

UNIVERSIDAD NACIONAL DE INGENIERÍA
FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA



TESIS

**MODELAMIENTO E IMPLEMENTACIÓN DE UN SISTEMA
DE SENSADO DE PH DE BAJO CONSUMO DE ENERGÍA
APLICADO AL CONTROL DE CALIDAD DE AGUA DE RÍOS**

**PARA OBTENER EL TÍTULO PROFESIONAL DE
INGENIERO ELECTRÓNICO**

ELABORADO POR:

LUIGHI ANTHONY VITÓN ZORRILLA

ASESOR

AURELIO MORALES VILLANUEVA, PHD.

LIMA – PERÚ

2018

MODELAMIENTO E IMPLEMENTACIÓN DE UN SISTEMA DE SENSADO DE PH
DE BAJO CONSUMO DE ENERGÍA APLICADO AL CONTROL DE CALIDAD DE
AGUA DE RÍOS

A mis padres María del Socorro y Segundo Herminio, de quienes recibí una esmerada formación y me han demostrado siempre su amor y afecto, a quienes debo los logros que he obtenido, incluyendo entre ellos este paso en mi vida profesional, y por quienes soy lo que soy.

AGRADECIMIENTOS

Agradecer ante todo a Dios por todas las bendiciones que he recibido de su parte, y por haberme concedido la constancia y perseverancia para poder llegar hasta la culminación de la tesis, cuyo desarrollo tuvo una trayectoria relativamente larga.

A mis padres, Socorro y Segundo, que me han brindado su apoyo en todo momento y me han dado la fortaleza para poder continuar en este camino, con la finalidad de poder concluir mi tesis y darme la posibilidad de titularme, culminando con esta importante etapa de mi formación profesional.

A mi familia con la he vivido en Lima por 7 años, mis primos Elizabeth y Carlos, que me han dado todas las facilidades para poder continuar en este camino y la comprensión de que la elaboración de la tesis constituye una tarea dedicada y, por tanto, necesita del tiempo y del espacio para poder realizarla.

A mis segundos padres, Elena y Julio, que aunque son mis tíos, me consideran un hijo suyo y se sienten orgullosos de los logros que he obtenido hasta ahora.

A toda mi familia que ha depositado su confianza en mi y me ha apoyado de alguna manera para llegar hasta este punto en mi carrera profesional.

A mis profesores de la universidad que me han proporcionado las herramientas para poder afrontar los retos de la vida profesional, cuyo nivel de exigencia fue vital para poder llegar hasta donde estoy.

A mis amigos y compañeros de universidad, quienes me han demostrado su afecto y apoyo, y confían en que puedo llegar más lejos aún.

Al Prof. Aurelio Morales, por aceptar ser el asesor de mi tesis, aún cuando no era un tema de su dominio, de quien aprendí bastante en la universidad en el curso de Sistemas Digitales II, y gracias al cual adquirí una serie de conocimientos que me han servido en la comprensión de los sistemas digitales en general y las tecnologías derivadas de éstos. Gracias a ello, se amplió mi perspectiva de lo que es la electrónica en sí y hasta donde se puede llegar con ella, afianzando mi amor por mi carrera.

A mis asesores en INICTEL-UNI, el Ing. Rubén Acosta, quien me eligió como su practicante en 2016 y fue quien propuso el tema que ahora es mi tesis de grado, y el Dr. Jinmi Lezama, que a parte de ser un gran amigo mio, también me ha orientado durante el desarrollo de mi tesis, dándole el seguimiento necesario y aportando con los comentarios y revisiones respectivas a cada uno de los escritos que elaboré. Agradecerle también por la confianza depositada en mi, y por creer que tengo un potencial que puedo explotar y que espero poder llegar a cubrir esas expectativas.

A mis compañeros y amigos de la coordinación III de la DIDT, parte del grupo de investigación de Sistemas Embebidos y Sensores para Internet de las Cosas (ES²-IoT), del cual formé parte, quienes siempre me han tratado con afecto y me han brindado su apoyo para poder culminar con mi tesis.

A mis compañeros y amigos de INICTEL-UNI, quienes me han tratado como uno de ellos, me han permitido integrarme a la institución, y me han brindado un clima laboral agradable, lo cual favoreció para el desarrollo de mi tesis.

A todas aquellas personas que de alguna manera han contribuido al desarrollo y culminación de este documento, y a quienes debo haber alcanzado esta etapa de mi formación como ingeniero, muchas gracias a todos ellos.

SUMARIO

El objetivo de la tesis "*Modelamiento e implementación de un sistema de sensado de pH de bajo consumo de energía aplicado al control de calidad de agua de ríos*", es el desarrollo de un sistema de sensado de pH de bajo consumo de energía el cual tenga como fin su incorporación a una red de sensores inalámbrica. El pH es una medida operacional muy importante en cuanto sus variaciones son indicadores de cambios en la composición del agua con efectos nocivos para los ecosistemas que viven y hacen uso de ésta, empleándose sistemas de sensado remotos para detectar estas variaciones. No obstante, los sistemas basados en sensores y técnicas convencionales usualmente tienen un elevado consumo de energía, lo cual implica tiempos reducidos de autonomía. Es por ello por lo que en la tesis se realiza la selección de un tecnología de sensado de bajo consumo a la vez que se efectúa la implementación de todo el sistema de acondicionamiento y control en el mismo dispositivo, empleando para ello un microcontrolador de señal mixta. Luego, para la comunicación con Internet, se hace uso de un microcontrolador que implementa un protocolo de bajo consumo dotándole de una conexión inalámbrica. El sistema construido es probado en el laboratorio, efectuando las mediciones de consumo de energía en cada una de sus etapas. Los resultados muestran un consumo inferior, en su parte analógica, respecto a otras soluciones comerciales, así como una adecuada sensibilidad y una comunicación continua con la plataforma de Internet. La solución presentada constituye una alternativa adecuada para el sensado de pH aplicado al control de calidad del agua, la cual puede extenderse a otras aplicaciones tales como agricultura y biomedicina.

ABSTRACT

The aim of the thesis "*Modeling and implementation of a low-power pH monitoring system applied to control the rivers water quality*", is the development of a low-power pH monitoring system which would be incorporated to a wireless sensor network. The pH is one of the most important operational measurements whose variations are indicators of water composition changes and, therefore, dangerous effects to the ecosystem which use them and live there, employing remote monitoring systems to detect those variations. However, systems based on conventional sensors and techniques, usually have higher levels of energy consumption, implying short autonomy periods. Hence, in this thesis, a selection of a low-power sensor technology is performed, as well as, the implementation of readout and processing systems in the same device through a mixed-signal microcontroller. In addition to this, for the communication with Internet, another microcontroller is used to implement a low-power wireless protocol. The system is tested in a laboratory environment, measuring the power levels at each stage. Results show a lower consumption, in its analogue component, than other commercial alternatives, as well as, an appropriate sensibility and a continuous communication with the Internet platform. The presented solution constitutes an adequate alternative for the pH sensing applied to the water quality control, which could be extended to other applications such as agriculture and biomedicine.

ÍNDICE

AGRADECIMIENTOS	IV
SUMARIO	VI
ABSTRACT	VII
GLOSARIO DE TÉRMINOS	XII
PRÓLOGO	1
CAPÍTULO 1	
INTRODUCCIÓN	4
1.1 Generalidades	4
1.1.1 El agua y las amenazas a su calidad	4
1.1.2 Monitoreo de la calidad del agua de ríos	5
1.1.3 Importancia de la medición de pH	6
1.2 Problemática	7
1.3 Objetivos	9
1.4 Hipótesis	9
1.5 Resumen del capítulo	10
CAPÍTULO 2	
FUNDAMENTO TEÓRICO	11
2.1 Indicador de potencial de hidrógeno (pH)	11
2.1.1 Fundamentos químicos del pH	11
2.1.2 Acerca de la definición del pH y su método de medición primario	12
2.2 Sensores para medición de pH	13
2.2.1 Fundamentos de sensores electroquímicos	14
2.2.2 Sensor de electrodo de vidrio	15
2.3 Sensor basado en transistor de efecto campo sensitivo a iones (ISFET)	16
2.3.1 Transistor de efecto campo de metal-oxido-semiconductor (MOSFET)	16
2.3.2 Interfaz electrolito-aislante-semiconductor (EIS)	17

2.3.3	Funcionamiento del ISFET	22
2.3.4	Tecnologías de fabricación y aplicaciones del ISFET	23
2.3.5	Principales desafíos al emplear el ISFET como sensor de pH	24
2.4	Sensores comerciales para medición de pH	25
2.5	Diseño de bajo consumo de energía	27
2.6	Microcontroladores de ultra bajo consumo	29
2.7	Redes de sensores inalámbricos	31
2.8	Modelamiento del sensor basado en ISFET	33
2.8.1	Modelamiento del MOSFET	33
2.8.2	Modelamiento de la interfaz EIS y del ISFET	35
2.8.3	Simulación del macromodelo del ISFET	38
2.9	Caracterización del sensor	41
2.10	Extracción de parámetros	42
2.10.1	Extracción de la tensión umbral	43
2.10.2	Extracción de la movilidad efectiva	46
2.11	Resumen del capítulo	48
CAPÍTULO 3		
DISEÑO DEL SISTEMA DE SENSADO DE PH		49
3.1	Diseño del sistema de medición para caracterización del ISFET	49
3.1.1	Enfoque conceptual y alternativas de diseño para el sistema de medición	50
3.1.2	Diseño del hardware	52
3.1.3	Diseño del software para el microcontrolador	53
3.1.4	Diseño del software de adquisición	55
3.1.5	Diseño del software de procesamiento	58
3.2	Diseño del sistema de acondicionamiento de señal	60
3.2.1	Selección de estructura de acondicionamiento	61
3.2.2	Diseño del circuito de acondicionamiento para el ISFET	63
3.2.3	Simulación de circuito de acondicionamiento para el ISFET	67
3.2.4	Diseño del circuito de acondicionamiento para el sensor de temperatura	68
3.2.5	Diseño del sistema de conversión analógico-digital	71
3.3	Diseño del sistema de procesamiento de señal, compensación y control	73
3.3.1	Diseño del sistema de procesamiento	74

3.3.2	Diseño del sistema de compensación	77
3.3.3	Diseño del sistema de control	83
3.4	Diseño del sistema de comunicación inalámbrica	85
3.4.1	Selección del protocolo para Internet de las Cosas (IoT)	85
3.4.2	Selección del dispositivo para comunicación inalámbrica	87
3.4.3	Selección del sistema operativo para implementación del protocolo	88
3.4.4	Selección de la plataforma para comunicación con Internet	91
3.5	Diseño del sistema de energía	93
3.6	Diseño del sistema global	94
3.7	Resumen del capítulo	95

CAPÍTULO 4

IMPLEMENTACIÓN DEL SISTEMA DE SENSADO DE PH	96	
4.1	Implementación del sistema para caracterización del ISFET	96
4.1.1	Implementación del sistema de adquisición	96
4.1.2	Implementación del software de procesamiento	99
4.2	Implementación del sistema de acondicionamiento de señal	101
4.2.1	Implementación del circuito de acondicionamiento del ISFET	101
4.2.2	Implementación del sistema de acondicionamiento completo	104
4.3	Implementación del sistema de procesamiento y control	109
4.4	Implementación del sistema de comunicaciones inalámbrica	111
4.4.1	Comunicación entre tarjetas PSoC y SmartRF06	112
4.4.2	Transmisión de datos mediante protocolo 6LoWPAN	113
4.4.3	Envío de datos a la plataforma de Internet de las Cosas (IoT)	116
4.5	Implementación del hardware para el prototipo de sensado de pH	116
4.5.1	Especificación de componentes y esquema de implementación	117
4.5.2	Diseño del circuito impreso (PCB)	119
4.5.3	Ensamblado de componentes y resultados preliminares	122
4.6	Resumen del capítulo	125

CAPÍTULO 5

RESULTADOS EXPERIMENTALES	126	
5.1	Verificación de información del fabricante	126
5.2	Caracterización del dispositivo	128

5.2.1	Resultados del sistema de medición para caracterización del ISFET	128
5.2.2	Extracción de parámetros aplicado a curvas obtenidas del proceso de caracte- rización del ISFET	130
5.2.3	Verificación de la adaptación del modelo del ISFET	132
5.3	Resultados del sensado de pH con el sistema de acondicionamiento del ISFET	135
5.4	Consumo de energía	137
5.5	Calibración del sensor de temperatura y de pH	140
5.6	Transmisión inalámbrica y envío de datos a Internet	145
5.7	Resumen del capítulo	147
	CONCLUSIONES Y RECOMENDACIONES	150
	BIBLIOGRAFÍA	154
	ANEXO A	
	HOJA DE DATOS TÉCNICOS ISFET WINSENSE	168
	ANEXO B	
	HOJA DE DATOS TÉCNICOS PSOC 5LP	174
	ANEXO C	
	HOJA DE DATOS TÉCNICOS CC2650	178
	ANEXO D	
	MODELO DEL ISFET EN SPICE	183
	ANEXO E	
	PROGRAMA IMPLEMENTADO EN PSOC 5LP	186
	ANEXO F	
	PROGRAMA IMPLEMENTADO EN CC2650	197
	ANEXO G	
	SOFTWARE PARA DISEÑO Y SIMULACIÓN	203

GLOSARIO DE TÉRMINOS

ADC	Analog-to-Digital Converter, conversor analógico-digital.
AMS	Analog Mixed Signal, señal analógica y mixta.
ASIC	Application-specific Integrated Circuit, circuito integrado para aplicaciones específicas.
CVCC	Constant-Voltage and Constant-Current, tensión-constante y corriente-constante.
DAC	Digital-to-Analog Converter, conversor digital-analógico.
DSP	Digital Signal Processor, procesador de señales digitales.
EDA	Electronic Design Automation, automatización del diseño electrónico.
EIS	Electrolite-Insulator-Semiconductor, electrolito-aislante-semiconductor.
ELR	Extrapolation in the Linear Region, extrapolación en la región lineal.
FPGA	Field-programmable Gate Array, arreglo de compuertas programables por campo.
IEEE	Institute of Electric and Electronic Engineers, Instituto de Ingeniería Eléctrica y Electrónica.
IHP	Inner Helmholtz Plane, plano interno de Helmholtz.
IoT	Internet of Things, Internet de las cosas.

ISFET	Ion Sensitive Field Effect Transistor, transistor de efecto campo sensitivo a iones.
LPWAN	Low Power Wide Area, red inalámbrica de baja potencia y rango amplio.
MOSFET	Metal-Oxide-Semiconductor Field Effect Transistor, transistor de efecto campo de metal-óxido-semiconductor.
OHP	Outer Helmholtz Plane, plano externo de Helmholtz.
OPAMP	Operational Amplifier, amplificador operacional.
RFID	Radio Frequency Identification, identificación por radio frecuencia.
SCE	Saturated Calomel Electrode, electrodo de calomel saturado.
SHE	Standard Hydrogen Electrode, electrodo de hidrógeno estándar.
SoC	System on a Chip, sistema integrado.
SSCE	Silver/Silver Chloride Electrode, electrodo de plata/cloruro de plata.
TIA	Transimpedance Amplifier, amplificador de transimpedancia.
UART	Universal Asynchronous Receiver-Transmitter, transmisor-receptor asíncrono universal.
WFAN	Wireless Field (or Factory) Area Network, red inalámbrica de área industrial.
WHAN	Wireless Home Area Network, red inalámbrica de área doméstica.
WLAN	Wireless Local Area Network, red inalámbrica de área local.
WNAN	Wireless Neighborhood Area Network, red inalámbrica de área de campo.
WPAN	Wireless Personal Area Network, red inalámbrica de área personal.

WSN Wireless Sensor Networks, redes de sensores inalámbricos.

WWAN Wireless Wide Area Network, red inalámbrica de área de rango amplio.

PRÓLOGO

El documento presentado constituye la tesis de grado para la obtención del título de ingeniero electrónico en la Universidad Nacional de Ingeniería, el cual fue desarrollado en el periodo desde agosto del 2016, cuando fue aprobado el perfil, hasta enero del 2018, momento en que se culmina su redacción.

El trabajo cuenta con un total de 6 capítulos, los cuales comprenden la *Introducción*, *Fundamento Teórico*, *Diseño del sistema de sensado de pH*, *Implementación del sistema de sensado de pH*, *Resultados experimentales* y *Conclusiones y recomendaciones*; además de la *Bibliografía* y los *Anexos*.

En la *Introducción*, se presenta el tema del agua así como las amenazas a las cuales se encuentra sometida resaltando la importancia del monitoreo para una adecuada gestión de este recurso. Luego se presenta la problemática así como el planteamiento de la alternativa solución a través del sistema de sensado de pH propuesto. Finalmente, se definen los objetivos y las hipótesis formuladas para el trabajo.

En el *Fundamento teórico*, se presentan los conceptos y tecnologías empleadas en el trabajo de la tesis, tales como el pH, el sensor empleado para la medición de este parámetro, los microcontroladores de bajo consumo, las redes de sensores inalámbricos y el modelamiento del sensor, así como los métodos para su caracterización y extracción de parámetros.

En el *Diseño del sistema de sensado de pH*, se presentan las etapas consideradas para el diseño del sistema, tanto del sistema para la caracterización del sensor, así como del prototipo empleado para el sensado de pH, contemplando las partes correspondientes al acondicionamiento de señal, procesamiento y control, comunicaciones inalámbricas, sistema de energía y diseño global del prototipo.

En la *Implementación del sistema de sensado de pH*, se presenta la implementación de cada una de las etapas diseñadas, tanto en el sistema de caracterización con sus diferentes componentes, y en el sistema de sensado de pH con cada una de sus partes, empleando para ello las tecnologías de microcontroladores seleccionadas en el capítulo anterior. Así mismo,

se contempla en este capítulo la construcción del prototipo.

En los *Resultados experimentales*, se presentan los resultados obtenidos a través de los experimentos con el sensor y el prototipo construido en la implementación. Se realizan un conjunto de pruebas correspondientes al consumo de energía, la calibración de sensores y el envío a la plataforma de Internet, de acuerdo a los objetivos trazados para la tesis.

