cdot \left(0.99836 + \frac{0.91394}{Q_{L1}} - \frac{1.0316}{Q_{L1}^2} \right) + \frac{0.7}{L_{RFC} \cdot (\pi f_o)^2} \quad (1.21)$$

$$C_1 = \left(\frac{1}{2\pi f_o R_{Lref}} \right) \cdot \left(\frac{1}{Q_{L1} - 0.1048} \right) \cdot \left(1.002 + \frac{1.0147}{Q_{L1} - 1.7879} \right) \quad (1.22)$$

Cabe señalar que la precisión de estas expresiones se valora tomando en cuenta un ciclo de trabajo del 50%, un factor de calidad del inductor L_1 igual o mayor a 5 y que los componentes electrónicos son ideales.

1.4. Comunicación para AIMDs

Muchas aplicaciones de IMD y AIMD que son implantados requieren de una comunicación con otros dispositivos que normalmente se encuentran fuera del organismo, ya sea para recibir instrucciones y ejecutar tareas, como la dosificación de fármacos y rehabilitación de órganos; o bien para enviar datos biológico para la prevención y control de enfermedades [59].

En esta sección se dará un panorama general de los tipos de comunicación para dispositivos implantados, dando una introducción a las estrategias usadas para la interacción con unidades externas de trabajo.

1.4.1. Tipos de comunicación

En telecomunicaciones el intercambio de información y datos entre dos elementos puede clasificarse por dos tipos de transmisión: uplink y downlink. En la literatura, para la comunicación de un dispositivo implantable, uplink hace referencia a la transmisión de datos procedentes del dispositivo implantado hacia una unidad externa de comunicación. Por el contrario, una transferencia downlink consta del envío de información o instrucciones desde el sistema externo hacia el dispositivo implantado.

Por poner un ejemplo, para sensores biodegradables se han propuestos esquemas simples de comunicación uplink, donde un sistema externo obtiene la medición de variables físicas hechas por el biosensor a través de la reflexión de su impedancia cambiante [12, 60].

Para aplicaciones de AIMDs que integran unidades de adquisición de datos biológicos y sistemas de procesamiento, es preferible que la comunicación sea bidireccional, es decir, que exista una comunicación uplink y downlink. Algunos ejemplos de diferentes implementaciones de comunicación está desglosado en el siguiente cuadro, en el cual se puede comparar el tipo de comunicación implementada, la velocidad de transferencia de la información y la potencia necesaria para el envío de la misma.

Dispositivo	Nivel de Comunicación	Velocidad de transmisión	Potencia de transmisión
Marcapasos	Uplink	10kb/s	$P < 100\mu W$
Estimulador de vejiga	Uplink	300kb/s(máx)	$P_{máx} = 100mW$
Implante Coclear	Downlink	300kb/s	$P_{min} = 10mW$ $P_{máx} = 100mW$
Cápsula Endoscópica	Uplink	20Mb/s(máx)	$P_{máx} = 30 - 570mW$
Prótesis de retina	Downlink Uplink	2 - 40Mb/s 250kb/s(máx)	$P_{máx} = 40 - 250mW$

Cuadro 1.3: Aplicaciones de AIMD con diferentes niveles comunicaciones y velocidad de transferencia de datos.

1.4.2. Comunicación inalámbrica

En muchas ocasiones, los dispositivos implantables necesitan una interacción con otros sistemas para realizar acciones en momentos específicos, por ejemplo, recibir instrucciones o enviar información que puedan ayudar en el tratamiento de enfermedades, en el monitoreo y rehabilitación de órganos; para estos casos, es necesario que los IMDs tengan la facultad de comunicarse con unidades de trabajo externas.

Normalmente, la comunicación que se establece para tener accesibilidad a un AIMD es hecha a través de una unidad externa de comunicación, por ejemplo, un aparato médico dedicado a la recolección de información de AIMDs o dispositivos inteligentes portátiles como un smartphone.

A diferencia de los sistemas WPT, los sistemas de comunicación para IMDs y AIMDs, tienen un enfoque de aplicación orientado a lograr una comunicación confiable desde el

dispositivo implantado hasta el otro extremo de la comunicación y viceversa; es decir, que en el intercambio de información no haya pérdidas de las instrucciones o los datos que sean intercambiados.

El intercambio inalámbrico de información entre una unidad externa y el dispositivo implantado tiene varias ventajas ante una comunicación guiada (o sea, por cables), por ejemplo, es posible implementar AIMDs completamente remotos; se puede implementar una comunicación bidireccional, y es posible habilitar comunicaciones de muy baja potencia de transmisión que se encuentra por debajo de los niveles de SAR mencionados en la sección 1.3.2.

Debido a esto, la comunicación inalámbrica es vista como una alternativa potencial para interactuar con los dispositivos que se encuentran implantados en un organismo. A continuación se hablará de las diferentes metodologías que se emplean para habilitar una comunicación en dispositivos médicos implantables. La clasificación general que se ha usado para describir cada uno de los métodos se basa en la tasa de datos transferidos.

1.4.3. Transferencia de datos a alta frecuencia (HF)

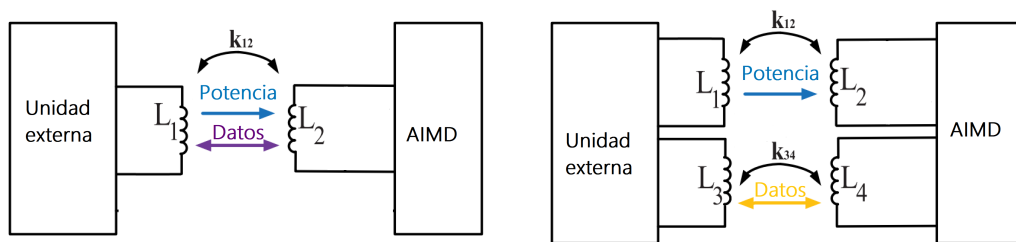
Como se mencionó en la sección 1.3, se puede genera una comunicación inalámbrica hacia AIMDs a través de acoplos capacitivos o inductivos. La transferencia de datos usando acoplos capacitivos ha tenido buenos resultados en cuanto a la fiabilidad y la tasa de intercambio de datos. El esquema general para lograr una transferencia de datos telemétricos es el mismo descrito para WPT. La diferencia radica en la implementación de una señal modulada para obtener los datos.

Las propiedades sobresalientes de un enlace capacitivo para comunicación de datos bidireccionales son: la generación de un campo eléctrico concentrado entre las placas del acoplo, lo que reduce el origen de interferencias electromagnéticas con otros sistemas; se puede usar este sistema para transferir datos de forma bidireccional y potencia eléctrica al mismo tiempo; y se puede conseguir una simpleza en el diseño de un sistema WPT y de comunicación.

El principio del funcionamiento de una comunicación inalámbrica basada en acoplos inductivos es semejante a lo visto en la transferencia de potencia por acoplos inductivos. Sin embargo, para generar una comunicación, las señales presentes en el enlace son cambiantes de acuerdo a la información que se quiere enviar o recibir. Esta técnica es conocida como modulación, y en sistemas de acoplos inductivos suele usarse el cambio de la amplitud de la señal o el cambio de la impedancia reflejada para transferir información de un extremo a otro.

Una comunicación mediante un enlace inductivo es posible integrarse a un sistema de WPT con acoplo inductivo. En la figura 1.13 a) se muestra la idea general de la transmisión de potencia y de datos mediante un sólo acoplo. Sin embargo, la limitante de esta técnica es la velocidad de transferencia de datos, pues se ve limitada por la frecuencia f_o del sistema de potencia [49].

Por otro lado, se pueden tener sistemas independientes de WPT y de comunicación inalámbrica por acoplos inductivos; este esquema está ilustrado en b). La ventaja de esta implementación es que ambos sistemas pueden optimizarse para la realización de cada tarea y también, es posible alcanzar tasas de velocidad en la transferencia de datos sin estar limitado por la frecuencia de operación del sistema WPT [59].



(a) Enlace inductivo de transmisión de potencia y datos.

(b) Enlace inductivo doble o independiente.

Figura 1.13: Esquema de comunicación de datos y transferencia de potencia de un enlace inductivo.

1.4.4. Transmisión de datos por muy alta frecuencia (VHF) y ultra alta frecuencia (UHF)

Otra metodología para habilitar comunicaciones inalámbricas es usando la transmisión de energía mediante ondas de radiofrecuencia. Esto permite enviar y recibir información a distancias mayores e intercambiar mayor cantidad de información, la idea del esquema de comunicación por radiofrecuencia puede ilustrarse en la siguiente figura.

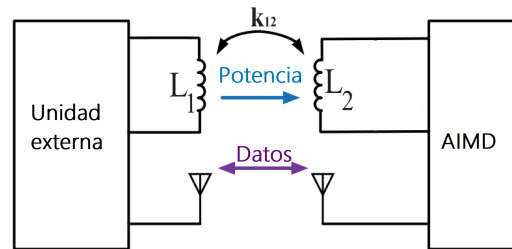


Figura 1.14: Esquema de comunicación de datos por radiofrecuencia y transferencia de potencia con enlace inductivo.

La transferencia de datos a frecuencias muy altas (VHF, por sus siglas en inglés y comprendida entre 30MHz y 300MHz) y ultra altas (UHF, entre 300MHz y 3GHz) son una alternativa para lograr transferir datos digitales a mayor rapidez, alcanzando velocidades de transmisión cercanas o superiores a 1Mbps⁵ [61].

La transmisión de datos por radiofrecuencia (RF) es implementada con antenas de radiofrecuencia, y estas pueden miniaturizarse, de modo que el tamaño del dispositivo puede incluso mantener sus dimensiones físicas.

Dentro de las metodologías sobresalientes para habilitar una comunicación a VHF y UHF para dispositivos implantables, se encuentran Wi-fi, Bluetooth, Zigbee, UWB y NIR. Cada

⁵Un Mbps es la unidad de velocidad de información transmitida dada como un millón de bits por segundo.

una de ellas presenta características que pueden ser aprovechadas para aplicaciones específicas con comunicaciones de alta velocidad [62]. A continuación se dará una introducción a la tecnología Bluetooth y sus alcances en aplicaciones para AIMDs.

1.4.5. Comunicación para un AIMD mediante Bluetooth

Bluetooth es una especificación industrial para redes inalámbricas que tiene alcances de varios metros para una comunicación inalámbrica entre dos dispositivos emparejados. Ha sido usada para interconectar dispositivos inteligentes como smartphones, relojes y computadoras. También se ha implementado para entrelazar sistemas de audio y vídeo; para crear sistemas de automatización de viviendas y edificios; y para crear comunicaciones inalámbricas para un AIMD.

Pensado en reducir los consumos energéticos para la transferencia de datos, la especificación fue extendida para crear un sistema de bajo consumo energético, conocida como Bluetooth Low-Energy (BLE). Al igual que Bluetooth convencional, esta tecnología trabaja a una frecuencia de 2.4GHz y es capaz de transferir datos con una tasa que alcanza 1Mbps; la diferencia radica en la potencia reducida requerida para BLE, siendo la mínima de 5mW [50, 61].

Bluetooth y BLE se consideran seguros para un organismo, pues cumple con los límites especificados por la Comisión Internacional de Protección Contra la Radiación No Ionizante (ICNIRP, por sus siglas en inglés) [63]. Esto puede aprovecharse para implementar aplicaciones médicas de implantes con comunicación Bluetooth, pues no se comprometen los tejidos ni órganos a energías ionizantes con esta especificación de comunicación inalámbrica.

Un inconveniente del uso de BLE es que el ambiente de implantación profunda puede provocar interferencias significativas que propicien la pérdida de información. Como se dio a conocer en el trabajo de Michael J. Christoe *et al.* en 2021 [64], las atenuaciones de RF en un ambiente de implantación son significantes para profundidades mayores a 88mm.

El desarrollo de Michael también apunta a comunicaciones más confiables con la versión

5.0 de BLE, debido a que esta versión maneja una mayor sensibilidad en la recepción de datos para mejorar la recepción de datos sin la necesidad de incrementar la potencia de transmisión, y un sistema inteligente para la corrección de errores (conocido en inglés como FEC) debido a las atenuaciones de la señal provocadas por el ambiente de implantación [64, 65].

Además, se ha demostrado que es posible realizar aplicaciones de implantes médicos con comunicación BLE, como el trabajo desarrollado por Sami *et al.* [66], que usaron el estándar Qi y la BLE 5.0V para tener un mejor control de la carga de baterías para AIMDs; y el trabajo de Junho *et al.* [67], que obtuvo buenos resultados experimentales in vivo con ratas con un dispositivo implantable energizado por WPT y comunicado con BLE en tiempo real para la realización de electrocardiogramas.

Estos resultados prometedores han llevado a proponer sistemas eficientes y confiables para la transmisión de información mediante la especificación de BLE; de manera que, para la actual propuesta de la implementación de un sistema de comunicación para AIMDs, se ha optado por usar esta tecnología.

2. Diseño y construcción de la Estación Electrónica (EPICI)

En el Instituto de Ingeniería de la UNAM se están desarrollando prototipos de dispositivos BioMEMS con la idea de que en un futuro pueda ser posible la implementación de esta tecnología como Dispositivos Médicos Implantables Activos (AIMD).

En el esfuerzo de continuar con el desarrollo de esta tecnología, han surgido dos requerimientos esenciales para el funcionamiento de los prototipos: en primer lugar que sean energizados, y en segundo lugar, que los prototipos puedan enviar los datos biológico generados a partir del monitoreo y la conversión de los mismos, mediante una comunicación establecida por una unidad de trabajo externa.

Ante estas dos necesidades, se ha propuesto como solución la construcción de una Estación Electrónica para satisfacer dichos requerimientos; es decir, se propone diseñar, fabricar y probar un sistema electrónico que suministre potencia eléctrica a prototipos de dispositivos BioMEMS y, de igual forma, sea capaz de habilitar una comunicación con los prototipos para recabar la información procedente de ellos.

2.1. Requerimientos y especificaciones

Resulta importante distinguir que el enfoque primordial en el desarrollo de la propuesta de solución abarca la posible aplicación futura de implantación de los dispositivos BioMEMS, por lo que la implementación de la transferencia de potencia y obtención de datos deben ser adecuada al entorno de un Dispositivo Médico Implantable Activo.

Con lo anterior, la Estación Electrónica debe desarrollarse pensando en que esté dedicada para AIMDs, pero en primera instancia, que cubra las necesidades de los prototipos de dispositivos, pues estos, a pesar de no tener un carácter implantable, son un primer acercamiento a un dispositivo BioMEMS de implantación.

Para satisfacer la primera necesidad de los prototipos de dispositivos BioMEMS, en la sección 1.2 se desglosaron las ventajas de la energización de dispositivos implantables mediante WPT y las potenciales repercusiones de un suministro de potencia por transmisión percutánea hacia AIMDs. Por tanto, con la investigación previa, se propone energizar a los prototipos por una de las técnicas de transferencia de potencia inalámbrica expuestas en la sección 1.3.

La segunda necesidad es la recolección de información del prototipo. En la sección 1.4 se enunciaron las diferentes formas de comunicación y con base en esta investigación, resulta idóneo plantear un sistema de comunicación inalámbrico, donde la comunicación sea robusta para minimizar un margen de error en el intercambio de información entre el prototipo de dispositivo BioMEMS y la Estación Electrónica.

Abordando las restricciones de diseño, el elemento receptor de potencia inalámbrica tiene restricciones en sus dimensiones; esto se ha establecido debido a los objetivos propuestos para futuros desarrollos del equipo de trabajo del Laboratorio de Pruebas no destructivas del Instituto de Ingeniería.

2.1. REQUERIMIENTOS Y ESPECIFICACIONES

Se ha pensado en un receptor planar con un área no mayor a 3.5cm^2 o un receptor cilíndrico con un diámetro no mayor a 3mm y de longitud de 1.8cm , con el fin de desarrollar un acoplo adecuado para una implantación subcutánea futura.

Así mismo, teniendo en cuenta que el grosor de la piel va desde 0.5mm hasta 4mm , la distancia máxima para la transmisión inalámbrica d será igual al grosor máximo de la misma más una separación adicional para aislar el elemento transmisor y evitar un contacto directo con el organismo; esta distancia adicional se propone de 4 a 6mm , por lo que d tendrá un rango de 10 a 8 milímetros.

En cuanto al prototipo de dispositivo BioMEMS, se ha preparado una tarjeta electrónica que tiene un microcontrolador CC2640R2F de bajo consumo de potencia. En la figura 2.1 se pueden apreciar las dimensiones de la tarjeta electrónica así como las terminales eléctricas del microcontrolador.

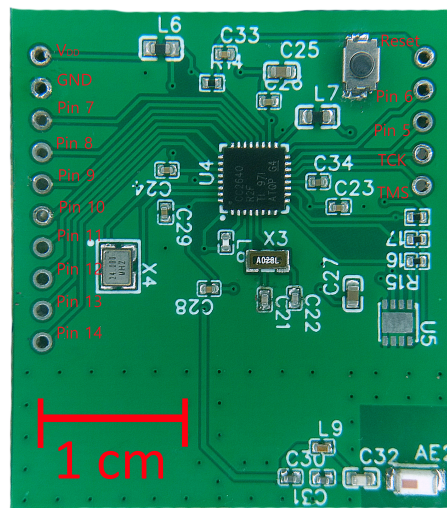


Figura 2.1: Descripción gráfica del prototipo de Dispositivo BioMEMS.

Como introducción de las características eléctricas de este prototipo, su demanda de potencia eléctrica, se ha medido aplicando un voltaje nominal de 3.3V , teniendo como resultado

CAPITULO 2. DISEÑO Y CONSTRUCCIÓN DE EEPICI

un consumo promedio de $10mW$ en su modo *Activo*; en otras palabras, esta es la potencia que demanda el prototipo para funcionar (el modo activo incluye el uso de BLE del bloque ilustrado en la figura 2.2).

Además, este microcontrolador cuenta con una capa de hardware independiente de baja energía (ilustrado por el bloque rosa en la figura 2.2) que puede operar en condiciones de ultra baja potencia (y puede realizar tareas sencillas como la conversión analógica digital de una variable física); para este caso, las mediciones de potencia realizadas corroboran que el consumo en su modo de operación *Standby* es de $4.95\mu W$ en promedio.

Por otro lado, se ha pensado que además de energizar al prototipo directamente con la transferencia de potencia desde la Estación Electrónica, se mantenga operativo de manera autónoma con un tiempo acotado sin el suministro WPT directamente. Para que el prototipo pueda tener un tiempo limitado de operatividad autónoma, éste puede energizarse mediante un capacitor que almacene la energía suficiente proveniente de la transferencia de potencia de la Estación Electrónica.

Por último, este microcontrolador tiene incorporado el hardware necesario para implementar una comunicación inalámbrica de radiofrecuencia de manera personalizada, este hardware está dedicado para crear una comunicación mediante el estándar Bluetooth de baja energía (BLE).

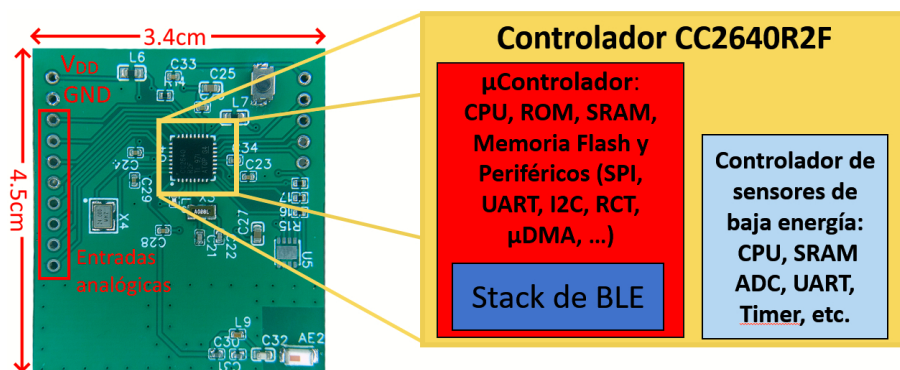


Figura 2.2: Descripción de la estructura del microcontrolador.

2.2. Diseño del sistema de Transferencia de Potencia Inalámbrica

Respecto al sistema WPT, se necesita diseñar dicho sistema con base en las características eléctricas y físicas del prototipo para que éste pueda mantenerse operativo. También se debe de considerar la demanda eléctrica para cargar el capacitor que funcionará como almacenador eléctrico del prototipo; en ese sentido, con estas dos ideas principales se puede definir la potencia total que tiene que suministrar la Estación Electrónica.

Partiendo de la operatividad autónoma del prototipo, se plantea que éste tenga una duración en modo *Standby* de hasta 30 minutos, para que éste pueda tomar mediciones de variables físicas como la temperatura ambiental. Para hacer posible esto, se plantea agregar un capacitor electrolítico que almacene la energía suficiente y que pueda suministrar potencia hacia el prototipo.

Obteniendo el almacenamiento de carga eléctrica consumida en 30 minutos por el prototipo en modo *Standby*, es posible obtener la capacitancia necesaria que debe tener el capacitor para lograr la operatividad autónoma del prototipo. A partir de la ecuación (1.1), se puede obtener dicho valor:

$$ACE = \frac{4.95\mu W}{3.3V} \times 0.5h = 0.75\mu Ah \quad (2.1)$$

y la conversión de esta cantidad de energía almacenada a carga eléctrica, por la ecuación (1.2), es:

$$q_e = \frac{0.75 \times 1C}{270\mu Ah} = 2.77mC \quad (2.2)$$

así, obteniendo la capacitancia del capacitor con la ecuación (1.3), esta resulta ser de:

$$C = \frac{2.77mC}{3.3V} = 841\mu F \quad (2.3)$$

Considerando la propuesta de energizar al prototipo y cargar eléctricamente a un capacitor, el escenario de mayor demanda energética se da al suministrar potencia para el prototipo cuando se encuentra en el estado *Activo* y, al mismo tiempo, al cargar el capacitor cuando éste se encuentra totalmente descargado. Es por esto que es necesario hacer un análisis de la potencia necesaria para cargar al capacitor y sumarla a la potencia que demanda el prototipo; y de esta manera obtener la potencia máxima que necesita entregar la Estación Electrónica.

Para cargar el capacitor, se propone el circuito de la figura 1.1. Proponiendo un tiempo de carga de 0.6 segundos, es necesario recurrir a la constante de tiempo τ ; evaluando la carga eléctrica obtenida anteriormente, el voltaje aplicado y la capacitancia encontrada, con la expresión (1.5) se tiene:

$$\tau = RC = -\frac{0.6\text{seg}}{\ln\left[1 - \frac{3.2\text{mC}}{1\text{mF} \times 3.3\text{V}}\right]} = 0.1716 \text{ [segundos]} \quad (2.4)$$

con lo que, al despejar R , su valor es:

$$R = \frac{\tau}{C} = \frac{0.1716\text{s}}{841\mu\text{F}} = 204\Omega \quad (2.5)$$

La resistencia R_{carga} del circuito de la figura 1.1 será aproximadamente a 204Ω , pues cuando éste se encuentra totalmente descargado, la resistencia en serie (R_c) del modelo del capacitor electrolítico es muy baja (siendo valores en el orden de decenas de mili-Ohms).

Como se mencionó anteriormente, la máxima corriente que debe de ser entregada para satisfacer el correcto comportamiento del circuito propuesto por la figura 1.1 está descrito por la simplificación de (1.7), siendo este valor de:

$$I_{\text{máx.carga}} = \frac{3.3\text{V}}{204\Omega} = 16.17\text{mA} \quad (2.6)$$

Así mismo, en términos de la potencia demandada para cargar el capacitor, está resulta ser:

$$P_{\text{máx.carga}} = 3.3\text{V} \times 16.17\text{mA} = 53.4\text{mW} \quad (2.7)$$

2.2. DISEÑO DEL SISTEMA DE TRANSFERENCIA DE POTENCIA INALÁMBRICA

Para obtener la potencia total que se debe suministrar al prototipo y al sistema de carga del capacitor, hay que sumar el consumo eléctrico promedio del prototipo (recordando que es de 10mW) a la potencia $P_{m\acute{a}x.carga}$; por lo que la potencia total del sistema de Transferencia de Potencia Inalámbrica (P_{WPT}), es de:

$$P_{WPT} = 10mW + 53.4mW = 63.4mW \quad (2.8)$$

Por último, con la expresión (1.13), la carga equivalente del sistema esta descrita como;

$$R_L = \frac{(V_{aplicado})^2}{P_{WPT}} = \frac{3.3V^2}{63.4mW} = 171.8\Omega \quad (2.9)$$

En resumen, con la cuantificación de la potencia demandada, las posibles técnicas para implementar una transferencia de potencia inalámbrica son de acoplo acústico e inductivo, ya que está demostrado, como se expuso en la sección 1.3, que son capaces de entregar cantidades de potencias mayores a la potencia máxima obtenida $P_{m\acute{a}x.WPT}$, para dimensiones de antenas receptoras de tamaños menores a la restricción antes mencionada.

También es posible descartar el uso de un acoplo óptico y de radiofrecuencia, debido a que actualmente estas estrategias no pueden entregar estas cantidades de potencia eléctrica teniendo en cuenta las dimensiones de las antenas receptoras. Por último, es inviable la implementación de un acoplo capacitivo debido a la separación de varios milímetros entre el acoplo que este desarrollo requiere.

Para concluir, teniendo en cuenta que el desarrollo de sistemas WPT con acoplos inductivos implican menores costos de producción y mayor facilidad de manufacturación en comparación con los costos más elevados al implementar una transmisión de potencia con acoplos acústicos [68], se ha concluido diseñar, desarrollar y probar un sistema WPT por acoplo inductivo.

2.2.1. Diseño del acoplo inductivo

Para diseñar el sistema de Transmisión de Potencia Inalámbrica, es necesario comenzar por el diseño del acoplo inductivo resonante. En este trabajo, se tomarán como referencia los desarrollos teóricos realizados por Soltani *et al.* [52] y por Mutashar *et al.* [69] expuestos en la sección 1.3.

El acoplo inductivo resonante constará principalmente de dos arreglos LC, uno del lado de transmisión y otro del lado de recepción. Para su diseño, se hará énfasis en dos características principales: en que el elemento receptor L_2 , no debe exceder las dimensiones establecidas, respetando los requerimientos de implantación antes mencionados; y en la potencia que debe de entregar el propio sistema, asegurando que el enlace pueda transferir la potencia calculada previamente.

Siguiendo el análisis teórico de Soltani, la geometría de los inductores puede ser rectangular o cuadrada, o bien circular de acuerdo con el análisis de Mutashar. Para reducir las dimensiones por lado de la antena receptora, se propone una geometría cuadrada, con una longitud menor a 18mm por lado, para no superar la restricción en las dimensiones del elemento receptor.

El método para obtener los parámetros del inductor transmisor L_1 y del inductor receptor L_2 se describe en el diagrama de flujo 2.3. Con este proceso iterativo es posible diseñar los dos inductores adecuados para la transmisión, es decir, que cumplan las restricciones de dimensiones físicas, que puedan transmitir o captar la potencia necesaria, y que puedan ser implementados por algún equipo disponible en el laboratorio de pruebas no destructivas del Instituto de Ingeniería¹.

¹Se ha tomado en cuenta usar la fresadora MODELA MDX-50 para el desarrollo de los inductores L_1 y L_2 en caso de no encontrarlos en el mercado.

2.2. DISEÑO DEL SISTEMA DE TRANSFERENCIA DE POTENCIA INALÁMBRICA

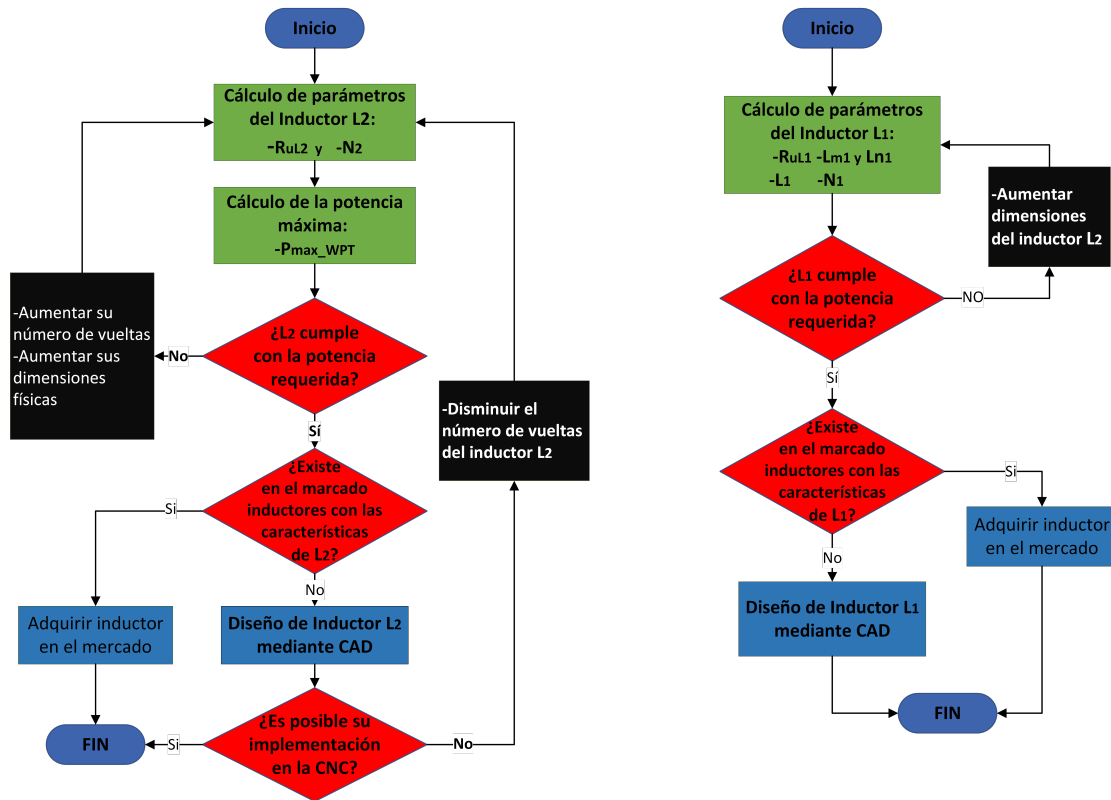


Figura 2.3: Diagrama de flujo del diseño de Inductores L1 y L2.