En las *Conclusiones y recomendaciones*, se enumeran cada una de las conclusiones obtenidas gracias al desarrollo de la tesis, respecto al problema que se plantea resolver. Así mismo, se añaden algunas recomendaciones para el funcionamiento adecuado del prototipo, mencionando además las proyecciones para el trabajo realizado.

En la *Bibliografía*, se añaden todas las fuentes de información en las cuales se ha basado para la elaboración del documento, entre artículos de investigación, libros, tesis, reportes técnicos, hojas de datos del fabricante, manuales, páginas web, entre otros.

En los *Anexos*, se considera la inclusión de las hojas de datos técnicos de los componentes más importantes del sistema, así como el listado de los programas más relevantes escritos para la implementación del prototipo.

El presente trabajo ha sido desarrollado gracias al auspicio del Instituto Nacional de Investigación y Capacitación en Telecomunicaciones de la Universidad Nacional de Ingeniería (INICTEL-UNI), el cual dentro del programa de prácticas profesionales ha proporcionado el espacio y las herramientas para su realización.

Como parte de la Universidad Nacional de Ingeniería, me gustaría agradecer a mi profesor, PhD. Aurelio Morales Villanueva por haber aceptado ser el asesor de mi tesis y haber efectuado la revisión respectiva de las diferentes versiones del documento, dando las observaciones y correcciones necesarias para adecuarlo a las exigencias de la universidad, hasta obtener una versión final, apropiada para su presentación a la universidad.

Como parte de INICTEL-UNI, me gustaría agradecer al Ing. Rubén Acosta, quien fue mi tutor de prácticas, y fue quien me propuso este tema para desarrollarlo como mi tesis de grado. Así mismo, un agradecimiento especial al PhD. Jinmi Lezama, quien me ha brindado su asesoría durante el desarrollo de la tesis, supervisando la elaboración de los prototipos, y revisando cada uno de los escritos que elaboré, los cuales, con ciertas modificaciones, forman parte del documento final. Gracias a sus observaciones se dio la orientación adecuada además de limitar el alcance necesario para el trabajo. Agradecerle también por incentivar el avance efectuado, así como motivar a la presentación de artículos derivados de la tesis, en los

cuales también participó en su redacción, figurando como co-autor. En la implementación del prototipo, agradecer al Ing. Ricardo Yauri, por las orientaciones en las técnicas de soldadura así como facilitar la información de los sensores y prototipos anteriores desarrollados en el área; agradecer al Bach. Hector Miyashiro, quien me ayudó en el soldado del microcontrolador y me enseñó como replicar el proceso con otros componentes; agradecer al Ing. David Ponce, por darme algunas orientaciones para lograr una soldadura adecuada de los diferentes componentes y brindar un ejemplo de como se debería realizar dicho proceso. En las pruebas con el sensor, un agradecimiento al Dr. David Laván, quien me permitió hacer uso de su laboratorio para la caracterización del sensor y las pruebas para calibración de los sensores, así como por demostrar interés en los resultados obtenidos con el prototipo construido. En la redacción del documento, un agradecimiento especial a la MSc. Antuanet Adanaqué, por haberme ayudado a reducir la dimensión de la primera versión completa del documento a través de la supresión de fragmentos de información que no eran muy relevantes para el escrito global, dejando solo lo necesario para su entendimiento y flujo general de la tesis. Así mismo, por proporcionar las recomendaciones a fin de disminuir el espacio ocupado por elementos dentro del documento a partir de su experiencia en la elaboración de su tesis, al ser ingeniera egresada de nuestra facultad. En ámbito de gestión, me gustaría agradecer a mi coordinador, Ing. Milton Ríos, y al director de investigación, Ing. Daniel Díaz, por permitirme continuar en INICTEL-UNI con el fin de terminar mi tesis. Así mismo, al personal administrativo, en especial a Yanet Begazo, quien agilizó la compra de los componentes necesarios para la construcción del prototipo.

Por otra parte, me gustaría hacer una mención especial al Dr. Gerard Santillán, quien me recomendó elegir este tema de tesis, en particular, por el dispositivo que emplea para el sensado, el ISFET, el cual tiene mucho potencial para explotar. Así mismo, agradecerle por la información compartida sobre diseño analógico y sobre la extracción de parámetros del MOSFET, que fue de ayuda para efectuar el modelamiento del sensor.

CAPÍTULO 1

INTRODUCCIÓN

En este capítulo se abordan los aspectos generales correspondientes a la temática de discusión en la tesis. En él se enmarca la problemática a resolver y, a su vez, se trazan los objetivos a alcanzar con su culminación. Al final, se enuncia la hipótesis que se busca demostrar a través del presente trabajo.

1.1 Generalidades

En esta primera sección se aborda los temas correspondientes al agua, las amenazas a las que se encuentra sometida, las tareas que se pueden efectuar para su protección, considerando de entre ellas el monitoreo, así como los parámetros empleados para su control de calidad, resaltando la importancia del pH.

1.1.1 El agua y las amenazas a su calidad

El agua es uno de los recursos más preciados de los cuales el hombre puede disponer para su beneficio. De hecho, también es uno de los más abundantes del planeta, cubriendo alrededor del 75 % de su superficie. Sin embargo, solo el 2.5 % del total constituye agua dulce, encontrándose el 68.7 % en estado congelado permanentemente. Por otra parte, la principal fuente de agua dulce constituida por lagos y ríos solo representa el 0.26 % del total de agua disponible en la hidrósfera [1].

Latinoamérica es un lugar privilegiado, en especial nuestro país, debido a la alta disponibilidad de este recurso. En este caso, cualquier "crisis de agua" se debe más a deficiencias en la administración del recurso que a motivos de escasez [2]. Entre los problemas que debe afrontar destacan, por un lado, la concentración de población en zonas secas o semihúmedas que, a su vez, demandan mayor cantidad y calidad de agua; y, por otro lado, las prácticas insostenibles e informales del uso del agua, su contaminación y los conflictos entre grandes infraestructuras (minería e hidroeléctricas) y grupos poblaciones locales [2].

La suma de estos factores contribuyen a una disminución en la disponibilidad de las

fuentes de agua saludables. Es por ello que un paso para mejorar en la gestión de recursos hídricos consiste en reconocer estos fallos e identificar sus principales amenazas.

El agua es afectada por diversos agentes, algunos de ellos con un impacto directo sobre ella, mientras que otros, con una influencia indirecta. Entre los que producen un efecto directo se encuentra la contaminación en sus diferentes formas. La contaminación del agua tiene efectos negativos que afectan la naturaleza y la vida del hombre. Un muestra de ello es que el agua contaminada causa aproximadamente 14000 muertes al día debido a su falta de tratamiento en países en vías de desarrollo [3].

Dado este antecedente, es necesario desarrollar una adecuada gestión y control de la contaminación. De allí que adquiera una relevancia preponderante las tareas de monitoreo, a fin de determinar su calidad y elaborar proyecciones acerca de las condiciones en las que se encontrará dentro de algunos años.

1.1.2 Monitoreo de la calidad del agua de ríos

Según Bartram & Ballance (como es citado en [4]), el monitoreo se define como una actividad a largo plazo, la cual consiste en la medida estandarizada y la observación de los medios acuáticos, con la finalidad de determinar su estado y sus tendencias.

En el caso de los ríos, estas fuentes de agua tienen características hidrológicas, químicas y biológicas propias, las cuales, a su vez, están relacionadas con los parámetros que se emplean para determinar su calidad, y definen los métodos que se deben considerar para su monitoreo [5].

Los esfuerzos para el control de calidad de agua de ríos deben estar enmarcados dentro de un programa de monitoreo, cuyo diseño requiere de varias etapas [6] que comprenden: el planeamiento, diseño, implementación, análisis y reporte de resultados.

El presente estudio corresponde a un desarrollo técnico que encaja en la etapa de implementación del programa de monitoreo dentro del procedimiento de trabajo de campo. El análisis de resultados y su respectiva publicación, por otra parte, escapan al alcance de las atribuciones de la institución donde es concebido (INICTEL-UNI).

En la etapa de diseño del programa de monitoreo es donde se determina los parámetros que se van a considerar para determinar la calidad del agua, los cuales dependen del tipo de estudio que se realice. Los parámetros pueden clasificarse según su naturaleza en físicos, como la temperatura y la transparencia; fisicoquímicos, como el pH, el oxígeno disuelto y la conductividad eléctrica; químicos, como los metales pesados, alkaninos, nitratos y fluoruros;

y microbiológicos, como agentes patógenos termotolerantes [7].

Por lo general, los físicos y fisicoquímicos requieren de una medición in-situ, realizándose directamente en el trabajo de campo, mientras que los químicos y microbiológicos requieren de estudios especializados, por lo que el proceso se extiende del trabajo de campo, donde se adquieren las muestras, al análisis de éstas en un laboratorio.

1.1.3 Importancia de la medición de pH

El pH en sí mismo no tiene efecto directo sobre la salud de las personas, no obstante, es una de las medidas operacionales más importantes para determinar la calidad del agua [8]. Este valor no debe encontrarse en los extremos de la escala, puesto que valores muy altos indicarían que el agua no puede ser desinfectada efectivamente con procedimientos convencionales de potabilización (cloro), mientras que valores muy bajos indicarían propiedades corrosivas [9].

Aunque los humanos presentan una mejor tolerancia a valores de pH (4-11), valores fuera de este rango causan irritaciones a la piel y los ojos, mientras que inferiores a 2.5 provocan daños irreversibles a revestimiento de órganos.

Dado que se trata de una medida relativa que varía de acuerdo la composición del agua, y también debido a los materiales empleados en sus redes de distribución, es difícil proporcionar un valor exacto como óptimo para su uso. No obstante, por lo general, se encuentra en el rango de 6.5 a 8.5. Estos niveles también están normados, y en caso de la legislación peruana, coincide con el rango mencionado anteriormente [10], [11]. La OMS por su parte, sugiere un rango que se extiende de 6.5 a 9.5 [12].

Con el valor de pH se puede determinar si el agua es apta para consumo en caso se encuentre dentro del rango sugerido, fuera de él se correría el riesgo de sufrir intoxicación o lastimar irreversiblemente los órganos internos. Detectar las variaciones de pH también es importante a razón de que ecosistemas dentro del agua no tienen la misma tolerancia que el hombre y diferencias súbitas superiores a 1.4 pH causarían su muerte [13]. Adicionalmente, el pH puede servir como indicador indirecto de presencia de patógenos tales como la *Escherichia Coli*, empleado para detectar contaminación por agentes microbiológicos [14].

Si bien es cierto, la sola medición del pH no es suficiente para determinar la idoneidad de una fuente de agua, sirve como punto de partida para poder alertar sobre los cambios de su estado.

1.2 Problemática

La tarea de monitoreo de la calidad del agua se encuentra respaldada en varios países del mundo por políticas y legislaciones, las cuales, a su vez, convocan a instituciones gubernamentales y privadas a fin de vigilar el estado de este recurso. Nuestro país también cuenta con instituciones gubernamentales (Autoridad Nacional del Agua, ANA) y legislaciones que están comprometidos con este tema [15]. No obstante, debido su relativamente reciente creación (2009), aún se encuentra en etapa de consolidación.

El Perú es un país privilegiado por la abundancia de recursos hídricos, característica que ha sido ampliamente empleada para sus distintas actividades económicas. No obstante ello, no lo hace exento de los problemas que amenazan la calidad de agua, entre ellos: aguas domésticas, vertimientos de industrias extractivas, explotación aurífera, explotación de petróleo, entre otras [16]. Es por ello que la tarea de gestión de recursos, incluyendo el monitoreo de éstos, adquiere relevancia para fines de prevención y protección.

Por una parte, gracias a las actividades realizadas por el ANA e instituciones colaboradoras, tales como el Ministerio del Ambiente, el Ministerio de Agricultura, entre otras, se ha logrado tener un importante progreso en la elaboración y ejecución del Plan Nacional de Gestión de Recursos Hídricos en el Perú [17].

Si bien se han realizado estudios para comprobar su calidad, tal como se muestra en [18], éstos demandan de una mayor frecuencia y rapidez de acceso a los datos sobre su estado. Aún más, considerando que los ríos, como se precisó anteriormente, son ecosistemas de alto dinamismo, para los cuales se debe tomar en cuenta varios factores de variación en el tiempo y en el espacio. Esto se complica más al incluir el objetivo de modelamiento de su comportamiento, por lo que la recopilación de datos muy distanciados temporalmente no ayudarían a predecir con certeza sus estados futuros.

Incrementar la frecuencia de monitoreo no siempre es una tarea sencilla de completar debido a que usualmente requiere que las mediciones se realicen in-situ y, aunque se puede promover la participación de las comunidades locales para efectuarlas [19], entra a tallar el error humano en los datos obtenidos. En ese caso, la solución tecnológica más apropiada sería la implementación de una red de sensores.

La solución de una red de sensores puede abordarse desde diferentes perspectivas. Por un lado, se encuentran soluciones comerciales, tales como las ofrecidas por Libellium [20], la cual provee una plataforma capaz de medir más de 12 parámetros entre físicos y químicos. No

obstante, si se considera el costo que ello demanda y la cantidad de estaciones que deberían situarse [6], probablemente no sea la alternativa más favorable en relación al balance costo-beneficio.

En el Perú también hay esfuerzos importantes. Por un lado, se encuentra el proyecto desarrollado en INICTEL-UNI denominado "Sistema de telemetría para el monitoreo de los recursos naturales y del ambiente en el Perú", que consiste en un sistema de monitoreo de la calidad del agua para crianza de peces amazónicos [21], el cual monitorea cuatro parámetros fisicoquímicos: pH, temperatura, oxígeno disuelto y conductividad eléctrica. Por otra parte, se encuentra el proyecto WAPOSAT [22], el cual surgió en la Universidad Nacional de Ingeniería (UNI) y tiene por objetivo el monitoreo de recursos hídricos en tiempo casi real mediante una red de sensores inalámbricos. Para ello emplea estaciones de monitoreo, las cuales cuentan con el conjunto de sensores receptores de los parámetros considerados, además de analizar las muestras para detectar impurezas.

Uno de los principales problemas que acarrea soluciones de esta naturaleza es la fuente de energía que, al tratarse de baterías, tiene una duración limitada, y si bien se puede extender mediante el empleo de paneles solares, sería mucho mejor si se pudiera trabajar desde el punto de vista del consumo intrínseco del sistema.

Uno de los sensores que llama más la atención es el de pH, no solo por su importancia en la determinación de calidad de agua, sino también, debido al consumo que representa dentro del sistema. En el proyecto desarrollado por INICTEL, de los cuatro sensores que se han empleado, el que consume más energía es precisamente el sensor de pH. Por ende, es preciso trabajar en este sensor y contribuir en la medida posible a la reducción de su consumo.

Por otra parte, los sensores cada vez son más pequeños y se requiere, por tanto, que estén más integrados. Gracias al desarrollo de la tecnología de sistemas embebidos, esto es factible, dando lugar a que todo el sistema se reduzca a unos pocos dispositivos de este tipo; implicando también una reducción en el consumo de energía, en los costos, así como en el espacio ocupado por éstos.

Por último, dada la importancia que están adquiriendo las tecnologías vinculadas con el concepto de Internet de las Cosas (IoT), con frecuencia se requiere que las soluciones propuestas, de alguna forma, tengan conexión con esta red global. Al estar conectado a Internet se puede facilitar el acceso de la información proporcionada por el sistema de monitoreo a los usuarios y autoridades competentes, así como ofrecer la posibilidad de

tomar medidas oportunas en caso de presentarse alguna incidencia. Es por ello que resulta significativo dotarle de esta característica.

1.3 Objetivos

El objetivo principal trazado para el presente trabajo es el modelamiento e implementación de un sistema de sensado de pH de bajo consumo de energía, desarrollado sobre una tecnología embebida y con envío de datos a Internet.

Así mismo, se definen objetivos específicos, los cuales comprenden:

- *Modelamiento y caracterización del sensor de pH.* Contempla la elección del sensor, la obtención de su modelo, las mediciones para obtener curvas características y la obtención de parámetros para caracterización.
- *Modelamiento del circuito de acondicionamiento para el sensor de pH.* Contempla el análisis de circuitos de adaptación conocidos, el diseño de una mejora para reducir el consumo orientado al sensor en particular, su simulación y verificación de las características deseadas.
- *Implementación del sistema en un dispositivo integrado para el acondicionamiento de señal y control.* Vinculado estrechamente al anterior. Contempla la adaptación del diseño para emplear los bloques constitutivos de una tecnología embebida, además de la selección del dispositivo adecuado para cubrir con los requerimientos del sistema.
- *Implementación de la conexión del sistema a Internet.* Incluye la selección de la tecnología inalámbrica empleada para la comunicación del sistema. Contempla la implementación y adecuado ensamblaje con el diseño de acondicionamiento de señal. Emplea una plataforma de evaluación (página web o aplicación móvil) a fin de verificar el envío de datos a Internet.

1.4 Hipótesis

La hipótesis general de la tesis se enuncia como sigue: "Es posible construir un sistema de sensado de pH para el monitoreo de calidad de agua, de bajo consumo de energía (menor a 8mA¹), que pueda ser incorporado a una red de sensores y cuente con conectividad a Internet".

¹Valor de consumo de un sensor de pH comercial de electrodo de vidrio con su sistema de adquisición

Derivadas de la anterior se formulan las siguientes hipótesis específicas:

- Es posible modelar y caracterizar el sensor seleccionado para la medición de pH.
- Es factible modelar y simular el sistema de acondicionamiento de bajo consumo.
- Es posible implementar el sistema diseñado en una tecnología embebida.
- Es posible comunicar el sistema de sensado a Internet.

1.5 Resumen del capítulo

El agua es un recurso fundamental para la supervivencia del hombre, lo cual se manifiesta en su marcada influencia en cada uno de los aspectos de su vida. Sin embargo, constantemente se ve amenazada por diferentes fuentes de contaminación. Una medida importante para prevenir y protegerla de estas amenazas es el desarrollo de un sistema de gestión de recursos, donde se contemplen tareas de monitoreo a fin de obtener la información sobre su estado.