Para obtener los valores teóricos del diseño del acoplo inductivo, se usó el programa Matlab, esto con el fin de facilitar las iteraciones necesarias descritas por el diagrama de flujo anterior. En el anexo A.1 se encuentra el código donde se calculan las propiedades eléctricas y físicas de los inductores, de la misma manera, se cuantifica la potencia máxima entregable y la eficiencia en función de la frecuencia de operación del sistema.

Retomando la idea de fabricar los elementos receptor y transmisor de forma rápida, barata, confiable y funcional, se ha propuesto desarrollar inductores a partir de placas fenólicas, usadas para la fabricación de circuitos impresos. Estas placas tienen un espesor de la capa de cobre de $t_x = 35\mu m$ y una resistividad del conductor de $\rho = 1.71 \times 10^{-8}\Omega m$.

Para obtener las propiedades físicas y eléctricas del inductor L_2 , se ha tomando como referencia el modelo eléctrico de la figura 1.7, siendo la resistencia intrínseca del inductor

receptor R_{uL2} de:

$$R_{uL2} = \frac{4 \times 1.71 \times 10^{-8}}{0.54 \times 35\mu m} = 3.6m\Omega \quad (2.10)$$

El número de vueltas para el inductor receptor está descrito en el desarrollo [52]; donde se tiene:

$$N_2 = \sqrt{\frac{R_L}{R_{uL2}}} = \sqrt{\frac{172\Omega}{3.6m\Omega}} = 218 \text{ [vueltas]} \quad (2.11)$$

Con el equipo disponible en el laboratorio, resulta imposible hacer un inductor de geometría planar cuadrada en espiral, sobre una tabla fenólica, de máximo 18mm de longitud por lado y con 218 vueltas. Sin embargo, se buscó obtener la mayor cantidad de vueltas posibles respetando estas dimensiones.

Usando el proceso iterativo del diagrama de flujo 2.3, se llegó a obtener hasta 20 vueltas para un inductor de 17.5mm de dimensión por lado. Además, con la orientación del desarrollo [52] se obtuvieron las dimensiones físicas del inductor: $d_{ext} = 17.5mm$ y $d_{int} = 5.13mm$; con relación de diámetros $\phi_2 = 0.54$ y diámetro promedio de $11.3mm$.

Obteniendo el valor de la inductancia de L_2 , con la ecuación (1.9), el resultado teórico es:

$$L_2 = \frac{C_1\mu(20)^2(11.3mm)}{2} \cdot \left[\ln\left(\frac{1.27}{\phi_2}\right) + 0.18\phi_2 + 0.13(\phi_2)^2 \right] = 5.36\mu H \quad (2.12)$$

Con los parámetros obtenidos de L_2 , la potencia máxima permitida por este elemento como receptor de potencia, a partir de la ecuación (1.16), es:

$$P_{\acute{m}ax_WPT} = \frac{(1.63 \times (0.0175m)^2 \times 20)^2 (2\pi)^4}{(3.6m\Omega \times 20^2)} \Bigg|_{f_o \leq f_c} = 107mW \quad (2.13)$$

Teniendo en cuenta una frecuencia límite f_c :

$$f_c = \frac{\sqrt{3}}{2\pi} \times \frac{172\Omega + 3.6m\Omega}{5.36\mu H} = 8.8MHz \quad (2.14)$$

2.2. DISEÑO DEL SISTEMA DE TRANSFERENCIA DE POTENCIA INALÁMBRICA

En la gráfica 2.4, como resultado del código de Matlab mencionando anteriormente, se puede apreciar la potencia máxima que puede ser entregada al prototipo de dispositivo BioMEMS hasta la frecuencia f_c , ver la sección 1.3.2.

De la misma manera, se ha graficado la eficiencia teórica que podría tener el enlace o acoplo inductivo (ver la curva azul). A pesar de que el enfoque de este trabajo no es obtener una alta eficiencia en la transmisión de potencia inalámbrica, cabe resaltar que es deseable que la frecuencia de trabajo del sistema sea inferior pero cercana a f_c , para que sea factible la transferencia de potencia de un extremo al otro.

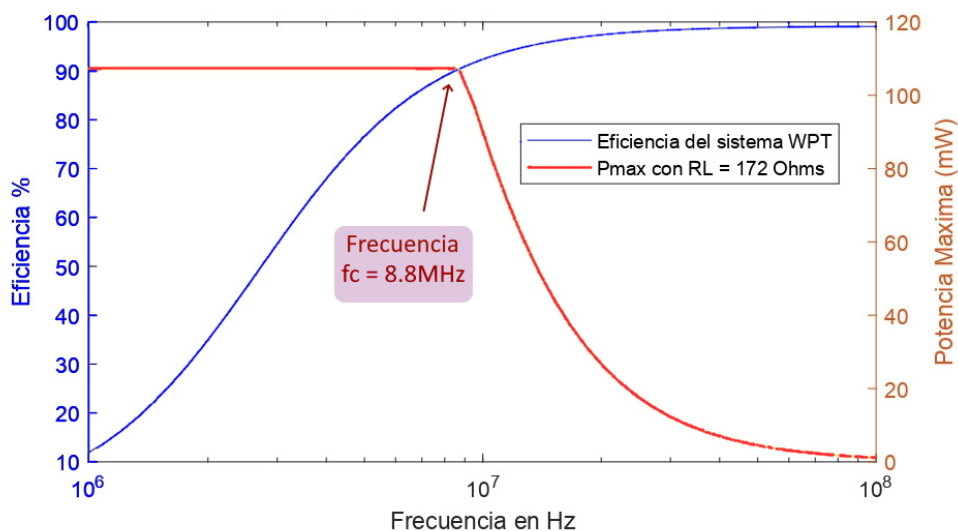


Figura 2.4: Gráfica de la potencia y eficiencia del acoplo inductivo.

Por otro lado, para obtener las dimensiones físicas y propiedades eléctricas del inductor L_1 como elemento transmisor del sistema WPT, de [52] obtenemos su diámetro externo D_{ext} y el diámetro interno D_{in} a partir de la distancia máxima de separación entre L_1 y L_2 .

Es decir, el diámetro externo es 5.24 veces la distancia de separación d y el diámetro interno es de 0.13 veces la misma separación d . Por tanto, las dimensiones de L_1 quedan como: $D_{ext} = 5.24 \cdot 10mm = 52.4mm$ y $D_{int} = 0.26 \cdot 10mm = 2.6mm$. Por lo que ϕ_1 es de 0.92 y su diámetro promedio de 27.3mm.

Nuevamente, el número de vueltas del inductor puede determinarse por la relación de la impedancia real de salida del suministro de potencia hacia L_1 y la resistencia intrínseca del inductor R_{uL1} , siendo esta de:

$$R_{uL1} = \frac{4 \times 1.71 \times 10^{-8}}{0.92 \times 35mum} = 2.1m\Omega \quad (2.15)$$

por tanto, el número de vueltas para el inductor L_1 es de:

$$N_1 = \sqrt{\frac{R_s}{R_{L1}}} = \sqrt{\frac{1\Omega}{2.1m\Omega}} = 21.6 \approx 22 \text{ [vueltas]} \quad (2.16)$$

Una vez obtenido el número de vueltas, la geometría del inductor y sus dimensiones, es posible calcular la inductancia teórica de este elemento a partir de (1.9), siendo de un valor:

$$L_1 = \frac{C_1\mu(11)^2(27.3mm)}{2} \cdot \left[\ln\left(\frac{1.27}{\phi_1}\right) + 0.18\phi_1 + 0.13(\phi_1)^2 \right] = 11\mu H \quad (2.17)$$

2.2.2. Diseño del circuito resonante

Hasta el momento se han obtenido los parámetros de los elementos transmisor y receptor para su fabricación. Sin embargo, para obtener un acoplo inductivo de mayor desempeño, se necesita hacer un circuito resonante para cada uno de estos elementos. Como se vio en la sección 1.3, para hacer un acoplo inductivo como resonante es suficiente hacer un arreglo como los de la figura 1.8.

Como la frecuencia de resonancia define la sintonización de los circuitos resonantes, es necesario conocer su valor antes de diseñar el circuito resonante para cada lado de la transmisión. En este caso, la frecuencia de resonancia f_o puede ser propuesta entre un rango comprendido de 1MHz hasta f_c , esto haciendo énfasis en que la transferencia sea factible de acuerdo a la gráfica obtenida de la figura 2.4.

Proponiendo una frecuencia f_o de 2.4MHz, el arreglo LC paralelo para L_2 y el arreglo LC serie para L_1 está descrito por la ecuación (1.17), de modo que para sintonizar al inductor L_1 a la frecuencia f_o , el capacitor C_1 será de:

2.2. DISEÑO DEL SISTEMA DE TRANSFERENCIA DE POTENCIA INALÁMBRICA

$$C_1 = \frac{1}{L_1(2\pi f_o)^2} = \frac{1}{11\mu H(2\pi 2.4MHz)^2} = 0.4nF \quad (2.18)$$

Mientras que el capacitor C_2 , para sintonizar al inductor L_2 corresponde a:

$$C_2 = \frac{1}{L_2(2\pi f_o)^2} = \frac{1}{5.36\mu H(2\pi 2.4MHz)^2} = 0.82nF \quad (2.19)$$

Como datos adicionales, el factor de calidad teórico de L_1 y de L_2 son de:

$$Q_1 = \frac{2\pi 2.4MHz \cdot 11\mu H}{2.1m\Omega \times 22} = 167 \quad Q_2 = \frac{2\pi 2.4MHz \times 5.36\mu H}{3.6m\Omega \times 20^2} = 55 \quad (2.20)$$

2.2.3. Diseño del rectificador-regulador de voltaje para el prototipo

Debido a que el prototipo de dispositivo BioMEMS debe ser energizado con un voltaje dc, es necesario que el voltaje inducido por el elemento receptor sea rectificado y regulado; por ello se plantea un circuito rectificador y regulador de voltaje, para que el receptor entregue el voltaje de 3.3V como máximo.

El esquema básico del circuito rectificador-regulador de voltaje puede ilustrarse a continuación. Como consideraciones de diseño, para la rectificación de onda completa, los diodos rectificadores son tipo Schottky; el capacitor de filtrado es para minimizar el rizo del voltaje directo; y el regulador de voltaje LDO es colocado para limitar el voltaje que se entregará al prototipo.

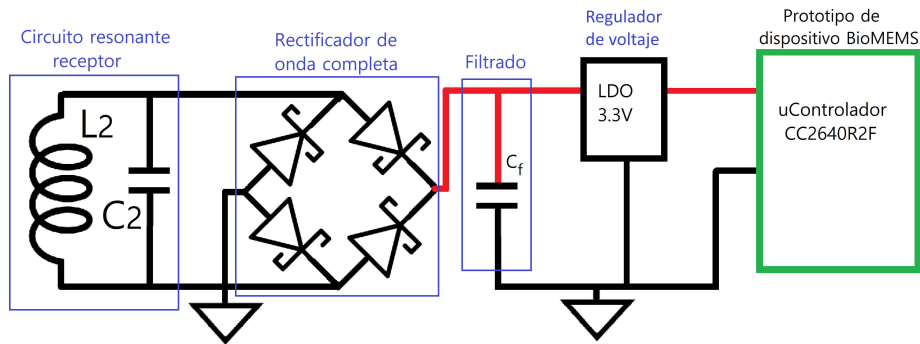


Figura 2.5: Circuito rectificador-regulador de voltaje para alimentar al prototipo.

2.2.4. Diseño de la etapa de potencia

Una vez obtenido el diseño del acoplo inductivo resonante, se necesita una etapa de amplificación para entregar la energía suficiente al transmisor y así lograr la transmisión de potencia hasta el extremo del receptor. Es por esto que esta sección se dedicará al diseño de un amplificador de potencia para energizar al elemento transmisor L_1 .

Como se vio en la sección 1.3.3, esta etapa puede implementarse con un amplificador de potencia de Conmutación. Para este trabajo se ha elegido usar un amplificador de clase E, debido a la facilidad de manejar una señal de entrada cuadrada, por ejemplo, una señal PWM proveniente de un dispositivo de Control que probablemente se necesite más adelante.

En primer lugar, la potencia que debería entregar el amplificador debería ser la misma que la potencia que se entregue al lado receptor, siendo esta de $P_{WPT} = 63.4mW$. No obstante, para poder entregar dicha potencia, el amplificador deberá entregar una cantidad mayor para compensar las pérdidas de potencia que habrá en la transmisión de energía a través del sistema WPT.

Estas pérdidas se originan por los factores que son descritos a continuación y para cada una de ellas se propone una cantidad de potencia complementaria para compensar dicha pérdida:

2.2. DISEÑO DEL SISTEMA DE TRANSFERENCIA DE POTENCIA INALÁMBRICA

- Por las reactancias generadas en cada circuito resonante real del sistema WPT, pues en la práctica es complejo alcanzar una sintonización perfecta y no tener este tipo de pérdidas; por lo que se agrega un 30 % de la potencia P_{WPT} .
- Por la absorción de energía SAR en los tejidos, adicionando una cantidad adicional estimada² de un 10 % de P_{WPT} .
- Por la eficiencia del acoplo inductivo (siendo $\eta \leq 0.5$ acorde a [56]). Teniendo una estimación de η , por experimentación empírica, de este acoplo inductivo de aproximadamente el 30 %, es necesario añadir³ un 230 %.

Es por ello que se estima que el amplificador entregue una potencia, según lo anterior de:

$$\begin{aligned}
 P_{apc} &= 63.4mW + (63.4mW \cdot 0.3) + (63.4mW \cdot 0.1) + (63.4mW \cdot 2.3) \\
 P_{apc} &= 234mW
 \end{aligned}
 \tag{2.21}$$

Además, se propone que el amplificador opere dentro del rango máximo de su operación, lo que requiere una sobre dimensión de esta potencia obtenida. Se propone que el rango máximo de operación sea del doble de P_{apc} , por lo que el amplificador debe de ser capaz de entregar hasta $P_{APC} = 470mW$.

Para el diseño del amplificador clase E, se referencia el circuito del amplificador clase E para un acoplo inductivo, descrito en las figuras 1.11 y 1.12. Primeramente, usando un estimado de 0.2 como factor de acoplamiento⁴ y la ecuación (1.18), la carga equivalente $R_L = 171.8\Omega$ se ve reflejada en el amplificador con un valor de:

$$R_{Lref} = \frac{(2\pi 2.4MHz)^2 (0.2^2 5.36\mu H \cdot 11\mu H)}{171.8 + R_{uL2}} = 3.73\Omega
 \tag{2.22}$$

²Esta cantidad se ha estimado a partir simulaciones de los tejidos de la piel hechos en [70] y [71].

³El cálculo matemático se encuentra descrito en el anexo B.1.

⁴Se ha estimado este valor con respecto al trabajo [72], donde las condiciones del acoplo inductivo, como la dimensión de L_1 respecto a la dimensión de L_2 , la geometría cuadrada en espira y la distancia d tienen valores similares a los parámetros obtenidos en el análisis de este trabajo.

CAPITULO 2. DISEÑO Y CONSTRUCCIÓN DE EEPICI

De este modo, a partir de la ecuación (1.20), se deberá suministrar un voltaje al amplificador de:

$$V_{DD} = \sqrt{470mW \times 3.73\Omega \cdot \frac{1 + \pi^2/4}{2}} = 1.96V \approx 2V \quad (2.23)$$

Teniendo en cuenta una demanda del amplificador de 235mA y permitiendo una variación de corriente alterna de hasta 15mA, debidas a la conmutación del amplificador, la atenuación de estas corriente de ac es descrita como:

$$dB = 20\log\left(\frac{235mA}{15mA}\right) = -24dB \quad (2.24)$$

A partir de una simulación, se obtuvo esta supresión de corrientes con un inductor RFC mayor o igual a $100\mu H$.

Fue calculado el capacitor shunt C_p a partir de (1.21), y el capacitor del arreglo resonante C_1 con la expresión (1.22). Los valores son los siguientes.

$$C_p = \left(\frac{1}{(2\pi 2.4MHz) \cdot (1 + \frac{\pi^2}{4}) \cdot (\frac{\pi}{2})} \right) \cdot \left(0.99836 + \frac{0.91394}{167} - \frac{1.0316}{167^2} \right) + \frac{0.7}{100\mu H \cdot (\pi f_o)^2}$$

$$C_p = 1.01nF$$

$$C_1 = \left(\frac{1}{2\pi 2.4MHz \cdot 12.47\Omega} \right) \cdot \left(\frac{1}{167 - 0.1048} \right) \cdot \left(1.002 + \frac{1.0147}{167 - 1.7879} \right)$$

$$C_1 = 32pF$$

El transistor MOSFET de canal-N QS6k1 tiene las características adecuadas para el desarrollo de este amplificador de potencia, pues es un transistor diseñado para aplicaciones de conmutación, puede trabajar con señales de entrada con amplitudes mínimas de 1.2V y hasta una frecuencia de operación de aproximadamente 30MHz.

2.2.5. Límites de potencia del amplificador debido a la absorción de energía por los tejidos de la piel

Nuevamente se necesita asegurar que se estén respetando los niveles de SAR para esta transmisión de potencia. Para este desarrollo se considerará H de $\frac{36.4A/m}{f_o}$ (establecido por el estándar [53]) como el límite para una exposición local; esto ya que L_1 , debido a sus dimensiones obtenidas (0.055m por lado) no es capaz de generar un campo magnético que envuelva completamente el cuerpo humano.

Partiendo de esta restricción, la densidad de campo magnético máxima permitida que puede producir el inductor L_1 es de:

$$B_{m\acute{a}x} = \frac{36.3A/m \times \pi 4 \times 10^{-7}}{2.4MHz} = 19\mu T \quad (2.25)$$

por lo que el flujo de campo magnético del transmisor L_1 resulta ser de:

$$\varphi_{max} = B_{m\acute{a}x} \times d_{ext}^2 = 19\mu T \times (0.055m)^2 = 57.7nWb \quad (2.26)$$

y finalmente, el voltaje pico máximo permitido en el transmisor L_1 es de:

$$\begin{aligned} v_{max_L1} &= \varphi \times (N_1 \cdot 2\pi \cdot f_o) \\ v_{p_max_L1} &= 57.7nWb \times (22 \cdot 2\pi \cdot 2.4MHz) \times \sqrt{2} \\ v_{p_max_L1} &= 26.76V \quad \rightarrow \quad V_{pp_max_L1} = 53.5V \end{aligned} \quad (2.27)$$

Con lo anterior, se puede afirmar que para la implementación de un amplificador de WPT con acoplo inductivo enfocado a energizar un IMD o AIMD y usando el inductor transmisor previamente diseñado, el voltaje en este elemento no debe de superar los $53.5V_{pp}$.

2.3. Diseño del sistema de Comunicación y de Control

2.3.1. Diseño de la Comunicación Inalámbrica

Para recabar los datos provenientes del prototipo de dispositivo BioMEMS, es necesario hacer una comunicación inalámbrica entre éste y la Estación Electrónica. En la sección 2.1 se mencionó que el prototipo tiene incorporado el Stack para habilitar una comunicación mediante Bluetooth de baja energía (BLE). Con este recurso, es posible hacer una comunicación Bluetooth remota y de corto alcance pero robusta para la obtención de los datos biológicos.

Esta etapa de comunicación inalámbrica constará de dos elementos esenciales. El primero, como parte de la Estación Electrónica, estará encargado de hacer la comunicación Bluetooth y solicitar los datos con y al prototipo de dispositivo BioMEMS. Por otro lado, el prototipo será el segundo elemento listo para conectarse con la Estación Electrónica y enviar la información o los datos cuando sean solicitados.

Para implementar esta comunicación, se propone que la Estación Electrónica controle la comunicación, por lo que deberá buscar al prototipo, y si es necesario, establecer una comunicación con él, una vez realizado esto, podrá solicitar los datos provenientes del propio prototipo. En el diagrama de flujo de la figura 2.6 se describe ordenadamente los pasos que la Estación debe ejecutar para realizar una comunicación inalámbrica exitosa.

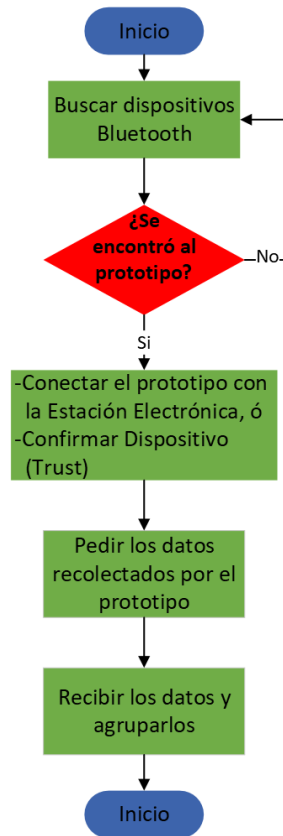


Figura 2.6: Diagrama de flujo del proceso de la Estación Electrónica para una comunicación Bluetooth con el prototipo.

Por otro lado, para concretar la comunicación inalámbrica, el microcontrolador del prototipo de dispositivo BioMEMS debe contener un firmware⁵ para que la Estación Electrónica pueda establecer la comunicación entre ellos. También es necesario que este firmware realice las siguientes tareas: obtener valores de una variable física mediante la conversión Analógica-Digital, almacenar datos digitales en la memoria Flash del microcontrolador y, cuando lo requiera la Estación, permitir el envío de los datos almacenados.

Es por ello que en el siguiente diagrama de flujo se describen, a grandes rasgos, los procesos que el microcontrolador de prototipo debe de ejecutar, siendo estos: la conversión

⁵Entendiendo que un firmware es un tipo de programa que proporciona el control para el hardware específico del dispositivo.

CAPITULO 2. DISEÑO Y CONSTRUCCIÓN DE EEPICI

una variable física, para almacenar uno o más datos de forma digital; la configuración del Stack Bluetooth, para poder habilitar una comunicación inalámbrica BLE; y la transmisión de datos recolectados, cuando estos sean solicitados por la Estación.

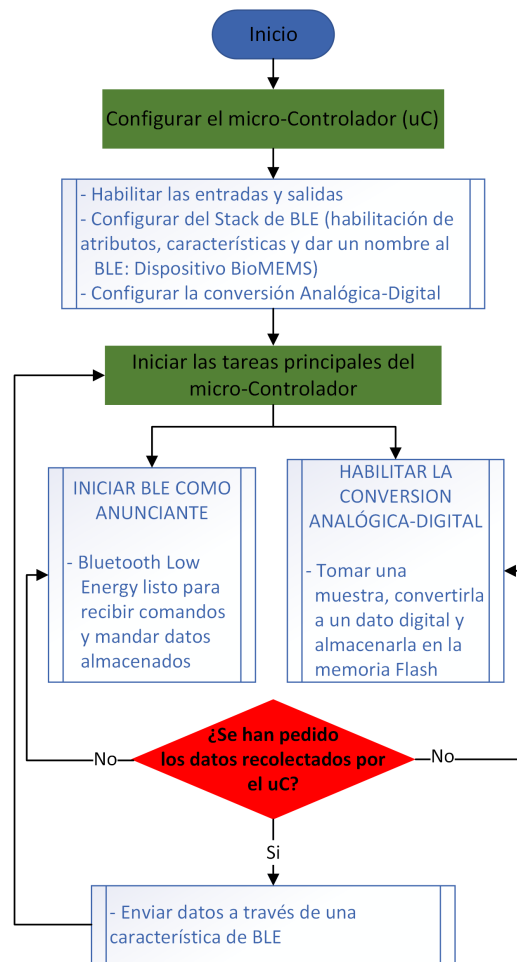


Figura 2.7: Diagrama de flujo del programa para el microcontrolador del prototipo del dispositivo BioMEMS.

2.3.2. Propuesta del sistema de Control

Hasta el momento, las dos etapas anteriores se han diseñado como sistemas independientes. Nos obstante, se puede emplear un dispositivo electrónico como una Unidad de Control que ejecute y controle estas dos etapas, de esta manera es posible consolidar el concepto de una Estación Electrónica de Potencia Inalámbrica y de Comunicación Inalámbrica (EEPICI).

Para tener un sistema de Control se puede usar un microcontrolador, un FPGA o una microcomputadora. Una Raspberry Pi 3, por ejemplo, es una micro-Computadora tiene varios periféricos⁶ integrados de interés para este trabajo: tiene Ethernet y/o Wifi, y Bluetooth, mismos que pueden ser usados para conectar la Estación a Internet y para crear la comunicación con el prototipo, respectivamente. Por otro lado, también cuenta con funciones alternas de fácil acceso, como salidas PWM de hasta 9.4MHz, que puede aprovecharse para habilitar fácilmente el suministro de potencia.

Considerando lo anterior, se propone usar una Raspberry Pi (Raspi) para implementar el sistema de control, pues con ella es sencillo establecer comunicaciones inalámbricas y habilitar sistemas que trabajan a muy altas frecuencias, como el amplificador de potencia diseñado en la sección 2.2.4. Además, al ser un dispositivo económico que puede ejecutar múltiples tareas a la vez, y por su potencia de procesamiento, es posible añadir tareas más complejas a la EEPICI, en caso de requerirlo en un futuro.

⁶Haciendo referencia que un periférico es una parte de un dispositivo que no forma parte de su unidad central.

2.4. Diseño de alimentación eléctrica para la Estación

Para energizar la EEPICI, se necesita conocer el voltaje y la potencia adecuados para energizar la Raspi y el amplificador de potencia. En primer lugar. La Raspi debe ser alimentada con un voltaje continuo de 5.1V, y su consumo de potencia ronda entre los 1.5 y 5 Watts, mientras que el amplificador de potencia debe de ser alimentado con un voltaje $V_{DD} = 2.0V$, con un consumo máximo de $470mW$.

Con esta información, se propone usar una fuente de voltaje directo de 5V para alimentar al sistema de control e implementar un convertidor de voltaje Step-Down para obtener un voltaje $V_{DD} = 2V$. Usando esta forma de alimentación, la demanda de corriente total de la EEPICI es de:

$$I_{EEPICI} = \frac{5W}{5V} + \frac{0.47W}{2V} = 1.23A \quad (2.28)$$

Con los parámetros anteriores, se puede elegir una fuente de voltaje que entregue 5.1V con una corriente de al menos el doble de I_{EEPICI} , para tener un correcto funcionamiento de la Estación.

Abordando la alimentación para el amplificador de potencia, la figura 2.8 ilustra un circuito convertidor de voltaje Step-Down que provee el fabricante⁷ con los valores de los elementos eléctricos para un correcto funcionamiento, y donde la relación para obtener V_{DD} está dado por:

$$V_{dc-dc} = V_{DD} = 1.23 \times \left(1 + \frac{R_2}{R_1}\right) \quad (2.29)$$

que para obtener el voltaje de 2V, R_1 y R_2 pueden ser de $47k\Omega$ y $33k\Omega$, respectivamente.

⁷Ver la hoja de datos LM2596 SIMPLE SWITCHER® Power Converter 150-kHz 3-A Step-Down Voltage Regulator.

2.4. DISEÑO DE ALIMENTACIÓN ELÉCTRICA PARA LA ESTACIÓN

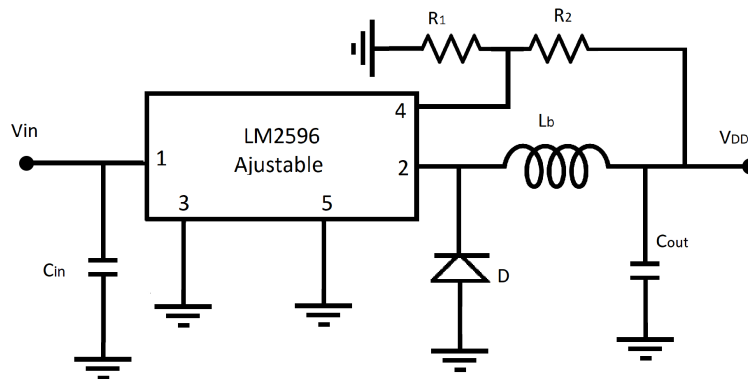


Figura 2.8: Circuito rectificador-regulador de voltaje de DC.

2.5. Fabricación del prototipo de EEPICI

En esta sección se desglosará la elaboración y fabricación de los inductores para el acoplo inductivo resonante y las tarjetas electrónicas para la rectificación-regulación de voltaje en el lado receptor y para la etapa de potencia. Así mismo, para habilitar una comunicación Bluetooth, parte de esta sección está dedicada al desarrollo del Software de Control de la EEPICI y su interfaz gráfica, así como la implementación del firmware para el microcontrolador del prototipo de dispositivo BioMEMS.

2.5.1. Fabricación de los inductores L_1 y L_2 y las tarjetas electrónicas

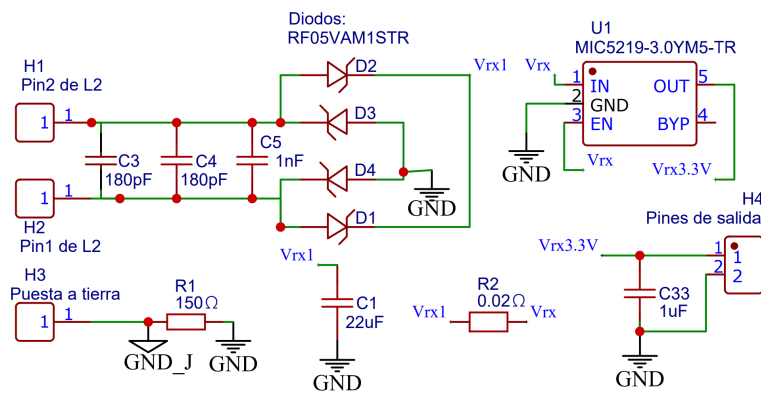
La fabricación de las tarjetas electrónicas y los inductores L_1 y L_2 se hicieron sobre tablas fenólicas FR4 con revestimiento de cobre de $35\mu m$ de espesor, dando como resultado tarjetas electrónicas conocidas como PCBs. Cabe destacar que se creó una técnica de fresado para la fabricación de estas tarjetas electrónicas con la fresadora CNC Modela MDX-50 del laboratorio de Pruebas no Destructivas del Instituto de Ingeniería.

De forma general, para la elaboración de cada inductor y los circuitos mencionados, se usó el software *EasyEDA* para crear el esquema electrónico de los inductores y los circuitos, y con ellos elaborar la disposición de los elementos electrónicos y las pistas acorde a cada esquema. Con ayuda de *Fusion* se creó un modelo 3D de las rutas elaboradas. Este modelo 3D se importó en el programa *SRPPlayer* para crear un archivo con las rutas automatizadas del fresado, y por último, este archivo se cargó en la máquina CNC para fresar las tablas FR4 y así para obtener las PCB correspondientes.

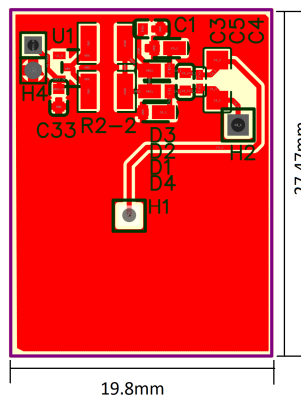
En primer lugar, el esquema correspondiente al inductor L_2 y su circuito rectificador-regulador se ilustran en la figura 2.9 a). En este esquema se representan las terminaciones del inductor L_2 como el *Pin1* y el *Pin2*. Cabe destacar que los capacitores C_3 , C_4 y C_5 están colocados de tal esta forma para obtener la capacitancia teórica C_2 ó, poder implementar

2.5. FABRICACIÓN DEL PROTOTIPO DE EEPICI

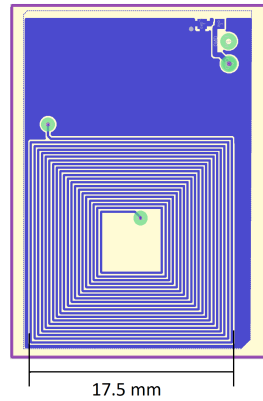
una nueva capacitancia en caso de ser necesario. Por otro lado, la disposición de los componentes electrónicos se diseñó de tal forma que se obtuviera una tarjeta electrónica de doble capa, donde el lado superior contiene la disposición de todo el circuito rectificador-regulador, ilustrado en b), mientras que del lado inferior se encuentra el inductor L_2 , ilustrado en c).



(a) Esquemático del circuito receptor y L_2 .



(b) PCB: capa superior.



(c) PCB: capa inferior.

Figura 2.9: Diseño de la tarjeta electrónica receptora.

Por otro lado, el esquema eléctrico del inductor L_1 sólo consta de dos terminales, pero puede visualizarse la generación de las pistas en la siguiente figura, donde el ancho de cada pista es de aproximadamente el doble de la separación que hay entre ellas.

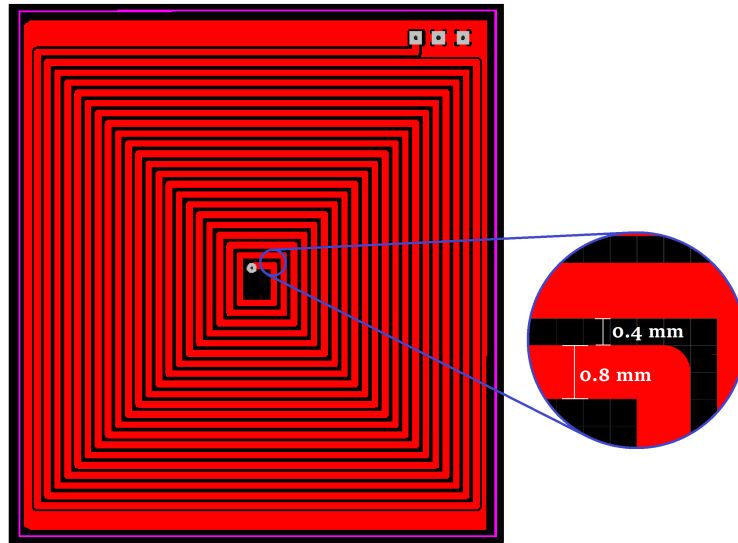
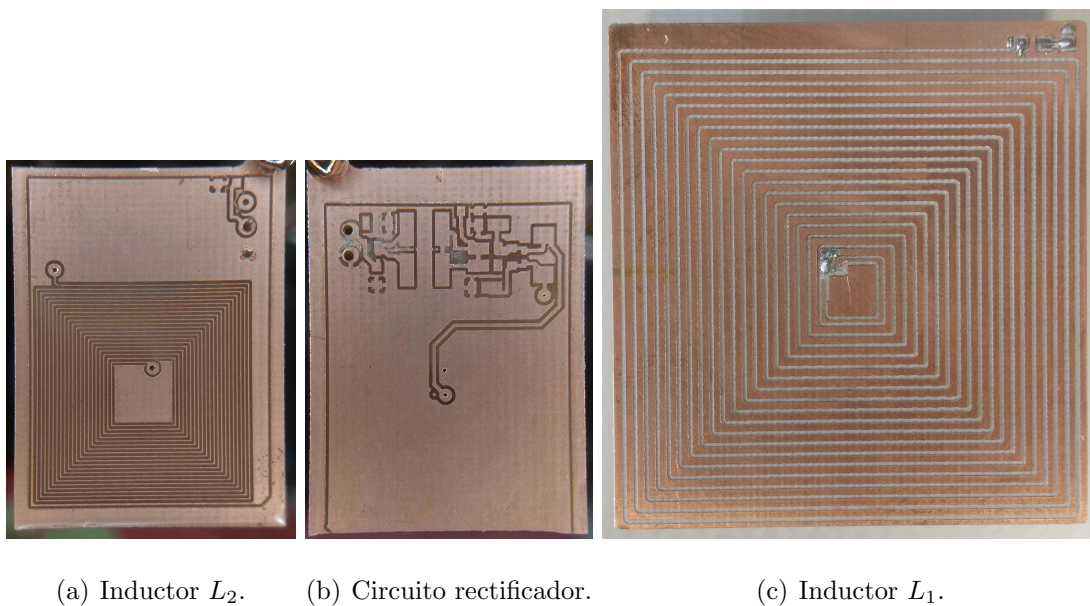


Figura 2.10: Modelo PCB del inductor L_1 .

Los resultados de la fabricación de cada tarjeta electrónica puede ilustrarse en la figura 2.11: siendo en a) la capa 1, y b) la capa 2, del circuito receptor, respectivamente. Mientras que el inductor L_1 consta de una sola cara y está ilustrada en c).



(a) Inductor L_2 .

(b) Circuito rectificador.

(c) Inductor L_1 .

Figura 2.11: Fabricación de PCBs: Inductor L_1 , L_2 y circuito rectificador-regulador.

2.5. FABRICACIÓN DEL PROTOTIPO DE EEPICI

Debido a que el amplificador de potencia está diseñado a partir de las características de los inductores teóricos, como su inductancia y su factor de calidad, es necesario medir estos parámetros de los inductores fabricados, y si es necesario, volver a calcular los elementos del amplificador que dependen de estos parámetros. Para obtener los parámetros de cada inductor, se armó un circuito resonante como la siguiente figura, donde C_{1-2} es el capacitor correspondiente para L_1 o para L_2 , correspondientemente. A este circuito se aplicó un barrido de frecuencia $V_{barrido}$ para encontrar la amplitud máxima de voltaje en el circuito resonante V_{res} , a una frecuencia de resonancia experimental $f_{o.exp}$, y con este último valor, obtener la inductancia real de cada inductor a través de la ecuación (1.17).

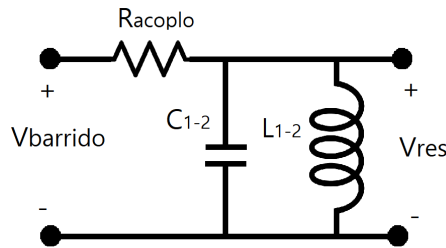


Figura 2.12: Esquema de circuito eléctrico para encontrar L_1 y L_2 real.

Por otro lado, se usó un multímetro de banco para encontrar la resistencia propia de cada inductor y obtener su factor de calidad. Para resumir los parámetros obtenidos de los inductores fabricados, en el cuadro 2.1 se muestra una comparación de los valores teóricos (como la inductancia propia, la resistencia propia y su factor de calidad) respecto a los valores obtenidos en la fabricación de los mismos.

CAPITULO 2. DISEÑO Y CONSTRUCCIÓN DE EEPICI

Parámetro	Valor teórico	Valor de fabricación	Error relativo
L_1	$11.1\mu H$	$6.6\mu H$	0.4
R_1	1Ω	2.2Ω	1.2
Q_1	168	45	0.73
L_2	$5.36\mu H$	$3.4\mu H$	0.38
R_2	1.45Ω	10Ω	5.92
Q_2	59	4.75	0.92

Cuadro 2.1: Comparación de los parámetros teóricos y obtenidos en la fabricación de los inductores.

Para que la respuesta del amplificador de potencia fuese la esperada, fue necesario volver a calcular los valores para C_p , C_1 y C_2 mediante sus expresiones correspondientes considerando los valores de la fabricación. En el siguiente cuadro se muestran los nuevos valores a partir de los parámetros obtenidos en la fabricación de L_1 y L_2 .

Elemento	Cálculo teórico	Nuevo cálculo
C_1	400pF	664pF
C_2	0.82nF	1.29nF
C_p	1.025nF	875pF

Cuadro 2.2: Nuevos valores de capacitores para la etapa de potencia.

Simulación de la etapa de potencia

Antes de desarrollar e implementar la tarjeta electrónica de la etapa de potencia, con ayuda del programa *LTSpice*, se simuló el amplificador de clase E con los valores modificados anteriormente. Se usó un modelo de simulación para el transistor QS6k1, los inductores L_1 y L_2 contienen sus resistencias propias dentro de su modelo, y estos últimos están acoplados mediante k1 de valor 0.2. Además, la carga R_L fue seleccionada de 172Ω y de 110Ω para simular la respuesta del amplificador al cargar el capacitor (de almacenamiento de energía para la operatividad del prototipo sin la WPT) y al energizar directamente al prototipo, respectivamente.

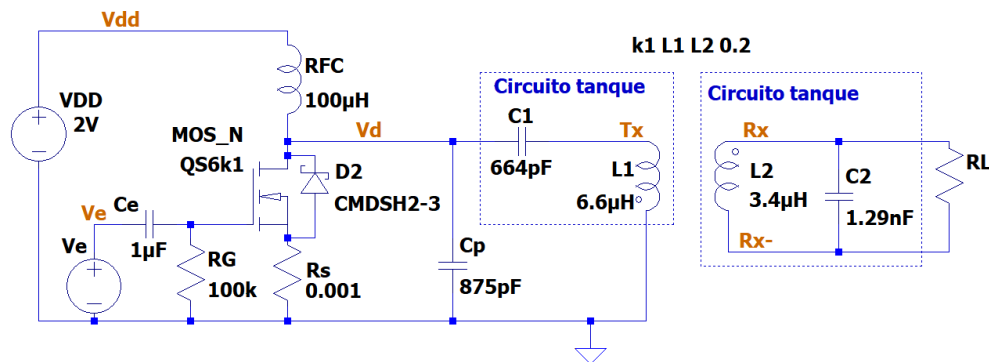
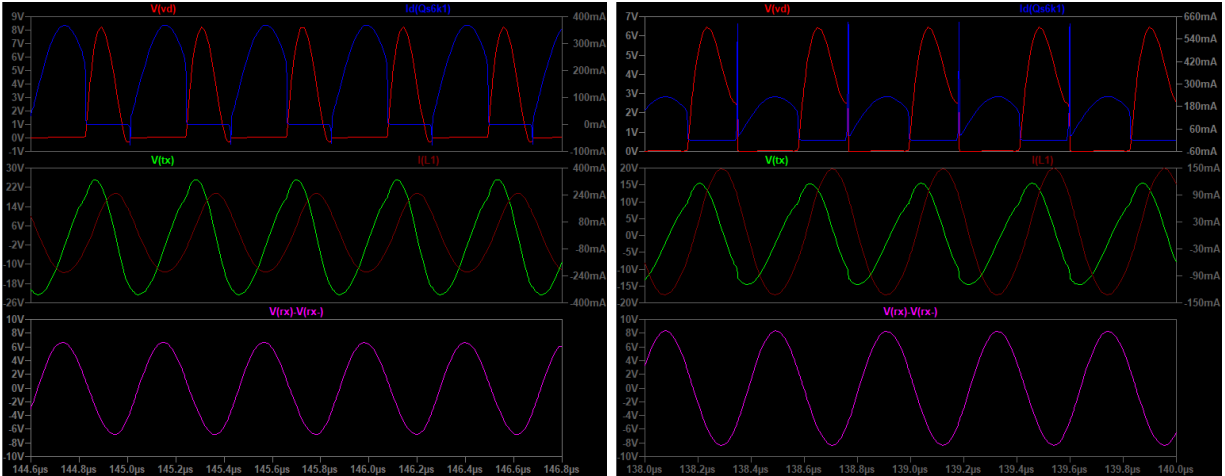


Figura 2.13: Circuito de amplificador de potencia con acoplo inductivo.

La simulación obtenida puede verse en la figura 2.14. Puede observarse, tanto en a) como en b) la generación de un voltaje en el inductor L_1 de hasta $55V_{pp}$ y $30V_{pp}$, respectivamente. De igual manera, el voltaje inducido al inductor L_2 con un voltaje de $14V_{pp}$ y $16V_{pp}$. Con lo que se estima que probablemente, la etapa de potencia pueda infringir los límites seguros de SAR para la transmisión de potencia inalámbrica. Sin embargo, esto se verá en los resultados de las pruebas experimentales.



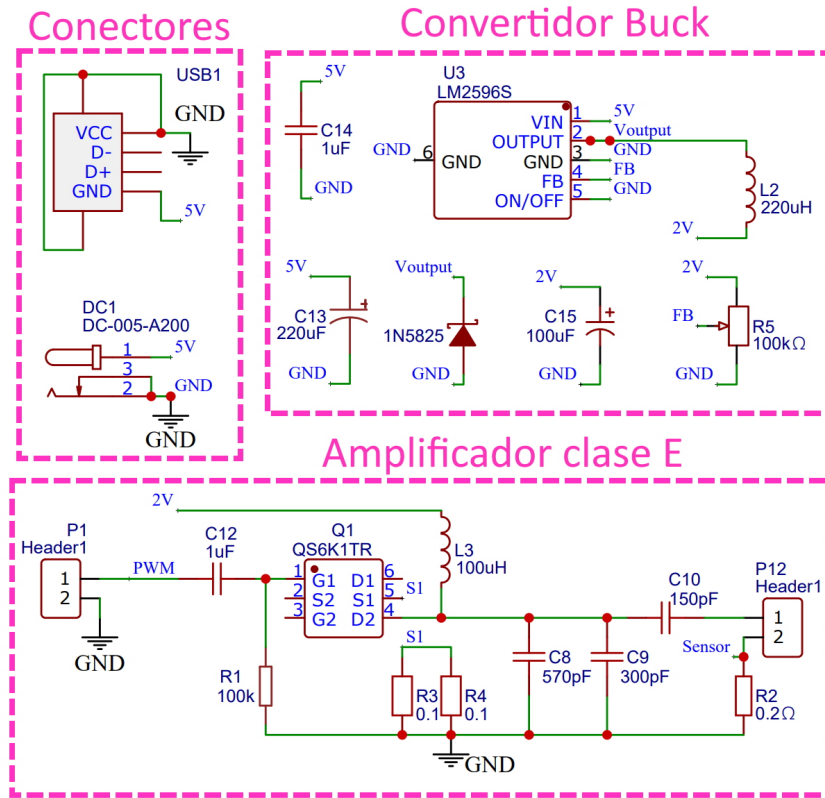
(a) Respuesta del amplificador teniendo una carga de $R_L = 172\Omega$ (b) Respuesta del amplificador para la carga de $R_L = 1110\Omega$

Figura 2.14: Gráfica en el tiempo de la respuesta del amplificador de potencia para diferentes cargas eléctricas resistivas.

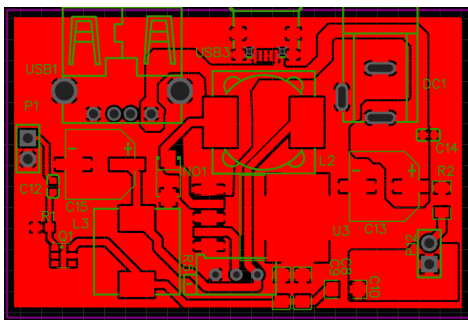
Fabricación de la tarjeta electrónica de la etapa de potencia

La tarjeta electrónica de la etapa de potencia, que integra el amplificador de clase E, con los valores modificados del cuadro 2.2, y el circuito correspondiente al convertidor de voltaje Step-Down, fue diseñada de la misma forma que las inductores de potencia y el circuito rectificador-regulador de voltaje. En la figura 2.15 a) se ilustra su esquemático correspondiente a estos circuitos, en b) se muestra el arreglo de la PCB para su fabricación, y en c) la tarjeta fabricada.

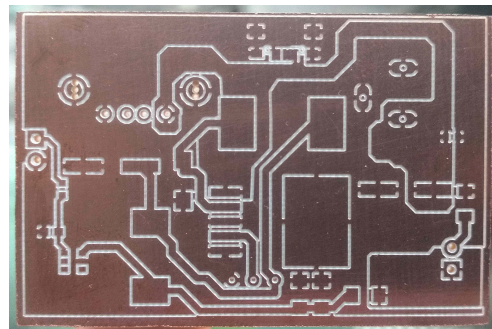
2.5. FABRICACIÓN DEL PROTOTIPO DE EEPICI



(a) Esquemático de la etapa de potencia.



(b) Diseño de PCB de la etapa de potencia.



(c) Tarjeta electrónica fabricada.

Figura 2.15: Desarrollo de la tarjeta electrónica de la etapa de potencia.

Continuando con la fabricación de las tarjetas electrónicas, se soldaron los componentes electrónicos de la tarjeta de la etapa de potencia y de el circuito rectificador-regulador de voltaje del elemento receptor. A continuación se ilustra el resultado final del desarrollo

CAPITULO 2. DISEÑO Y CONSTRUCCIÓN DE EEPICI

de las tarjetas electrónicas: el rectificador-regulador de voltaje del elemento receptor y el amplificador de potencia de clase E junto con su convertidor de voltaje Step-Down como fuente de alimentación.

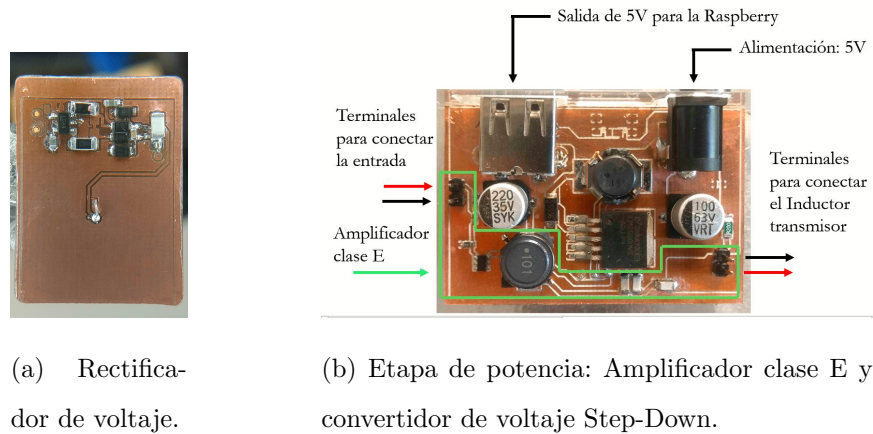


Figura 2.16: Resultado final de las tarjetas electrónicas.

Además, la tarjeta electrónica del elemento receptor y el circuito rectificador-regulador se encapsuló en silicón semitransparente para ser conectado directamente al prototipo de dispositivo BioMEMS, como se muestra a continuación.

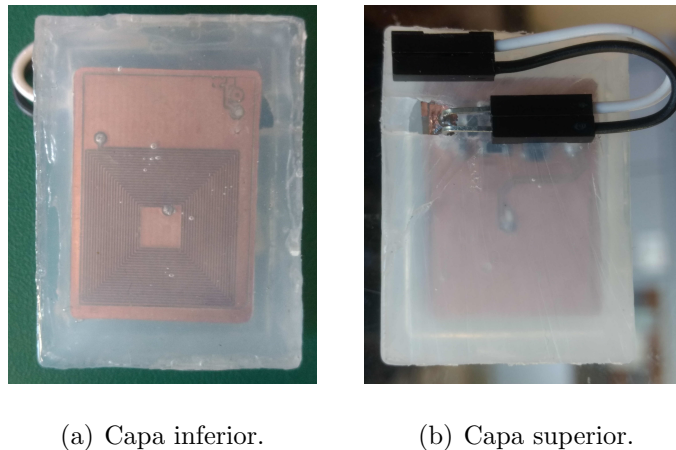


Figura 2.17: Encapsulado del elemento receptor.

Retos y desafíos de elaboración y fabricación

En el desarrollo de los inductores en tarjetas fenólicas hubo retos de fabricación. A lo largo de las iteraciones en la fabricación del inductor L_2 se buscaron algunas mejoras, por ejemplo, que el circuito rectificador-regulador de voltaje no quedara expuesto directamente al flujo de campo magnético generado por el inductor L_1 , por ello, a partir de la versión 3 se buscó que este circuito estuviera por detrás del lado de L_2 y protegido por la capa anterior aterrizada al común eléctrico del circuito; esto puede verse en la figura 2.18.

Otra dificultad de fabricación, que fue de la mano junto al diseño de L_2 , fue conseguir la mayor cantidad de vueltas posibles para su tamaño límite de 3.5cm^2 . Se puede decir que para su fabricación, la resolución de la fresadora se operó cerca de su resolución límite, esto provocó en repetidas ocasiones no concretar un resultado esperado, sin embargo, al final el número de vueltas se pudo aumentar de 15 a 20 a partir de la versión 3.

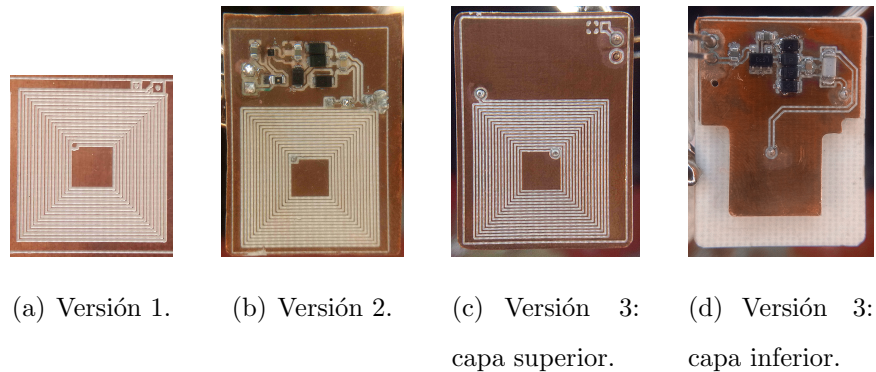
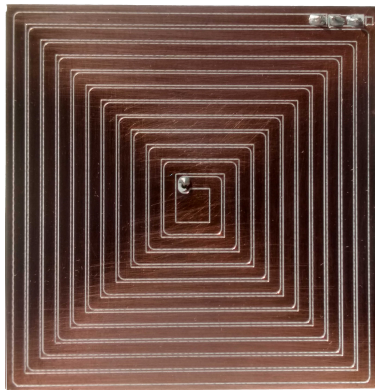


Figura 2.18: Versiones anteriores del inductor L_2 . Destacando las características R_{sL_2} y N_2 de cada versión: a) $L = 2.048\mu\text{H}$, $R_{sL} = 6.3\Omega$ $N = 15$; b) $L = 2.4\mu\text{H}$, $R_{sL} = 5.2\Omega$ $N = 15$; c) y d) $L = 2.048\mu\text{H}$, $R_{sL} = 3.5\Omega$ $N = 20$.

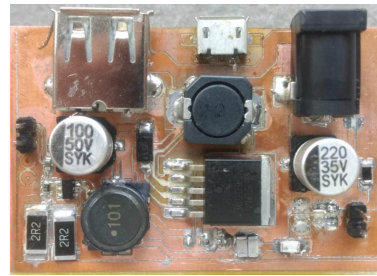
CAPITULO 2. DISEÑO Y CONSTRUCCIÓN DE EEPICI

Por otro lado, en la figura 2.19 a) se ilustra una primera versión de L_1 que, para transmitir la suficiente potencia a L_2 , éste superaba por mucho los niveles seguros de SAR, debido al número de vueltas con las que se diseñó. De acuerdo a la sección 2.2.5, se realizó un inductor que tuviera mayor número de vueltas y con ello se mitigaran los riesgos potenciales de infringir los niveles seguros de radiación de campo magnético generados por este inductor. Además, con una nueva versión, se mejoró la inductancia propia de $2.9\mu H$ a $6.6\mu H$

Acerca del circuito de potencia, primeramente se implementó un amplificador clase C alimentado con un voltaje de hasta $13.5V$, que en las pruebas preliminares se descartó debido a un sobrecalentamiento del transistor (ver figura 2.19 b). Por otro lado, fue complicado conseguir la fabricación del amplificador de potencia con los valores exactos de las capacitancias calculadas, debido a que el amplificador tiene una respuesta muy diferente ante mínimas variaciones de C_1 y C_p .



(a) Inductor Versión 1.



(b) etapa de potencia, penúltima versión.

Figura 2.19: Versiones anteriores de L_1 con $N = 9$, $L = 2.9\mu H$, y $R = 1\Omega$; y del amplificador clase C con circuito elevador de voltaje.

2.5.2. Desarrollo de software de control

Para la elaboración del software de control, se pensó en hacer un programa junto con una interfaz gráfica que pudiera ejecutarse en la tarjeta Raspberry Pi, y mediante un display con touch incorporado a la misma, se pudiera controlar la EEPICI.

Sin embargo, por el momento se ha realizado una aplicación en Matlab, que puede ejecutarse desde un ordenador y controlar directamente la EEPICI a través de la Raspberry Pi. En otras palabras, desde esta aplicación se tiene el control de la Estación Electrónica mandando los comandos necesarios para realizar la transmisión de potencia inalámbrica, buscar el prototipo de dispositivo BioMEMS, extraer los datos recabados por este último y, mandar, por correo electrónico, estos datos en un archivo de hoja de cálculo.

La interfaz gráfica del software de control se ilustra a continuación, y el código de programación para el desarrollo de dicha aplicación se encuentra desglosado en el anexo A.2. Como puntos a resaltar, el sistema WPT es habilitada mediante este software, pues genera una señal PWM de 2.4MHz a la salida de la terminal 12 del puerto GPIO de la Raspberry Pi; esta señal PWM es entonces usada por la etapa de potencia como señal de entrada para transmitir potencia a esta frecuencia (este código se encuentra de la línea 103 a la 119).

Por otro lado, una instrucción es enviada a la Raspi para solicitar los datos procedentes del prototipo. La Raspi, mediante su dispositivo Bluetooth, establece un intercambio de información con el prototipo para obtener los datos. Esto último se hace a través de la lectura de un atributo genérico del BLE del prototipo, de identificador único ID como *FFF5*. Es entonces que el BLE del microcontrolador envía los datos a través de este atributo, y de esta manera; esta instrucción está descrita en la función *ObtenerdatosButtonPush*, concretamente en la línea 149.