En el caso de los ríos, estas tareas tienen sus propios desafíos debido a las características intrínsecas del medio. Así mismo, requiere de una identificación de los parámetros que han de considerarse para su implementación. Dentro de estas características, adquiere una importante relevancia la medición del pH.

En el Perú, los sistemas de monitoreo aún requieren de más tiempo para consolidarse. Aunque existen esfuerzos importantes que contribuyen a esta labor, usualmente las soluciones no están optimizadas en el consumo de energía por el lado del sensor.

En el presente trabajo se propone la implementación de un sistema de monitoreo, cuyo bajo consumo está orientado al sensor y la electrónica empleada para su acondicionamiento. Adicionalmente, contempla su comunicación a Internet, con la finalidad de proveer con prontitud la información obtenida a las instituciones y personas competentes.

En este capítulo se ha abordado el contexto general en el cual se desarrolla la tesis, la importancia del monitoreo de calidad de agua, así como de la problemática actual en nuestro país, planteando una solución a través de herramientas tecnológicas orientadas a la implementación de un sistema de sensado de pH bajo consumo de energía, lo cual constituye el panorama global de aplicación de los conceptos que se abordarán a detalle en el siguiente capítulo.

CAPÍTULO 2

FUNDAMENTO TEÓRICO

En el presente capítulo se explica los conceptos básicos acerca del pH, los sensores empleados para medir esta magnitud, el diseño de bajo consumo de energía, las redes de sensores y finalmente el modelamiento del sensor de pH escogido.

2.1 Indicador de potencial de hidrógeno (pH)

El indicador de pH es una medida adimensional de la concentración de iones de hidronio (H_3O^+). Está definido para cualquier medio como el logaritmo decimal del recíproco de la actividad de ion de hidrógeno, α_{H^+} , en dicho medio [23], como se muestra en (2.1):

$$\text{pH} = -\log_{10} \alpha_{\text{H}^+} = \log_{10} \left(\frac{1}{\alpha_{\text{H}^+}} \right) \quad (2.1)$$

El indicador de pH se emplea para determinar si una solución es ácido o base.

A fin de abordar con propiedad el concepto del pH, se considera conveniente exponer otros conceptos químicos relacionados con esta magnitud los cuales se describen a continuación.

2.1.1 Fundamentos químicos del pH

El término pH se aplica generalmente a soluciones, las cuales son mezclas homogéneas de un soluto en un solvente, que típicamente es agua. Las soluciones ácidas o básicas (alcalinas) son producto de una reacción química, en la cual se disocian los ácidos o bases al combinarse con el agua, liberando iones hidronio o hidroxilo (OH^-), dependiendo del caso.

Este tipo de reacciones químicas son reversibles y, por tanto, sus productos pueden reaccionar para obtener las sustancias que los generaron. Una vez que la concentración de productos y reactantes se estabiliza, se llega a un estado de equilibrio químico dinámico. Las concentraciones de los elementos en este estado se encuentran en una proporción fija denominada constante de equilibrio.

En el caso de las reacciones de formación de soluciones ácidas o básicas, el elemento que reacciona es el agua, el cual puede adoptar el comportamiento de base o ácido según

el caso, por lo que recibe la denominación de anfiprótico. En este proceso, la constante de equilibrio adquiere el nombre de constante de autoprotólisis K_w , la cual tiene un valor de 10^{-14} a temperatura ambiente (25°C). La relación se puede apreciar en (2.2) y (2.3):



$$K_w = [\text{H}_3\text{O}^+][\text{OH}^-] = 10^{-7} \times 10^{-7} = 10^{-14} \quad (2.3)$$

La escala de pH es una medida que facilita la cuantificación de la concentración de iones hidronio disociados en la reacción, y se encuentra intrínsecamente relacionada con la constante K_w al establecerse el equilibrio dinámico en el agua.

2.1.2 Acerca de la definición del pH y su método de medición primario

La definición del pH mostrada en la ecuación (2.1) emplea el término de actividad de una sustancia en lugar de concentración (como fue enunciado originalmente por Sorensen en 1909 [24]), debido a que es incorrecto aplicar la función de logaritmo a una magnitud con unidades. La actividad del ion usualmente es expresada como a_{H} , y se define como lo muestra (2.4):

$$a_{\text{H}} = c_{\text{H}}\gamma_{\text{H}}/c^0 = m_{\text{H}}\gamma_{\text{H}}/m^0 \quad (2.4)$$

donde c^0 es equivalente a 1 mol L^{-1} , m^0 es equivalente a 1 mol kg^{-1} , c_{H} es la concentración de iones de hidronio, m_{H} es la molalidad, mientras que γ_{H} y γ_{H} son los coeficientes de actividad para cada caso [25].

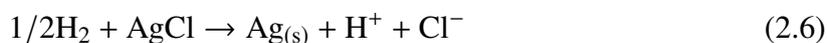
No obstante, de acuerdo a lo expresado por la *International Union of Pure and Applied Chemistry* (IUPAC), la ecuación proporcionada solo corresponde a una definición hipotética, puesto que la actividad de un ion no puede ser medido mediante un método termodinámico válido [24]. Es por ello que se le otorga una definición operacional, la cual esta relacionada con el método de medición que se emplea para obtenerlo.

Es así como, para poder definir con propiedad la escala de pH, es necesario establecer un procedimiento que pueda relacionarla con las magnitudes fundamentales del Sistema Internacional de Unidades (SI). El método primario citado por la IUPAC, consistente en una celda electroquímica denominada celda de Harned, logra establecer esta relación respecto a valores de tensión y concentración directamente medibles. Esta celda tiene la estructura expresada por el diagrama mostrado en la ecuación (2.5):



donde se aprecia la presencia de los dos electrodos de platino y de plata/cloruro de plata, comunicados mediante un buffer S con electrolitos de ion de cloro (comúnmente KCl o NaCl).

En la celda se produce una reacción química espontánea, como lo muestra (2.6):



cuyo potencial está determinado según la ecuación de Nernst, presentada en (2.7):

$$E_1 = E^0 - \frac{RT}{F} \ln 10 \log [(m_{\text{H}}\gamma_{\text{H}}/m^0)(m_{\text{Cl}}\gamma_{\text{Cl}}/m^0)] \quad (2.7)$$

De esta ecuación se puede obtener el valor de pH a partir de despejar el valor de la concentración de H, tal como lo muestra (2.8):

$$\text{pH} = -\log(\alpha_{\text{H}}) = \log(m_{\text{H}}\gamma_{\text{H}}/m^0) = (E - E^0) \left(\frac{nF}{RT \ln 10} \right) + \ln(m_{\text{Cl}}\gamma_{\text{Cl}}/m^0) \quad (2.8)$$

Si bien es cierto, es posible determinar el valor del potencial estándar E^0 [24], una estrategia más sencilla consiste en tomar la diferencia de dos valores de pH, primero de una solución de la cual ya se conoce su valor (generalmente un buffer), y luego de otra que se requiere conocer. Esta operación se expresaría a través de la ecuación (2.9) [26]:

$$\text{pH}(X) - \text{pH}(S) = \frac{E_X - E_S}{\ln 10 RT/F} \quad (2.9)$$

donde S corresponde a la solución conocida y X es la solución de la cual se desea saber el pH. En ambos casos, es en estas soluciones donde se sitúa el electrodo de hidrógeno [26].

2.2 Sensores para medición de pH

Para la medición de pH existen métodos, los cuáles se pueden clasificar en electroquímicos y no electroquímicos [27]. Entre estos últimos, el más conocido es el método por colorimetría, que emplea materiales sensibles al valor del pH, cambiando de color al sumergirse en una solución ácida o básica. Estos indicadores son compuestos, ácidos débiles o bases débiles, que se encuentran como tintes naturales y cambian de color ante una determinada concentración de iones de hidrógeno [23].

En el caso de los electroquímicos, éstos emplean sensores que traducen la información desde el dominio químico al dominio eléctrico. Estos sensores, por lo general, son activos dado que requieren recibir energía eléctrica en forma de tensión para poder excitar la muestra y recibir la información de su pH. Entre los ejemplos de sensores electroquímicos empleados

para la medición de pH, se encuentran los electrodos de vidrio, los basados en semiconductores y en electrodos metálicos u óxido metálicos[27]. Adicionalmente, existen otras técnicas más recientes que involucran a estructuras nanométricas, polímeros y grafeno [28].

Según su forma de operación, los sensores electroquímicos pueden ser de tres tipos: potenciométricos (medición de tensión), amperométricos (medición de corriente) y conductométricos (medición de conductividad). Los sensores de pH usualmente se encuentran dentro de los primeros.

Por tanto, para abordar el tema de los sensores de pH es importante reconocer algunos conceptos referidos a la naturaleza propia de los sensores electroquímicos, así como de las interfaces que los constituyen, las cuales son uno de sus elementos esenciales.

2.2.1 Fundamentos de sensores electroquímicos

Los sensores electroquímicos son dispositivos que responden a la concentración de los elementos de la muestra mediante una señal eléctrica. Sus fundamentos residen en la electroquímica, requiriendo, por tanto, las mismas reglas para operar apropiadamente [29]:

- Debe establecerse un circuito eléctrico cerrado.
- Debe cumplirse la condición de neutralidad de cargas.
- Se producen diferentes tipos de transferencia de carga, la cual puede ser electrónica, iónica o la combinación de ambas.

En cuanto a su estructura, los sensores electroquímicos están basados en celdas electroquímicas donde se presentan interfaces entre el electrodo y la solución, las cuales adquieren elevada importancia para entender su funcionamiento. Una interfaz se define como la región de unión de dos fases de diferentes propiedades físicas [29]. Las interfaces se pueden clasificar en dos tipos: polarizadas o no polarizadas. Son del primer tipo cuando no existe una transferencia de carga en la interfaz (ideales), mientras que son del segundo, cuando sí se produce. Son estas últimas las que permiten relacionar el potencial de la interfaz únicamente por la ecuación de Nernst (2.7).

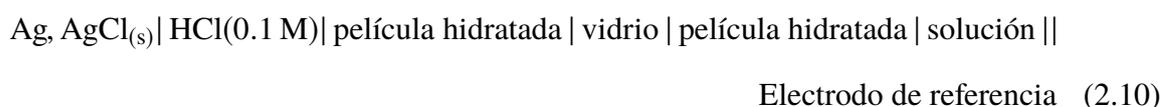
En el caso de los sensores potenciométricos, se pueden diferenciar varios tipos de interfaces, de las cuales, para la medición de pH, generalmente se emplean del tipo ion sensitivas. Estas interfaces tienen la particularidad de ser sensitivas a un tipo determinado de iones, los cuales terminan ocupando los sitios de unión en la superficie de la interfaz.

Por último, una celda potenciométrica está conformada, adicionalmente al electrodo principal (donde se producen las reacciones de interés), por un electrodo de referencia; el cual, además de cerrar el circuito como una condición para el proceso electroquímico, cumple con determinadas características de estabilidad, reversibilidad y reproducibilidad [29]. Existen varios tipos de electrodo de referencia. El recomendado es el electrodo de hidrógeno estándar (SHE) sobre el cual también se ha establecido un potencial equivalente a 0 [24]. No obstante, dada la dificultad para construirlo, su elevado costo, además del riesgo al emplearlo, existen otras alternativas para electrodos de referencia, tales como el electrodo de calomel saturado (SCE) y electrodo de plata/cloruro de plata (SSCE) [30], cada uno con una tensión definida respecto al electrodo estándar (0.242V para el SCE y 0.197 para el SSCE).

2.2.2 Sensor de electrodo de vidrio

No solo es uno de los sensores más conocidos, sino también es reconocido como método secundario para medición de pH según la IUPAC [24]. El sensor basado en electrodo de vidrio está conformado en realidad por dos electrodos, uno que es propiamente el electrodo de bulbo de vidrio, mientras que otro es el electrodo de referencia. Usualmente, ambos electrodos se encuentran empaquetados en el mismo dispositivo, donde el electrodo de referencia se ubica en un recipiente concéntrico al de bulbo de vidrio, que tiene contacto con la solución a través de un diafragma [23].

Estructuralmente, el sensor basado en electrodo de vidrio está conformado por un celda electroquímica, expresada por la ecuación (2.10) [29]:



Como se puede apreciar en (2.10), el contacto del electrodo de vidrio con la solución acuosa ocasiona la creación de una película hidratada delgada en ambas superficies del cristal, tanto interna como externa. En dichas películas hidratadas se realizan dos procesos, uno correspondiente al intercambio de iones entre la película hidratada y la solución, y otro a la difusión de todos los iones participantes a través del vidrio. Ambos procesos contribuyen a la sensibilidad de sensor [29].

La tensión de salida del electrodo de bulbo vidrio se relaciona con el valor de pH a partir de la ecuación de Nernst, tal como lo muestra (2.11):

$$\Delta E = C + \frac{RT}{F} \ln(10)\text{pH} \quad (2.11)$$

A fin de evitar el cálculo analítico de la constante C , basta con efectuar la calibración del sensor mediante soluciones buffer de pH conocidos, respecto a cuya regresión lineal se calcula el valor de pH de la solución a medir.

A pesar de ser uno de los dispositivos más empleados para la medición del pH, tiene serios inconvenientes al momento de miniaturizarlos y emplearlos en mecanismos que requieran de menor espacio, como, por ejemplo, en aplicaciones orientadas a biomedicina [31]. Adicionalmente, el tiempo de respuesta del sensor es relativamente lento, puesto que, basado en su funcionamiento, se tiene que realizar un proceso de difusión de los iones a través del cristal, lo cual es intrínsecamente lento, haciendo que no sea aplicable para determinados escenarios donde el tiempo de respuesta es una de las condiciones primordiales, a fin de reducir también el consumo de energía empleado en el sensor.

Es así como aparece otra alternativa, la cual está basada en semiconductores y presenta un menor tiempo de respuesta, además de ser más robusto y miniaturizable, aunque con ciertos retos a considerar en el diseño de la electrónica de adaptación. Este último será empleado en el presente trabajo y, por tanto, merece un análisis más profundo, el cual se desarrollará en la siguiente sección.

2.3 Sensor basado en transistor de efecto campo sensitivo a iones (ISFET)

El sensor basado en transistor de efecto campo sensitivo a iones (ISFET) es un sensor potenciométrico basado en semiconductores, con un amplio desarrollo que le ha permitido alcanzar niveles de optimización, desde los materiales empleados para su fabricación, hasta los circuitos de acondicionamiento para efectuar la medición de pH.

El ISFET [32] basa su funcionamiento en el MOSFET convencional, al cual se le ha adicionado una interfaz ion selectiva. Por tanto, para poder entender su forma de operación, es necesario abordar ambos conceptos y luego integrarlos en el dispositivo final.

2.3.1 Transistor de efecto campo de metal-oxido-semiconductor (MOSFET)

El transistor de efecto campo de metal-oxido-semiconductor (MOSFET) es un dispositivo de cuatro terminales basado en semiconductores, cuyo funcionamiento se basa en la modulación de la conductividad de una resistencia semiconductor, a partir de la aplicación de una tensión eléctrica o un campo eléctrico en forma perpendicular a la extensión de dicho dispositivo, recibiendo este proceso el nombre de principio de modulación de conductividad o efecto campo [33].

El MOSFET cuenta con dos difusiones a través de las cuales se realiza la conexión con el canal formado en el semiconductor del sustrato (denominado así al semiconductor donde se forma el canal). Al aplicar una diferencia de tensión entre ambas difusiones, se produce una movilidad en las cargas libres generadas, dando origen a una corriente. Esta corriente tiene la expresión presentada en (2.12):

$$i_D = \frac{W}{L} \mu_e \frac{\epsilon_0}{t_0} \left(v_{GS} - V_{FB} - |2\phi_p| + \frac{t_0}{\epsilon_0} Q^*(v_{SB}) - \frac{v_{DS}}{2} \right) v_{DS} \quad (2.12)$$

donde W es el ancho del canal, L es la longitud del canal, $Q^*(v_{SB})$ son las cargas distribuidas en el canal, v_{GS} es la tensión entre compuerta y fuente, y v_{DS} es la tensión entre drenador y fuente. Esta ecuación es la más empleada al referirse al funcionamiento del MOSFET, la cual suele expresarse como se muestra en (2.13) [34]:

$$i_D = \frac{W}{L} \mu C_{ox} \left[v_{GS} - V_{TH} - \frac{v_{DS}}{2} \right] v_{DS} \quad (2.13)$$

donde $C_{ox} = \epsilon_0/t_0$ es la capacitancia por unidad de área del aislante y el V_{TH} hace referencia a la tensión umbral, cuya ecuación lo expresa (2.14):

$$V_{TH} = V_{FB} + |2\phi_p| - \frac{Q^*(v_{BS})}{C_{ox}} \quad (2.14)$$

2.3.2 Interfaz electrolito-aislante-semiconductor (EIS)

La interfaz electrolito-aislante-semiconductor (EIS) se fundamenta en el principio de los sitios de unión, el cual fue presentado por primera vez por Yates en 1974 [35], quien propone un método para el cálculo de las densidades de carga superficiales, debido a los sitios de unión en la interfaz entre el aislante y el electrolito.