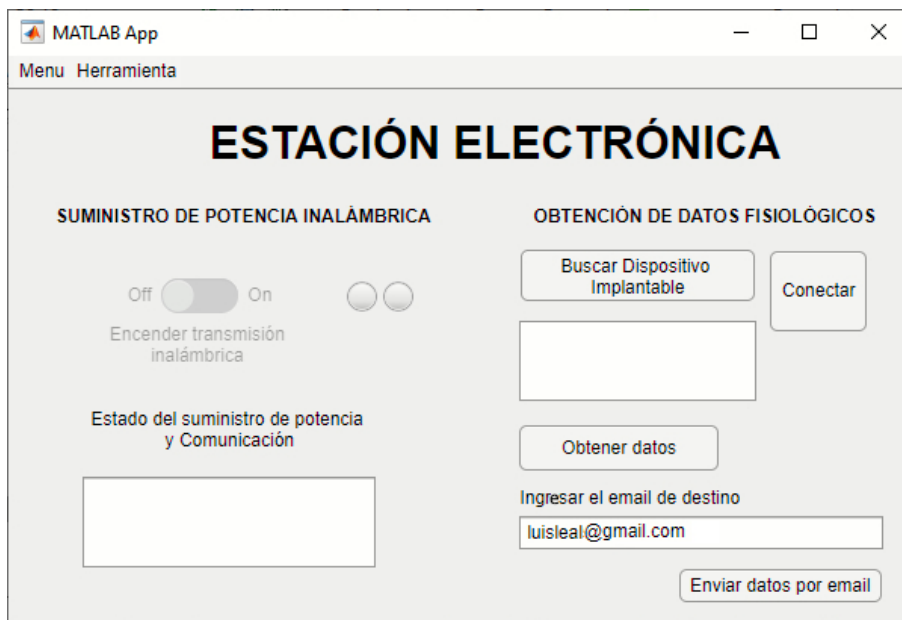


Figura 2.20: Interfaz gráfica del Software de Control para la EEPICI.

Para usar el software de control, primero es necesario conectar la Raspi al ordenador, esto se hace seleccionando: Menú \rightarrow Conectar Estación EEPICI. Una vez conectada, se habilita el Switch *Encender transmisión inalámbrica* para habilitar la transmisión de potencia. Al acercar el prototipo al inductor $L1$, éste se energiza y se pueden obtener los datos del microcontrolador con el botón *Obtener Datos*. Además, estos datos pueden enviarse en un archivo de excel por correo electrónico usando el botón *Enviar datos por email*.

2.5.3. Desarrollo del firmware del prototipo de dispositivo BioMEMS

El desarrollo del firmware se dividió en dos principales tareas, la primera es ejecutada por el Controlador de Sensores (ver figura 2.2) para realizar el monitoreo y el almacenamiento de los datos de una variable física; y la segunda es ejecutada por el microcontrolador principal del prototipo para habilitar la comunicación BLE y enviar los datos almacenados.

2.5. FABRICACIÓN DEL PROTOTIPO DE EEPICI

Se generó la primera parte del firmware para el Controlador de Sensores. Con ayuda del entorno de desarrollo *Sensor Controller Studio* se configuró al Controlador habilitando un periférico de conversión Analógica Digital ADC, configurando la terminal 7 del prototipo como entrada del ADC y autorizando una comunicación con el microcontrolador principal.

La tarea del Controlador de Sensores consiste en realizar 4 veces el proceso enlistado a continuación⁸:

1. Activar el ADC para tomar una muestra y convertirla a un valor digital.
2. Almacenar este valor convertido en una variable.
3. Esperar 1 segundo para tomar una nueva muestra.

Una vez tomadas las 4 muestras en un lapso de 4 segundos, es generada una interrupción hacia el microcontrolador principal. En el anexo C.1 se encuentra el código de programación para la ejecución de esta tarea.

Para desarrollar la segunda parte del firmware, se trabajó en el entorno *Code Composer Studio*, donde se importó el kit de desarrollo de software (SDK, por sus siglas en inglés) de nombre *Simplelink_Peripheral_cc2640r2lp_app*⁹. Basado en este proyecto, se terminó el desarrollo del firmware para integrar la tarea del Controlador de Sensores y la habilitación de una comunicación BLE.

Esta parte del desarrollo del firmware se encuentra desglosado en el anexo C.2 y C.3. En el archivo principal *Simple_peripheral.c* se agregó la biblioteca *scif.h* para importar el monitoreo del Controlador de Sensores, y se describieron las características y acciones para controlar la tarea de monitoreo y almacenamiento de los datos. Estas acciones están declaradas y descritas como:

⁸La cantidad de repeticiones del proceso es arbitrario, pues pueden ser más o menos dependiendo de una aplicación específica. La idea es que cada muestra se tome en un tiempo determinado y que represente un solo dato, es decir, que esta tarea monitoree una variable fisiológica de forma periódica y en periodos largos o corto de tiempo, mas no necesariamente dentro de un segundo.

⁹La finalidad de implementar el firmware del prototipo sobre este kit, o proyecto base, fue poder desarrollar el firmware completo sin la necesidad de configurar todo el Stack de BLE, debido a la complejidad que esto requiere.

- `SensorController_Init`: para inicializar las configuraciones del Controlador de Sensores y posteriormente ejecutar la función `RunTaskOnce` (líneas 331-345).
- `RunTaskOnce`: para ejecutar una vez la tarea del Controlador de Sensores (líneas 347-366).
- `scTaskAlertCallback`: para que se almacenen los datos procedentes de la conversión ADC, hecha por el Controlador de Sensores, y ejecutar nuevamente la función `RunTaskOnce`. Esta función es ejecutada a partir de una interrupción proveniente del Controlador de Sensores (líneas 309-330).
- `SimplePeripheral_init`: para inicializar y configurar el Stack BLE, agregando las siguientes acciones:
 - Nombrar al dispositivo BLE como *Dispositivo BioMEMS* (líneas 150-151 y 186).
 - Dar acceso al emparejamiento con la contraseña *12345* (líneas 461 y 466).
 - Habilitar la visibilidad del dispositivo (líneas 454 y 464).
 - ejecutar la función `SensorController_Init` (línea 478).
- `SimplePeripheral_taskFxn`: para enviar los datos almacenados mediante el atributo de BLE con el identificador *FFF5* (líneas 598-612).

El proceso para cargar el firmware, que contiene las dos tareas descritas anteriormente, hacia el prototipo de dispositivo BioMEMS está mostrado en el anexo C.4. En este anexo se describe brevemente la revisión del prototipo cargar un firmware *Code composer Studio* mediante el depurador de pruebas Debug XDS110.

3. Prueba de concepto de EEPICI

En esta sección se describe la prueba del concepto de la Estación Electrónica de Potencia Inalámbrica y Comunicación Inalámbrica, Primero se hicieron las pruebas por separado del sistema de transferencia de potencia y del sistema de comunicación inalámbrica, como se verá a continuación. Después se juntaron los dos sistemas como un sistema íntegro para realizar una prueba experimental del concepto de EEPICI.

3.1. Diseño experimental

3.1.1. Transferencia de potencia inalámbrica

Para el desarrollo de las pruebas experimentales de la transferencia de potencia inalámbrica, se diseñó y fabricó un soporte para lograr un emparejamiento entre los inductores L_1 y L_2 en posición específica y poder ajustar la distancia entre ellos, como puede ilustrarse a continuación.

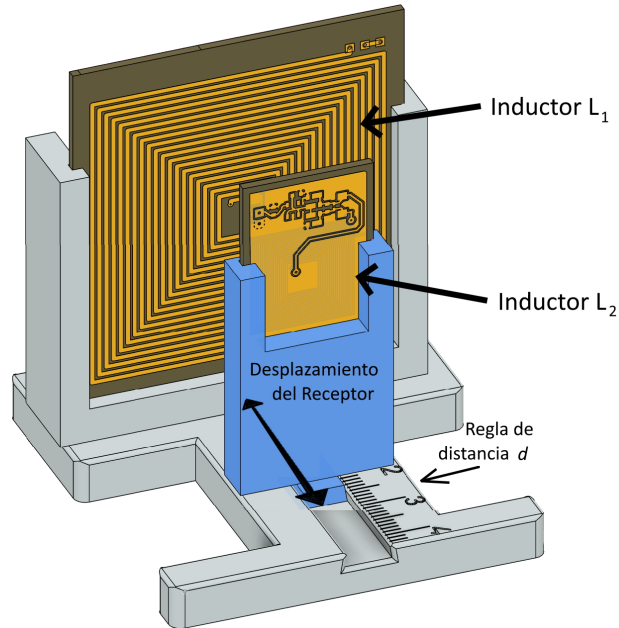


Figura 3.1: Base para los elementos del acoplamiento inductivo.

El primer experimento consistió en hacer pruebas de la transmisión de potencia conectando diferentes cargas resistivas R_L , al circuito rectificador-regulador del elemento receptor, para medir el voltaje V_R entregado por receptor y calcular la potencia que éste entrega. Concretamente se realizaron 3 pruebas: una primera para $R_L = \infty$, lo que equivale a no conectar una carga al sistema; una segunda para $R_L = 180\Omega$, como representación de la carga de mayor demanda de potencia descrita en la sección 2.2 en la página 63; y una tercera conectando el prototipo de dispositivo BioMEMS como la carga R_L .

Para cada prueba se observó el voltaje V_{Tpp} en el inductor L_1 , para validar o desaprobar que los niveles de SAR, relacionados con la intensidad de campo magnético generada por el elemento transmisor L_1 , son menores a los límites calculados en la sección 2.2.5.

Posteriormente se conectó una carga resistiva variable (un potenciómetro) para medir el valor de la carga R_{Lmax} ante una atenuación de -3dB en V_{Rmin} , o el equivalente a una caída desde 3.3V hasta 2.2V. Esto representa la potencia máxima que puede entregar el sistema de transferencia de potencia inalámbrica, y puede calcularse como $\frac{(2.2V)^2}{R_{Lmax}}$.

Este primer experimento se realizó a una distancia de separación entre los inductores de magnitud $d = 8mm$, y un voltaje de alimentación al amplificador de potencia $V_{DD} = 2V$.

Posteriormente, se realizó un segundo experimento repitiendo todo el procedimiento descrito anteriormente para un voltaje de alimentación al amplificador de potencia de $V_{DD} = 3.4V$.

Por último, se realizó un experimento adicional para medir los mismos parámetros de interés, cuando es conectado el prototipo de dispositivo BioMEMS como carga del sistema para los siguientes parámetros: $d = 6mm$ y $V_{DD} = 2V$. esto con el fin de comparar estos resultados con los obtenidos en las pruebas anteriores con el prototipo como R_L .

3.1.2. Comunicación inalámbrica para la obtención de datos

Para probar el sistema de comunicación inalámbrica, se llevó a cabo la ejecución del software de control para habilitar la comunicación entre el prototipo de dispositivo BioMEMS y la Raspberry Pi, como se muestra en la figura 3.2.

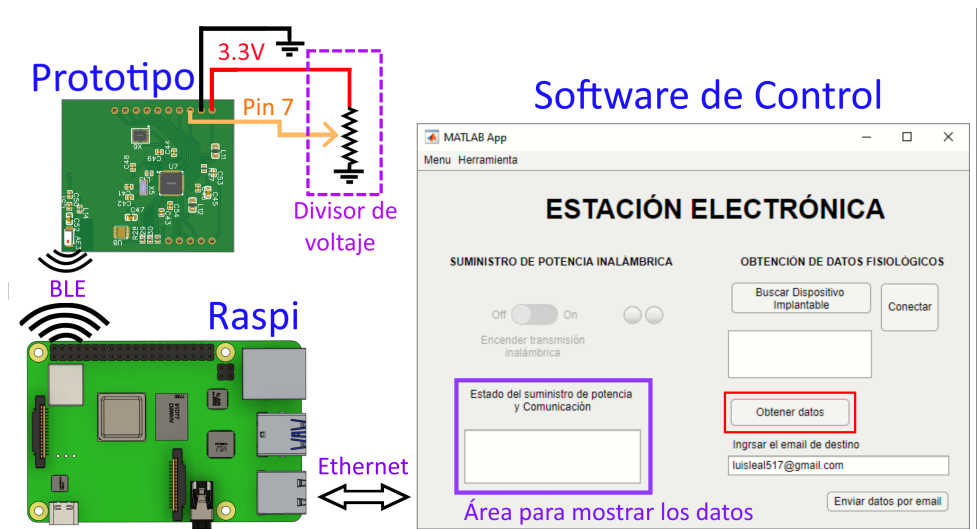


Figura 3.2: Esquema general para la prueba de obtención de datos.

La prueba del monitoreo de una variable física (tal como la temperatura corporal, la presión arterial, etc.), se realizó a partir de datos de prueba generados por un divisor de voltaje polarizando un potenciómetro como se muestra en la figura anterior. Mientras que la obtención de estos datos de prueba, se solicitaron desde el software de control, mediante la instrucción *Obtener datos*.

Para probar la conversión y obtención de datos de prueba, se propusieron dos eventos. El primero fue mantener el potenciómetro en una posición fija mientras el prototipo de dispositivo BioMEMS monitoreaba y almacenaba los 4 datos de prueba para después solicitarlos desde la interfaz gráfica. El segundo evento es aumentar paulatinamente el valor del potenciómetro mientras el controlador de sensores toma los 4 datos, y nuevamente solicitarlos desde la interfaz gráfica.

Por último, se exploró enviar los datos recolectados por la Estación Electrónica hacia un servicio de correo electrónico al ingresar una dirección de correo electrónico en el campo inferior derecho de la interfaz gráfica y seleccionar la instrucción *Enviar datos por email*.

3.2. Resultados

3.2.1. Transferencia de potencia inalámbrica

Pertinente al primer experimento, en la figura 3.3a) y 3.3b) se puede visualizar el montaje para las pruebas correspondientes usando una carga ∞ y una de 180Ω , respectivamente. Sin conectar una carga eléctrica, se obtuvo $V_{Tpp} = 48V$ y $V_R = 3.3V$. Por otro lado, con $R_L = 180\Omega$, V_{Tpp} es de $42V$ y V_R de $1.38V$.

En la prueba del prototipo como carga del sistema, se usó la carga constante¹ del prototipo de dispositivo BioMEMS para minimizar el voltaje de rizo en V_R entregado al prototipo.

¹Esto se logró colocando un capacitor en paralelo al microcontrolador del prototipo, para mitigar el comportamiento de éste como una carga dinámica, que es causada por la variación de corriente debido a los cambios de estado u operatividad del microcontrolador del prototipo.

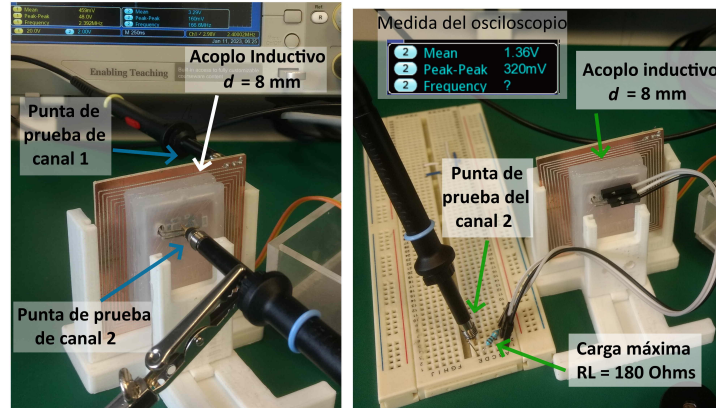
(a) Sistema WPT sin carga (b) Sistema WPT $R_L = 180\Omega$

Figura 3.3: Prueba de transmisión de potencia del primer experimento.

Como resultado, el sistema de transmisión pudo energizar al prototipo con $V_R = 2.58V$ y el voltaje en el inductor transmisor V_{Tpp} fue de $51.2V$. El montaje para esta prueba se ilustra a la izquierda de la figura 3.4, mientras que a la derecha se observa la forma de onda de V_{Tpp} y V_R se mantiene como un valor prácticamente constante.

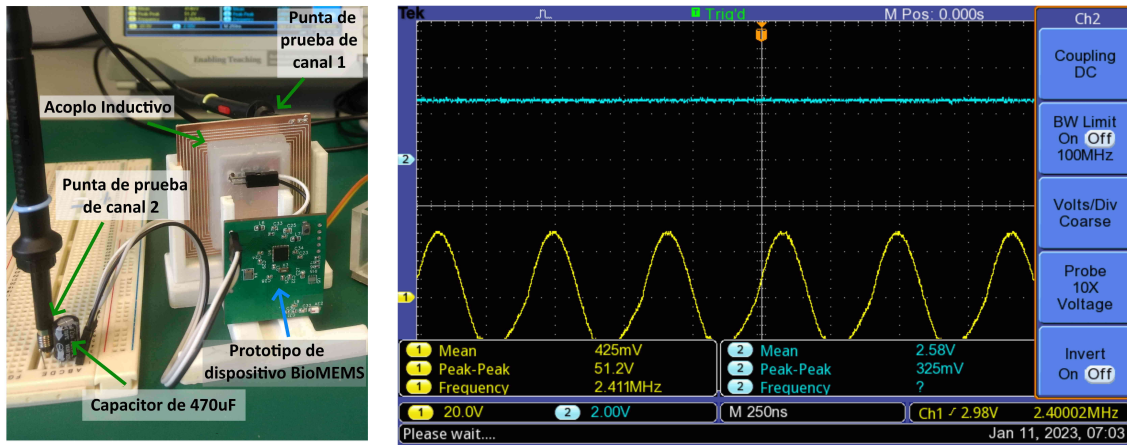


Figura 3.4: Resultado de la transmisión de potencia con el prototipo como carga.

Usando una carga resistiva variable, el voltaje entregado por el receptor empieza a decrecer con una carga menor o igual a $R_{Llimite}$ de $3.17k\Omega$, entregando una potencia de $3.43mW$. Además, R_{Lmax} fue de 510Ω , entregando así una potencia máxima de $9.49mW$.

CAPÍTULO 3. PRUEBA DE CONCEPTO DE EEPICI

El segundo experimento, recordando los parámetros $d = 8mm$ y $V_{DD} = 3.4V$, se llevó a cabo de la misma manera que en el primer experimento. Cuando el sistema no tiene carga, el voltaje inducido es $V_R = 3.3V$ y el voltaje en el inductor transmisor de $V_{Tpp} = 62.3V$. Mientras que para la carga $R_L = 180\Omega$, estos valores son: $V_R = 2.1V$ y $V_{Tpp} = 55.5V$.

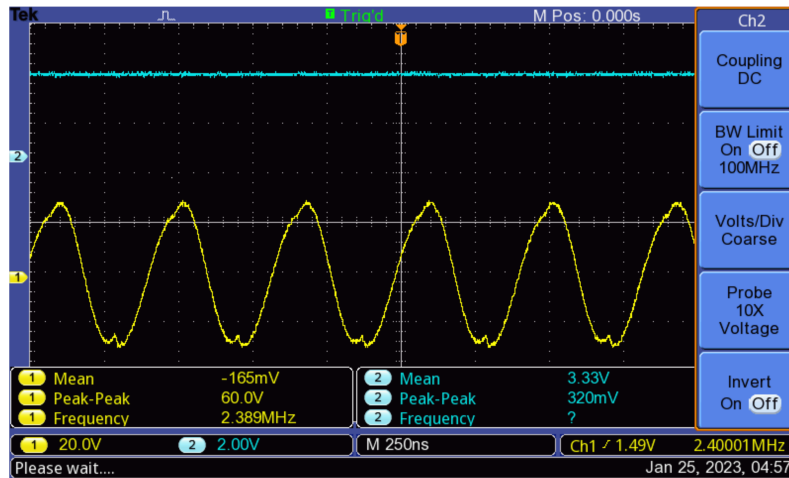


Figura 3.5: Nueva prueba de transmisión de potencia con el prototipo como carga R_L .

Por otro lado, al colocar al prototipo de dispositivo BioMEMS como la carga del sistema WPT, éste es energizado con un voltaje $V_R = 3.33V$, mientras que el voltaje en el inductor es $V_{Tpp} = 60V$.

Nuevamente, variando el valor del potenciómetro conectado al receptor como carga, se encontró que V_R comienza a decrecer para un valor resistivo de $1.43k\Omega$, entregando así una potencia de $7.61mW$. Ante una carga de 510Ω , se obtuvo un $V_R = 2.4V$ y por tanto $11.3mW$ de potencia entregada. Por último, la potencia máxima entregada fue de $11.8mW$, cuando V_R cae a V_{Rmin} , con una carga resistiva de 410Ω .

Por último, con el experimento adicional, para $d = 6mm$ y $V_{DD} = 2V$, se midió el voltaje inducido en el lado receptor, siendo este de $V_R = 3.29V$, mientras que el voltaje en el inductor transmisor fue de $44.2V$, como se ilustra a continuación.

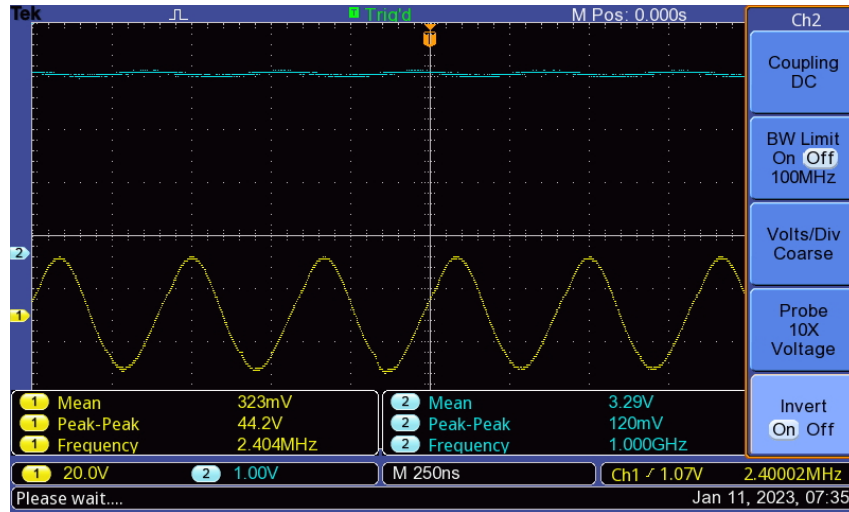


Figura 3.6: Prueba adicional del prototipo como carga R_L .

En resumen, la siguiente figura ilustra las gráficas del voltaje V_R entregado por el receptor en función de la carga conectada al mismo. El primer experimento corresponde a la curva azul, con una distancia de separación en la transmisión de $d = 8mm$ y la potencia de 470mW entregada por el amplificador suministrando $V_{DD} = 2v$; el experimento 2 es representado por la curva roja, la misma separación del acoplo ($d = 8mm$) y aumentando la potencia del amplificador al aplicar $V_{DD} = 3.4v$; y la prueba adicional por la curva verde, con $d = 6mm$ y $V_{DD} = 2v$. Para estos tres comportamientos, se hizo énfasis en V_{Tpp} cuando la carga conectada al sistema es el prototipo.

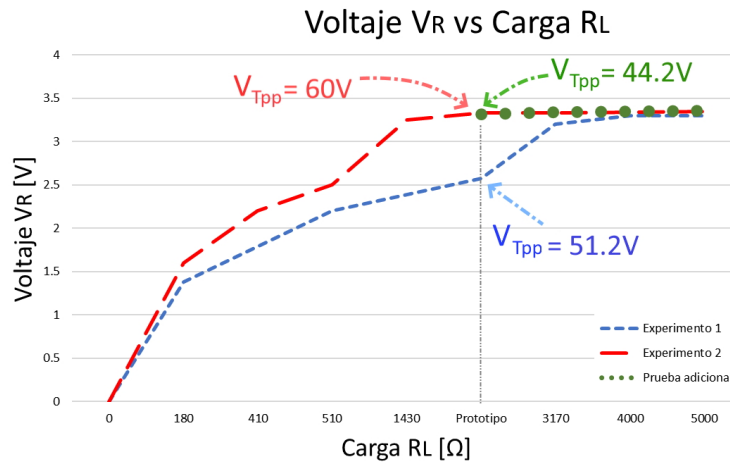
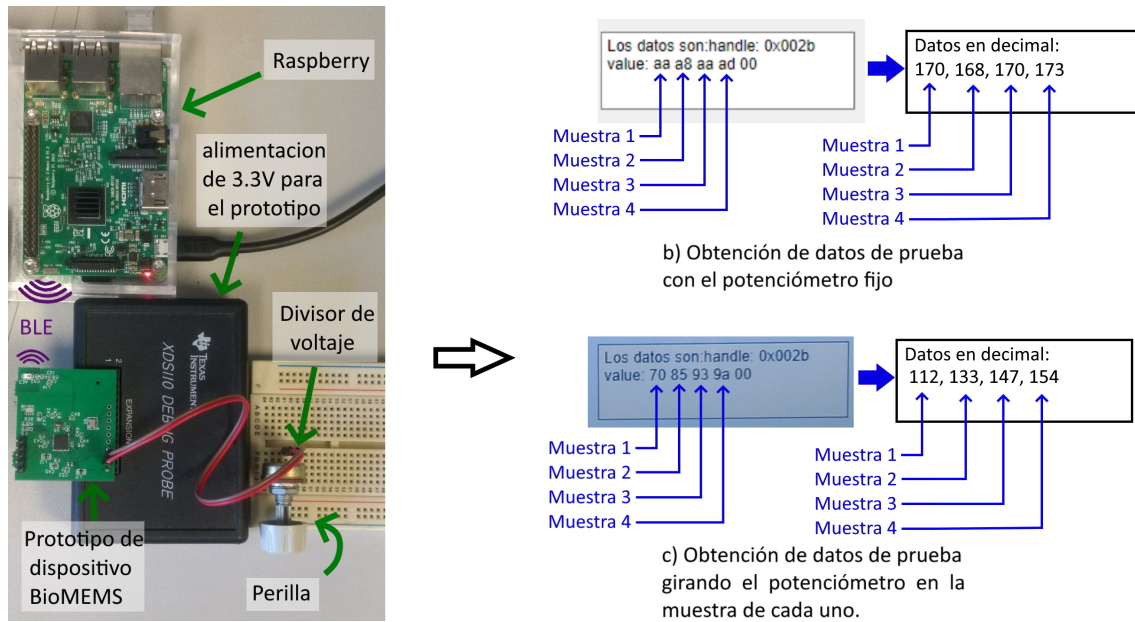


Figura 3.7: Comparación del comportamiento del V_R ante variaciones de la carga R_L .

3.2.2. Comunicación inalámbrica

El prototipo de dispositivo BioMEMS se energizó con el dispositivo *Debug Probe XDS110*, a un voltaje fijo de 3.3V. En la figura 3.8 se observa el arreglo físico para la prueba de la comunicación inalámbrica y la obtención de datos.

Se puede observar que en la figura 3.8 b) las 4 muestras son casi del mismo valor, donde se nota que hay una pequeña diferencia entre estos valores a causa del margen de error en la conversión de cada muestra hecha por el ADC. Por otro lado, en 3.8 c) se observa un incremento de los valores de cada muestra al mover el potenciómetro de posición mientras que el convertidor ejecuta cada muestra.



(a) Montaje del sistema de comunicación inalámbrica

Figura 3.8: Prueba de comunicación inalámbrica para la obtención de datos de la variable física.

Además, se probó el envío de estos datos obtenidos al correo *luisleal***@gmail.com*. Por lo que se comprobó la recepción de los datos en un archivo de excel como se muestra a continuación.

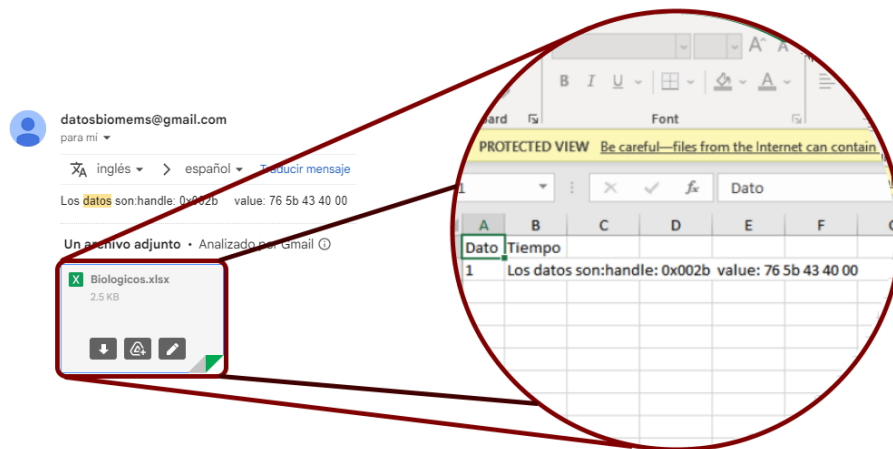


Figura 3.9: Recepción de datos de prueba por correo electrónico.

3.3. Discusión

Referente al primer experimento de la transferencia de potencia inalámbrica se puede resaltar que es posible suministrar una potencia con un voltaje constante para cargas mayores a $R_{Llimite} = 3.17k\Omega$. Sin embargo, para energizar al prototipo de dispositivo BioMEMS, siendo éste una carga menor a $R_{Llimite}$, V_R sufre una caída de voltaje, llegando a ser de $2.58V$.

Lo anterior indica que la transmisión de potencia no es suficiente para energizar al prototipo de manera óptima. Esto puede deberse a que a lo largo de la transmisión de potencia haya mas pérdidas de energía que las estimadas en los cálculos de la sección 2.2.4, por tanto, la potencia que puede entregarse en el extremo receptor es menor a la que demanda el prototipo.

En cuanto a los límites seguros de SAR pertinentes a la transmisión de energía, se puede constatar que $V_{Tpp} = 51.2V$ es menor a $V_{pp-max-L1}$, recordando que este voltaje límite es de $53.5V$, como se había esperado en el análisis de la etapa de potencia.

Por otro lado, para el segundo experimento, al incrementar la potencia al amplificador suministrando $V_{DD} = 3.4V$, se pudo transmitir la potencia suficiente para energizar al prototipo de forma ideal, es decir, que éste fuese alimentado con un voltaje de $3.3V$. Esto permite una mayor estabilidad para crear la comunicación BLE con la Raspberry y por tanto, minimizar el riesgo de pérdidas de los datos de prueba.

En contra parte, el voltaje V_{Tpp} , siendo de $60V_{pp}$ supera la restricción $V_{max-pp-L1}$. Esto significa que es posible que en esta condición de WTP se excedan los niveles seguros de SAR establecidos por IEEE si un organismo o parte de él se encuentra envuelto en el campo magnético generado por L_1 .

Una alternativa para energizar al prototipo de forma idónea y al mismo tiempo no exceder $V_{max-pp-L1}$, es mejorando las características de L_1 ; pues al aumentar N_1 del inductor transmisor y/o su área, de acuerdo a las ecuaciones (2.26) y (2.27), $V_{pp-max-L1}$ puede ser mayor y por ende respetar el limite de SAR relacionados con el amplificador de potencia.

Por último, con el experimento adicional se mostró que disminuyendo la distancia d , es posible alimentar al prototipo de dispositivo BioMEMS implantable de forma adecuada y sin superar $V_{max_pp_L1}$. Esto se debe a que hay un mayor factor de acoplamiento entre los inductores, en comparación con el primer experimento. Sabiendo que el parámetro d es un valor propuesto en las especificaciones, si se permite ser de 6mm o menor, esta puede ser una opción mas viable para respetar las condiciones de implantación y energizar adecuadamente al prototipo.

Referente en la comunicación BLE, se pueden discutir algunos puntos. El primero es que para un fin práctico, se crearon datos prueba con la manipulación de un divisor de voltaje, es decir, cambiando a voluntad la variable física. Sin embargo, para una aplicación médica, se podrían monitorear variables fisiológicas reales con ayuda de sensores miniatura de tecnología BioMEMS incorporados al prototipo.

El segundo punto es sobre el firmware desarrollado para el microcontrolador CC2640R2F del prototipo. Pues este firmware puede adecuarse fácilmente para satisfacer las necesidades de una aplicación médica específica orientada al monitoreo de variables fisiológicas; por ejemplo, se puede agregar más tipos de mediciones, cambiar la frecuencia con que se toman las muestras y modificar la cantidad de muestras de cada variable fisiológica.

Un desarrollo deseable para implementar, a futuro, en el firmware es agregar la funcionalidad de lograr que el prototipo de dispositivo BioMEMS opere en su modo de baja energía *Standby*; pues existe un beneficio potencial al reducir el consumo eléctrico del prototipo manteniéndolo operativo durante un periodo de tiempo específico sin la energización a través del sistema WPT.

Por último, el desarrollo del software de control, a pesar de no estar diseñado para ejecutarse directamente desde la tarjeta Raspberry Pi, contiene la estructura básica de las instrucciones que directamente pueden ejecutarse como comandos en el sistema operativo Rasbian de la Raspi. Con ello, ahora se tiene la oportunidad de replicar este software de control para crear un nuevo programa que sea ejecutado desde la Estación Electrónica.

3.4. Prueba integral de la EEPICI

La Estación Electrónica se ensambló sobre una caja de acrílico, diseñada especialmente para juntar los sistemas de Potencia Inalámbrica y Comunicación Inalámbrica, ilustrada en la figura 3.10.

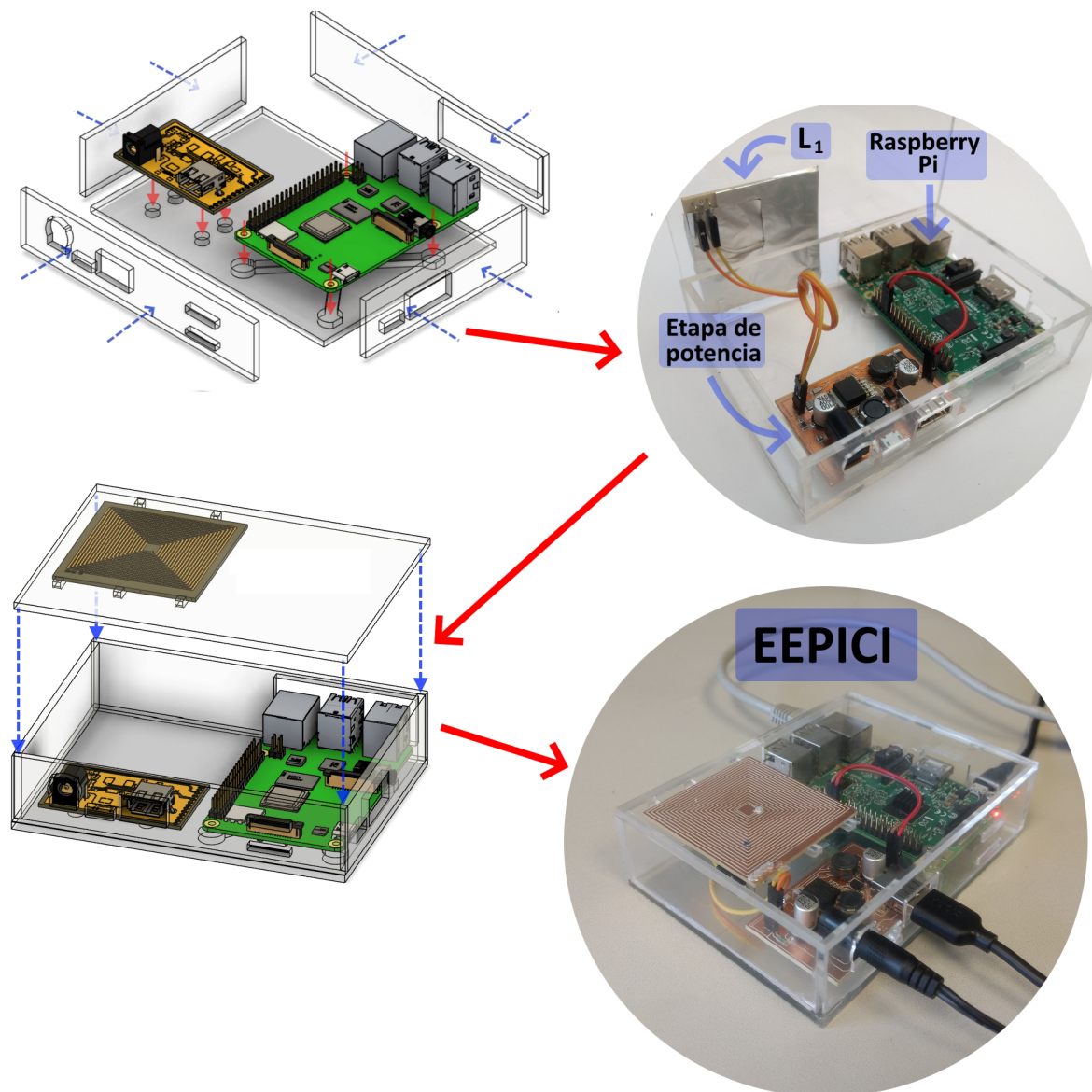


Figura 3.10: Montaje de EEPICI.

3.4.1. Resultado experimental

La prueba final de la Estación Electrónica consistió en energizar al prototipo de dispositivo BioMEMS y obtener los datos de pruebas generados por éste. En las figuras 3.11 y 3.12 se puede observar la energización del prototipo y la recepción de 4 datos consecutivos.

En la primera figura estos valores corresponden a los datos de prueba generados por el monitoreo con el potenciómetro fijo. En la segunda figura se ilustra nuevamente la energización y la obtención de datos de prueba; a diferencia de la primera figura, estos datos fueron generados incrementando el giro del potenciómetro mientras el prototipo realizaba el monitoreo.

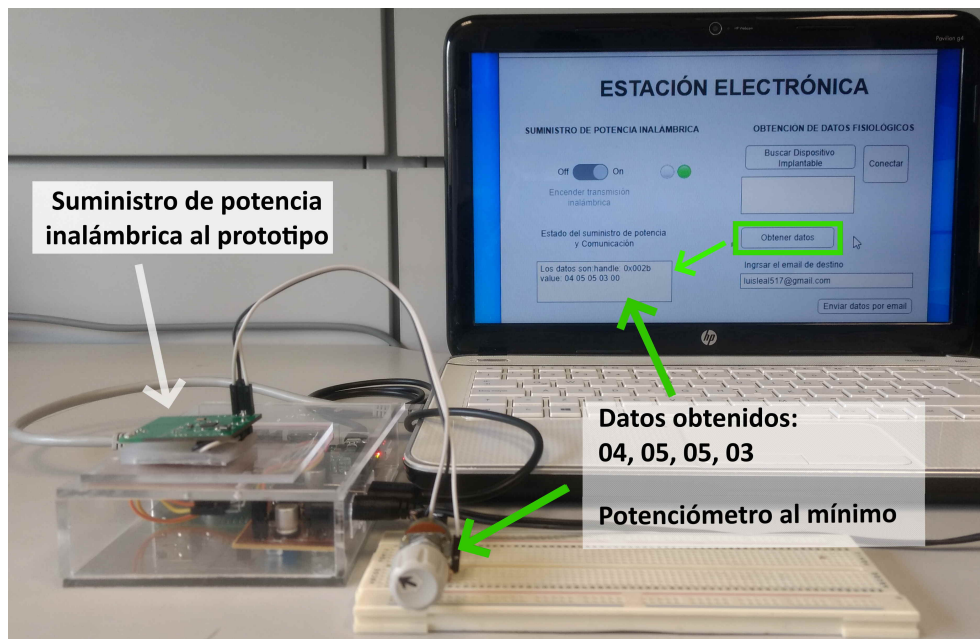


Figura 3.11: Prueba de la EEPICI energizando al prototipo y obteniendo los datos de prueba.

CAPÍTULO 3. PRUEBA DE CONCEPTO DE EEPICI

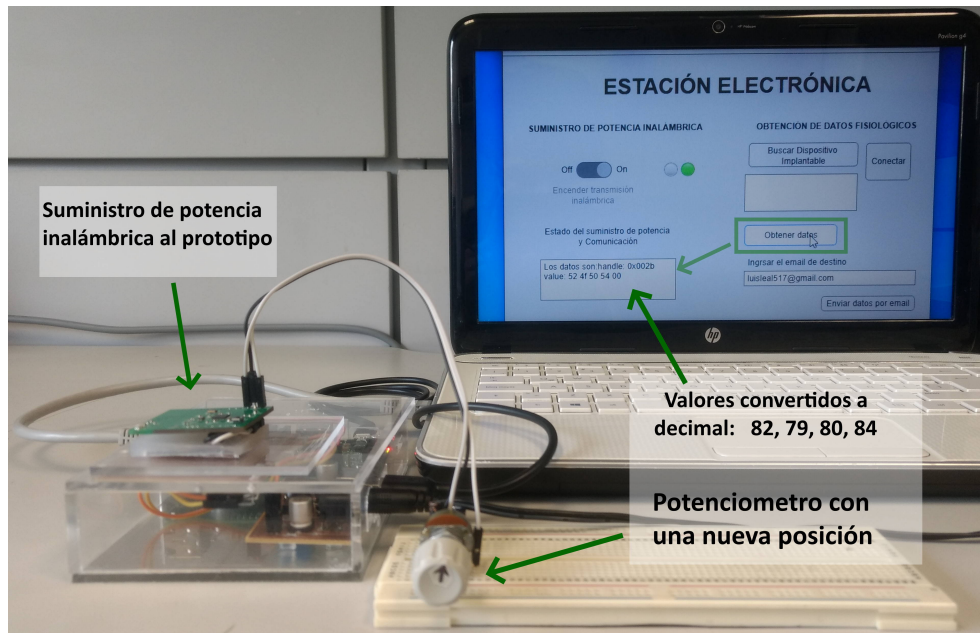


Figura 3.12: Prueba de la EEPICI energizando al prototipo y obteniendo los datos de prueba variables.

4. Conclusiones

En este trabajo se diseñó, desarrolló y probó una Estación Electrónica para transferir potencia inalámbrica y obtener datos de prueba a través de una comunicación inalámbrica hacia y desde un prototipo de dispositivo BioMEMS.

Primeramente, se investigaron las estrategias de transferencia de energía eléctrica enfocadas a dispositivos médicos implantables y las formas de transmisión de información para estos dispositivos. A partir de esta investigación, así como de los requisitos para energizar apropiadamente al prototipo y de las condiciones de una posible implantación futura, se realizó el diseño de un sistema de transferencia de potencia por acoplo inductivo de transmisores y receptores de potencia.

Igualmente, se presentaron algunos métodos del suministro de potencia hacia sistemas de acoplos inductivos, y se diseñó una etapa de potencia adecuada al sistema previamente diseñado bajo los requisitos y las condiciones anteriores. Además, se elaboró una metodología de fabricación de tarjetas electrónicas, misma que se usó en la elaboración de la etapa de potencia y los elementos transmisor y receptor.

Para el microcontrolador del prototipo se desarrolló un firmware que logró recolectar información de una variable física y permitió establecer una comunicación Bluetooth para enviar esta información como datos digitales de prueba.

CAPÍTULO 4. CONCLUSIONES

Al mismo tiempo, se elaboró una estrategia electrónica de control con su interfaz gráfica para solicitar los datos recolectados por el prototipo y almacenar estos datos de prueba en una hoja de cálculo para ser enviados hacia un servicio de correo electrónico. Además, en el software de control también se describió el código necesario para encender o apagar la transmisión de potencia inalámbrica.

La Estación Electrónica se ensambló y probó, y de ello se puede concluir lo siguiente:

En primer lugar, que es posible energizar al prototipo de dispositivo BioMEMS a una distancia máxima de 8mm, manteniendo en el lado del transmisor de potencia, una cantidad de energía emitida por debajo de los niveles seguros de radiación por campo magnético; con esto se consigue preservar la integridad de organismos biológicos, como los tejidos piel y fibras musculares. Además, bajo estas condiciones se logró entregar una potencia máxima de hasta $11.3mW$. No obstante, se comprobó que para energizar al prototipo adecuadamente, es necesario que la distancia de transmisión sea como máximo de 6mm ó, por otro lado, aumentar la cantidad del suministro de potencia al sistema de acoplo inductivo, aunque esto último infringe los límites seguros de SAR.

En segundo lugar, se exploró la habilitación de una comunicación inalámbrica por Bluetooth entre el prototipo de dispositivo BioMEMS y la Estación Electrónica; dicha comunicación fue aprovechada para obtener los datos de prueba que el prototipo recolectó de una variable física.

Aunque los experimentos realizados demuestran que es posible energizar al prototipo de dispositivo BioMEMS bajo las condiciones de implantación que respeta los niveles seguros de SAR y a una distancia máxima de 6mm, es necesario la caracterización de la cantidad de energía que es absorbida por los distintos tipos de tejidos, de fibras y de células de un organismo, y cómo este fenómeno en un ambiente real puede afectar el rendimiento de la potencia entregada al prototipo.

Finalmente, el desarrollo de esta Estación Electrónica, visto como un primer acercamiento para la recolección de datos procedentes de un prototipo y la transmisión de potencia hacia estos, forma un punto de partida esencial para el desarrollo de dispositivos implantables con tecnología BioMEMS.

Referencias

- [1] M. Gray, J. Meehan, C. Ward, S. P. Langdon, I. H. Kunkler, A. Murray, and D. Argyle, “Implantable biosensors and their contribution to the future of precision medicine,” *Veterinary Journal*, vol. 239, pp. 21–29, 9 2018.
- [2] R. Singh, M. J. Bathaei, E. Istif, and L. Beker, “A review of bioresorbable implantable medical devices: Materials, fabrication, and implementation,” *Advanced Healthcare Materials*, vol. 9, 9 2020.
- [3] K. Xu, S. Li, S. Dong, S. Zhang, G. Pan, G. Wang, L. Shi, W. Guo, C. Yu, and J. Luo, “Bioresorbable electrode array for electrophysiological and pressure signal recording in the brain.,” *Advanced Healthcare Materials*, vol. 8, no. 15, 2019.
- [4] K. Scholten and E. Meng, “A review of implantable biosensors for closed-loop glucose control and other drug delivery applications,” *International Journal of Pharmaceutics*, vol. 544, pp. 319–334, 6 2018.
- [5] D. R. Moore and R. V. Shannon, “Beyond cochlear implants: awakening the deafened brain,” *Nature Neuroscience*, vol. 12, pp. 686–691, 2009.
- [6] A. L. Benabid, S. Chabardes, J. Mitrofanis, and P. Pollak, “Deep brain stimulation of the subthalamic nucleus for the treatment of parkinson’s disease,” *The Lancet Neurology*, vol. 8, no. 1, pp. 67–81, 2009.

-
- [7] A. Jackson and J. B. Zimmermann, “Neural interfaces for the brain and spinal cord—restoring motor function,” *Nature Reviews Neurology*, vol. 8, no. 12, pp. 690–699, 2012.
- [8] K. N. Bocan and E. Sejdić, “Adaptive transcutaneous power transfer to implantable devices: A state of the art review,” *Sensors (Switzerland)*, vol. 16, 3 2016.
- [9] Y.-H. Joung, “Development of implantable medical devices: from an engineering perspective,” *International neurourology journal*, vol. 17, no. 3, p. 98, 2013.
- [10] X. Huang, D. Wang, Z. Yuan, W. Xie, Y. Wu, R. Li, Y. Zhao, D. Luo, L. Cen, B. Chen, *et al.*, “A fully biodegradable battery for self-powered transient implants,” *Small*, vol. 14, no. 28, p. 1800994, 2018.
- [11] J. Coffel and E. Nuxoll, “Biomems for biosensors and closed-loop drug delivery,” *International Journal of Pharmaceutics*, vol. 544, pp. 335–349, 6 2018.
- [12] M. D. Santis and I. Cacciotti, “Wireless implantable and biodegradable sensors for post-surgery monitoring: Current status and future perspectives,” *Nanotechnology*, vol. 31, 4 2020.
- [13] M. Luo, A. W. Martinez, C. Song, F. Herrault, and M. G. Allen, “A microfabricated wireless rf pressure sensor made completely of biodegradable materials,” *Journal of Microelectromechanical Systems*, vol. 23, no. 1, pp. 4–13, 2014.
- [14] B. D. Nelson, S. S. Karipott, Y. Wang, and K. G. Ong, “Wireless technologies for implantable devices,” *Sensors*, vol. 20, p. 4604, 8 2020.
- [15] Y. Hao and R. Foster, “Wireless body sensor networks for health-monitoring applications,” *Physiological measurement*, vol. 29, no. 11, p. R27, 2008.
- [16] J. A. Germn Girela, Diego Rosso, “Extracción percutánea de cables de marcapasos cardíaco y cardiodesfibriladores por vía percutánea percutaneous,” pp. 5–9.

REFERENCIAS

- [17] B. Cronin and M. K. Essandoh, “Update on cardiovascular implantable electronic devices for anesthesiologists,” *Journal of cardiothoracic and vascular anesthesia*, vol. 32, no. 4, pp. 1871–1884, 2018.
- [18] S. R. Khan, S. K. Pavuluri, G. Cummins, and M. P. Desmulliez, “Wireless power transfer techniques for implantable medical devices: A review,” *Sensors (Switzerland)*, vol. 20, pp. 1–58, 6 2020.
- [19] J. D. Kim, C. Sun, and I. S. Suh, “A proposal on wireless power transfer for medical implantable applications based on reviews,” pp. 166–169, IEEE Computer Society, 2014.
- [20] C. Dagdeviren, B. D. Yang, Y. Su, P. L. Tran, P. Joe, E. Anderson, J. Xia, V. Doraiswamy, B. Dehdashti, X. Feng, *et al.*, “Conformal piezoelectric energy harvesting and storage from motions of the heart, lung, and diaphragm,” *Proceedings of the National Academy of Sciences*, vol. 111, no. 5, pp. 1927–1932, 2014.
- [21] Z. Li, G. Zhu, R. Yang, A. C. Wang, and Z. L. Wang, “Muscle-driven in vivo nanogenerator,” *Advanced materials*, vol. 22, no. 23, pp. 2534–2537, 2010.
- [22] X. Cheng, X. Xue, Y. Ma, M. Han, W. Zhang, Z. Xu, H. Zhang, and H. Zhang, “Implantable and self-powered blood pressure monitoring based on a piezoelectric thinfilm: Simulated, in vitro and in vivo studies,” *Nano Energy*, vol. 22, pp. 453–460, 2016.
- [23] B. D. Ratner, A. S. Hoffman, F. J. Schoen, and J. E. Lemons, “Biomaterials science: an introduction to materials in medicine,” *San Diego, California*, pp. 162–4, 2004.
- [24] X. Qian, H. Hao, B. Ma, X. Wen, C. Hu, and L. Li, “Implanted rechargeable electroencephalography (eeg) device,” *Electronics Letters*, vol. 50, no. 20, pp. 1419–1421, 2014.
- [25] H. L. Krauss, C. W. Bostian, and F. H. Raab, *Solid state radio engineering*. John Wiley & Sons, 1991.

-
- [26] Q. Xu, T. Wang, S. Mao, W. Jia, Z.-H. Mao, and M. Sun, “Wireless power transfer for miniature implantable biomedical devices,” in *Recent Wireless Power Transfer Technologies*, IntechOpen, 2019.
- [27] P. W. Lamberti, “Las investigaciones de heinrich hertz sobre las ondas electromagnéticas,” *Revista de Enseñanza de la Física*, vol. 10, no. 2, pp. 37–47, 1997.
- [28] H. J. Kim, H. Hirayama, S. Kim, K. J. Han, R. Zhang, and J. W. Choi, “Review of near-field wireless power and communication for biomedical applications,” *IEEE Access*, vol. 5, pp. 21264–21285, 9 2017.
- [29] M. Ali, N. Ramli, and H. Nugroho, “A critical analysis of a wireless power transmission (wpt) with an improvement method for a non-radiative wpt,” *Indonesian Journal of Electrical Engineering and Informatics*, vol. 8, pp. 273–288, 6 2020.
- [30] K. Goto, T. Nakagawa, O. Nakamura, and S. Kawata, “An implantable power supply with an optically rechargeable lithium battery,” *IEEE Transactions on Biomedical Engineering*, vol. 48, no. 7, pp. 830–833, 2001.
- [31] M. Mujeeb-U-Rahman, D. Adalian, C.-F. Chang, and A. Scherer, “Optical power transfer and communication methods for wireless implantable sensing platforms,” *Journal of biomedical optics*, vol. 20, no. 9, p. 095012, 2015.
- [32] H. Basaeri, D. B. Christensen, and S. Roundy, “A review of acoustic power transfer for bio-medical implants,” *Smart Materials and Structures*, vol. 25, no. 12, p. 123001, 2016.
- [33] A. Sanni and A. Vilches, “Powering low-power implants using PZT transducer discs operated in the radial mode,” *Smart Materials and Structures*, vol. 22, p. 115005, oct 2013.

REFERENCIAS

- [34] S. Arra, J. Leskinen, J. Heikkila, and J. Vanhala, “Ultrasonic power and data link for wireless implantable applications,” in *2007 2nd International Symposium on Wireless Pervasive Computing*, 2007.
- [35] P. J. Larson and B. C. Towe, “Miniature ultrasonically powered wireless nerve cuff stimulator,” in *2011 5th International IEEE/EMBS Conference on Neural Engineering*, pp. 265–268, 2011.
- [36] F. Mazzilli, C. Lafon, and C. Dehollain, “A 10.5 cm ultrasound link for deep implanted medical devices,” *IEEE Transactions on Biomedical Circuits and Systems*, vol. 8, no. 5, pp. 738–750, 2014.
- [37] R. Shadid and S. Noghianian, “A literature survey on wireless power transfer for biomedical devices,” *International Journal of Antennas and Propagation*, vol. 2018, 2018.
- [38] S. Y. R. Hui, W. Zhong, and C. K. Lee, “A critical review of recent progress in mid-range wireless power transfer,” *IEEE Transactions on Power Electronics*, vol. 29, no. 9, pp. 4500–4511, 2013.
- [39] K. Agarwal, R. Jegadeesan, Y.-X. Guo, and N. V. Thakor, “Wireless power transfer strategies for implantable bioelectronics,” *IEEE reviews in biomedical engineering*, vol. 10, pp. 136–161, 2017.
- [40] D. Rozario, V. K. Pathipati, A. Ram, N. A. Azeez, and S. S. Williamson, “Modified resonant converters for contactless capacitive power transfer systems used in ev charging applications,” in *IECON 2016-42nd Annual Conference of the IEEE Industrial Electronics Society*, pp. 4510–4517, IEEE, 2016.
- [41] A. M. Sodagar and P. Amiri, “Capacitive coupling for power and data telemetry to implantable biomedical microsystems,” pp. 411–414, 2009.

-
- [42] R. Narayanamoorthi, “Modeling of capacitive resonant wireless power and data transfer to deep biomedical implants,” *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 9, no. 7, pp. 1253–1263, 2019.
- [43] M. Manoufali, K. Bialkowski, B. J. Mohammed, P. C. Mills, and A. Abbosh, “Near-field inductive-coupling link to power a three-dimensional millimeter-size antenna for brain implantable medical devices,” *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 1, pp. 4–14, 2018.
- [44] P. Feng, P. Yeon, Y. Cheng, M. Ghovanloo, and T. G. Constandinou, “Chip-scale coils for millimeter-sized bio-implants,” *IEEE Transactions on Biomedical Circuits and Systems*, vol. 12, no. 5, pp. 1088–1099, 2018.
- [45] G. Pan and L. Wang, “Swallowable wireless capsule endoscopy: Progress and technical challenges,” *Gastroenterology research and practice*, vol. 2012, 2011.
- [46] A. Mabrouki, M. Latrach, and V. Lorrain, “High efficiency low power rectifier design using zero bias schottky diodes,” in *2014 IEEE Faible Tension Faible Consommation*, pp. 1–4, 2014.
- [47] T. Sun, X. Xie, G. Li, Y. Gu, Y. Deng, and Z. Wang, “A two-hop wireless power transfer system with an efficiency-enhanced power receiver for motion-free capsule endoscopy inspection,” *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 11, pp. 3247–3254, 2012.
- [48] U.-M. Jow and M. Ghovanloo, “Modeling and optimization of printed spiral coils in air, saline, and muscle tissue environments,” *IEEE transactions on biomedical circuits and systems*, vol. 3, no. 5, pp. 339–347, 2009.
- [49] S. Mutashar, M. A. Hannan, S. A. Samad, and A. Hussain, “Analysis and optimization of spiral circular inductive coupling link for bio-implanted applications on air and within human tissue,” *Sensors*, vol. 14, no. 7, pp. 11522–11541, 2014.

REFERENCIAS

- [50] L. Cai and P. Gutruf, “Soft, wireless and subdermally implantable recording and neuromodulation tools,” *Journal of Neural Engineering*, vol. 18, 8 2021.
- [51] S. Mohan, M. del Mar Hershenson, S. Boyd, and T. Lee, “Simple accurate expressions for planar spiral inductances,” *IEEE Journal of Solid-State Circuits*, vol. 34, no. 10, pp. 1419–1424, 1999.
- [52] N. Soltani, M. ElAnsary, J. Xu, J. S. Filho, and R. Genov, “Safety-optimized inductive powering of implantable medical devices: Tutorial and comprehensive design guide,” *IEEE Transactions on Biomedical Circuits and Systems*, vol. 15, no. 6, pp. 1354–1367, 2021.
- [53] IEEE, “Ieee standard for safety levels with respect to human exposure to electric, magnetic and electromagnetic fields, 0 hz to 300 ghz,” *The Institute of Electrical and Electronics Engineers New York, NY*, no. C95. 1, pp. 1–312, 2019.
- [54] A. P. Malvino and D. J. Bates, *Principios de electrónica*, vol. 2. McGraw-Hill, 1991, pp. 381, 382.
- [55] P. Harden, “The handiman’s guide to mosfet switched mode amplifiers,” 2003. Scribd. <https://es.scribd.com/document/15467409/MOSFET-Switched-Mode-Amplifiers> (accedido en enero 2, 2023).
- [56] M. Machnoor and G. Lazzi, “Wireless power transfer: Types of reflected impedances and maximum power transfer theorem,” *IEEE Antennas and Wireless Propagation Letters*, vol. 19, no. 10, pp. 1709–1713, 2020.
- [57] R. Narayanamoorthi, A. Vimala Juliet, B. Chokkalingam, S. Padmanaban, Z. M. Leonowicz, *et al.*, “Class e power amplifier design and optimization for the capacitive coupled wireless power transfer system in biomedical implants,” *Energies*, vol. 10, no. 9, pp. 1–20, 2017.