La interacción entre el aislante y el electrolito genera una densidad de carga en la superficie σ_o , la cual produce un potencial eléctrico en la interfaz φ_o . Esta carga, a su vez, está compensada por una carga distribuida a lo largo del electrolito. Puesto que existe un conjunto de iones adheridos a la superficie del aislante, su concentración $[H^+]_S$ es diferente a la del resto de la solución $[H^+]$. Esto se verifica por la relación de Boltzmann [36], mostrada en (2.15):

$$[H^+]_S = [H^+] e^{-\frac{q\varphi_o}{KT}} \quad (2.15)$$

El potencial generado en la superficie del aislante φ_o disminuye en el electrolito conforme se aleja de la interfaz a través de una capa difusa, fenómeno explicado a través del modelo de Gouy-Chapman-Stern. Según este modelo, esto se debe a las cargas en el electrolito,

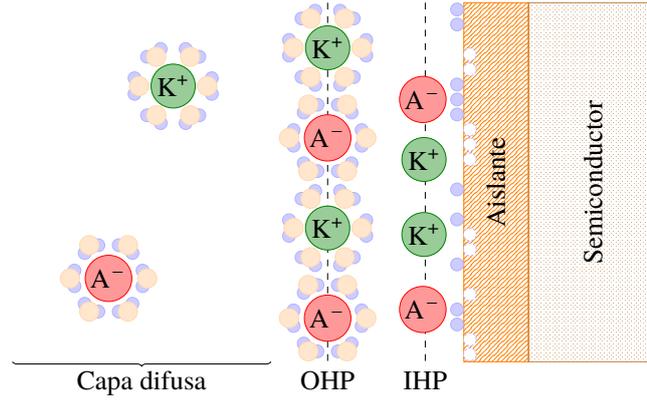
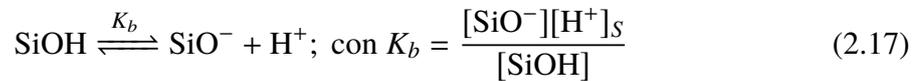
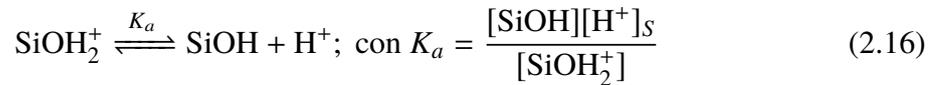


Fig. 2.1: Estructura de la interfaz EIS, formada por la capa IHP, OHP y difusa.

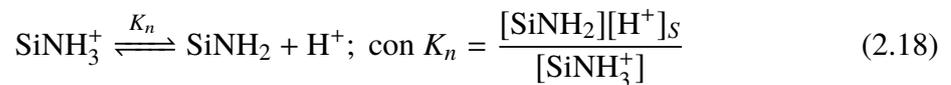
las cuales están distribuidos en capas: una capa difusa, formada por los iones hidratados distribuidos en la solución, y dos planos de carga denominados plano externo de Helmholtz (OHP), que marca la frontera de aproximación de los iones hidratados, y el plano interno de Helmholtz (IHP), constituido por los iones no hidratados que han logrado atravesar el plano externo, formando sitios de unión con la superficie del aislante (proceso denominado *surface complexation*) [37]. En la Fig. 2.1 se muestra un diagrama con estas regiones.

Para la cuantificación de dichas cargas, se procede partiendo de su relación con los sitios de unión en la superficie del aislante.

Los sitios de unión se forman a partir de la ionización de las moléculas de la superficie inicialmente neutras, las cuales al aceptar o donar protones (iones de hidrógeno), generan sitios con carga positiva o negativa, tal como se muestra en las ecuaciones (2.16) y (2.17) [38], [39]:



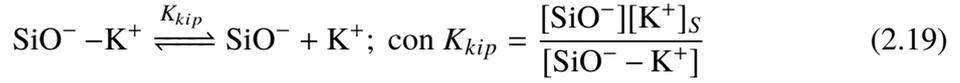
donde SiOH_2^+ , SiOH y SiO^- corresponden a los sitios con carga positiva, neutra y negativa, respectivamente. Grattarola [40] introdujo una ecuación adicional en caso el aislante fuera Si_3N_4 , donde los sitios no solo estaban conformados por componentes del silanol (SiOH), sino también por aminas, cuya reacción se muestra en (2.18):



donde SiNH_3^+ y SiNH_2 corresponden a los sitios de unión de las aminas. Los iones formados en los sitios de unión son los que determinan la carga en la superficie y adquieren la denominación

de iones determinantes del potencial.

Por otra parte, los iones adsorbidos, formando la IHP producto del proceso de *surface complexation*, compensan hasta cierto punto la carga neta de la superficie, donde las reacciones que muestran su interacción con los sitios de unión se expresan a través de las ecuaciones (2.19) y (2.20) [37], [38], [41]:



Si se considera el total de sitios de unión igual a N_s , de los cuales N_{sil} corresponde al total de sitios debido al silanol, mientras N_{nit} corresponde al total de sitios debido a la amina, se tendrían las ecuaciones (2.21), (2.22) y (2.23):

$$[\text{SiOH}_2^+] + [\text{SiOH}] + [\text{SiO}^-] + [\text{SiO}^- - \text{K}^+] + [\text{SiOH}_2^+ - \text{A}^-] = N_{\text{sil}} \quad (2.21)$$

$$[\text{SiNH}_3^+] + [\text{SiNH}_2] = N_{\text{nit}} \quad (2.22)$$

$$N_{\text{sil}} + N_{\text{nit}} = N_s \quad (2.23)$$

Como se puede apreciar en (2.21), la contribución en los sitios del silanol se deben, en parte, a los sitios de unión libres, así como los que ya han sido ocupados por los iones adsorbidos de la solución. Con estas ecuaciones, se puede realizar una transformación para obtener una expresión donde cada uno de los elementos dependan de los términos característicos de la solución, tales como K_a , K_b , K_{kip} , K_{aip} , $[\text{A}^-]$ y $[\text{K}^+]$. Para ello, en primera instancia, se transforman las ecuaciones de las constantes de equilibrio (2.16), (2.17), (2.18), (2.19) y (2.20), para que dependan de los sitios neutros. Lo cual, reemplazado en (2.22), resulta en las ecuaciones (2.24) y (2.25):

$$[\text{SiOH}] \left(\frac{[\text{H}^+]_S}{K_a} \left[1 + \frac{[\text{A}^-]_S}{K_{aip}} \right] + \frac{K_b}{[\text{H}^+]_S} \left[1 + \frac{[\text{K}^+]_S}{K_{kip}} \right] + 1 \right) = N_{\text{sil}} \quad (2.24)$$

$$[\text{SiNH}_2] \left(\frac{[\text{H}^+]_S}{K_n} + 1 \right) = N_{\text{nit}} \quad (2.25)$$

Reemplazando los valores de los sitios neutros mostrados en (2.24) y (2.25) en las ecuaciones anteriores, se obtiene las relaciones para cada tipo de sitio de unión. Posteriormente, se puede obtener la densidad de cargas en la superficie (σ_0) como la suma de la contribución de los iones determinantes del potencial, resultando la ecuación (2.26):

$$\sigma_0 = q([\text{SiOH}_2^+] - [\text{SiO}^-] + [\text{SiNH}_3^+])$$

$$\sigma_0 = qN_{\text{sil}} \left(\frac{[\text{H}^+]_S [\text{H}^+]_S - K_a K_b}{[\text{H}^+]_S [\text{H}^+]_S (1 + K_{\text{aip}}^{-1} [\text{A}^-]_S) + K_b K_a (1 + K_{\text{kip}}^{-1} [\text{K}^+]_S) + K_a [\text{H}^+]_S} \right) + qN_{\text{nit}} \left(\frac{[\text{H}^+]_S}{[\text{H}^+]_S + K_n} \right) \quad (2.26)$$

Por otra parte, la carga de los iones adsorbidos formando la IHP (σ_b) se obtiene de la sumatoria de los iones del electrolito, lo cual se expresa en (2.27):

$$\sigma_b = q([\text{K}^+] - [\text{A}^-]) \quad (2.27)$$

Posteriormente, en la capa difusa, la ecuación de la carga (σ_d) se obtiene a partir de la teoría de Gouy-Chapman, mostrada en (2.28) [38], [42]:

$$\sigma_d = -\sqrt{8\varepsilon_r \varepsilon_0 KTC} \sinh\left(\frac{q\varphi_d}{2KT}\right) \quad (2.28)$$

donde ε_r es la permitividad relativa de la solución, ε_0 es la permitividad eléctrica del vacío, K es la temperatura, T es la temperatura y C es la concentración de los iones del electrolito en el *bulk* de la solución¹.

Es importante mencionar que las concentraciones $[\text{K}^+]$ y $[\text{A}^-]$ están relacionadas con la concentración C a través de las ecuaciones (2.29) y (2.30) [38]:

$$[\text{K}^+] = C e^{-\frac{q\varphi_b}{KT}} \quad (2.29)$$

$$[\text{A}^-] = C e^{\frac{q\varphi_b}{KT}} \quad (2.30)$$

Finalmente, en el semiconductor también se puede hallar el valor de la densidad de carga (σ_s) a partir de sus concentraciones intrínsecas, como se muestra en (2.31) [38], [41], [43]:

$$\sigma_s = \sqrt{2\varepsilon_s \varepsilon_0 KT} \left(p_0 \left[e^{-\frac{q\varphi_s}{kT}} + \frac{q\varphi_s}{kT} + 1 \right] + n_0 \left[e^{\frac{q\varphi_s}{kT}} - \frac{q\varphi_s}{kT} - 1 \right] \right)^{1/2} \quad (2.31)$$

donde ε_s es la permitividad relativa del silicio. Luego, por el principio de neutralidad de cargas, se tiene, además, la relación entre las densidades de carga en la estructura EIS, presentada por (2.32):

$$\sigma_0 + \sigma_d + \sigma_b + \sigma_s = 0 \quad (2.32)$$

Adicionalmente, las densidades de carga expresadas por las ecuaciones (2.26), (2.27), (2.28) y (2.31), relacionadas, a su vez, por la ecuación (2.32), cuentan también con ecuaciones

¹Se denomina *bulk* a la región de la solución alejada de la OHP.

que los vinculan con los potenciales a lo largo de la interfaz EIS. De acuerdo a lo mencionado en la teoría del MOSFET, en el aislante se produce una capacitancia C_{ox} , la cual está relacionada con la carga del semiconductor (σ_d). Esto se expresa a través de la ecuación (2.33):

$$\varphi_s - \varphi_0 = \frac{\sigma_s}{C_{ox}} \quad (2.33)$$

Por otra parte, en cada una de las capas de Helmholtz se producen capacitancias: C_1 , entre la capa interna (IHP) y la superficie del aislante, y C_2 , entre la capa interna (IHP) y capa externa (OHP); las cuales están relacionadas con las cargas en cada una de las capas a través de las ecuaciones (2.34) y (2.35):

$$\varphi_0 - \varphi_b = \frac{\sigma_0 + \sigma_b}{C_1} \quad (2.34)$$

$$\varphi_b - \varphi_d = -\frac{\sigma_d}{C_2} \quad (2.35)$$

El objetivo de establecer las relaciones de las cargas en las diferentes secciones de la interfaz EIS, a la vez que los potenciales en cada uno de los niveles, es obtener una relación del potencial respecto al valor de pH, de tal forma que $\varphi_0 = f(pH)$. Bousse[44] fue uno de los primeros en encontrar esta relación, definiendo para ello una constante β , como se presenta en (2.36):

$$\beta = \frac{q^2 N_{sil} \delta_0}{C_{eq} KT}; \quad \delta_0 = 2\sqrt{2K_b/K_a} \quad (2.36)$$

quedando la relación entre el valor de φ_0 y el pH expresada como en (2.37):

$$\varphi_0 = \frac{KT}{q} \ln(10) \frac{\beta}{1 + \beta} (pH - pH_{pzc}) \quad (2.37)$$

donde pH_{pzc} representa la medida de concentración para la cual el potencial en la superficie de la interfaz es cero y la carga neta es cero.

Otro enfoque, proporcionado por Bergveld [45] (citado en [36], [37]) para hallar la relación entre el potencial φ_0 y el pH, es a través de la capacidad de buffer de la superficie del aislante β_s^2 . La relación resultante es la mostrada por la ecuación (2.38):

$$\frac{d\varphi_0}{dpH} = -\frac{\ln(10) \frac{KT}{q}}{\ln(10) \frac{KT}{q} \frac{C_{eq}}{\beta_s q} + 1} = -\ln(10) \frac{KT}{q} \alpha \quad (2.38)$$

²Se define capacidad de buffer como la resistencia que ofrece un buffer para cambiar su pH al añadirse una cantidad de base o ácido fuerte. La relación cuantitativa fue introducida por Van Slyke, quien definió a la capacidad de buffer como la concentración en equivalente gramo por litro para que cambie el valor de pH en 1[46].

donde α viene a ser la sensibilidad del ISFET, la cual se aproxima a 1 cuando el valor de la capacidad del buffer β_s es muy alta. De aquí que el máximo de la expresión (2.38) correspondería a la constante de proporcionalidad de la ecuación de Nernst(2.7) equivalente a 59.3 mV a 25°C [45].

2.3.3 Funcionamiento del ISFET

En las dos anteriores secciones se ha discutido características tanto del MOSFET, como de la interfaz EIS, considerándolos elementos separados. Ahora corresponde a combinar ambas contribuciones formando un solo elemento: el ISFET.

Puesto que la interfaz EIS se encuentra conectada en el lugar de la compuerta del MOSFET, la contribución de ésta dentro de las ecuaciones del dispositivo se pueden apreciar a nivel de la distribución de sus potenciales. Este análisis es realizado por Bergveld [31], a partir de la ecuación de la tensión de *flat-band*, mostrada en (2.39):

$$V_{\text{FB}} = \frac{\phi_m - \phi_{\text{Si}}}{q} - \frac{Q_{\text{ss}} + Q_{\text{os}}}{C_{\text{ox}}} \quad (2.39)$$

donde Q_{ss} corresponde a densidad de carga fija en los estados superficiales del semiconductor y Q_{sc} es la carga por unidad de área en la región de agotamiento en la superficie del semiconductor³.

En el caso del ISFET, la contribución del potencial intrínseco del metal ϕ_m debe ser reemplazado por la contribución de la interfaz EIS. Esta contribución comprende: el potencial del electrodo de referencia E_{ref} , donde se incluye el valor del potencial intrínseco del metal; y los potenciales debido a la interacción del electrolito con el aislante, correspondientes al potencial dipolar de la superficie de la solución χ_{sol} , que es constante, y el potencial superficial debido a la densidad de cargas en la interfaz φ_0 , el cual tiene una dependencia respecto al pH, tal como se mostró en la ecuación (2.37). Con esta modificación se llega a la expresión (2.40):

$$V_{\text{FB(ISFET)}} = E_{\text{ref}} - \varphi_0 + \chi_{\text{sol}} - \frac{\phi_{\text{sil}}}{q} - \frac{Q_{\text{ss}} + Q_{\text{os}}}{C_{\text{ox}}} \quad (2.40)$$

Esta nueva tensión puede reemplazarse en la ecuación de la tensión umbral del MOSFET,

³Se denomina estados superficiales a los estados de energía que se forman en la superficie de un material, los cuales son diferentes a los estados de la región que se adentra en él, conocida como espacios de carga. Estos estados pueden asumir el rol de donadores o aceptores de carga, dependiendo del nivel de energía que tengan respecto al nivel de energía intrínseco del material. En el MOSFET se presentan estados superficiales en la interfaz entre el semiconductor y el aislante [47].

resultado un expresión similar para el ISFET, como lo muestra (2.41):

$$V_{TH(ISFET)} = V_{FB(ISFET)} + |2\phi_p| - \frac{Q^*(v_{BS})}{C_{ox}} \quad (2.41)$$

Grattarola [40] propone una forma alternativa de expresar la tensión umbral, mostrada por la ecuación (2.42), la cual, si bien no tiene una correspondencia directa con los elementos que forman parte del sistema, se adapta mejor para fines de su modelamiento:

$$V_{TH(ISFET)} = V_{TH(MOS)} + E_{ref} + \varphi_{lj} + \chi_{sol} - \varphi_0 - \frac{\varphi_m}{q} \quad (2.42)$$

En este caso, se agrega un término φ_{lj} , el cual corresponde a la diferencia de potencial en la unión líquida.

Considerando este valor para la tensión umbral del dispositivo, las ecuaciones correspondientes a la corriente del MOSFET (2.13) siguen siendo válidas para el ISFET, con la diferencia que, en este caso, la tensión umbral es dependiente del valor del pH de la solución donde se encuentra sumergido el dispositivo.

2.3.4 Tecnologías de fabricación y aplicaciones del ISFET

Las tecnologías de fabricación hacen referencia a los materiales y metodologías empleadas en la construcción del ISFET. A lo largo de su historia se han registrado varios materiales utilizados como membranas sensitivas entre el semiconductor y el electrolito, con determinadas características de sensibilidad y rango de medición. El primer material empleado fue el óxido de silicio (SiO_2) [32], el cual tiene un rango de operación de 2-5pH y una sensibilidad entre 25 a 48 mV/pH [37] (otros estudios han mostrado una sensibilidad inferior de 20-35mV/pH [48]). Posteriormente, se empleó el nitruro de silicio Si_3N_4 [49], con un rango más amplio de 2-12pH y una sensibilidad de 46-56mV/pH. Otro material empleado es el óxido de aluminio (Al_2O_3) [50], con un rango igual que el anterior pero con una mayor sensibilidad de 53-57mV/pH [51]. También se encuentra el óxido de tantalio (Ta_2O_5), el cual presenta igualmente un rango de 2-12pH pero una sensibilidad superior a los anteriores de 56-57mV/pH [51]. Este material es el que posee la mayor capacidad de buffer, lo cual hace que su sensibilidad sea cercana al ideal de Nernst [45]. No obstante ello, de los anteriores, el más empleado para la fabricación de sensores ISFET es el nitruro de silicio. Esto debido a que es un material común en el proceso de fabricación de la tecnología CMOS, la cual es empleada para la fabricación de circuitos integrados, facilitando la integración del ISFET y su circuito de acondicionamiento en un mismo chip [52].

En cuanto a las aplicaciones del ISFET, éstas han sido variadas desde el momento de su concepción. Bergveld [32] inicialmente lo construyó para realizar mediciones neurofisiológicas, en las cuales se necesitaba identificar las causas del potencial eléctrico producidos en las neuronas, empleando el ISFET para medir su actividad iónica [53]. Otras aplicaciones, también presentadas por Bergveld[54], comprenden su empleo como sensores en la punta de catéteres para medición de parámetros fisiológicos de sistemas vivos, tal como la medición de la presión en la sangre a partir del valor del pH. También para medidas extracorporales de las concentraciones de iones en la sangre, lo cual aprovecha la posibilidad de medir en volúmenes muy pequeños y a la vez evita problemas de biocompatibilidad, entre otras.