-
- [58] N. Sokal, “Class-e switching-mode high-efficiency tuned rf/microwave power amplifier: improved design equations,” in *2000 IEEE MTT-S International Microwave Symposium Digest (Cat. No.00CH37017)*, vol. 2, pp. 779–782 vol.2, 2000.
- [59] A. Trigui, S. Hached, A. C. Ammari, Y. Savaria, and M. Sawan, “Maximizing data transmission rate for implantable devices over a single inductive link: Methodological review,” *IEEE Reviews in Biomedical Engineering*, vol. 12, pp. 72–87, 2019.
- [60] D. Svård, A. Cichocki, and A. Alvandpour, “Design and evaluation of a capacitively coupled sensor readout circuit, toward contact-less eeg and eeg,” in *2010 Biomedical Circuits and Systems Conference (BioCAS)*, pp. 302–305, 2010.
- [61] J. Rosenthal, A. Pike, and M. S. Reynolds, “A 1 mbps 158 pj/bit bluetooth low energy (ble) compatible backscatter communication uplink for wireless neural recording in an animal cage environment,” in *2019 IEEE International Conference on RFID (RFID)*, pp. 1–6, IEEE, 2019.
- [62] Y. Hao and R. Foster, “Wireless body sensor networks for health-monitoring applications,” *Physiological measurement*, vol. 29, no. 11, p. R27, 2008.
- [63] I. C. on Non-Ionizing Radiation Protection *et al.*, “Icnirp statement on the “guidelines for limiting exposure to time-varying electric, magnetic, and electromagnetic fields (up to 300 ghz)”,” *Health physics*, vol. 97, no. 3, pp. 257–258, 2009.
- [64] M. J. Christoe, J. Yuan, A. Michael, and K. Kalantar-Zadeh, “Bluetooth signal attenuation analysis in human body tissue analogues,” *IEEE Access*, vol. 9, pp. 85144–85150, 2021.
- [65] M. Woolley, “Bluetooth core specification version 5.0 feature enhancements,” *Bluetooth SIG*, 2021.

- [66] S. Hached, A. Trigui, I. El Khalloufi, M. Sawan, O. Loutochin, and J. Corcos, "A bluetooth-based low-energy qi-compliant battery charger for implantable medical devices," in *2014 IEEE International Symposium on Bioelectronics and Bioinformatics (IEEE ISBB 2014)*, pp. 1–4, 2014.
- [67] J. Kim, H. Kim, D. Kim, H.-J. Park, K. Ban, S. Ahn, and S.-M. Park, "A wireless power transfer based implantable ecg monitoring device," *Energies*, vol. 13, no. 4, p. 905, 2020.
- [68] D. Ahn and S. Hong, "Wireless power transmission with self-regulated output voltage for biomedical implant," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 5, pp. 2225–2235, 2014.
- [69] S. Mutashar, M. A. Hannan, S. A. Samad, and A. Hussain, "Design of spiral circular coils in wet and dry tissue for bio-implanted micro-system applications," *Progress In Electromagnetics Research M*, vol. 32, pp. 181–200, 2013.
- [70] K. van Oosterhout, M. Paulides, H. Pflug, S. Beumer, and R. Mestrom, "An approximate electromagnetic model for optimizing wireless charging of biomedical implants," *IEEE Transactions on Biomedical Engineering*, vol. 69, no. 6, pp. 1954–1963, 2022.
- [71] A. I. Al-Kalbani, M. R. Yuce, and J.-M. Redouté, "A biosafety comparison between capacitive and inductive coupling in biomedical implants," *IEEE Antennas and Wireless Propagation Letters*, vol. 13, pp. 1168–1171, 2014.
- [72] E. A. Mindubaev and I. S. Uzlov, "Experimental study of inductive energy transfer in strong coupling mode," in *2019 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EICoN Rus)*, pp. 2341–2344, 2019.

A. Código de Matlab

A.1. Código para la obtención de parámetros teóricos de L_1 y L_2 y potencia máxima permitida

Este código corresponde a los cálculos teóricos de los inductores L_1 y L_2 , que corresponden a los relevantes como la inductancia propia, sus resistencias intrínseca y propia, su factor de calidad y la potencia que cada uno puede entregar y recibir, respectivamente. Además, en este código se encuentra la generación de la gráfica que ilustra la potencia de transmisión inalámbrica máxima permitida así como la eficiencia en la transmisión, en función de la frecuencia a la que se transmite la energía.

```
1 %Parametros de relacion de diametros:
2 phi2 = 0.54;
3 phi1 = 0.92;
4
5 %Parametros de tablas fenolica de cobre
6 rho = 1.71*10^-8; %ohm m resistividad para el cobre
7 trx = 0.000035; %m 35um
8 ttx = 0.000035; %m 35um
9
10 %Parametros de Implantacion:
11 OD2 = 0.0175; %m Diametro de bobina Rx, restriccion de implantacion
12 RL = 172; %potencia entregada a una carga total: %204//1100
13 d1 = 0.01; %m Distancia entre bobinas, maxima de 1.0cm
14 Rs = 1; %Resistencia aproximada del inductor L1 en serie con la
    resistencia del mosfet: 0.8+5.2
15
```

ANEXO A. CÓDIGO DE MATLAB

```
16 %Factor de acoplamiento para enlace inductivo
17 k=0.1;          %Valor maximo (comprende un rango entre 0.1 y 0.01) para dl
18
19 %Coeficientes de expresion de Wheeler para inductor cuadrado
20 C1 = 1.27; C2 = 2.07; C3 = 0.18; C4 = 0.13;
21 u = pi*4*10^-7; %Constante de permeabilidad electrica del vacio
22 dprom2 = 0.5*OD2*(1+(1-phi2)/(1+phi2)); %parametro diametro promedio de L2
23
24 %Calculo de Parametros intrinsecos del inductor L2
25 R_2 = 4*rho/(phi2*trx); %resistencia por
    vuelta
26 L_2 = C1*u*dprom2*(log(C2/phi2)+C3*phi2+C4*(phi2)^2)/2; %Inductancia por
    vuelta
27
28 %Calculo de numero de vueltas de bobina Rx, pueden ser menos porque no
    afecta
29 %mucho en resultados de eficiencia y Pmax entregada
30 N22 = sqrt(RL/R_2); %Calculo aproximado a partir de una fabricacion
    que
31 N2 = 20; %es posible hacer en la maquina CNC Modela MDX-50
32
33 %Dimensiones del Inductor L1 OR1: Radio externo IR1: Radio interno
34 OR1 = 2.62*d1;
35 IR1 = 0.1*d1;
36
37 %Dimensiones de Inductor L1 OD1:Diametro externo; ID1: Diametro interno
38 OD1 = 2*OR1;
39 ID1 = 2*IR1;
40
41 %Dimensiones del Inductor L2 ID2: Diametro interno
42 ID2 = 0.37*OD2;
43 dprom1 = 0.5*OD1*(1+(1-phi1)/(1+phi1)); %parametro diametro promedio de L1
44
45 %Calculo de parametros intrinsecos para Inductor L1
46 R_1_1 = 4*rho/(phi1*ttx); %Resistencia por
    vuelta
47 L_1 = C1*u*dprom1*(log(C2/phi1)+C3*phi1+C4*(phi1)^2)/2; %Inductancia por
    vuelta
48
49 %Valores propios de Inductor L2
50 Rs2 = R_2*N2^2;
51 L2 = N2^2*L_2;
52
53 %Valores propios de Inductor L1
54 N1_1 = sqrt(Rs/R_1_1);
55 Rs1_1 = R_1_1*N1_1^2;
56 L1_1 = N1_1^2*L_1;
57
58 %Variable Frecuencia de operacion: para ver la respuesta del sistema en un
59 fop = logspace(6,8); %rango de frecuencias entre 1MHz a 100MHz
60
61 %Calculo de la potencia maxima (mW) desde el Inductor L1 con restriccion
    SAR
62 PTmax = 1000*((2*pi)^4*(OD2*OD2)^2*1.63^2)/(R_2);
63
64 %Calculo de la potencia maxima (mW) para el Inductor L2 con restriccion de
    SAR
65 PRmax = 1000*((2*pi)^4*(OD2*OD2)^2*1.63^2)/(R_2)*(4./(1+((2*pi.*fop*L2)./(
    RL+Rs2)).^2));
66
```

A.1. CÓDIGO PARA LA OBTENCIÓN DE PARÁMETROS TEÓRICOS DE L_1 Y L_2 Y POTENCIA MÁXIMA PERMITIDA

```

67 %Calculo de la potencia maxima (mW) en la Transmision de Potencia
    Inalambrica
68 %en funcion de la frecuencia
69 n = length(fop);
70 Ptotalmax = PRmax;
71 for i=1:n
72     if PRmax(i) > PTmax%
73         Ptotalmax(i) = PTmax;
74     end
75 end
76 Q1 = 2*pi*2400000*L1_1/Rs1_1;
77 Q2 = 2*pi*2400000*L2/Rs2;
78 %Eficiencia en la transmision de potencia en funcion de la frecuencia
79 n1 = 100*(1-2/(R_2*N2^2 + RL)*pi*L2.*fop/(2*pi.*fop*L2/(Rs2)))./(1+(k^2*2*
    pi.*fop*L1_1/(Rs1_1)*2/(R_2*N2^2 + RL)*pi*L2.*fop).^-1);
80
81 %Limite en la frecuencia para la transferencia de potencia maxima
82 fc = (sqrt(3))/(2*pi)*(RL+Rs2)/(L2);
83
84 figure(1) %Generacion de grafica de Potencia maxima y eficiencia en
    funcion de fpot
85 semilogx(fop,n1,'b');
86 hold on
87 ylabel('Eficiencia %');
88 yyaxis right;
89 semilogx(fop, Ptotalmax, 'y');
90 ylabel('Potencia Maxima (mW)');
91 xlabel('Frecuencia en Hz');
92 legend('Eficiencia del sistema WPT', 'Pmax con RL = 172 Ohms','Location',
    'best');
93 Disp2 = sprintf('Potencia maxima: %f mW hasta una frecuencia: %f MHz',
    PTmax, fc/1000000);
94 disp(Disp2);

```

A.2. Código para la implementación de la aplicación del Software Control de EEPICI

El siguiente código de Matlab genera una interfaz gráfica con la que es posible mandar las instrucciones para realizar la transmisión de potencia inalámbrica, buscar el prototipo de dispositivo BioMEMS, extraer los datos recabados por este último y, mandar estos datos en un archivo excel mediante correo electrónico.

```

1 classdef EEPICI < matlab.apps.AppBase
2
3     % Properties that correspond to app components
4     properties (Access = public)
5         UIFigure                matlab.ui.Figure
6         Menu                    matlab.ui.container.Menu
7         ConectarEstacinEEPICIMenu  matlab.ui.container.Menu
8         SalirMenu              matlab.ui.container.Menu
9         HerramientaMenu        matlab.ui.container.Menu
10        AyudaMenu              matlab.ui.container.Menu
11        OBTENCINDEDATOSFISIOLOGICOSLabel  matlab.ui.control.Label
12        SUMINISTRODEPOTENCIAINALMBRICALabel  matlab.ui.control.Label
13        ESTACINELECTRNICALabel  matlab.ui.control.Label
14        EstadodelsuministrodepotenciayComunicacinTextArea  matlab.ui.
            control.TextArea
15        EstadodelsuministrodepotenciayComunicacinLabel  matlab.ui.control.
            Label
16        PotenciaEntregadaGauge    matlab.ui.control.SemicircularGauge
17        PotenciaEntregadaGaugeLabel  matlab.ui.control.Label
18        Lamp2                    matlab.ui.control.Lamp
19        Lamp                    matlab.ui.control.Lamp
20        EnviardatosporemailButton  matlab.ui.control.Button
21        IngrsaremaildedestinoEditField  matlab.ui.control.EditField
22        IngrsaremaildedestinoEditFieldLabel  matlab.ui.control.Label
23        ObtenerdatosButton        matlab.ui.control.Button
24        ConectarButton           matlab.ui.control.Button
25        ListBox                  matlab.ui.control.ListBox
26        BuscarDispositivoImplantableButton  matlab.ui.control.Button
27        EncendertransmisininalmbricaSwitch  matlab.ui.control.Switch
28        EncendertransmisininalmbricaSwitchLabel  matlab.ui.control.Label
29    end
30
31    properties (Access = private)
32        EEPICI
33        led
34        data_temp
35        data
36        TablaExcel
37        archivo
38        mail
39        contrasena
40        email
41        ListDevices% Description
42        aux=0;
43    end
44    methods (Access = private)

```

A.2. CÓDIGO PARA LA IMPLEMENTACIÓN DE LA APLICACIÓN DEL SOFTWARE CONTROL DE EEPICI

```
45     end
46     % Callbacks that handle component events
47     methods (Access = private)
48     % Menu selected function: SalirMenu
49     function SalirMenuSelected(app, event)
50         %Funcion para salir del programa
51         clear app.EPICI;
52         close(app.UIFigure);
53     end
54
55     function ConectarEstacinEEPICIMenuSelected(app, event)
56     %Funcion para conectar Raspberry Pi a Matlab
57         clear app.EEPICI;
58         try
59             app.EEPICI = raspi();
60         catch
61             app.aux = 1;
62         end
63         if app.aux== 1
64             app.EstadodelsuministrodepotenciayComunicacinTextArea.
65                 Value = 'Sin conexion con Estacion Electronica';
66             fig = uifigure;
67             accion = uiconfirm(fig,'Hubo un error al tratar de
68                 Conectar la Estacion Electronica?','Error de
69                 Emparejamiento',...
70                 'Icon','error', 'Options',{'-Intentar de Nuevo', '
71                 Aceptar'});
72             if accion == "-Intentar de Nuevo"
73                 close(fig);
74                 ConectarEstacinEEPICIMenuSelected(app, event);
75             else
76                 close(fig);
77             end
78         else
79             app.EncendertransmisininalmbricaSwitch.Enable = 'on';
80             app.Lamp2.Color = [0.74,0.08,0.18];
81             app.led = app.EEPICI.AvailableLEDs{1};
82             writeLED(app.EEPICI, app.led, 0);
83             app.EstadodelsuministrodepotenciayComunicacinTextArea.
84                 Value = 'Estacion Conectada';
85         end
86     end
87     %Funcion para habilitar Bluetooth y escanear dispositivo
88     function BuscarDispositivoImplantableButtonPushed(app, event)
89         app.ListBox.Items = {'BioMEMS'};
90     end
91
92     function ConectarButtonPushed(app, event)
93     %Funcion para emparejar y conectar con prototipo BioMEMS
94         try
95             system(app.EEPICI, 'sudo bluetoothctl trust 04:EE:03:B4
96                 :77:42');
97         catch
98             app.aux = 3;
99         end
100        if app.aux == 3
101            app.aux = 0;
102            app.EstadodelsuministrodepotenciayComunicacinTextArea.Value
103                = 'El Prototipo no esta energizado';
104        else
```

ANEXO A. CÓDIGO DE MATLAB

```
98         app.EstadodelsuministrodepotenciayComunicacinTextArea.Value
          = 'El Prototipo se encuentra disponible';
99     end
100 end
101 %Funcion para encender o pagar la transmision de potencia
102 function EncendertransmisininalmbricaSwitchValueChanged(app, event
    )
103     value = app.EncendertransmisininalmbricaSwitch.Value;
104     if value == "On" %Codigo para usar
        PWM
105         system(app.EEPICI, 'gpio mode 26 pwm'); %Uso de GPIO 12
106         system(app.EEPICI, 'gpio pwmc 4'); %Uso de 2.4MHz (9.6
            MHz/4)
107         system(app.EEPICI, 'gpio pwm 26 512');%Asignando relor de
            9.2MHz
108         writeLED(app.EEPICI, app.led, 1);
109         app.EstadodelsuministrodepotenciayComunicacinTextArea.
            Value = 'Transmision de Potencia Activada';
110         app.Lamp.Color = [0 1 0];
111         app.Lamp2.Color = [0.9 0.9 0.9];
112     elseif value == "Off"
113         system(app.EEPICI, 'gpio mode 26 in'); %Apagar PWM
114         writeLED(app.EEPICI, app.led, 0);
115         app.EstadodelsuministrodepotenciayComunicacinTextArea.
            Value = 'Transmision de Potencia desactivada';
116         app.Lamp.Color = [0.9 0.9 0.9];
117         app.Lamp2.Color = [0.74,0.08,0.18];
118     end
119 end
120 %Funcion para mandar el archivo de excel por correo electronico:
121 function EnviardatosporemailButtonPushed(app, event)
122     app.mail = 'datosbiomems@gmail.com'; %correo
        remitente
123     app.contrasena = 'witebbsdgxftqfty'; %Proporcionanda por
        Google
124
125     %Configuracion del correo y servicio
126     setpref('Internet','E_mail', app.mail);
127     setpref('Internet','SMTP_Server','smtp.gmail.com');
128     setpref('Internet','SMTP_Username', app.mail);
129     setpref('Internet','SMTP_Password',app.contrasena);
130
131     %Comandos para generar el servicio de envio de correo
132     props = java.lang.System.getProperties;
133     props.setProperty('mail.smtp.auth','true');
134     props.setProperty('mail.smtp.socketFactory.class','javax.net.
        ssl.SSLSocketFactory');
135     props.setProperty('mail.smtp.socketFactory.port','465');
136
137     %Envio de correo electronico
138     app.email = strcat(app.IngrsarelemaildedestinoEditField.Value)
        ;
139     sendmail(app.email,'Resultados de datos biologicos (IINGEN)',
        app.data,'C:\Users\luisl\Documents\MATLAB\Biologicos.xlsx')
        ;
140     app.NoconectadaLabel.Text = 'email enviado';
141 end
142 %Funcion para obtener los datos biologicos procedentes del
    microcontrolador
```


A.2. CÓDIGO PARA LA IMPLEMENTACIÓN DE LA APLICACIÓN DEL SOFTWARE CONTROL DE EEPICI

```

143     function ObtenerdatosButtonPushed(app, event)
144         try %Comando gatttool que genera la lectura de la
145             %caracteristica "0000fff5-0000-1000-8000-00805f9b34fb"
146             %de bluetooth del prototipo. Esta caracteristica apunta
147             %a los datos almacenados en el microcontrolador y los
148             %envia por la caracteristica
149             app.data_temp = system(app.EEPICI, 'gatttool -b 04:EE:03:
                B4:77:42 --char-read -u 0000fff5-0000-1000-8000-00805
                f9b34fb');
150         catch
151             app.aux = 2;
152         end
153         if app.aux== 2
154             app.EstadodelsuministrodepotenciayComunicacinTextArea.
                Value = 'Se perdio la comunicacion con el Dispositivo
                BioMEMS';
155             fig2 = uifigure;
156             accion = uiconfirm(fig2,'El dispositivo BioMEMS no esta
                disponible','Error de comunicacion',...
                'Icon','error', 'Options',{ 'Aceptar'});
157             if accion == "Aceptar"
158                 close(fig2);
159             end
160         else
161             app.data = strcat('Los datos son: ',app.data_temp);
162             app.archivo = 'Biologicos.xlsx';
163             app.TablaExcel = {'Dato', 'Tiempo'; '1',app.data};
164             writecell(app.TablaExcel,app.archivo);
165             app.EstadodelsuministrodepotenciayComunicacinTextArea.
                Value = app.data;
166         end
167     end
168 end
169 % Component initialization
170 methods (Access = private)
171     % Create UIFigure and components
172     function createComponents(app)
173         % Create UIFigure and hide until all components are created
174         app.UIFigure = uifigure('Visible', 'off');
175         app.UIFigure.Position = [100 100 613 397];
176         app.UIFigure.Name = 'MATLAB App';
177         app.UIFigure.Scrollable = 'on';
178         app.UIFigure.Tag = 'Figure';
179     % Create Menu
180         app.Menu = uimenu(app.UIFigure);
181         app.Menu.Text = 'Menu'; % Create
182         ConectarEstacinEEPICIMenu
183         app.ConectarEstacinEEPICIMenu = uimenu(app.Menu);
184         app.ConectarEstacinEEPICIMenu.MenuSelectedFcn =
                createCallbackFcn(app, @ConectarEstacinEEPICIMenuSelected,
                true);
185         app.ConectarEstacinEEPICIMenu.Text = 'Conectar Estacion EEPICI
                ';
186     % Create SalirMenu
187         app.SalirMenu = uimenu(app.Menu);
188         app.SalirMenu.MenuSelectedFcn = createCallbackFcn(app,
                @SalirMenuSelected, true);
189         app.SalirMenu.Text = 'Salir';
190     % Create HerramientaMenu

```

ANEXO A. CÓDIGO DE MATLAB

```
191     app.HerramientaMenu = uimenu(app.UIFigure);
192     app.HerramientaMenu.Text = 'Herramienta';
193 % Create AyudaMenu
194     app.AyudaMenu = uimenu(app.HerramientaMenu);
195     app.AyudaMenu.Text = 'Ayuda';
196 % Create EncendertransmisininalmbricaSwitchLabel
197     app.EncendertransmisininalmbricaSwitchLabel = uilabel(app.
198     UIFigure);
199     app.EncendertransmisininalmbricaSwitchLabel.
200     HorizontalAlignment = 'center';
201     app.EncendertransmisininalmbricaSwitchLabel.Enable = 'off';
202     app.EncendertransmisininalmbricaSwitchLabel.Position = [68 189
203     122 28];
204     app.EncendertransmisininalmbricaSwitchLabel.Text = {'Encender
205     transmision'; 'inalambrica'};
206 % Create EncendertransmisininalmbricaSwitch
207     app.EncendertransmisininalmbricaSwitch = uiswitch(app.UIFigure
208     , 'slider');
209     app.EncendertransmisininalmbricaSwitch.ValueChangedFcn =
210     createCallbackFcn(app,
211     @EncendertransmisininalmbricaSwitchValueChanged, true);
212     app.EncendertransmisininalmbricaSwitch.Tag = 'SW_WPT_ON_OFF';
213     app.EncendertransmisininalmbricaSwitch.Enable = 'off';
214     app.EncendertransmisininalmbricaSwitch.Position = [105 227 52
215     23];
216 % Create BuscarDispositivoImplantableButton
217     app.BuscarDispositivoImplantableButton = uibutton(app.UIFigure
218     , 'push');
219     app.BuscarDispositivoImplantableButton.ButtonPushedFcn =
220     createCallbackFcn(app,
221     @BuscarDispositivoImplantableButtonPushed, true);
222     app.BuscarDispositivoImplantableButton.Tag = '
223     B_ScanDispositivos';
224     app.BuscarDispositivoImplantableButton.Position = [350 240 155
225     34];
226     app.BuscarDispositivoImplantableButton.Text = {'Buscar
227     Dispositivo '; 'Implantable'; ''};
228 % Create ListBox
229     app.ListBox = uilistbox(app.UIFigure);
230     app.ListBox.Items = {};
231     app.ListBox.Tag = 'LI_ListaDispositivos';
232     app.ListBox.Position = [346 168 160 53];
233     app.ListBox.Value = {};
234 % Create ConectarButton
235     app.ConectarButton = uibutton(app.UIFigure, 'push');
236     app.ConectarButton.ButtonPushedFcn = createCallbackFcn(app,
237     @ConectarButtonPushed, true);
238     app.ConectarButton.Tag = 'B_ConectarD';
239     app.ConectarButton.Position = [516 220 64 53];
240     app.ConectarButton.Text = 'Conectar';
241 % Create ObtenerdatosButton
242     app.ObtenerdatosButton = uibutton(app.UIFigure, 'push');
243     app.ObtenerdatosButton.ButtonPushedFcn = createCallbackFcn(app
244     , @ObtenerdatosButtonPushed, true);
245     app.ObtenerdatosButton.Tag = 'B_ObtenerDatos';
246     app.ObtenerdatosButton.Position = [349 115 132 30];
247     app.ObtenerdatosButton.Text = 'Obtener datos';
248 % Create IngresarelemailedestinoEditFieldLabel
```

A.2. CÓDIGO PARA LA IMPLEMENTACIÓN DE LA APLICACIÓN DEL SOFTWARE CONTROL DE EEPICI

```

233     app.IngrsarelemailedestinoEditFieldLabel = uilabel(app.
        UIFigure);
234     app.IngrsarelemailedestinoEditFieldLabel.HorizontalAlignment
        = 'right';
235     app.IngrsarelemailedestinoEditFieldLabel.Position = [346 83
        146 22];
236     app.IngrsarelemailedestinoEditFieldLabel.Text = 'Ingrsar el
        email de destino';
237 % Create IngrsarelemailedestinoEditField
238     app.IngrsarelemailedestinoEditField = uieditfield(app.
        UIFigure, 'text');
239     app.IngrsarelemailedestinoEditField.Tag = 'T_email';
240     app.IngrsarelemailedestinoEditField.Position = [346 60 246
        22];
241     app.IngrsarelemailedestinoEditField.Value = 'luisleal***
        @gmail.com';
242 % Create EnviardatosporemailButton
243     app.EnviardatosporemailButton = uibutton(app.UIFigure, 'push')
        ;
244     app.EnviardatosporemailButton.ButtonPushedFcn =
        createCallbackFcn(app, @EnviardatosporemailButtonPushed,
        true);
245     app.EnviardatosporemailButton.Tag = 'B_EnviarDatos';
246     app.EnviardatosporemailButton.Position = [456 20 135 22];
247     app.EnviardatosporemailButton.Text = 'Enviar datos por email';
248 % Create Lamp
249     app.Lamp = uilamp(app.UIFigure);
250     app.Lamp.Tag = 'LM_ON';
251     app.Lamp.Position = [254 228 20 20];
252     app.Lamp.Color = [0.902 0.902 0.902];
253 % Create Lamp2
254     app.Lamp2 = uilamp(app.UIFigure);
255     app.Lamp2.Tag = 'LM_OFF';
256     app.Lamp2.Position = [230 228 20 20];
257     app.Lamp2.Color = [0.902 0.902 0.902];
258 % Create PotenciaEntregadaGaugeLabel
259     app.PotenciaEntregadaGaugeLabel = uilabel(app.UIFigure);
260     app.PotenciaEntregadaGaugeLabel.HorizontalAlignment = 'center'
        ;
261     app.PotenciaEntregadaGaugeLabel.Visible = 'off';
262     app.PotenciaEntregadaGaugeLabel.Position = [265 17 64 28];
263     app.PotenciaEntregadaGaugeLabel.Text = {'Potencia'; 'Entregada
        '};
264 % Create PotenciaEntregadaGauge
265     app.PotenciaEntregadaGauge = uigauge(app.UIFigure, '
        semicircular');
266     app.PotenciaEntregadaGauge.Tag = 'G_PotenciaD';
267     app.PotenciaEntregadaGauge.Visible = 'off';
268     app.PotenciaEntregadaGauge.ScaleColors = [1 0 0;1 1 0;0.3922
        0.8314 0.0745];
269     app.PotenciaEntregadaGauge.ScaleColorLimits = [0 30;30 60;60
        100];
270     app.PotenciaEntregadaGauge.Position = [272 44 51 27];
271 % Create EstadodelsuministrodepotenciayComunicacinLabel
272     app.EstadodelsuministrodepotenciayComunicacinLabel = uilabel(
        app.UIFigure);
273     app.EstadodelsuministrodepotenciayComunicacinLabel.
        HorizontalAlignment = 'center';

```

ANEXO A. CÓDIGO DE MATLAB

```
274     app.EstadodelsuministrodepotenciayComunicacinLabel.Position =
275     [58 119 186 48];
276     app.EstadodelsuministrodepotenciayComunicacinLabel.Text = {'
277     Estado del suministro de potencia '; 'y Comunicacion'};
278     % Create EstadodelsuministrodepotenciayComunicacinTextArea
279     app.EstadodelsuministrodepotenciayComunicacinTextArea =
280     uitextarea(app.UIFigure);
281     app.EstadodelsuministrodepotenciayComunicacinTextArea.Tag = '
282     TA_InfoWPT';
283     app.EstadodelsuministrodepotenciayComunicacinTextArea.Position
284     = [53 51 196 60];
285     % Create ESTACINELECTRNICALabel
286     app.ESTACINELECTRNICALabel = uilabel(app.UIFigure);
287     app.ESTACINELECTRNICALabel.FontSize = 30;
288     app.ESTACINELECTRNICALabel.FontWeight = 'bold';
289     app.ESTACINELECTRNICALabel.Position = [143 338 389 36];
290     app.ESTACINELECTRNICALabel.Text = 'ESTACIoN ELECTRoNICA';
291     % Create SUMINISTRODEPOTENCIAINALMBRICALabel
292     app.SUMINISTRODEPOTENCIAINALMBRICALabel = uilabel(app.UIFigure
293     );
294     app.SUMINISTRODEPOTENCIAINALMBRICALabel.FontWeight = 'bold';
295     app.SUMINISTRODEPOTENCIAINALMBRICALabel.Position = [36 286 256
296     22];
297     app.SUMINISTRODEPOTENCIAINALMBRICALabel.Text = 'SUMINISTRO DE
298     POTENCIA INALaMBRICA';
299     % Create OBTENCINDEDATOSFISIOLOGICOSLabel
300     app.OBTENCINDEDATOSFISIOLOGICOSLabel = uilabel(app.UIFigure);
301     app.OBTENCINDEDATOSFISIOLOGICOSLabel.FontWeight = 'bold';
302     app.OBTENCINDEDATOSFISIOLOGICOSLabel.Position = [360 286 234
303     22];
304     app.OBTENCINDEDATOSFISIOLOGICOSLabel.Text = 'OBTENCIoN DE DATOS
305     FISIOLoGICOS';
306     % Show the figure after all components are created
307     app.UIFigure.Visible = 'on';
308     end
309     end
310     % App creation and deletion
311     methods (Access = public)
312     % Construct app
313     function app = EPICI
314     % Create UIFigure and components
315     createComponents(app)
316     % Register the app with App Designer
317     registerApp(app, app.UIFigure)
318     if nargin == 0
319         clear app
320     end
321     end
322     function delete(app)
323     % Delete UIFigure when app is deleted
324     delete(app.UIFigure)
325     end
326     end
327     end
```

B. Cálculos matemáticos

B.1. Potencia adicional debido a la eficiencia del acoplo inductivo

Considerando que el acoplo inductivo tiene una eficiencia de $\eta = 30\%$, o expresado como $\eta = 0.3$, la cantidad de potencia entregada en el extremo receptor sería $\eta \cdot P_{WPT}$, como se muestra en la ecuación (B.1). Sin embargo, en el extremo receptor es necesario que sea entregada la potencia P_{WPT} . Por lo que se plantea la relación (B.2) para calcular la cantidad de potencia total que se debe suministrar desde el extremo receptor para entregar dicha potencia hasta el final del acoplo inductivo, como se muestra a continuación.

$$\begin{array}{rcl} P_{Transmitida} & & P_{Recibida} \\ P_{WPT} & \rightarrow & 0.3 \cdot P_{WPT} \end{array} \quad (B.1)$$

$$P_{Total} \rightarrow P_{WPT} \quad (B.2)$$

Para calcular la potencia total, se hace una interpolación (o regla de tres) con las expresiones anteriores:

$$P_{Total} = \frac{P_{WPT} \cdot P_{WPT}}{0.3} = 3.3P_{WPT} \quad (B.3)$$

ANEXO B. CÁLCULOS MATEMÁTICOS

Ahora, para obtener sólo la potencia de compensación, se hace la siguiente resta:

$$P_{compensacion} = P_{Total} - P_{WPT} = 3.3P_{WPT} - P_{WPT} = 2.3P_{WPT} \quad (B.4)$$

Lo anterior indica que la potencia de compensación representa 230% de P_{WPT} , es decir, para poder entregar en el lado receptor la potencia estimada, es necesario, adicionar 2.3 veces P_{WPT} en el lado transmisor.

C. Códigos de C

C.1. Desarrollo del Monitoreo y Almacenamiento en Sensor Controller Studio

La inicialización del controlador de sensores está descrito para el proyecto *ADC Data Logger for LaunchPad* en *Sensor Controller Studio*, en el siguiente código, mismo que debe colocarse en el apartado *Initialization code* del proyecto.

```
1 //Codigo de inicio de tarea
2 adcSelectGpioInput(AUXIO_A_SENSOR_OUTPUT); //GPIO7 como entrada ADC
3 fwScheduleTask(1); //Crea la tarea numero 1
```

El siguiente código corresponde a la ejecución de la tarea hace una muestra de la variable física, la convierte a un dato digital y la almacena. Dicho código corresponde al apartado *Execution Code* del proyecto.

```
1 //Esta es la tarea principal o tarea 1 del Controlador de Sensores
2
3 fwDelayUs(100000, FW_DELAY_RANGE_100_MS); //Crea un delay inicial
4 //habilita adc
5 adcEnableSync(ADC_REF_FIXED, ADC_SAMPLE_TIME_2P7_US, ADC_TRIGGER_MANUAL);
6 U16 n=0;
7 U16 del=0; //Para el conteo del buffer
8
9
10 n = output.head;
11 adcGenManualTrigger(); //Se activa el ADC
```

ANEXO C. CÓDIGOS DE C

```
12 adcReadFifo(output.pSamples[n]);           //lee el valor almacenado en fifo
13 output.Value[n] = output.pSamples[n];     //Almacena valor en variable
14 utilIncrAndWrap(n, BUFFER_SIZE; output.head); //Aumento para nueva muestra
15     while (del < delay) {                 //Delay de 1 segundo
16         fwDelayUs(100000, FW_DELAY_RANGE_100_MS);
17         del=del+1;
18     }
19 n = output.head;
20 adcGenManualTrigger();                    //Se activa el ADC
21 adcReadFifo(output.pSamples[n]);         //lee el valor almacenado en fifo
22 output.Value[n] = output.pSamples[n];    //Almacena valor en variable
23 utilIncrAndWrap(n, BUFFER_SIZE; output.head); //Aumento para nueva muestra
24     while (del < delay) {                 //Delay de 1 segundo
25         fwDelayUs(100000, FW_DELAY_RANGE_100_MS);
26         del=del+1;
27     }
28 n = output.head;
29 adcGenManualTrigger();                    //Se activa el ADC
30 adcReadFifo(output.pSamples[n]);         //lee el valor almacenado en fifo
31 output.Value[n] = output.pSamples[n];    //Almacena valor en variable
32 utilIncrAndWrap(n, BUFFER_SIZE; output.head); //Aumento para nueva muestra
33     while (del < delay) {                 //Delay de 1 segundo
34         fwDelayUs(100000, FW_DELAY_RANGE_100_MS);
35         del=del+1;
36     }
37 n = output.head;
38 adcGenManualTrigger();                    //Se activa el ADC
39 adcReadFifo(output.pSamples[n]);         //lee el valor almacenado en fifo
40 output.Value[n] = output.pSamples[n];    //Almacena valor en variable
41 utilIncrAndWrap(n, BUFFER_SIZE; output.head); //Aumento para nueva muestra
42     while (del < delay) {                 //Delay de 1 segundo
43         fwDelayUs(100000, FW_DELAY_RANGE_100_MS);
44         del=del+1;
45     }
46 adcDisable();                             //desactiva el adc (lo deja suspendido)
47 fwGenAlertInterrupt(); //Manda interrupci n al microcontrolador
    principal
48 utilSwapEndianess(0; output.head);
```

C.2. Código del archivo ejecutable main.c

El siguiente código es un archivo fuente que es parte del proyecto *Simpleperipheral*. En este código se modificó la forma de encender al microcontrolador y mantenerlo activo en todo momento, sin que entre en el estado operativo de *Standby*.

```
1 // INCLUDES
2 #include <xdc/runtime/Error.h>
```


C.2. CÓDIGO DEL ARCHIVO EJECUTABLE MAIN.C

```
3 #include <ti/drivers/Power.h>
4 #include <ti/drivers/power/PowerCC26XX.h>
5 #include <ti/sysbios/BIOS.h>
6 #include <ti/sysbios/knl/Clock.h>
7 #include <icall.h>
8 #include "bcomdef.h"
9 #include "peripheral.h"
10 #include "simple_peripheral.h"
11 #include <inc/hw_memmap.h>
12 #include <driverlib/vims.h>
13 #include "ble_user_config.h"
14
15 /* ARCHIVOS DE CABEZA PARA ENCENDER UN LED DE PRUEBA */
16 #include <ti/drivers/PIN.h>
17 #include <ti/drivers/pin/PINCC26XX.h>
18
19 // BLE user defined configuration
20 #ifndef USE_DEFAULT_USER_CFG
21 #ifdef ICALL_JT
22 icall_userCfg_t user0Cfg = BLE_USER_CFG;
23 #endif // USE_DEFAULT_USER_CFG
24 //*****VARIABLE PARA MANEJAR EL PIN 8 COMO UN LED DE PRUEBA*****
25
26 PIN_Handle hPins = 0; //MANEJADOR DE PIN DE SALIDA PARA LED DE PRUEBA
27
28 int main()
29 {
30 Board_initGeneral(); //INICIALIZADOR DE PINES DE LA TARJETA
31 //HABILITA EL PIN 8 COMO SALIDA CON UN VALOR BAJO
32 PIN_setOutputEnable(hPins, Board_PIN_LED1, true);
33 PIN_setOutputValue(hPins, Board_PIN_LED1, Board_GPIO_LED_OFF);
34
35 // Enable iCache prefetching
36 VIMSConfigure(VIMS_BASE, TRUE, TRUE);
37 // Enable cache
38 VIMSMODESet(VIMS_BASE, VIMS_MODE_ENABLED);
39
40 #ifdef ICALL_JT
41 /* Update User Configuration of the stack */
42 user0Cfg.appServiceInfo->timerTickPeriod = Clock_tickPeriod;
43 user0Cfg.appServiceInfo->timerMaxMillisecond = ICall_getMaxMsecs();
44 #endif /* ICALL_JT */
45
46 /* Initialize ICall module */
47 ICall_init();
48
49 /* Start tasks of external images - Priority 5 */
50 ICall_createRemoteTasks();
51
52 /* Kick off profile - Priority 3 */
53 GAPRole_createTask();
54
55 //CREACION DE LA TAREA 2, HABILITACION DE LA COMUNICACION BLE
56 SimplePeripheral_createTask();
57
58 /* enable interrupts and start SYS/BIOS */
59 BIOS_start();
60 return 0;
61 }
```

ANEXO C. CÓDIGOS DE C

```
62 /*
63  @brief      Error handler to be hooked into TI-RTOS.
64  */
65 void smallErrorHook(Error_Block *eb)
66 {
67     for (;;)
68 }
```

C.3. Código del archivo ejecutable `simple_peripheral.c`

Este código corresponde al archivo fuente principal del kit de desarrollo *Simpleperipheral*. En este se desarrolló propiamente el firmware del microcontrolador para su funcionalidad de monitoreo y conectividad BLE.

```

1 // INCLUDES
2 #include <string.h>
3 #include <ti/sysbios/knl/Task.h>
4 #include <ti/sysbios/knl/Clock.h>
5 #include <ti/sysbios/knl/Event.h>
6 #include <ti/sysbios/knl/Queue.h>
7 #include <ti/display/Display.h>
8 #include <icall.h>
9 #include "util.h"
10 #include "att_rsp.h"
11
12 /* This Header file contains all BLE API and icall structure definition */
13 #include "icall_ble_api.h"
14 #include "devinfoservice.h"
15 #include "simple_gatt_profile.h"
16 #include "ll_common.h"
17
18 #include "peripheral.h"
19 #include "board_key.h"
20
21 #include "board.h"
22 #include "simple_peripheral.h"
23
24 /* ARCHIVOS FUENTE PARA INCLUIR FUUNCIONES DE ADC DEL SENSOR CONTROLLER*/
25
26 #include "scif.h"
27 //*****
28
29 // Advertising interval when device is discoverable (625us, 160=100ms)
30 #define DEFAULT_ADVERTISING_INTERVAL 160
31
32 // General discoverable mode: advertise indefinitely
33 #define DEFAULT_DISCOVERABLE_MODE GAP_ADTYPE_FLAGS_GENERAL
34
35 // Minimum connection interval (units of 1.25ms, 80=100ms) for automatic
36 // parameter update request
37 #define DEFAULT_DESIRED_MIN_CONN_INTERVAL 80
38
39 // Maximum connection interval ( of 1.25ms, 800=1000ms) for automatic
40 // parameter update request
41 #define DEFAULT_DESIRED_MAX_CONN_INTERVAL 800
42
43 // Slave latency to use for automatic parameter update request
44 #define DEFAULT_DESIRED_SLAVE_LATENCY 0
45
46 // Supervision timeout value (of 10ms, 1000=10s) for automatic parameter
47 // update request

```

ANEXO C. CÓDIGOS DE C

```
48 #define DEFAULT_DESIRED_CONN_TIMEOUT          1000
49
50 // After the connection is formed, the peripheral waits until the central
51 // device asks for its preferred connection parameters
52 #define DEFAULT_ENABLE_UPDATE_REQUEST
53     GAPROLE_LINK_PARAM_UPDATE_WAIT_REMOTE_PARAMS
54 // Connection Pause Peripheral time value (in seconds)
55 #define DEFAULT_CONN_PAUSE_PERIPHERAL        6
56
57 // How often to perform periodic event (in msec)
58 #define SBP_PERIODIC_EVT_PERIOD              5000
59
60 // Application specific event ID for HCI Connection Event End Events
61 #define SBP_HCI_CONN_EVT_END_EVT            0x0001
62
63 // Type of Display to open
64 #if !defined(Display_DISABLE_ALL)
65     #if defined(BOARD_DISPLAY_USE_LCD) && (BOARD_DISPLAY_USE_LCD!=0)
66         #define SBP_DISPLAY_TYPE Display_Type_LCD
67     #elif defined(BOARD_DISPLAY_USE_UART) && (BOARD_DISPLAY_USE_UART!=0)
68         #define SBP_DISPLAY_TYPE Display_Type_UART
69     #else // !BOARD_DISPLAY_USE_LCD && !BOARD_DISPLAY_USE_UART
70         #define SBP_DISPLAY_TYPE 0 // Option not supported
71     #endif // BOARD_DISPLAY_USE_LCD && BOARD_DISPLAY_USE_UART
72 #else // BOARD_DISPLAY_USE_LCD && BOARD_DISPLAY_USE_UART
73     #define SBP_DISPLAY_TYPE 0 // No Display
74 #endif // !Display_DISABLE_ALL
75
76 // Task configuration
77 #define SBP_TASK_PRIORITY                    1
78 #ifndef SBP_TASK_STACK_SIZE
79 #define SBP_TASK_STACK_SIZE                  644
80 #endif
81 // Application events
82 #define SBP_STATE_CHANGE_EVT                 0x0001
83 #define SBP_CHAR_CHANGE_EVT                 0x0002
84 #define SBP_PAIRING_STATE_EVT               0x0004
85 #define SBP_PASSCODE_NEEDED_EVT            0x0008
86 #define SBP_CONN_EVT                        0x0010
87
88 // Internal Events for RTOS application
89 #define SBP_ICALL_EVT                        ICALL_MSG_EVENT_ID
90 #define SBP_QUEUE_EVT                       UTIL_QUEUE_EVENT_ID
91 #define SBP_PERIODIC_EVT                    Event_Id_00
92 #define SBP_DATA_READY_EVT                  Event_Id_01
93
94 // Bitwise OR of all events to pend on
95 #define SBP_ALL_EVENTS                       (SBP_ICALL_EVT           | \
96                                             SBP_QUEUE_EVT           | \
97                                             SBP_DATA_READY_EVT      | \
98                                             SBP_PERIODIC_EVT)
99
100 // Set the register cause to the registration bit-mask
101 #define CONNECTION_EVENT_REGISTER_BIT_SET(RegisterCause) (
102     connectionEventRegisterCauseBitMap |= RegisterCause )
103 // Remove the register cause from the registration bit-mask
104 #define CONNECTION_EVENT_REGISTER_BIT_REMOVE(RegisterCause) (
105     connectionEventRegisterCauseBitMap &= (~RegisterCause) )
106 // Gets whether the current App is registered to the receive connection
107 // events
```

C.3. CÓDIGO DEL ARCHIVO EJECUTABLE SIMPLE_PERIPHERIAL.C

```
105 #define CONNECTION_EVENT_IS_REGISTERED (connectionEventRegisterCauseBitMap
    > 0)
106 // Gets whether the RegisterCause was registered to receive connection
    event
107 #define CONNECTION_EVENT_REGISTRATION_CAUSE(RegisterCause) (
    connectionEventRegisterCauseBitMap & RegisterCause )
108
109 // App event passed from profiles.
110 typedef struct
111 {
112     appEvtHdr_t hdr; // event header.
113     uint8_t *pData; // event data
114 } sbpEvt_t;
115
116 /*****
117  * GLOBAL VARIABLES
118  */
119 uint16_t i=0; //VARIABLE PARA LA EN SIMA MUESTRA
120 uint8_t muestras[5]={0,0,0,0,0}; //ARREGLO QUE ALMACENA 4 VALORES
121 uint8_t muestraUnica=0;
122
123 // Display Interface
124 Display_Handle dispHandle = NULL;
125
126 //Entity ID globally used to check for source and destination of messages
127 static ICall_EntityID selfEntity;
128
129 // Event globally used to post local events and pend on system and
130 // local events.
131 static ICall_SyncHandle syncEvent;
132
133 // Clock instances for internal periodic events.
134 static Clock_Struct periodicClock;
135
136 // Queue object used for app messages
137 static Queue_Struct appMsg;
138 static Queue_Handle appMsgQueue;
139
140 // Task configuration
141 Task_Struct sbpTask;
142 Char sbpTaskStack[SBP_TASK_STACK_SIZE];
143
144 // Scan response data (max size = 31 bytes)
145 static uint8_t scanRspData[] =
146 {
147     // complete name
148     0x14, // length of this data
149     GAP_ADTYPE_LOCAL_NAME_COMPLETE,
150     'D','i','s','p','o','s','i','t','i','v','o','_',
151     'B','i','o','M','E','M','S',
152
153     // connection interval range
154     0x05, // length of this data
155     GAP_ADTYPE_SLAVE_CONN_INTERVAL_RANGE,
156     LO_UINT16(DEFAULT_DESIRED_MIN_CONN_INTERVAL), // 100ms
157     HI_UINT16(DEFAULT_DESIRED_MIN_CONN_INTERVAL),
158     LO_UINT16(DEFAULT_DESIRED_MAX_CONN_INTERVAL), // 1s
159     HI_UINT16(DEFAULT_DESIRED_MAX_CONN_INTERVAL),
160
161     // Tx power level
162     0x02, // length of this data
163     GAP_ADTYPE_POWER_LEVEL,
```

ANEXO C. CÓDIGOS DE C

```
164     0          // 0dBm
165 };
166
167 // Advertisement data (max size = 31 bytes, though this is
168 // best kept short to conserve power while advertising)
169 static uint8_t advertData[] =
170 {
171     // Flags: this field sets the device to use general discoverable
172     // mode (advertises indefinitely) instead of general
173     // discoverable mode (advertise for 30 seconds at a time)
174     0x02, // length of this data
175     GAP_ADTYPE_FLAGS,
176     DEFAULT_DISCOVERABLE_MODE | GAP_ADTYPE_FLAGS_BREDR_NOT_SUPPORTED,
177
178     // service UUID, to notify central devices what services are included
179     // in this peripheral
180     0x03, // length of this data
181     GAP_ADTYPE_16BIT_MORE, // some of the UUID's, but not all
182     LO_UINT16(SIMPLEPROFILE_SERV_UUID),
183     HI_UINT16(SIMPLEPROFILE_SERV_UUID)
184 };
185
186 // GAP GATT Attributes
187 static uint8_t attDeviceName[GAP_DEVICE_NAME_LEN] = "Term metro BioMEMS";
188
189 /*****
190 * LOCAL FUNCTIONS
191 */
192
193 static void SimplePeripheral_init( void );
194 static void SimplePeripheral_taskFxn(UArg a0, UArg a1);
195
196 static uint8_t SimplePeripheral_processStackMsg(ICall_Hdr *pMsg);
197 static uint8_t SimplePeripheral_processGATTMsg(gattMsgEvent_t *pMsg);
198 static void SimplePeripheral_processAppMsg(sbpEvt_t *pMsg);
199 static void SimplePeripheral_processStateChangeEvt(gaprole_States_t
200     newState);
201 static void SimplePeripheral_processCharValueChangeEvt(uint8_t paramID);
202 static void SimplePeripheral_performPeriodicTask(void);
203 static void SimplePeripheral_clockHandler(UArg arg);
204
205 static void SimplePeripheral_passcodeCB(uint8_t *deviceAddr,
206     uint16_t connHandle, uint8_t uiInputs,
207     uint8_t uiOutputs, uint32_t numComparison);
208 static void SimplePeripheral_pairStateCB(uint16_t connHandle,
209     uint8_t state, uint8_t status);
210 static void SimplePeripheral_processPairState(uint8_t state,
211     uint8_t status);
212 static void SimplePeripheral_processPasscode(uint8_t uiOutputs);
213
214 static void SimplePeripheral_stateChangeCB(gaprole_States_t newState);
215 static void SimplePeripheral_charValueChangeCB(uint8_t paramID);
216 static uint8_t SimplePeripheral_enqueueMsg(uint8_t event, uint8_t state,
217     uint8_t *pData);
218
219 static void SimplePeripheral_connEvtCB(Gap_ConnEventRpt_t *pReport);
220 static void SimplePeripheral_processConnEvt(Gap_ConnEventRpt_t *pReport);
221
222 //FUNCIONES RELACIONADAS AL CONTROLADOR DE SENSORES
223 void scCtrlReadyCallback(void); //FUCCION DE CONTROL DE TAREAS LISTO
```

C.3. CÓDIGO DEL ARCHIVO EJECUTABLE SIMPLE_PERIPHERAL.C

```
223 void scTaskAlertCallback(void); //FUNCION DE CONTROL DE INTERRUPCIONES
224
225 void SensorController_Init(void); //FUNCION DE INCIO DEL CONTROLADOR DE S
226 void Run_Task_Once(void); //EJECUCI N DE LA TAREA DE MONITOREO
227 //Y ALMACENAMIENTO UNA VEZ
228
229 // Peripheral GAPRole Callbacks
230 static gapRolesCBs_t SimplePeripheral_gapRoleCBs =
231 {
232     SimplePeripheral_stateChangeCB}; // GAPRole State Change Callbacks
233
234
235 // GAP Bond Manager Callbacks
236 // These are set to NULL since they are not needed. The application
237 // is set up to only perform justworks pairing.
238 static gapBondCBs_t simplePeripheral_BondMgrCBs =
239 {
240     SimplePeripheral_passcodeCB, SimplePeripheral_pairStateCB};
241
242 // Simple GATT Profile Callbacks
243 static simpleProfileCBs_t SimplePeripheral_simpleProfileCBs =
244 {
245     SimplePeripheral_charValueChangeCB
246 };
247 /*****
248 * The following typedef and global handle the registration to connection
249 * event
250 */
251 typedef enum
252 {
253     NOT_REGISTER = 0,
254     FOR_AOA_SCAN = 1,
255     FOR_ATT_RSP = 2,
256     FOR_AOA_SEND = 4,
257     FOR_TOF_SEND = 8
258 }connectionEventRegisterCause_u;
259 // Handle the registration and un-registration for the connection event
260 uint32_t connectionEventRegisterCauseBitMap = NOT_REGISTER;
261
262 /*****
263 * @fn SimplePeripheral_RegisterToAllConnectionEvent()
264 * @brief register to receive connection events for all the connection
265 * @return @ref SUCCESS
266 */
267 bStatus_t SimplePeripheral_RegisterToAllConnectionEvent (
268     connectionEventRegisterCause_u connectionEventRegisterCause)
269 {
270     bStatus_t status = SUCCESS;
271
272     // in case there is no registration for the connection event
273     if (!CONNECTION_EVENT_IS_REGISTERED) //, make the registration
274     {
275         status = GAP_RegisterConnEventCb(SimplePeripheral_connEvtCB,
276             GAP_CB_REGISTER, LINKDB_CONNHANDLE_ALL);
277     }
278     if(status == SUCCESS)
279     {
280         //add the reason bit to the bitamap.
281         CONNECTION_EVENT_REGISTER_BIT_SET(connectionEventRegisterCause);
282     }
283 }
```

ANEXO C. CÓDIGOS DE C

```
283     return(status);
284 }
285
286 /*****
287 * @fn      SimplePeripheral_UnRegisterToAllConnectionEvent()
288 * @brief   Unregister connection events
289 * @return  @ref SUCCESS
290 */
291 bStatus_t SimplePeripheral_UnRegisterToAllConnectionEvent (
292     connectionEventRegisterCause_u connectionEventRegisterCause)
293 {
294     bStatus_t status = SUCCESS;
295
296     CONNECTION_EVENT_REGISTER_BIT_REMOVE(connectionEventRegisterCause);
297     if (!CONNECTION_EVENT_IS_REGISTERED)
298     {
299         GAP_RegisterConnEventCb(SimplePeripheral_connEvtCB,
300                                 GAP_CB_UNREGISTER, LINKDB_CONNHANDLE_ALL);
301     }
302     return(status);
303 }
304 /*****
305 * @fn      scCtrlReadyCallback
306 * @brief   FUNCION DE NOTIFICACION DE INICIALIZACION DE LA TAREA 1
307 */
308 void scCtrlReadyCallback(void) {}
309 /*****
310 * @fn      scTaskAlertCallback
311 * @brief   INTERRUPCION PARA EL ALMACENAMIENTO DE DATOS ANALOGICOS
312 * EN PROGRAMA PRINCIPAL
313 */
314 void scTaskAlertCallback(void)
315 {
316     scifClearAlertIntSource();           //LIMPIA BANDERA DE INTERRUPCION
317     //ALMACENA EL INDICE DEL BUFFER DEL SENSOR CONTROLLER
318     i = scifTaskData.adcDataLogger.output.head;
319     int inc=0;
320     for(inc=0; inc <= 3; inc++){
321         //ALMACENA EL VALOR CONVERTIDO OBTENIDO DE VALOR
322         muestraUnica = scifTaskData.adcDataLogger.output.Value[inc];
323         //ALMACENA EN UN ARREGLO DE LA APLICACION PRINCIPAL LOS VALORES
324         muestras[inc] = muestraUnica;
325     }
326     scifAckAlertEvents();               //LIMPIA LA ALERTA DE EVENTOS
327     Run_Task_Once();
328     //GENERA UN EVENTO PARA ENVIAR DATOS RECOLECTADOS
329     Event_post(syncEvent, SBP_DATA_READY_EVT);
330 }
331 /*****
332 * @fn      SensorController_Init:
333 * @brief   CREACION DE LA FUNCION DEL INICIO DEL CONTROLADOR SE SENSORES
334 */
335 void SensorController_Init(void)
336 {
337     //INICIO DE SCIF PARA PODER EJECUTAR VARIAS TAREAS A LA VEZ
338     scifOsaiInit();
339     scifOsaiRegisterCtrlReadyCallback(scCtrlReadyCallback);
340     scifOsaiRegisterTaskAlertCallback(scTaskAlertCallback);
341
342     //FUNCION DE INICIALIZACION DEL CONTROLADOR DE SENSORES
```


C.3. CÓDIGO DEL ARCHIVO EJECUTABLE SIMPLE_PERIPHERAL.C

```
343     //INCLUYE EL PIN 7 COMO ENTRADA PARA ADC
344     scifInit(&scifDriverSetup);
345     Run_Task_Once();           //EJECUTA LA TAREA 1 UNA VEZ
346 }
347 /*****
348 * @fn      Run_Task_Once:
349 * @brief   EJECUCION DE LA TAREA UNA SOLA VEZ
350 */
351 void Run_Task_Once()
352 {
353     //DELAY DE 20 MILISEGUNDOS: ESTABILIZAR LA CONFIGURACION DE SC
354     if (scifWaitOnNbl(20000) != SCIF_SUCCESS)
355     {
356         while(1); //SI EL CONTRLADOR DE SENSOR NO PUEDE INICIAR LA TAREA
357     }
358     else
359     {
360         //VUALVE A CONFIGURAR LAS ASLIDAS, ENTRADAS Y DATOS, DE OTRO
361         //MODO NO VUELVE A TOMAR DATOS ( IMPORTANTE !!!)
362         scifResetTaskStructs(1 << SCIF_ADC_DATA_LOGGER_TASK_ID,
363             (1 << SCIF_STRUCT_INPUT) | (1 << SCIF_STRUCT_OUTPUT));
364         ///EJECUTA LA TAREA UNA VEZ MAS
365         scifExecuteTasksOnceNbl(1 << SCIF_ADC_DATA_LOGGER_TASK_ID);
366     }
367 }
368 /*****
369 * @fn      SimplePeripheral_createTask
370 * @brief   Task creation function for the Simple Peripheral.
371 */
372 void SimplePeripheral_createTask(void)
373 {
374     Task_Params taskParams;
375     // Configure task
376     Task_Params_init(&taskParams);
377     taskParams.stack = sbpTaskStack;
378     taskParams.stackSize = SBP_TASK_STACK_SIZE;
379     taskParams.priority = SBP_TASK_PRIORITY;
380     Task_construct(&sbpTask, SimplePeripheral_taskFxn, &taskParams, NULL);
381 }
382 /*****
383 * @fn      SimplePeripheral_init
384 *
385 * @brief   Called during initialization and contains application
386 *          specific initialization (ie. hardware initialization/setup,
387 *          table initialization, power up notification, etc), and
388 *          profile initialization/setup.
389 */
390 static void SimplePeripheral_init(void)
391 {
392     ICall_registerApp(&selfEntity, &syncEvent);
393
394     // Create an RTOS queue for message from profile to be sent to app.
395     appMsgQueue = Util_constructQueue(&appMsg);
396     // Create one-shot clocks for internal periodic events.
397     Util_constructClock(&periodicClock, SimplePeripheral_clockHandler,
398         SBP_PERIODIC_EVT_PERIOD, 0, false, SBP_PERIODIC_EVT);
399
400     dispHandle = Display_open(SBP_DISPLAY_TYPE, NULL);
401
402     GAP_SetParamValue(TGAP_CONN_PAUSE_PERIPHERAL,
```