Su avance es significativo en diferentes áreas tales como en biomedicina, siendo empleado para el monitoreo de regiones extremadamente estrechas, como el caso del tracto intestinal humano [55]; así como en agricultura de precisión, donde se emplea en conjunto con un sensor de electroconductividad y de temperatura, integrados en un mismo chip [56].

Su aplicación en monitoreo puede extenderse a otros entornos, como en el de caso parámetros medioambientales en el control de la calidad de agua. Una muestra se presenta en [57], donde se ha implementado una red de sensores inalámbricos para la medición *in situ* del pH agua, empleando sensores ISFET comunicados a través de una red de comunicaciones móviles. El sistema ofrece una solución de bajo costo y bajo consumo, lo cual es apropiado para largos periodos de monitoreo. Aplicaciones de este tipo son tomadas como antecedentes de la viabilidad de la propuesta planteada en el presente trabajo.

2.3.5 Principales desafíos al emplear el ISFET como sensor de pH

El ISFET presenta muchas ventajas respecto a los sensores tradicionales basados en bulbo de vidrio, tales como la posibilidad de miniaturizarlos a escalas suficientemente pequeñas, a fin de incluirlos en circuitos integrados, así como tener un tiempo de respuesta mucho más rápido que los sensores de electrodo de vidrio y ser más robustos, no requiriendo de un mantenimiento continuo [58]. No obstante, también cuenta con algunos retos a considerar en el diseño del sensor, así como de la electrónica de adaptación, debido al comportamiento no ideal del dispositivo cuando es sometido, ya sea, a periodos continuos de medición o, simplemente, debido a cambios en las condiciones de temperatura.

Moser *et al.* [59] hace un resumen de los principales retos a afrontar al momento de trabajar con dispositivos ISFET. Estos retos se describen en la TABLA N° 2.1.

Si bien no han sido considerados en el estudio anterior [59], el ISFET también cuenta con

TABLA N° 2.1: Principales retos para utilización del ISFET

Desafío	Descripción
<i>Efecto de cargas atrapadas</i>	Resultado de la transferencia de cargas en el proceso de fabricación de los chips. Esta carga atrapada afecta a la tensión umbral, desplazándola en un valor constante respecto a su tensión nominal.
<i>Desviación de la corriente</i>	Incremento progresivo de la corriente cuando se realiza mediciones prolongadas de una misma solución con el mismo valor de pH.
<i>Atenuación capacitiva</i>	Debido a la implementación del sensor en una tecnología estándar CMOS, lo cual resulta en una atenuación adicional de la señal. Se produce al emplear como membrana sensitiva al Si ₃ N ₄ .
<i>Sensibilidad ante la temperatura</i>	Al estar basado en semiconductores, el dispositivo también cuenta con una estrecha dependencia de la temperatura de operación.
<i>Respuesta al ruido</i>	No solo se debe al ruido introducido por el MOSFET, sino también debido a la interfaz química del sensor.

otros desafíos, tales como el fenómeno de histéresis o efecto de memoria, lo cual muestra la aparición de diferentes tensiones cuando se realizan mediciones sucesivas de la misma solución, variando el ancho de la curva de histéresis para diferentes valores de pH [58]. Adicionalmente, el dispositivo presenta sensibilidad ante la luz debido a que está basado en semiconductores, material que puede ser excitado por esta fuente, proporcionándole suficiente energía como para cambiar su estado de conducción. El efecto se acentúa aún más en el caso del ISFET debido a su compuerta flotante, afectando el valor de la tensión umbral [60].

Los retos presentados anteriormente corresponden a las características intrínsecas de los sensores basados en ISFET, los cuales ya han sido estudiados y pueden ser mitigados a través de algunas estrategias de compensación y procesamiento de señal.

2.4 Sensores comerciales para medición de pH

Es importante identificar los sensores que se emplearán para la implementación de la solución propuesta en el presente trabajo, los cuales, acorde a lo planteado inicialmente, deben ser de bajo consumo de energía.

Para la medición de pH existen una gran variedad de sensores y pueden estar basados en diferentes mecanismos de funcionamiento, tal como fue mostrado en la sección 2.2. Así por ejemplo, existen sensores de pH cuyo funcionamiento reside en imágenes, las cuales ofrecen no solo la medición de pH, sino también una visualización bidimensional de la actividad química de las soluciones [61]. Sin embargo, los más conocidos y por tanto más utilizados siguen siendo los basados en bulbo de vidrio y en ISFET.

TABLA N° 2.2: Características de sensores ISFET

Sensor	Característica	Valor
ISFET Sensor Sentron [62]	Sensor	Libre de vidrio, Transistor Efecto Campo Ion selectivo (ISFET)
	Precisión	+/-0.01pH
	Rango	0.00 – 14.00
	Variación	Máx. 0.14pH
	Consumo típico	100nA
Sensor CS526 Campbell [63]	Rango	0.00 – 14.00
	Alimentación	5Vdc
	Consumo máximo	15mA
	Salida	Lógica TTL, 2400 bps, 8 bits de dato, sin bit de paridad y un bit de stop
	Máxima longitud del cable	100m
	Longitud	102mm
	Diámetro	16mm
Sensor TF 396 Emerson [64]	Rango	2 – 12
	Elemento de medida	ISFET
	Variación	+0.05pH
	Tiempo de respuesta	99 % en 30 segundos
	Rango de temperatura	0-100°C
	Rango de presión	-10 a 100psig (32 a 790kPa)
	Longitud de cable	6m, 18m
Micro sonda HACH [65]	Diámetro	4.8mm
	Juntura	Fibra
	Longitud	130mm
	Tipo de sonda	ISFET
	Rango	0-14pH
	Rango temperatura	Continuo uso 0-60°C

Para el caso de los sensores basados en ISFET, correspondiente a la tecnología empleada en el presente trabajo, existen algunas manufacturas que ofrecen dichos sensores para aplicaciones industriales y educacionales, tales como se muestran en la TABLA N° 2.2. De estos sensores, cuatro presentan un rango completo de medición (0-14pH), mientras que uno de ellos tiene la mayor sensibilidad (0.01pH).

No obstante sus propiedades sorprendentes, los sensores presentados cuentan con un precio elevado debido a sus características de fabricación muy exigentes, las cuales exceden a las necesarias para la aplicación que se le dará en el presente estudio. Además, dado que la finalidad es la construcción de un prototipo para el sensado, la opción más adecuada sería emplear un sensor diseñado para aplicaciones en laboratorio. En este caso se han identificado

TABLA N° 2.3: Sensores ISFET para prototipos

Sensores	Característica	Valor
ISFET IMB (Barcelona) [66], [67]	Tensión de drenador	0.5-2V
	Corriente de operación	0.1-1.0mA
	Trasconductancia	>0.5mA/V
	Tensión umbral	-2.0-1.0V
	Sensibilidad	≥55mV/pH
	Rango lineal	1-13pH
	Precisión	0.05pH
	Variación de largo plazo	≤1.0mV/h
	Tiempo de vida	8 meses en inmersión continua
WINSENSE (Tailandia) [68]	Tensión de drenador	0.3V
	Salida analógica	1-2V
	Corriente de operación	25-30uA
	Trasconductancia	>0.5mA/V
	Temperatura	0°C - 100°C
	Sensibilidad	≥50mV/pH
	Rango lineal	2-12pH
	Precisión	0.01pH
	Tiempo de respuesta	10s continua

dos manufacturas, las cuales se muestran en TABLA N° 2.3.

De ambas alternativas, la que ofrece mejores características en el balance costo-beneficio, además de consumir menor energía (tomando en cuenta los valores de tensión y corriente de operación), es el proporcionado por WINSENSE, por lo que en el presente trabajo se emplearán los sensores ISFET de esta empresa.

2.5 Diseño de bajo consumo de energía

Uno de los retos más importantes al momento de diseñar un sistema de monitoreo, es el consumo de energía. En el caso de sistemas que están orientados a operar en ambientes donde el acceso de energía es restringido, el principal criterio de diseño consiste en reducir el consumo de los componentes que lo constituyen, a fin de reducir el consumo del sistema total. Para reducir el consumo de dichos componentes existen varias estrategias que se pueden emplear de acuerdo al tipo de tecnología que empleen y en caso se traten de sistemas digitales o analógicos. Existe todo un estudio para abordar el bajo consumo, el cual tiene sus fundamentos en los conceptos de energía y potencia en un sistema eléctrico. Este estudio conlleva hacia el diseño por niveles, escogiendo el más adecuado de acuerdo al tipo de sistema

TABLA N° 2.4: Niveles de diseño de baja potencia

Nivel	Descripción
TECNOLOGÍA	Nivel más bajo de diseño, donde se abordan las características propias de los transistores tales como su tensión umbral. Este nivel está orientado a los parámetros de fabricación de los dispositivos semi-conductores.
CIRCUITOS Y LÓGICA	Segundo nivel de diseño. Se aborda el arreglo de transistores formando circuitos, los cuales se orientan a la reducción de consumo de energía.
ARQUITECTURA	Tercer nivel de diseño. Se evalúa cómo los circuitos y módulos lógicos se encuentran dispuestos dentro de un sistema electrónico a fin de conseguir el objetivo de bajo consumo de energía. A este nivel se encuentran los procesadores.
ALGORITMO	Cuarto nivel de diseño. Se emplean algunos mecanismos de estimación del consumo del algoritmo, a través de los cuales se identifica dos clases de disipación: una orientada a su disipación inherente y la otra referida al los recursos empleados en su implementación.
SISTEMA	Último nivel de diseño. Orientado a las estrategias de particionamiento y estados de operación del sistema global. En este caso, la desactivación de los módulos no usados es una de las estrategias comúnmente empleadas para reducir el consumo de energía.

que se requiera implementar.

El Prof. Rabaey de la Universidad de Berkeley menciona en su libro sobre metodologías de diseño de baja potencia [69], los diferentes niveles desde los cuales se puede abordar el reto de construir un sistema de bajo consumo de energía. Si bien es cierto, están orientados hacia el diseño digital, pueden extrapolarse, hasta cierto punto, al diseño analógico y mixto. Estos niveles se muestran en la TABLA N° 2.4.

Por otra parte, en la industria, el consumo de potencia ha sido abordado desde diferentes frentes, los cuales contemplan, principalmente, la reducción del tamaño del chip, así como la capacitancia del encapsulado, escalando la fuente de alimentación, usando estrategias de administración del uso de energía y el empleo de mejores técnicas de diseño [69]. En el presente trabajo no es posible abordar las estrategias que impliquen una mejora tecnológica, como el caso de la reducción del integrado o un escalamiento de la fuente de alimentación, puesto que implicaría la fabricación de un nuevo chip con estas características. No obstante, sí se podría optar por mejores técnicas de diseño, lo cual comprenderían los niveles a partir del circuito hasta el sistema general, incorporando técnicas que impliquen una óptima utilización de energía, a partir del algoritmo implementado en el microprocesador empleado para la

implementación del sistema.

Es importante, por ello, hacer una buena selección de los microcontroladores, puesto que deben cumplir con el requerimiento de consumir el mínimo de energía.

2.6 Microcontroladores de ultra bajo consumo

Los microcontroladores son dispositivos que cuentan con un microprocesador (o unidad central de procesamiento, CPU) y unidades periféricas, tales como memorias, puertos de entrada/salida, contadores, temporizadores, ADCs, DACs, y otros bloques funcionales que les permiten comunicarse y ejercer acción sobre su entorno externo [70].

Los microcontroladores existen para diferentes propósitos, los cuales varían sus características de acuerdo a la aplicación que se les otorga. Para el caso de los orientados a trabajar con sensores, existen microcontroladores que están diseñados para presentar un bajo consumo y, por ende, facilitan su inserción en una red de sensores. Una revisión con los principales microcontroladores de bajo consumo se presenta en la TABLA N° 2.5.

Por otra parte, existe un grupo especial dentro del conjunto de microcontroladores, diseñados para implementar sistemas de señal analógica y mixta (AMS). Esta característica especial evitaría el uso de circuitos integrados dedicados para la implementación de OPAMs, filtros, entre otros módulos analógicos necesarios en la interfaz de los sensores, integrando toda la implementación en un solo dispositivo. Entre los microcontroladores de este tipo se encuentran los mostrados en la TABLA N° 2.6.

Es necesario tener en cuenta que, si bien los microcontroladores cuentan con una variedad de bloques analógicos configurables, tienen limitaciones en cuanto a características eléctricas y de configuración que los hace útiles para aplicaciones muy específicas.

En el caso del PSoC®, éste corresponde a una marca comercial de Cypress, la cual comprende un conjunto de microcontroladores basados en ARM® Cortex® y dispone de la característica de programar bloques digitales y configurar elementos analógicos. Su introducción en el mercado no es muy reciente, no obstante, su empleo es relativamente reducido comparado con microcontroladores de marcas más conocidas como Microchip o Texas Instruments. Dentro de esta familia de microcontroladores, existen varios tipos, siendo los más actuales el PSoC 4 y PSoC 5, cuyas características se presentan en la TABLA N° 2.7.

Si bien es cierto, de los tipos de PSoC, el que presenta características de menor consumo de energía es la familia PSoC4, ésta no dispone de muchos módulos analógicos por lo que,

TABLA N° 2.5: Principales microcontroladores de bajo consumo

Manufactura	Dispositivos	Características
STMICROELECTRONICS	Serie STM8L (8-bits) y serie STM32L (32 bits), agrupada en STM32L4, L1 y L0 [71]	STM32L0 basado en ARM® Cortex®-M0+, con consumo dinámico inferior a 87uA/MHz y 250 nA en modo de ultra bajo consumo.
TEXAS INSTRUMENTS	Familia MSP430™ y sus variantes optimizadas en tiempo de respuesta (MSP430F1x/ F2x/ G2x) o bajo consumo (MSP430F5x/ FR5x/ FR2x) [72]	Llega a obtener un consumo mínimo de 120uA/MHz y 0.7uA en modo de reposo.
MICROCHIP	Familias de 8 bits (PIC10, PIC12, PIC16, PIC18 y algunos híbridos con AVR), 16 bits (PIC24 y dsPIC®) y 32 bits (PIC32 y dispositivos SAM) [73]	Corrientes en estado de reposo por debajo de 9nA y corrientes de activación por debajo de 30uA/MHz.
AMBIQ MICRO	Microcontroladores Apollo, en versiones Apollo 1 y Apollo 2 [74]	Basados en ARM Cortex M4F, con un consumo de corriente activo de 35uA/MHz y 143nA en estado de reposo.
CYPRESS	Dispositivos de 8, 16 y 32 bits, basadas en procesadores propietarios (RISC o CISC) así como otros basados en ARM® de 32 bits [75]	Microcontroladores FM0+ basados en ARM® Cortex®-M0+ con un consumo de corriente activo de 40uA/MHz y 0.6uA en modo de reposo.

TABLA N° 2.6: Microcontroladores de señal mixta

Microcontroladores	Características
FAMILIA PSOC (CYPRESS)	Microprocesador ARM con bloques analógicos de alto rendimiento, dispositivos lógicos programables, conexiones y ruteo programables, y módulos CapSense®. Destacan el PSoC4 y PSoC5.
MSP430FR2x (TEXAS INSTRUMENTS)	Familia de 16 bits, cuentan con amplificadores operacionales y amplificadores de transimpedancia. Tienen un consumo de 126uA/MHz en modo activo y un consumo de 770nA en modo de reposo. Destacan el MSP430FR2310 y el MSP430FR2311.
PIC16F777 (MICROCHIP)	Microcontroladores de 8 bits, disponen de amplificadores operacionales y comparadores.

para la implementación del diseño del proyecto, se emplea la versión más completa proveída por la familia PSoC5.

TABLA N° 2.7: Familias de microcontroladores PSoC

Familia	Características
PSoC 4 [76]	<ul style="list-style-type: none"> ■ Basados en ARM Cortex-M0 y Cortex-M0+. ■ Familia 4000: incluye posibilidad de conectar con módulos CapSense®. ■ Familia 4100: incorporan bloques configurables, sólo analógicos. ■ Familia 4200: incluye también bloques digitales configurables (PLD). ■ 4200L es el microcontrolador con mayor capacidad en bloques analógicos. Dispone de 4 amplificadores operacionales, 6 comparadores, 1 ADC tipo SAR de 12 bits de resolución y 4 DACs de corriente. ■ Familias 4100BLE y 4200BLE (derivadas de 4100 y 4200) cuentan con comunicación inalámbrica Bluetooth (BLE).
PSoC 5 [77]	<ul style="list-style-type: none"> ■ Basados en ARM® Cortex®-M3. ■ Cuentan con bloques de filtros digitales, 24 canales de acceso directo a memoria (DMA), un procesador digital de punto fijo de 24 bits, más de 20 unidades de diseño para periféricos analógicos de precisión, amplificadores operacionales, ADC tipo SAR y Sigma-Delta, DACs de 8bits, que pueden ser de corriente o de tensión de acuerdo a la configuración.

2.7 Redes de sensores inalámbricos

Las redes de sensores inalámbricos (WSN) constituyen el eje de los programas de monitoreo de parámetros medioambientales. Se pueden definir como una red de nodos, los cuales trabajan de forma cooperativa con la finalidad de monitorear o controlar el medio que los circunda. Normalmente cuentan con tres componentes fundamentales, los cuales son: los nodos sensores, el dispositivo enrutador (denominado *gateway*) y el observador [78]. Los nodos y los *gateway* forman parte del emplazamiento de la red, los cuales se encuentran en contacto directo con el medio que se requiere monitorear, mientras el observador normalmente se encuentra en una estación remota. Los nodos sensores cuentan con cuatro elementos fundamentales: los sensores, la unidad de procesamiento (comúnmente un sistema embebido), el sistema de energía y el módulo de comunicación inalámbrica.