ANEXO C. CÓDIGOS DE C

```
403         DEFAULT_CONN_PAUSE_PERIPHERAL);
404
405 // Start the Device:
406 // Please Notice that in case of wanting to use the GAPRole_SetParameter
407 // function with GAPROLE_IRK or GAPROLE_SRK parameter - Perform
408 // these function calls before the GAPRole_StartDevice use.
409 //because Both cases are updating the gapRole_IRK & gapRole_SRK variables
410 VOID GAPRole_StartDevice(&SimplePeripheral_gapRoleCBs);
411 // Setup the Peripheral GAPRole Profile.
412 {
413     uint16_t advertOffTime = 0;
414     uint8_t enableUpdateRequest = DEFAULT_ENABLE_UPDATE_REQUEST;
415     uint16_t desiredMinInterval = DEFAULT_DESIRED_MIN_CONN_INTERVAL;
416     uint16_t desiredMaxInterval = DEFAULT_DESIRED_MAX_CONN_INTERVAL;
417     uint16_t desiredSlaveLatency = DEFAULT_DESIRED_SLAVE_LATENCY;
418     uint16_t desiredConnTimeout = DEFAULT_DESIRED_CONN_TIMEOUT;
419
420     GAPRole_SetParameter(GAPROLE_ADVERT_OFF_TIME, sizeof(uint16_t),
421                          &advertOffTime);
422
423     GAPRole_SetParameter(GAPROLE_SCAN_RSP_DATA, sizeof(scanRspData),
424                          scanRspData);
425     GAPRole_SetParameter(GAPROLE_ADVERT_DATA, sizeof(advertData),
426                          advertData);
427     GAPRole_SetParameter(GAPROLE_PARAM_UPDATE_ENABLE, sizeof(uint8_t),
428                          &enableUpdateRequest);
429     GAPRole_SetParameter(GAPROLE_MIN_CONN_INTERVAL, sizeof(uint16_t),
430                          &desiredMinInterval);
431     GAPRole_SetParameter(GAPROLE_MAX_CONN_INTERVAL, sizeof(uint16_t),
432                          &desiredMaxInterval);
433     GAPRole_SetParameter(GAPROLE_SLAVE_LATENCY, sizeof(uint16_t),
434                          &desiredSlaveLatency);
435     GAPRole_SetParameter(GAPROLE_TIMEOUT_MULTIPLIER, sizeof(uint16_t),
436                          &desiredConnTimeout);
437 }
438 // Set the Device Name characteristic in the GAP GATT Service
439 GGS_SetParameter(GGS_DEVICE_NAME_ATT, GAP_DEVICE_NAME_LEN,
440                 attDeviceName);
441
442 // Set GAP Parameters to set the advertising interval
443 {
444     uint16_t advInt = DEFAULT_ADVERTISING_INTERVAL;
445     GAP_SetParamValue(TGAP_LIM_DISC_ADV_INT_MIN, advInt);
446     GAP_SetParamValue(TGAP_LIM_DISC_ADV_INT_MAX, advInt);
447     GAP_SetParamValue(TGAP_GEN_DISC_ADV_INT_MIN, advInt);
448     GAP_SetParamValue(TGAP_GEN_DISC_ADV_INT_MAX, advInt);
449 }
450
451 // Setup the GAP Bond Manager.
452 {
453     //Don't send a pairing request after connecting; the peer device must
454     // initiate pairing
455     uint8_t pairMode = GAPBOND_PAIRING_MODE_WAIT_FOR_REQ;
456     // Use authenticated pairing: require passcode.
457     uint8_t mitm = TRUE;
458     // This device only has display capabilities. Therefore,
459     //it will display the passcode during pairing.
460     uint8_t ioCap = GAPBOND_IO_CAP_DISPLAY_ONLY;
461     // Request bonding
462     uint8_t bonding = TRUE;
```

C.3. CÓDIGO DEL ARCHIVO EJECUTABLE SIMPLE_PERIPHERAL.C

```
463     uint8_t replaceBonds = FALSE;
464
465     GAPBondMgr_SetParameter(GAPBOND_PAIRING_MODE,
466                             sizeof(uint8_t), &pairMode);
467     GAPBondMgr_SetParameter(GAPBOND_MITM_PROTECTION,
468                             sizeof(uint8_t), &mitm);
469     GAPBondMgr_SetParameter(GAPBOND_IO_CAPABILITIES,
470                             sizeof(uint8_t), &ioCap);
471     GAPBondMgr_SetParameter(GAPBOND_BONDING_ENABLED,
472                             sizeof(uint8_t), &bonding);
473     GAPBondMgr_SetParameter(GAPBOND_LRU_BOND_REPLACEMENT,
474                             sizeof(uint8_t), &replaceBonds);
475 }
476 // Initialize GATT attributes
477 GGS_AddService(GATT_ALL_SERVICES);           // GAP GATT Service
478 GATTServApp_AddService(GATT_ALL_SERVICES);   // GATT Service
479 DevInfo_AddService();                         // Device Information Serv
480 SimpleProfile_AddService(GATT_ALL_SERVICES);  // Simple GATT Profile
481
482 // Setup the SimpleProfile Characteristic Values
483 {
484     uint8_t charValue1 = 1;
485     uint8_t charValue2 = 2;
486     uint8_t charValue3 = 3;
487     uint8_t charValue4 = 4;
488     uint8_t charValue5[SIMPLEPROFILE_CHAR5_LEN] = { 1, 2, 3, 4, 5 };
489
490     SimpleProfile_SetParameter(SIMPLEPROFILE_CHAR1, sizeof(uint8_t),
491                               &charValue1);
492     SimpleProfile_SetParameter(SIMPLEPROFILE_CHAR2, sizeof(uint8_t),
493                               &charValue2);
494     SimpleProfile_SetParameter(SIMPLEPROFILE_CHAR3, sizeof(uint8_t),
495                               &charValue3);
496     SimpleProfile_SetParameter(SIMPLEPROFILE_CHAR4, sizeof(uint8_t),
497                               &charValue4);
498     SimpleProfile_SetParameter(SIMPLEPROFILE_CHAR5,
499                               SIMPLEPROFILE_CHAR5_LEN, charValue5);
500 }
501 // Register callback with SimpleGATTprofile
502 SimpleProfile_RegisterAppCBs(&SimplePeripheral_simpleProfileCBs);
503 // Start Bond Manager and register callback
504 VOID GAPBondMgr_Register(&simplePeripheral_BondMgrCBs);
505 // Register with GAP for HCI/Host messages. This is needed to receive
506 // events. For more information, see the section in the User's Guide:
507 // http://software-dl.ti.com/lprf/sdg-latest/html
508 GAP_RegisterForMsgs(selfEntity);
509 //Register for GATT local events and Responses pending for transmission
510 GATT_RegisterForMsgs(selfEntity);
511
512 //Set default values for Data Length Extension
513 {
514     //Set initial values to maximum, RX is set to max. by default
515     #define APP_SUGGESTED_PDU_SIZE 251 //default is 27 octets(TX)
516     #define APP_SUGGESTED_TX_TIME 2120 //default is 328us(TX)
517 }
518
519 #if !defined (USE_LL_CONN_PARAM_UPDATE)
520 // The HCI will generate an HCI event that will get received in the
521 // main loop
```

ANEXO C. CÓDIGOS DE C

```
522 HCI_LE_ReadLocalSupportedFeaturesCmd();
523 #endif // !defined (USE_LL_CONN_PARAM_UPDATE)
524 }
525 /*****
526 * @fn      SimplePeripheral_taskFxn
527 * @brief   Application task entry point for the Simple Peripheral.
528 * @return  None.
529 */
530 static void SimplePeripheral_taskFxn(UArg a0, UArg a1)
531 {
532     // Initialize application
533     SimplePeripheral_init();
534     // DAR INICIO A LA CONVERSI N ANALOGICA DIGITAL!!!
535     SensorController_Init();
536     // DAR INICIO A LA CONVERSI N ANALOGICA DIGITAL!!!
537     // Application main loop
538     for (;;)
539     {
540         uint32_t events;
541         // Waits for an event to be posted associated with the calling thread
542         // Note that an event associated with a thread is posted when a
543         // message is queued to the message receive queue of the thread
544         events = Event_pend(syncEvent, Event_Id_NONE, SBP_ALL_EVENTS,
545                             ICALL_TIMEOUT_FOREVER);
546         if (events)
547         {
548             ICall_EntityID dest;
549             ICall_ServiceEnum src;
550             ICall_HciExtEvt *pMsg = NULL;
551             //Fetch any available messages that might have been sent
552             if (ICall_fetchServiceMsg(&src, &dest,
553                                     (void **)&pMsg) == ICALL_ERRNO_SUCCESS)
554             {
555                 uint8 safeToDealloc = TRUE;
556                 if ((src == ICALL_SERVICE_CLASS_BLE) && (dest == selfEntity))
557                 {
558                     ICall_Stack_Event *pEvt = (ICall_Stack_Event *)pMsg;
559                     if (pEvt->signature != 0xffff)
560                     {
561                         // Process inter-task message
562                         safeToDealloc = SimplePeripheral_processStackMsg((ICall_Hdr *)
563                                     pMsg);
564                     }
565                 }
566                 if (pMsg && safeToDealloc)
567                 {
568                     ICall_freeMsg(pMsg);
569                 }
570             }
571             // If RTOS queue is not empty, process app message.
572             if (events & SBP_QUEUE_EVT)
573             {
574                 while (!Queue_empty(appMsgQueue))
575                 {
576                     sbpEvt_t *pMsg = (sbpEvt_t *)Util_dequeueMsg(appMsgQueue);
577                     if (pMsg)
578                     {
579                         // Process message.
```

C.3. CÓDIGO DEL ARCHIVO EJECUTABLE SIMPLE_PERIPHERAL.C

```
580     SimplePeripheral_processAppMsg(pMsg);
581     // Free the space from the message.
582     ICall_free(pMsg);
583 }
584 }
585 }
586
587 if (events & SBP_PERIODIC_EVT)
588 {
589     Util_startClock(&periodicClock);
590
591     // Perform periodic application task
592     SimplePeripheral_performPeriodicTask();
593 }
594
595 //ESTA ES LA CONDICION DEL EVENTO GENERADO POR LA INTERRUPCION
596 //DESDE EL CONTROLADOR DE SENSORES DE LA FUNCION
597 //"scTaskAlertCallback"
598 if(events & SBP_DATA_READY_EVT)
599 {
600     //MANDA LOS DATOS ALMACENADOS Y REFERENCIADOS POR
601     //"SIMPLEPROFILE_CHAR5", CON LONGITUD "SIMPLEPROFILE_DATA_LEN"
602     SimpleProfile_SetParameter(SIMPLEPROFILE_CHAR5,
603                               SIMPLEPROFILE_DATA_LEN,
604                               muestras);
605
606     uint8_t num = 175;
607     SimpleProfile_SetParameter(SIMPLEPROFILE_DATA_ID,
608                               sizeof(uint8_t),
609                               &num);
610 }
611 }
612 }
613 /*****
614 * @fn      SimplePeripheral_processStackMsg
615 * @return  TRUE if safe to deallocate incoming message, FALSE otherwise.
616 */
617 static uint8_t SimplePeripheral_processStackMsg(ICall_Hdr *pMsg)
618 {
619     uint8_t safeToDealloc = TRUE;
620
621     switch (pMsg->event)
622     {
623     case GATT_MSG_EVENT:
624         // Process GATT message
625         safeToDealloc = SimplePeripheral_processGATTMsg((gattMsgEvent_t *)
626                                                         pMsg);
627         break;
628     case HCI_GAP_EVENT_EVENT:
629     {
630
631         // Process HCI message
632         switch(pMsg->status)
633         {
634         case HCI_COMMAND_COMPLETE_EVENT_CODE:
635             // Process HCI Command Complete Event
636             {
637
638 #if !defined (USE_LL_CONN_PARAM_UPDATE)
```

ANEXO C. CÓDIGOS DE C

```
639 // This code will disable the use of the LL_CONNECTION_PARAM_REQ
640 // control procedure (for connection parameter updates, the
641 // L2CAP Connection Parameter Update procedure will be used
642 // instead).
643 // Parse Command Complete Event for opcode and status
644 hciEvt_CmdComplete_t* command_complete = (hciEvt_CmdComplete_t*) pMsg;
645 uint8_t pktStatus = command_complete->pReturnParam[0];
646 //find which command this command complete is for
647 switch (command_complete->cmdOpcode)
648 {
649     case HCI_LE_READ_LOCAL_SUPPORTED_FEATURES:
650         {
651             if (pktStatus == SUCCESS)
652             {
653                 uint8_t featSet[8];
654
655                 memcpy( featSet, &command_complete->pReturnParam[1], 8 );
656                 CLR_FEATURE_FLAG( featSet[0], LL_FEATURE_CONN_PARAMS_REQ );
657                 HCI_EXT_SetLocalSupportedFeaturesCmd( featSet );
658             }
659         }
660         break;
661     default:
662         //do nothing
663         break;
664 }
665 #endif // !defined (USE_LL_CONN_PARAM_UPDATE)
666
667     }
668     break;
669
670     case HCI_BLE_HARDWARE_ERROR_EVENT_CODE:
671         break;
672     default:
673         break;
674 }
675 }
676 }
677 break;
678 default:
679     // do nothing
680     break;
681 }
682 return (safeToDealloc);
683 }
684 /*****
685 * @fn      SimplePeripheral_processGATTMsg
686 * @brief   Process GATT messages and events.
687 */
688 static uint8_t SimplePeripheral_processGATTMsg(gattMsgEvent_t *pMsg)
689 {
690     // See if GATT server was unable to transmit an ATT response
691     if (attRsp_isAttRsp(pMsg))
692     {
693         if( SimplePeripheral_RegisterToAllConnectionEvent(FOR_ATT_RSP) ==
694             SUCCESS)
695         {
696             return (FALSE);
697         }
698     }
699 }
```

C.3. CÓDIGO DEL ARCHIVO EJECUTABLE SIMPLE_PERIPHERAL.C

```
698 // Free message payload. Needed only for ATT Protocol messages
699 GATT_bm_free(&pMsg->msg, pMsg->method);
700 return (TRUE);
701 }
702 /*****
703 * @fn      SimplePeripheral_processConnEvt
704 * @brief   Process connection event
705 */
706 static void SimplePeripheral_processConnEvt(Gap_ConnEventRpt_t *pReport)
707 {
708     if( CONNECTION_EVENT_REGISTRATION_CAUSE(FOR_ATT_RSP))
709     {
710         // The GATT server might have returned a blePending as it was trying
711         // to process an ATT Response. Now that we finished with this
712         // connection event, let's try sending any remaining ATT Responses
713         // on the next connection event.
714         // Try to retransmit pending ATT Response (if any)
715         if (attRsp_sendAttRsp() == SUCCESS)
716         {
717             SimplePeripheral_UnRegisterToAllConnectionEvent (FOR_ATT_RSP);
718         }
719     }
720 }
721 }
722 /*****
723 * @fn      SimplePeripheral_processAppMsg
724 * @brief   Process an incoming callback from a profile.
725 * @return  None.
726 */
727 static void SimplePeripheral_processAppMsg(sbpEvt_t *pMsg)
728 {
729     switch (pMsg->hdr.event)
730     {
731     case SBP_STATE_CHANGE_EVT:
732     {
733         SimplePeripheral_processStateChangeEvent((gaprole_States_t)pMsg->
734             hdr.state);
735     }
736     break;
737     case SBP_CHAR_CHANGE_EVT:
738     {
739         SimplePeripheral_processCharValueChangeEvent(pMsg->hdr.state);
740     }
741     break;
742     // Pairing event
743     case SBP_PAIRING_STATE_EVT:
744     {
745         SimplePeripheral_processPairState(pMsg->hdr.state, *pMsg->pData);
746         ICall_free(pMsg->pData);
747         break;
748     }
749     // Passcode event
750     case SBP_PASSCODE_NEEDED_EVT:
751     {
752         SimplePeripheral_processPasscode(*pMsg->pData);
753         ICall_free(pMsg->pData);
754         break;
755     }
756     case SBP_CONN_EVT:
```

ANEXO C. CÓDIGOS DE C

```
757     {
758         SimplePeripheral_processConnEvt((Gap_ConnEventRpt_t *) (pMsg->pData
759         ));
760         ICall_free(pMsg->pData);
761         break;
762     }
763     default:
764         // Do nothing.
765         break;
766 }
767 /*****
768 * @fn      SimplePeripheral_stateChangeCB
769 * @brief   Callback from GAP Role indicating a role state change.
770 */
771 static void SimplePeripheral_stateChangeCB(gaprole_States_t newState)
772 {
773     SimplePeripheral_enqueueMsg(SBP_STATE_CHANGE_EVT, newState, NULL);
774 }
775 /*****
776 * @fn      SimplePeripheral_processStateChangeEvt
777 * @brief   Process a pending GAP Role state change event.
778 * @return  None.
779 */
780 static void SimplePeripheral_processStateChangeEvt(gaprole_States_t
781           newState)
782 {
783     switch ( newState )
784     {
785     case GAPROLE_STARTED:
786         {
787             uint8_t ownAddress[B_ADDR_LEN];
788             uint8_t systemId[DEVINFO_SYSTEM_ID_LEN];
789             GAPRole_GetParameter(GAPROLE_BD_ADDR, ownAddress);
790             // use 6 bytes of device address for 8 bytes of system ID value
791             systemId[0] = ownAddress[0];
792             systemId[1] = ownAddress[1];
793             systemId[2] = ownAddress[2];
794             // set middle bytes to zero
795             systemId[4] = 0x00;
796             systemId[3] = 0x00;
797             // shift three bytes up
798             systemId[7] = ownAddress[5];
799             systemId[6] = ownAddress[4];
800             systemId[5] = ownAddress[3];
801             DevInfo_SetParameter(DEVINFO_SYSTEM_ID, DEVINFO_SYSTEM_ID_LEN,
802                                 systemId);
803             // Device starts advertising upon initialization of GAP
804             uint8_t initialAdvertEnable = TRUE;
805             // Set the Peripheral GAPRole Parameters
806             GAPRole_SetParameter(GAPROLE_ADVERT_ENABLED, sizeof(uint8_t),
807                                 &initialAdvertEnable);
808         }
809         break;
810     case GAPROLE_ADVERTISING:
811         break;
812     case GAPROLE_CONNECTED:
813         {
814             linkDBInfo_t linkInfo;
```


C.3. CÓDIGO DEL ARCHIVO EJECUTABLE SIMPLE_PERIPHERAL.C

```
814     uint8_t numActive = 0;
815     Util_startClock(&periodicClock);
816     numActive = linkDB_NumActive();
817     // Use numActive to determine the connection handle of the last
818     // connection
819     if ( linkDB_GetInfo( numActive - 1, &linkInfo ) == SUCCESS )
820     {
821     }
822     else
823     {
824         uint8_t peerAddress[B_ADDR_LEN];
825         GAPRole_GetParameter(GAPROLE_CONN_BD_ADDR, peerAddress);
826     }
827 }
828 break;
829
830 case GAPROLE_CONNECTED_ADV:
831     break;
832
833 case GAPROLE_WAITING:
834     {
835         uint8_t advertReEnable = TRUE;
836         Util_stopClock(&periodicClock);
837         attRsp_freeAttRsp(bleNotConnected);
838         GAPRole_SetParameter(GAPROLE_ADVERT_ENABLED, sizeof(uint8_t),
839                             &advertReEnable);
840     }
841     break;
842
843 case GAPROLE_WAITING_AFTER_TIMEOUT:
844     attRsp_freeAttRsp(bleNotConnected);
845     break;
846
847 case GAPROLE_ERROR:
848     break;
849
850 default:
851     break;
852 }
853 }
854 /*****
855 * @fn          SimplePeripheral_charValueChangeCB
856 * @brief      Callback from Simple Profile indicating a characteristic
857 *             value change.
858 */
859 static void SimplePeripheral_charValueChangeCB(uint8_t paramID)
860 {
861     SimplePeripheral_enqueueMsg(SBP_CHAR_CHANGE_EVT, paramID, 0);
862 }
863 /*****
864 * @fn          SimplePeripheral_processCharValueChangeEvent
865 * @brief      Process a pending Simple Profile characteristic value change
866 *             event.
867 * @return     None.
868 */
869 static void SimplePeripheral_processCharValueChangeEvent(uint8_t paramID)
870 {
871     uint8_t newValue;
872     switch(paramID)
873     {
874         case SIMPLEPROFILE_CHAR1:
```

ANEXO C. CÓDIGOS DE C

```
875     SimpleProfile_GetParameter(SIMPLEPROFILE_CHAR1, &newValue);
876     break;
877 case SIMPLEPROFILE_CHAR3:
878     SimpleProfile_GetParameter(SIMPLEPROFILE_CHAR3, &newValue);
879     break;
880 default:
881     // should not reach here!
882     break;
883 }
884 }
885 /*****
886 * @fn      SimplePeripheral_performPeriodicTask
887 *
888 * @brief   Perform a periodic application task. This function gets
889 * called every five seconds (SBP_PERIODIC_EVT_PERIOD). In this example,
890 * the value of the third characteristic in the SimpleGATTProfile
891 * service is retrieved from the profile, and then copied into the
892 * value of the the fourth characteristic.
893 * @return  None.
894 */
895 static void SimplePeripheral_performPeriodicTask(void)
896 {
897     uint8_t valueToCopy;
898     //Call to retrieve the value of the third characteristic in the profile
899     if (SimpleProfile_GetParameter(SIMPLEPROFILE_CHAR3, &valueToCopy) ==
        SUCCESS)
900     {
901         // Call to set that value of the fourth characteristic in the profile
902         // Note that if notifications of the fourth characteristic have been
903         // enabled by a GATT client device, then a notification will be sent
904         // every time this function is called.
905         SimpleProfile_SetParameter(SIMPLEPROFILE_CHAR4, sizeof(uint8_t),
906                                   &valueToCopy);
907     }
908 }
909 /*****
910 * @fn      SimplePeripheral_pairStateCB
911 * @brief   Pairing state callback.
912 * @return  none
913 */
914 static void SimplePeripheral_pairStateCB(uint16_t connHandle,
915                                         uint8_t state, uint8_t status)
916 {
917     uint8_t *pData;
918     // Allocate space for the event data.
919     if ((pData = ICall_malloc(sizeof(uint8_t))))
920     {
921         *pData = status;
922         // Queue the event.
923         SimplePeripheral_enqueueMsg(SBP_PAIRING_STATE_EVT, state, pData);
924     }
925 }
926 /*****
927 * @fn      SimplePeripheral_processPairState
928 * @brief   Process the new pairing state.
929 * @return  none
930 */
931 static void SimplePeripheral_processPairState(uint8_t state,
932                                               uint8_t status)
933 {
```

C.3. CÓDIGO DEL ARCHIVO EJECUTABLE SIMPLE_PERIPHERAL.C

```
934 }
935 /*****
936 * @fn      SimplePeripheral_passcodeCB
937 * @brief   Passcode callback.
938 * @return  none
939 */
940 static void SimplePeripheral_passcodeCB(uint8_t *deviceAddr,
941                                       uint16_t connHandle,
942                                       uint8_t uiInputs,
943                                       uint8_t uiOutputs,
944                                       uint32_t numComparison)
945 {
946     uint8_t *pData;
947     // Allocate space for the passcode event.
948     if ((pData = ICall_malloc(sizeof(uint8_t))))
949     {
950         *pData = uiOutputs;
951         // Enqueue the event.
952         SimplePeripheral_enqueueMsg(SBP_PASSCODE_NEEDED_EVT, 0, pData);
953     }
954 }
955 /*****
956 * @fn      SimplePeripheral_processPasscode
957 * @brief   Process the Passcode request.
958 * @return  none
959 */
960 static void SimplePeripheral_processPasscode(uint8_t uiOutputs)
961 {
962     // This app uses a default passcode. A real-life scenario would handle
963     // pairing scenarios and likely generate this randomly.
964     uint32_t passcode = B_APP_DEFAULT_PASSCODE;
965     uint16_t connectionHandle;
966     GAPRole_GetParameter(GAPROLE_CONNHANDLE, &connectionHandle);
967     // Send passcode response
968     GAPBondMgr_PasscodeRsp(connectionHandle, SUCCESS, passcode);
969 }
970 /*****
971 * @fn      SimplePeripheral_clockHandler
972 * @brief   Handler function for clock timeouts.
973 * @param   arg - event type
974 * @return  None.
975 */
976 static void SimplePeripheral_clockHandler(UArg arg)
977 {
978     // Wake up the application.
979     Event_post(syncEvent, arg);
980 }
981 /*****
982 * @fn      SimplePeripheral_connEvtCB
983 * @brief   Connection event callback.
984 */
985 static void SimplePeripheral_connEvtCB(Gap_ConnEventRpt_t *pReport)
986 {
987     // Enqueue the event for processing in the app context.
988     if( SimplePeripheral_enqueueMsg(SBP_CONN_EVT, 0 ,(uint8_t *) pReport) ==
989         FALSE)
990     {
991         ICall_free(pReport);
992     }
993 }
```

```
993 /*****
994  *
995  * @brief   Creates a message and puts the message in RTOS queue.
996  * @param   event - message event.
997  */
998 static uint8_t SimplePeripheral_enqueueMsg(uint8_t event, uint8_t state,
999                                           uint8_t *pData)
1000 {
1001     sbpEvt_t *pMsg = ICall_malloc(sizeof(sbpEvt_t));
1002
1003     // Create dynamic pointer to message.
1004     if (pMsg)
1005     {
1006         pMsg->hdr.event = event;
1007         pMsg->hdr.state = state;
1008         pMsg->pData = pData;
1009
1010         // Enqueue the message.
1011         return Util_enqueueMsg(appMsgQueue, syncEvent, (uint8_t *)pMsg);
1012     }
1013     return FALSE;
1014 }
```

C.4. Programación del microcontrolador CC2640R2F (5x5mm)

C.4.1. Revisión de tarjeta electrónica para BioMEMS y depurador XDS110 Debug Probe

Cada tarjeta electrónica para BioMEMS contiene un microcontrolador de la familia BLE (Bluetooth Low Energy), que demanda un consumo de potencia muy bajo. El microcontrolador es el CC2640R2F de empaque RHB 5x5mm con 32 pines¹, por lo que estas tarjetas fueron personalizadas y es necesario adecuar el kit de desarrollo desde *Code Composer Studio*; pues estos ejemplos están diseñados para el microcontrolador de 48 pines de la tarjeta de desarrollo LAUNCHXL-CC2640R2.

¹ver página 11 del pdf CC2640R2F_Características de la documentación de 2020 BioMEMS Implantación (Drive).

C.4. PROGRAMACIÓN DEL MICROCONTROLADOR CC2640R2F (5X5MM)

Por otra parte, para que el microcontrolador opere de manera óptima en cuestiones de consumo de energía, como entrar en modo de IDLE y Standby, es necesario cambiar los capacitores $C41$ y $C42$; $C11$ y $C12$; o $C21$ y $C22$, por unos de valor de 12pF cada uno².

Para programar al microcontrolador CC2640R2F, y otros de su familia, como CC13XX, CC26X2, etc., es recomendable usar un depurador de Texas Instruments. que es el medio para poder cargar el firmware al microcontrolador. Este se comunica con la computadora o PC mediante USB, y puede programar y depurar microcontroladores mediante algunos protocolos de comunicación serial, como UART, JTAG, compact JTAG y SWD.



Figura C.1: Depurador XDS110.

C.4.2. Conexiones para la depuración del microcontrolador

Hay varias formas de hacer una conexión entre el depurador y la tarjeta, y son mediante el protocolo JTAG. Las terminales con que se puede programar al microcontrolador se muestran en las siguientes imágenes. Para alimentar la tarjeta electrónica, se ocuparon las terminales 25 y 27 para GND y Vcc del adaptador de expansión, respectivamente. Estas terminales se conectan en Vcc y GND (figura a) de la tarjeta electrónica.

²Cada tarjeta electrónica BioMEMS tiene un par de capacitores de acoplo para el oscilador de 32kHz, dependiendo de a qué tarjeta se le hará el reemplazo, corresponde un par de capacitores descritos para cambiar

ANEXO C. CÓDIGOS DE C

Para hacer las conexiones entre el depurador se usó el protocolo JTAGa través del adaptador de 14 terminales³. No es necesario respetar los colores de cada Jumper, pero sí que cada terminal sea la correcta de extremo a extremo.

Además, es necesario introducir un voltaje como detector de nivel de voltaje de trabajo hacia este adaptador (terminal de color morada); sin un voltaje de detección de nivel, el depurador XDS110 no reconoce una conexión con la tarjeta electrónica. Se puede obtener un voltaje de 3.3V desde el adaptador de expansión incorporado en el dispositivo XDS110 (terminal 29 de la figura c).

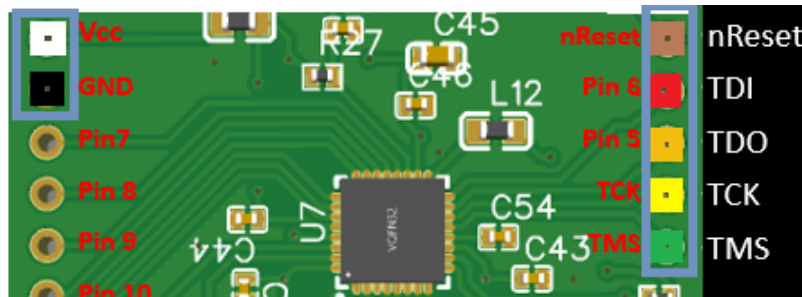


Figura C.2: Terminales de protocolo JTAG y de energización del microcontrolador.



(a) Adaptador de 14 pines con indicadores de pines
(b) Adaptador de expansión con indicadores de pines

Figura C.3: Configuración de conexiones JTAG.

³Terminales de JTAG mostradas por los colores verde, café, rojo, naranja y amarillo.

C.4.3. Configuración de depuración para XDS110 Debug Probe

La configuración para el protocolo de JTAG se encuentran dentro de una carpeta del proyecto, llamada *TargetConfigs*, específicamente en las configuraciones avanzadas del archivo *CC2640R2f.ccxml*.