Respecto la unidad de procesamiento, existen diferentes alternativas [79]. Tradicionalmente los nodos sensores se han basado en microcontroladores, no obstante, modernas aplicaciones con demandas de procesamiento en tiempo real, bajo consumo y bajo coste, han orientado los diseños a otras arquitecturas. El procesador de señales digitales (DSP), por ejemplo, constituye una buena opción al momento de realizar tareas de procesamiento elevadas, en donde demuestra tener mucha mayor eficiencia que los microcontroladores. Los circuitos integrados para aplicaciones específicas (ASICs), por su parte, constituyen la opción de mayor optimización, puesto que son construidos para una tarea dedicada, aunque

con una flexibilidad mínima. Los arreglos de compuertas programables por campo (FPGAs), emergen como una alternativa para la ejecución de aplicaciones computacionales complejas, no obstante, con un precio mas elevado y limitaciones para su integración y autonomía en un nodo sensor. Nuevas soluciones implementan el nodo sensor en un sistema integrado (SoC), los cuales incluyen la interfaz para el sensor, el procesador y el módulo de comunicaciones, constituyendo una nueva tendencia para la implementación de los nodos sensores, aunque con ciertas consideraciones respecto a su consumo de energía.

Respecto a las tecnologías inalámbricas empleadas en las redes de sensores, las más populares son Zigbee, Bluetooth y 6LoWPAN [78]. Zigbee, es un protocolo diseñado para ejecutar comunicaciones de alto nivel empleando baja potencia basado en el estándar IEEE 802.15.4 [80]. Este protocolo soporta redes robustas y no costosas con un número bastante amplio de nodos. Bluetooth fue diseñado también para sistemas de bajo consumo, no obstante, es apropiado para distancias cortas, así como para redes con pocos nodos activos [80]. Por su parte, 6LoWPAN (IPv6 sobre redes de área personal inalámbricas de baja potencia), es una tecnología de red o de adaptación de capa de red que permite que los paquetes de IPv6 puedan ser transportados eficientemente en capas de enlace pequeñas, como es el caso de las definidos por el estándar IEEE 802.15.4. Al tratarse de un estándar abierto, facilita su implementación, a la vez que tiene una mejor compatibilidad con otros protocolos de Internet [81].

La información producida por la red de sensores necesita llegar hacia el observador a través de algún medio de comunicación. Para este propósito, se emplea con frecuencia el Internet, lo cual conlleva a un nuevo concepto denominado Internet de las Cosas.

El Internet de las cosas (IoT) es un concepto empleado para referirse a objetos (usualmente sistemas embebidos) con la capacidad de generar información y comunicarse entre sí, con personas o servicios, a través de Internet a una escala global [82]. Este concepto data desde los años 90, no obstante, ha adquirido un énfasis enorme en los últimos años, logrando introducirse en los diferentes aspectos de la vida de las personas con proyecciones a evolucionar hasta el punto donde absolutamente todo esté conectado a Internet (Internet del Todo, IoE). Las redes de monitoreo de parámetros ambientales, los sistemas de automatización de viviendas y edificios, las redes de energía inteligentes (*smart grids*), entre otros, son claros ejemplos de tecnologías vinculadas al concepto de Internet de las Cosas.

Si bien se trata de un concepto amplio, las bases de su desarrollo dependen de las redes

de sensores inalámbricos (WSN) y de los dispositivos de identificación por radio frecuencia (RFID). Por tanto, cuenta con los mismos desafíos, tales como el consumo de potencia, la capacidad computacional, la escalabilidad, la tolerancia a fallos, entre otros. Adicionalmente, dada su condición de estar conectado a Internet, depende esencialmente de la disponibilidad de este elemento en cualquier lugar y en cualquier momento.

El presente trabajo es acerca de un sistema de monitoreo de parámetros ambientales, lo cual constituye parte de una red de sensores. Si bien es cierto, la red de sensores podría funcionar adecuadamente sin la necesidad de conectarse a Internet, dada la tendencia de las tecnologías a tener una conexión con la red global, es poco realista pensar en una solución a largo plazo que no cuente con este elemento como un requerimiento.

2.8 Modelamiento del sensor basado en ISFET

El ISFET se puede modelar como dos elementos, uno correspondiente al MOSFET y otro, a la interfaz electrolito-aislante-semiconductor (EIS), debido a la cual se debe su sensibilidad al pH de la solución. Para cada uno de estos componentes existen ecuaciones que los gobiernan, así como sus modelos correspondientes, los cuales en conjunto constituyen el modelo del sensor de pH basado en ISFET.

En primer lugar, se abordará el modelamiento del MOSFET, las alternativas que se tienen para realizarlo, así como la explicación de las ecuaciones que lo fundamentan. Posteriormente, se procede a modelar la interfaz EIS, complementado con el modelo anterior, formando el sistema completo correspondiente al ISFET.

2.8.1 Modelamiento del MOSFET

Respecto al modelamiento del MOSFET existen varios esfuerzos a lo largo de la historia de este dispositivo, los cuales se han enfocado en determinadas características, aumentando su precisión y adaptándolo a las nuevas tecnologías de fabricación.

Una lista con la evolución de los modelos de acuerdo a generaciones se puede apreciar en la TABLA N° 2.8. Conforme avanza la generación de modelos, también aumenta su precisión para tecnologías de semiconductor más avanzadas (menor longitud de canal), derivando también en una mayor cantidad de parámetros para describir el modelo. No obstante, en el caso del ISFET, puesto que se trata usualmente de dispositivos de canal largo, modelos de primeras generaciones serían suficientes para describirlo.

De los modelos anteriormente presentados, los modelos de nivel 1 y 2 del MOSFET, así

TABLA N° 2.8: Modelos del MOSFET [83]

Generación	Modelos
PRIMERA GENERACIÓN	<ul style="list-style-type: none"> ■ Modelo 1. Modelo cuadrático Shichman-Hodges ■ Modelo 2. Basado en características físicas. ■ Modelo 3. Semiempírico con ajuste de curvas.
SEGUNDA GENERACIÓN	<ul style="list-style-type: none"> ■ BSIM - Universidad de Berkeley. ■ BSIM 2 - Universidad de Berkeley.
TERCERA GENERACIÓN	<ul style="list-style-type: none"> ■ BSIM3 - Universidad de Berkeley. ■ BSIM4 - Universidad de Berkeley. ■ MM9 - Phillips Research. ■ EKV - EPFL.
CUARTA GENERACIÓN	<ul style="list-style-type: none"> ■ EKV3.0 - EPFL. ■ HiSIM - Universidad de Hiroshima. ■ MM11 - Phillips Research. ■ SP - Universidad de Pennsylvania.

como el modelo EKV, han sido empleados satisfactoriamente para simular la componente semiconductor de los ISFET. Los primeros descritos principalmente en SPICE, mientras el último orientado al lenguaje VHDL-AMS. En este caso, se presentarán las ecuaciones que rigen los parámetros de los primeros niveles del modelo del MOSFET.

Los tres primeros niveles del modelo del MOSFET, cuentan con dos grupos de parámetros: eléctricos y de procesamiento, de los cuales los primeros constituyen los parámetros derivados, mientras que los últimos son los primarios. Aunque ambos conjuntos pueden ser ingresados dentro del modelo, la información sobre los parámetros eléctricos prevalece sobre los computados a partir de los primarios [84].

Los parámetros eléctricos son 5 en total y se muestran en la Fig. 2.2. Estos parámetros son suficientes para describir el modelo del MOSFET de nivel 1, no obstante, usualmente es necesario emplear otros parámetros correspondientes a las características físicas y geométricas del dispositivo, conllevando al siguiente nivel del modelo del MOSFET.

Los parámetros primarios, por su parte, se pueden agrupar en un conjunto de parámetros de caracterización de la compuerta y del canal; otro conjunto, de caracterización de junturas en drenador y fuente; y un tercer grupo, con significado solo en el modelo de nivel 3 del MOSFET. Esta clasificación se puede apreciar en la Fig. 2.3. Cada uno de estos parámetros son útiles para definir las características del dispositivo y son también a partir de los cuales se calculan los parámetros eléctricos, otrora ingresados directamente. De éstos, los empleados más frecuentemente en los modelos del MOSFET son los mostrados en la TABLA N° 2.9.

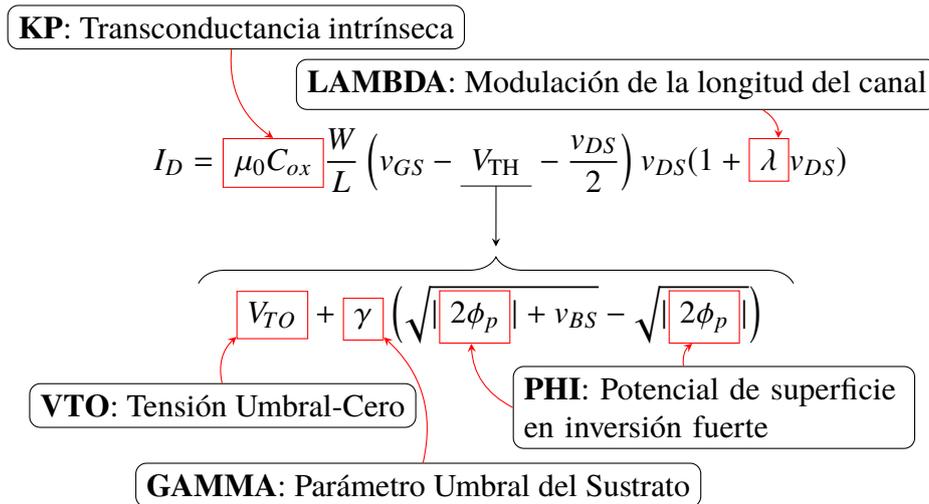


Fig. 2.2: Parámetros eléctricos del modelo del MOSFET

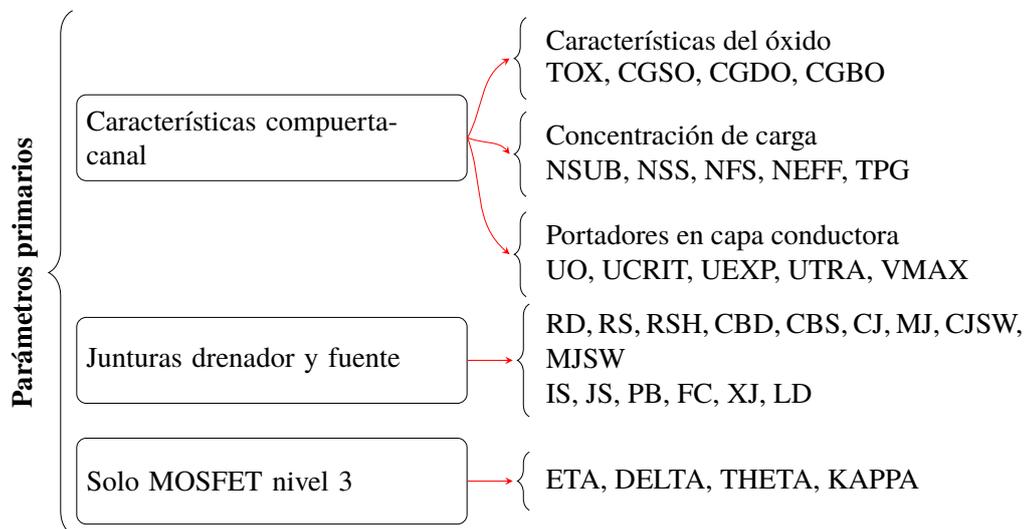


Fig. 2.3: Parámetros primarios del modelo del MOSFET

Algunos de estos parámetros se emplearán en el modelamiento de la parte semiconductor del ISFET, cuya interfaz EIS se modela en la siguiente sección.

2.8.2 Modelamiento de la interfaz EIS y del ISFET

El modelo empleado corresponde al macromodelo propuesto por Sergio Matinoia y Giuseppe Massobrio [85], el cual ha sido empleado con recurrencia en investigaciones posteriores para validar el comportamiento del ISFET. Este modelo utiliza el nivel 2 del MOSFET para simular la parte semiconductor del dispositivo, el cual ya se analizó previamente. En cuanto a la interfaz ion sensitiva, emplea modelos de capacitores conectados en serie, cada uno de ellos representando una determinada fase en la interfaz EIS.

La EIS presenta una distribución de cargas debido a la interacción de los iones de

TABLA N° 2.9: Principales parámetros primarios del MOSFET

Parámetro primario	Parámetros derivados
TOX: Ancho del óxido	$C_{ox} = \frac{\epsilon_0}{TOX}$ (2.43)
NSUB: Dopaje en el sustrato	$V_{TO} = V_{FB} + 2\phi_F + \frac{2\sqrt{q\epsilon_{Si}NSUB}\phi_F}{C_{ox}}$ (2.44)
	$GAMMA = \frac{2\sqrt{q\epsilon_{Si}NSUB}}{C_{ox}}$ (2.45)
	$PHI = 2\frac{KT}{2} \ln \frac{NSUB}{n_i}$ (2.46)
NSS: Densidad de carga efectiva en la superficie del semiconductor	$V_{FB} = \phi_{ms} - \frac{qNSS}{C_{ox}}$ (2.47)
U0: Movilidad de cargas en la superficie	$KP = C_{ox}U0$ (2.48)
UCRIT, UEXP, UTRA: Variación de la movilidad de la superficie	$\mu_s = U0 \left[\frac{UCRIT\epsilon_{Si}}{C_{ox}(V_{GS} - V_{TH} - UTRAv_{DS})} \right]^{UEXP}$ (2.49)
RD, RS: Resistencias de drenador y fuente	

hidrógeno con la superficie del aislante o membrana ion sensitiva. Esta estructura está formada por un conjunto de cargas distribuidas según la relación de Helmholtz, formando una capa difusa tridimensional, la cual se modela como un capacitor y cuyo valor se obtiene de la relación de Gouy-Chapman. En este caso, se hace una aproximación de la relación a un valor constante, como lo presenta la ecuación (2.50):

$$C_{Gouy} = \frac{d}{d\varphi_d} \sqrt{8\epsilon_r\epsilon_0KTC} \sinh\left(\frac{q\varphi_d}{2KT}\right) \approx \frac{\sqrt{8\epsilon_r\epsilon_0KTC}q}{2KT} \quad (2.50)$$

Luego entre la OHP y la IHP se presenta una estructura con distribución de carga en placas paralelas, la cual se modela como una capacitor de valor constante. De acuerdo a la bibliografía, este capacitor tiene un valor típico de 50uF [86]. Finalmente, entre la IHP y la superficie del aislante, se presenta otra estructura de placas paralelas de carga, la cual se modela también como un capacitor contante cuyo valor típico es de 20μF [39]. Puesto que se trata de capacitores de placas paralelas, sus capacitancias por unidad de superficie serían $C_1 = \epsilon_{IHP}d_{IHP}$ y $C_2 = \epsilon_{OHP}d_{OHP}$, respectivamente. Luego, la capacitancia equivalente sería tal como se expresa en la ecuación (2.51):

$$C_{Helm} = \frac{1}{\frac{1}{\epsilon_{IHP}d_{IHP}} + \frac{1}{\epsilon_{OHP}d_{OHP}}} = \frac{\epsilon_{IHP}\epsilon_{OHP}}{\epsilon_{IHP}d_{IHP} + \epsilon_{OHP}d_{OHP}} \quad (2.51)$$

Además, como estas capacitancias se encuentran en serie con la capacitancia de la capa difusa C_{Gouy} , luego el sistema tiene una capacitancia equivalente, mostrada en (2.52):

$$C_{eq} = \frac{C_{Helm}C_{Gouy}}{C_{Helm} + C_{Gouy}} \quad (2.52)$$

Por otro lado, el potencial entre la superficie del aislante y el electrodo de referencia presenta una dependencia respecto al valor del pH. Dada la complejidad de la ecuación y con la finalidad de acondicionarla para su implementación en un modelo de simulación, se hace uso de un recurso auxiliar consistente en fuentes dependientes del valor de pH, cuyos valores son empleados por otra fuente de tensión conectada entre las terminales de interés, correspondientes a la superficie del aislante (compuerta del modelo del MOSFET) y el electrodo de referencia.

Para llegar a estas relaciones, se parte de la ecuación de distribución de carga en la superficie (2.26). Luego de combinarla con la relación de Boltzman y dividirla por la concentración de *bulk* en la solución, se llega a la expresión (2.53):

$$\sigma_0 = qN_{sil} \left(\frac{e^{-\frac{2q\varphi_0}{KT}} - \frac{K_a K_b}{[H^+]^2}}{e^{-\frac{2q\varphi_0}{KT}} + e^{-\frac{q\varphi_0}{KT}} \frac{K_a}{[H^+]} + \frac{K_b K_a}{[H^+]^2}} \right) + qN_{nit} \left(\frac{e^{-\frac{q\varphi_0}{KT}}}{e^{-\frac{q\varphi_0}{KT}} + \frac{K_n}{[H^+]}} \right) \quad (2.53)$$

Posteriormente se pueden definir las relaciones (2.54) y (2.55):

$$EP1 = \ln \left(\frac{K_a}{[H^+]} \right) = \ln K_a - \ln([H^+]) = \ln K_a + \ln(10)\text{pH} \quad (2.54)$$

$$EP2 = \ln \left(\frac{K_a K_b}{[H^+]^2} \right) = \ln K_a K_b - \ln([H^+]^2) = \ln K_a K_b + 2 * \ln(10)\text{pH} \quad (2.55)$$

También se define la constante ET para facilidad de cálculo, mostrada en (2.56):

$$ET = \frac{q}{KT} \quad (2.56)$$

Los valores de EP1 y EP2 corresponden a las tensiones auxiliares empleadas dentro de SPICE. Adicionalmente, por el principio de neutralidad de cargas (2.32), la sumatoria de éstas debe ser igual a cero. Luego, al considerar la concentración de cargas en el IHP y en el semiconductor, inferiores a su contraparte en la capa difusa y la superficie del aislante, estas últimas resultan de la misma magnitud pero con el signo opuesto (debido a que su suma sería cero), llegando a la relación (2.57):

$$\sigma_0 = -\sigma_d = \varphi_0 C_{eq} \quad (2.57)$$

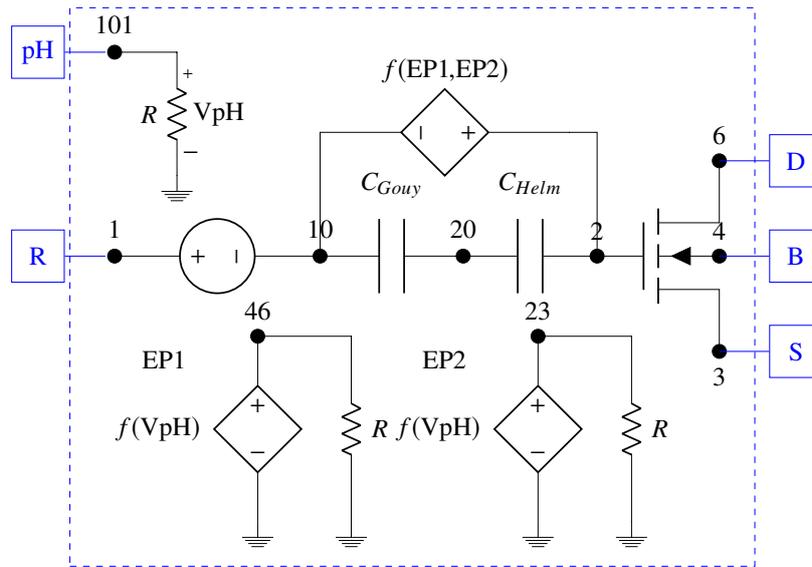


Fig. 2.4: Esquema del macromodelo del ISFET

Despejando la tensión de la anterior ecuación (2.57), se puede obtener una relación de esta tensión respecto a las variables EP1 y EP2 definidas anteriormente, tal como lo muestra (2.58):

$$\varphi_0 = \frac{q}{C_{eq}} N_{sil} \left(\frac{e^{-2\varphi_0 ET} - e^{EP1}}{e^{-2\varphi_0 ET} + e^{-\varphi_0 ET} e^{EP2} + e^{EP1}} \right) + q N_{nit} \left(\frac{e^{-\varphi_0 ET}}{e^{-\varphi_0 ET} + \frac{K_n}{K_a} e^{EP2}} \right) \quad (2.58)$$

Esta última ecuación representa una fuente de tensión dependiente de las fuentes EP1 y EP2, las cuales dependen a su vez del valor de pH.

De acuerdo a las relaciones obtenidas anteriormente, se puede construir el modelo del ISFET para SPICE tal como lo muestra la Fig. 2.4.

2.8.3 Simulación del macromodelo del ISFET

Luego de haber estudiado los fundamentos del modelo, se procede a realizar su comprobación a través de simulaciones. Para llevar a cabo esta tarea, se emplea un software EDA que procesa los modelos escritos en SPICE para obtener las curvas bajo determinados estímulos. El software elegido para ello fue SystemVision de Mentor Graphics.

La primera prueba ejecutada consiste en obtener la familia de curvas i_{DS} versus v_{GS} para variaciones de pH. En este caso se ha empleado el circuito de *testbench* mostrado en la Fig. 2.5. Como se puede apreciar, para obtener estas curvas es necesario mantener el valor de tensión en las terminales de V_{DS} , en este caso, con un valor de 0.5V; mientras se varía el valor de V_{GS} , en un rango de 0-4V. Esta prueba se ejecuta varias veces para diferentes valores de pH, desde 1 hasta 11.

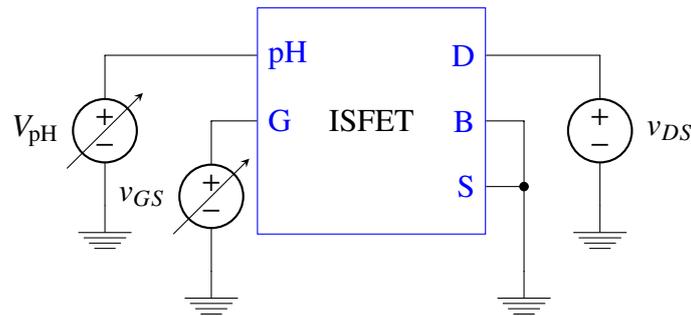


Fig. 2.5: Circuito de *testbench* empleado para la simulación de las curvas i_{DS} vs. v_{GS} con variaciones de pH.

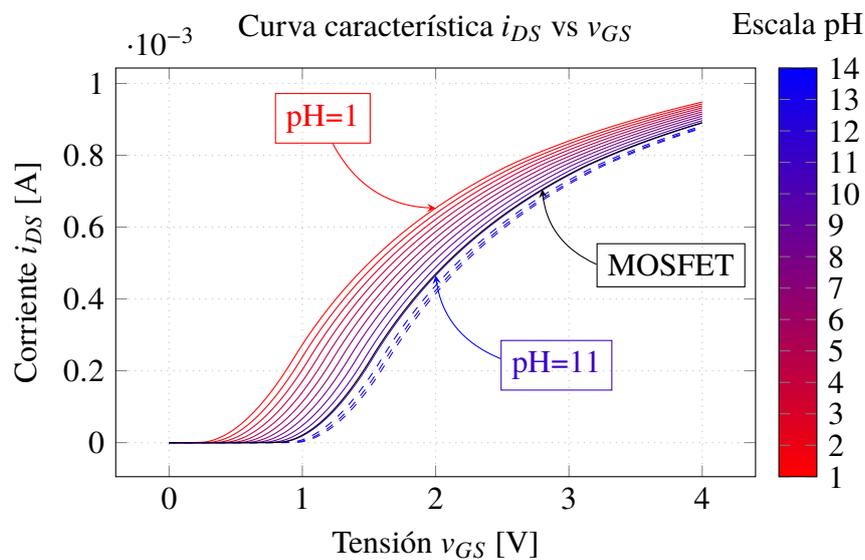


Fig. 2.6: Variación de la curva i_{DS} vs V_{GS} para diferentes valores de pH desde 1 hasta 11, con tensión de polarización $V_{DS} = 500\text{mV}$.

El resultado de la simulación se puede apreciar en Fig. 2.6. Allí se puede verificar que, para valores inferiores de pH (cuando la solución es más ácida), la tensión umbral del dispositivo tiende a disminuir, entrando a la zona de inversión fuerte para valores por debajo de los 0.5V. Conforme va aumentando el valor de pH (la solución se hace más básica), la tensión umbral aumenta hasta llegar a valores cercanos a 1V. Las variaciones de las tensiones umbrales para cada uno de los valores de pH presenta una tendencia lineal, lo cual es favorable al momento de realizar el sensado de pH mediante este dispositivo, puesto que al obtener el valor de tensión umbral, se podría obtener el valor de pH a partir de él.

Si bien es cierto, la escala de pH tiene un rango de 1-14, en este caso, el modelo del sensor es válido hasta un pH 11, puesto que para valores superiores se presenta una variación irregular de la tensión umbral respecto al pH. Esto se puede apreciar también

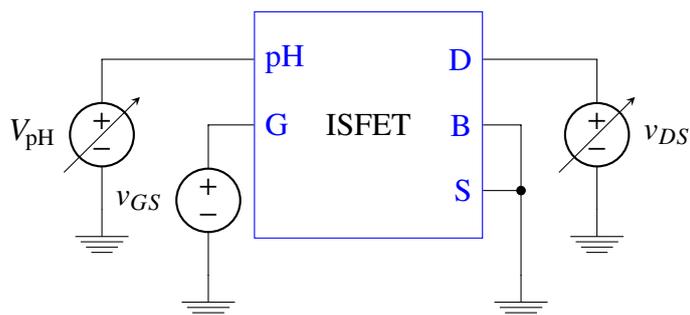


Fig. 2.7: Circuito de *testbench* empleado para la simulación de las curvas i_{DS} vs. v_{DS} con variaciones de pH

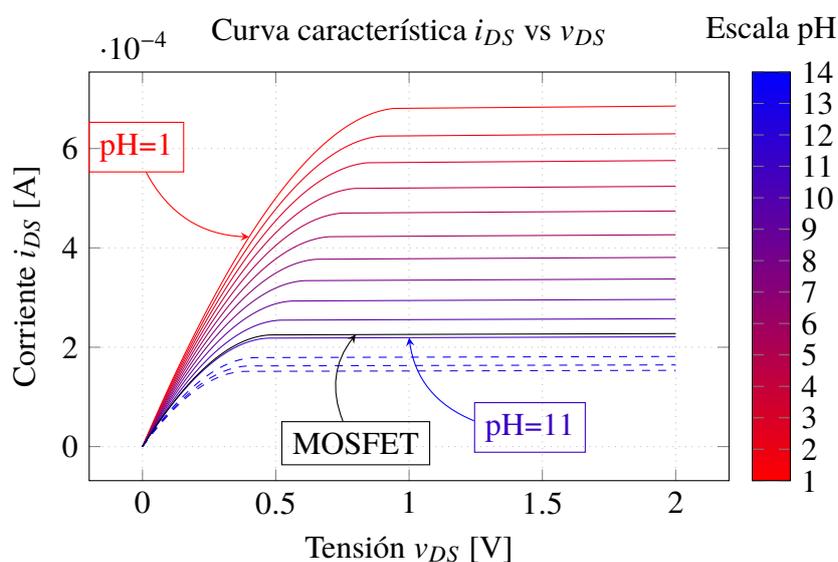


Fig. 2.8: Variación de la curva i_{DS} vs V_{DS} para diferentes valores de pH desde 1 hasta 11, con tensión de polarización $V_{GS} = 1.5V$.

en la gráfica, al observar el comportamiento de la corriente para valores de pH mayores, representadas mediante líneas discontinuas, las cuales aparecen más alejadas del resto de curvas para valores de pH inferiores. Adicionalmente, se puede identificar que este cambio de comportamiento tiene que ver con la tensión umbral del MOSFET que forma parte del modelo, el cual aparece representada mediante una curva negra, marcando el límite hasta donde el sensor presenta sensibilidad respecto al pH.

La segunda prueba consiste en obtener la familia de curvas de i_{DS} versus v_{DS} , para la cual se emplea el circuito de *testbench* mostrado en la Fig. 2.7. En este caso, la tensión que se mantiene constante es la tensión V_{GS} , siendo la elegida igual a 1.5V, mientras se ejecuta un barrido de tensión de V_{DS} , entre 0-2V. Esta prueba se efectúa también varias veces, obteniendo una curva por cada valor de pH.

El resultado de la prueba se puede apreciar en Fig. 2.8. En este caso, se verifica que si la solución es ácida, la corriente de saturación es más alta respecto a cuando la solución es básica. Así mismo, se puede verificar que el rango de variación es relativamente amplio desde los $200\mu\text{A}$ hasta los $700\mu\text{A}$. Al igual que en la curva anterior, el modelo presenta un adecuado comportamiento hasta el valor de pH 11, a partir del cual la variación de la corriente de saturación es irregular. El límite está igualmente marcado con la curva del componente MOSFET del modelo, representado también con una curva de color negro.

2.9 Caracterización del sensor

El proceso de caracterización del sensor consiste en realizar mediciones del dispositivo bajo ciertas condiciones estándares y obtener de dichas mediciones sus curvas características. Una vez obtenidas estas curvas, se pueden extraer los parámetros del sensor que posteriormente servirán para ingresarlo al modelo y simular el dispositivo real.

Las curvas características que se mencionan son las mismas empleadas en el caso del MOSFET, los cuales corresponden a la familia de curvas i_{DS} vs. v_{GS} y i_{DS} vs. v_{DS} . Cada una de estas curvas tiene sus propias propiedades y tienen una determinada utilidad al momento de realizar el proceso de extracción de parámetros.

En el caso de las curvas i_{DS} vs. v_{GS} , éstas se obtienen a partir de realizar un barrido de tensión en v_{GS} mientras se mantiene constante el valor de V_{DS} , efectuando las mediciones de tensión y de corriente, para después repetir el mismo proceso con un valor diferente de V_{DS} . Cada una de las curvas obtenidas pertenece a la familia de i_{DS} vs. v_{GS} y generalmente son empleadas para la extracción de la tensión umbral del dispositivo. Tomando como referencia la ecuación de la corriente del ISFET (la cual es la misma que del MOSFET (2.13) solo que con la tensión umbral dependiente del pH), se puede obtener la ecuación de las curvas de esta familia, mostrada en (2.59):

$$i_D = \frac{W}{L} \mu C_{ox} V_{DS} [v_{GS}] - \underbrace{\left[V_{TH} + \frac{V_{DS}}{2} \right]}_{\text{constante}} V_{DS} \quad (2.59)$$

De esta ecuación, se puede apreciar que se trata de líneas rectas, con una pendiente equivalente al factor de transconductancia multiplicado por la tensión V_{DS} .

En el caso de la familia de curvas i_{DS} vs. v_{DS} , se sigue la misma metodología que en el caso anterior, con la diferencia que ahora el barrido de tensión se aplica para la tensión v_{DS} , mientras se mantiene constante V_{GS} . Luego, se obtiene una curva diferente cada vez que se

varia el valor de tensión V_{GS} . Esta curva es importante en cuanto proporciona información acerca de las corrientes y tensiones de saturación. Adicionalmente, se pueden emplear para la extracción de la movilidad efectiva, así como el factor de modulación la longitud de canal en la región de saturación. Para obtener la ecuación de estas curvas, se parte nuevamente de la ecuación de la corriente del ISFET (2.13), solo que en este caso se distinguen las dos zonas de operación del dispositivo, correspondiente a la zona óhmica y la zona de saturación.

Para el caso de la zona óhmica, la ecuación sería la presentada en (2.60):

$$i_{DS} = K [V_{GS} - V_{TH}] v_{DS} - \frac{v_{DS}^2}{2} \quad (2.60)$$

En este caso, se puede apreciar que se trata de una ecuación cuadrática, la cual tiene concavidad hacia abajo, además de tener intersección en el origen debido a que la corriente a una tensión de $v_{DS} = 0$ sería igual a cero.

Por otra parte para la zona de saturación la ecuación sería la mostrada en (2.61):

$$i_{DS} = \frac{K}{2} [V_{GS} - V_{TH}] \quad (2.61)$$

La cual sería una constante, no obstante, también presenta un factor de modulación λ , el cual origina que la corriente tienda a incrementar a un factor constante, conforme aumenta la tensión entre drenador y fuente.

2.10 Extracción de parámetros

El objetivo de realizar el modelamiento y la simulación de un dispositivo o sistema, es efectuar predicciones de su comportamiento en determinados escenarios. No obstante, la precisión de estas predicciones están estrechamente vinculadas a los parámetros que se empleen en los modelos que los representan [87]. El modelo presentado del ISFET, tratándose de un modelo general, requiere de la modificación de ciertos parámetros de ajuste, los cuales son propios del dispositivo elegido para su implementación.

Es por ello que la extracción de parámetros constituye un paso importante en el proceso de modelamiento y simulación del dispositivo, debido a que permite obtener los mencionados parámetros característicos que describen su comportamiento.

Dado que el ISFET tiene similitudes respecto al MOSFET, se puede aplicar las mismas técnicas para la extracción de sus parámetros. Sin embargo, es necesario realizar algunos ajustes, dado que en su modelo existen contribuciones propias de su característica ion-sensitiva, las cuales difieren de un MOSFET convencional.

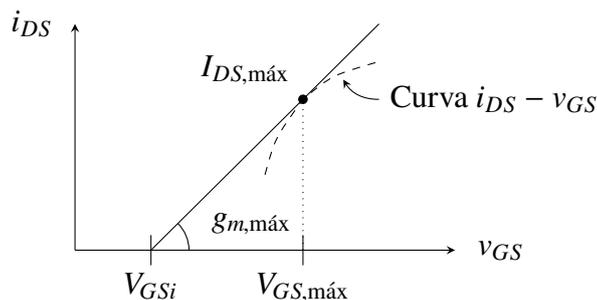


Fig. 2.9: Representación del método ELR

Respecto al MOSFET, existen diversos métodos para la extracción de parámetros, los cuales se obtienen desde los datos de las curvas corriente-tensión o capacitancia-tensión. Estos métodos se pueden clasificar en dos tipos: métodos de regresión lineal (analíticos) y los de optimización no lineal (numéricos). Así mismo, los métodos difieren de acuerdo al tipo de parámetro que se desee obtener. En el presente caso, los parámetros de interés son la tensión umbral y la movilidad efectiva, cuyos métodos de extracción se describen a continuación.

2.10.1 Extracción de la tensión umbral

Uno de los parámetros más importantes y, a la vez, uno de los primeros en el proceso de caracterización del dispositivo, es la tensión umbral. Tal como lo menciona Galup en su libro sobre modelamiento del MOSFET [88], existen al menos una docena de métodos empleados para la extracción de la tensión umbral. Una muestra de ello lo presenta Ortiz-Conde en [89]. Entre los principales se encuentran los presentados en la TABLA N° 2.10.

De los métodos presentados, se ha encontrado en la literatura científica, la aplicación de dos de ellos para la extracción de parámetros del ISFET: uno basado en el método TCR [92] y otro basado en el método ELR [93]. En el presente trabajo se empleará como referencia este último, cuya representación gráfica se muestra en la Fig. 2.9 y, según la cual, se siguen los siguientes pasos:

- Obtener la curva de la transconductancia cruzada g_m de la corriente de drenador i_{DS} respecto a la tensión entre compuerta y fuente v_{DS} . Esta curva corresponde a la derivada de la relación de estas dos magnitudes. Su ecuación se muestra en (2.62).

$$g_m = \frac{di_{DS}}{dv_{GS}} \quad (2.62)$$

- Obtener el punto máximo de la curva de transconductancia e identificar el valor de la tensión de $v_{GS,máx}$ para el cual se presenta este máximo.

TABLA N° 2.10: Métodos para la extracción de la tensión umbral V_{TH}

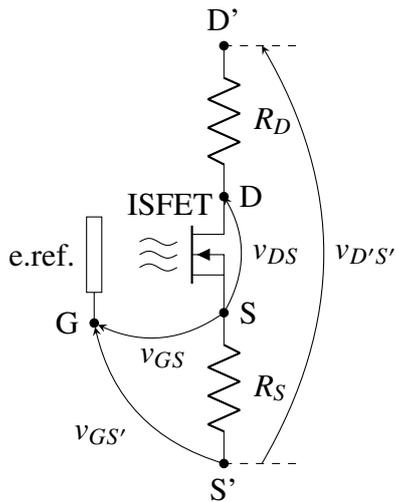
Método	Características
<i>Método de corriente constante (CC) [90]</i>	<ul style="list-style-type: none"> ■ Método mas empleado en la industria debido a su simplicidad. ■ Consiste en la excitación del MOSFET a través de una corriente de drenador constante I_T, la cual provoca la aparición de la tensión umbral en la compuerta, medida a través de un opamp en realimentación. ■ Para evitar una dependencia de su geometría, se suele emplear corrientes bajas del orden de 10-50nA.
<i>Extrapolación en la región lineal (ELR) [90]</i>	<ul style="list-style-type: none"> ■ Consiste en aproximar la región lineal en inversión fuerte de la curva I_{DS} versus V_{GS} a una recta cuyo punto de intersección con el eje V_{GS} corresponde a la tensión umbral con un desplazamiento de $V_{DS}/2$. ■ Usualmente emplea tensiones constantes de V_{DS} entre 50 a 100mV. ■ Se suele emplear como punto de intersección para la recta de extrapolación al punto de la curva I_{DS}-V_{GS} correspondiente a la tensión de compuerta-fuente donde se produce la máxima transconductancia. ■ La desventaja de este método es que solo es aplicable en los casos cuando las resistencias en drenador y fuente son despreciables.
<i>Extracción lineal de la transconductancia (GMLE) [90]</i>	<ul style="list-style-type: none"> ■ Consiste en la extrapolación lineal de la curva g_m versus V_{GS}, en el punto de máxima derivada. ■ Se analiza la región de la curva en inversión débil y moderada, la cual depende linealmente de la tensión V_{GS}
<i>Método de la segunda derivada o cambio en la transconductancia (SD o TC) [89]</i>	<ul style="list-style-type: none"> ■ Desarrollado para evitar la dependencia de las resistencias en serie en el drenador y la fuente. ■ Determina el valor de la tensión umbral para cuando la derivada de la transconductancia (d^2I_D/dV_{GS}^2) tiene su valor máximo.
<i>Variación de transconductancia respecto a la corriente (TCR) [91]</i>	<ul style="list-style-type: none"> ■ Basado en la curva de la variación de la transconductancia respecto a la corriente de drenador obtenido con valores bajos de V_{DS}. ■ El análisis se efectúa en las regiones de inversión débil y moderada. ■ Requiere la obtención de la curva g_m/I_D respecto a la tensión V_{GS}, a fin de ubicar su valor máximo y, finalmente, la tensión para la cual se produce el 0.5301 de este máximo, correspondiente a la tensión umbral.

- Ubicar el punto en la curva de la corriente para la tensión hallada y trazar la tangente. La pendiente de la recta corresponde al valor de máxima transconductancia $g_{m,máx}$.

- Ubicar el cruce por cero v_{GSi} en la recta de las tensiones v_{GS} . Luego, restar el valor de la tensión $v_{DS}/2$. El valor resultante corresponde a la tensión umbral, tal como lo expresa la ecuación (2.63):

$$V_{TH} = v_{GSi} - \frac{v_{DS}}{2} \quad (2.63)$$

El método ELR presenta la desventaja de ser dependiente de las resistencias en drenador y fuente. Esta dependencia influye en las tensiones aplicadas al dispositivo tal como lo muestra la Fig. 2.10. El resultado de su influencia se refleja en la ecuación de la tensión umbral, como



$$v_{D'S'} = v_{DS} + i_{DS}(R_S + R_D) \quad (2.65)$$

$$v_{GS'} = v_{GS} + i_{DS}(R_S) \quad (2.66)$$

Fig. 2.10: Efecto de las resistencias en los contactos del ISFET

se muestra en (2.64):

$$V_{TH} \approx V_{GSi} - \frac{V_{DS}}{2} + K(R_D + R_S)(V_{GS,m\acute{a}x} - V_{GSi})^2 - 2K^2(R_D + R_S)^2(V_{GS,m\acute{a}x} - V_{GSi})^3 \quad (2.64)$$

Adicionalmente, un enfoque mas general de la influencia de la tension V_{DS} proporcionado por Tsvividis [94], considera un factor α afectando este termino, el cual puede ser diferente de la unidad. Puesto que las resistencias seran constantes para un determinado dispositivo, ası como el punto de maxima transconductancia, la relacion en (2.64) se puede simplificar a la expresion mostrada en (2.67):

$$V_{TH} \approx V_{GSi} - \alpha \frac{V_{DS}}{2} + C \quad (2.67)$$

Finalmente, la ecuacion (2.67) expresa la tension umbral en funcion del valor de V_{DS} . Es importante sealar que V_{TH} es el valor de tension umbral para un valor de V_{DS} en particular, no obstante, experimentalmente se puede identificar que este valor varıa cuando cambia la tension en drenador y fuente de polarizacion. Aun mas, la tension que se ingresa en el modelo no es exactamente V_{TH} sino V_{TO} . Es por ello que para aproximar el valor de V_{TO} , se puede realizar una extrapolacion para varios valores de V_{DS} , de tal forma que cuando la recta de regresion cruce por cero $V_{DS} = 0$, entonces $V_{TH} = V_{TO}$.

Por otro lado, el V_{TH} y V_{TO} corresponden al valor del ISFET como todo un conjunto, mas dentro del modelo se ingresa el valor de V_{TO} correspondiente unicamente a la contribucion del MOSFET, por lo cual es necesario restar del valor de V_{TO} calculado previamente, el efecto de la capa sensitiva. Esto se puede apreciar en la Fig. 2.11. Puesto que se puede considerar una

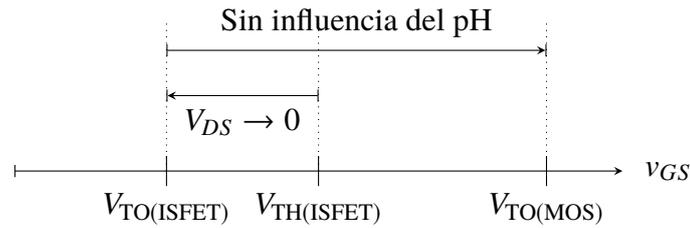


Fig. 2.11: Esquema de tensiones

influencia constante para un valor de pH definido, entonces es posible obtener esta tensión a partir de una simulación del modelo aun no ajustado, con el valor de pH empleado para obtener las muestras. Luego, este valor debe modificar a la tensión V_{TO} , como lo muestra (2.68):

$$V_{TO(MOS)} = V_{TO(ISFET)} - \underbrace{\left(E_{ref} + \varphi_{lj} + \chi_0 - \varphi_0 - \frac{\varphi_m}{q} \right)}_{\text{Obtenido de simulación}} \quad (2.68)$$

Para hallar el factor de transconductancia K , se realiza una operación matemática a las expresiones de i_{DS} y g_m , de tal forma que no dependa de las resistencias de drenador y fuente, tal como se muestra en (2.69):

$$\frac{i_{DS}^2}{g_m} = K v_{D'S'} \left(v_{GS'} - V_{TH} - \frac{v_{D'S'}}{2} \right)^2 \quad (2.69)$$

De esta manera, evaluando en el punto de máxima transconductancia, se obtendría la relación para K , presentada por la ecuación (2.70).

$$K = \frac{\frac{I_{DS,máx}^2}{g_{m,máx}}}{V_{DS} (V_{GS,máx} - V_{GSi})^2} \quad (2.70)$$

Finalmente, para la obtención de las resistencias de drenador y fuente, se puede realizar la evaluación en las ecuaciones de la corriente o transconductancia cruzada para el punto de máxima transconductancia.

2.10.2 Extracción de la movilidad efectiva

La movilidad efectiva, la cual forma parte del parámetro de transconductancia (K) en la ecuación de la corriente de drenador del ISFET, sufre una atenuación conforme la tensión entre drenador y compuerta aumenta. Esta atenuación, de acuerdo al modelo de nivel 2 del MOSFET, está expresada a través de la ecuación (2.71):

$$\mu_{eff} = U_0 \left[\frac{UCRIT \epsilon_{Si}}{C_{ox} (v_{GS} - V_{TH} - UTRAV_{DS})} \right]^{U_{EXP}} \quad (2.71)$$

TABLA N° 2.11: Métodos de extracción de u_{eff}

Método	Ecuaciones
<i>Transconductancia directa.</i> [95] Emplea las curvas I_{DS} vs. V_{DS} , a partir de las cuales se obtiene la transconductancia directa g_d para varios valores de V_{GS} . Luego, con estos valores se calcula la movilidad efectiva.	$g_d = \frac{di_{DS}}{dv_{DS}} = K(v_{GS} - V_{TH} - v_{DS}) \quad (2.74)$
	$\mu_{eff} = \frac{L}{WC_{ox}} \frac{g_d}{(v_{GS} - V_{TH} - v_{DS})} \quad (2.75)$
<i>Corriente de drenador.</i> [96] Emplea el valor de la corriente directamente en la zona de inversión fuerte, dividiéndola por el término dependiente de la tensión.	$\mu_{eff} = \frac{L}{W} \frac{i_{DS}(v_{GS})}{V_{DS}C_{ox}(v_{GS} - V_{TH})} \quad (2.76)$
<i>Transconductancia cruzada.</i> [97] Emplea la curva I_{DS} vs. V_{GS} igual que el método ELR.	$\mu_{eff} = \frac{L}{WC_{ox}} \frac{g_m}{v_{DS}} \quad (2.77)$
<i>Corriente cuadrática.</i> Evita influencia de las resistencias de drenador y fuente.	$\mu_{eff} = \frac{L}{WC_{ox}} \frac{i_{DS}^2/g_m}{V_{DS}(V_{GS} - V_{GSi})^2} \quad (2.78)$

Uno de los métodos que se pueden emplear para obtener los parámetros de esta ecuación consiste en aplicar la función logaritmo a ambos lados de la ecuación, como lo sugiere Vladimirescu [84], resultando la expresión mostrada en (2.72):

$$\log\left(\frac{u_{eff}}{U_0}\right) = UEXP \left[\log\left(\frac{UCRIT\epsilon_{Si}}{C_{ox}}\right) - \log(v_{GS} - V_{TH} - UTRAV_{DS}) \right] \quad (2.72)$$

Luego, se puede graficar el valor de $\log(\mu_{eff}/U_0)$ en el eje x respecto a $\log(v_{GS} - V_{TH} - UTRAV_{DS})$ en el eje y, con lo cual se obtiene el valor de UEXP y UCRIT. Posteriormente, para hallar el valor de U_0 se debe encontrar el valor de tensión v_{GS} que hace que el término exponencial sea igual a 1. Esto se logra cuando se establece la relación mostrada por (2.73):

$$v_{GS} = V_{TH} + UTRAV_{DS} + \frac{UCRIT\epsilon_{Si}}{C_{ox}} \quad (2.73)$$

donde el valor de la movilidad efectiva para este valor de V_{GS} es igual a U_0 .

De acuerdo a lo mostrado anteriormente, la obtención de la movilidad efectiva es relativamente sencilla a partir de la curva μ_{eff} respecto a v_{GS} . Luego, el problema se reduce a encontrar esta curva característica a partir de las mediciones. Para poder encontrar esta curva se han identificado cuatro métodos, los cuales se resumen en la TABLA N° 2.11.

Del conjunto de métodos presentados, se podrían emplear los tres primeros al tener la misma tendencia de la curva original, siendo el tercero, que emplea la transconductancia cruzada g_m , el más directo, puesto que no requiere de pasos adicionales, así como de otros valores

calculados previamente. Respecto a su variación en la relación a la curva teórica, la mayor similitud es en los valores de tensión cercanos a los puntos de máxima transconductancia.

2.11 Resumen del capítulo

El indicador de pH es una medida adimensional de la concentración de iones de hidronio en una solución, resultado de una reacción química producida en su proceso de formación. Para la obtención de la información de pH se emplean usualmente sensores, siendo los más utilizados los de naturaleza electroquímica, tales como el sensor de bulbo de vidrio y el ISFET. Este último es empleado para la tesis, y basa su funcionamiento en un MOSFET, al cual se ha adicionado una interfaz ion selectiva. Esta interfaz constituye el elemento más importante del ISFET y es debido a la cual presenta una sensibilidad respecto al pH. Este dispositivo tiene varias aplicaciones encontrándose también en el ámbito de monitoreo de calidad de agua. Dentro de los principales desafíos destacan su variación de corriente y sensibilidad ante la temperatura, mitigados mediante técnicas de acondicionamiento. Existen diferentes sensores comerciales basados en ISFET, siendo el fabricado por WINSENSE, el que posee el menor consumo de energía. En el diseño de bajo consumo se identifican niveles, siendo el más recurrente el nivel de sistema, para lo cual se requiere una buena selección de microcontroladores. De los existentes, se centra la atención en los de señal mixta, puesto que se requiere implementar también el circuito de acondicionamiento, destacando la familia PSoC de Cypress. La solución presentada forma parte de una red de sensores que constituye una tecnología dentro del ecosistema de Internet de las Cosas (IoT).

Para el modelamiento del ISFET, se emplea el macromodelo del dispositivo, modelando la interfaz como capacitores y fuentes de tensión dependientes. El resultado muestra las curvas características del sensor así como los rangos de operabilidad. En el diseño es preciso ajustar el modelo, por lo que se requiere un proceso de caracterización y extracción de parámetros, de donde se obtiene la tensión umbral y la movilidad efectiva, empleando técnicas similares al caso del MOSFET.

En el presente capítulo se ha abordado el concepto de pH, el sensor ISFET, microcontroladores de bajo consumo, redes de sensores inalámbricos, los cuales constituyen las tecnologías empleadas para el diseño del sistema de sensado de pH que se desarrollará en el siguiente capítulo. Así mismo, se ha efectuado el modelamiento y caracterización del sensor, empleado también en el diseño, específicamente del sistema de acondicionamiento de señal.

CAPÍTULO 3

DISEÑO DEL SISTEMA DE SENSADO DE PH

En el presente capítulo se aborda el diseño del sistema de sensado de pH, considerando cada una de sus partes, las cuales comprenden el diseño del sistema de caracterización para el sensor de pH, el sistema de adaptación de señal, el sistema de procesamiento y de control, el sistema de comunicaciones inalámbricas y el sistema complementario de energía.

3.1 Diseño del sistema de medición para caracterización del ISFET

Para evaluar el funcionamiento del sensor de pH es necesario realizar un proceso de caracterización, a partir del cual se obtiene los parámetros del dispositivo real que ingresan a su modelo y ajustan su comportamiento.

Por lo general, para realizar este procedimiento, se emplean equipos especializados que comprenden fuentes de medición (SMU) [98], así como analizadores paramétricos [99]. Estos equipos cuentan con características para una medición muy precisa, como en el caso de los SMUs, los cuales combinan las propiedades de una fuente de precisión y un multímetro digital de alta resolución, con una gran velocidad, operación en los cuatro cuadrantes de tensión y corriente, así como el barrido de tensión [100]. No obstante, estos dispositivos no siempre son accesibles debido a su alto costo, siendo inviable su adquisición para la caracterización de unos pocos sensores dentro de un proyecto modesto.

Una solución más accesible es propuesta por Ochoa *et al.* [101], quien hace uso de un microcontrolador y de algunos elementos adicionales para obtener las curvas del sensor. Entonces, se podría abordar el problema a partir de la construcción de un sistema de medición, el cual disponga de todas las herramientas necesarias para la caracterización del ISFET, además de ofrecer la posibilidad de mostrar casi en tiempo real esa información en la computadora, con la finalidad de monitorear las señales adquiridas y, a su vez, almacenar dicha información para un procesamiento posterior.

Es importante también considerar el empleo de herramientas de código abierto, con la

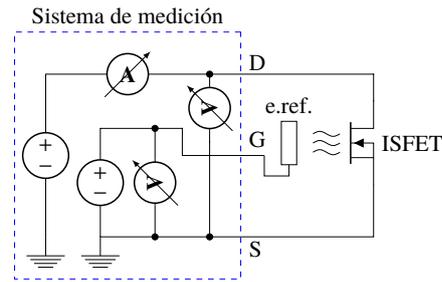


Fig. 3.1: Sistema de caracterización conceptual

finalidad de no restringir el uso de la solución propuesta por motivos de licencia. En este caso, no se utilizaría Matlab o LabVIEW, como lo han hecho en otras investigaciones [102], [103], sino que la solución se orientaría a una implementación directamente en un lenguaje de programación, tal como C++, Python o Java.

3.1.1 Enfoque conceptual y alternativas de diseño para el sistema de medición

Un sistema de medición implica una generación de tensiones para la polarización del sensor, así como la obtención de los parámetros eléctricos resultado de la estimulación con los primeros. Por tanto, para obtener las curvas características del dispositivo, se requiere generar una variación de dichas tensiones de polarización v_{GS} y v_{DS} , efectuando, a su vez, las mediciones de ambas tensiones, así como de la corriente que circula entre el drenador y fuente del dispositivo i_{DS} . El esquema conceptual del sistema de medición para lograr este objetivo se puede apreciar en la Fig. 3.1.

A fin de generar estas variaciones de tensión, así como para realizar las mediciones de los diferentes valores de corriente y tensión, se pueden plantear algunas alternativas para el diseño del sistema de medición.

Un primer enfoque para obtener el comportamiento deseado es a través del empleo de un osciloscopio y un multímetro con la función de amperímetro, con un esquema de conexión, tal como se muestra en la Fig. 3.2a. El empleo de equipos de medición brinda la ventaja de evitar la incertidumbre debido a errores de medición producidos por el instrumento, puesto que éstos cuentan con todos los estándares necesarios para ofrecer medidas confiables. No obstante, para fines de extracción de información, se requiere que el usuario realice el ingreso manual de los datos a la computadora introduciendo un error debido a la observación. Así mismo, el utilizar directamente las fuentes de tensión sin limitación de corriente (a los niveles exigidos por el ISFET, el cual se encuentra en el orden de los microamperios), no favorece a la protección del dispositivo.

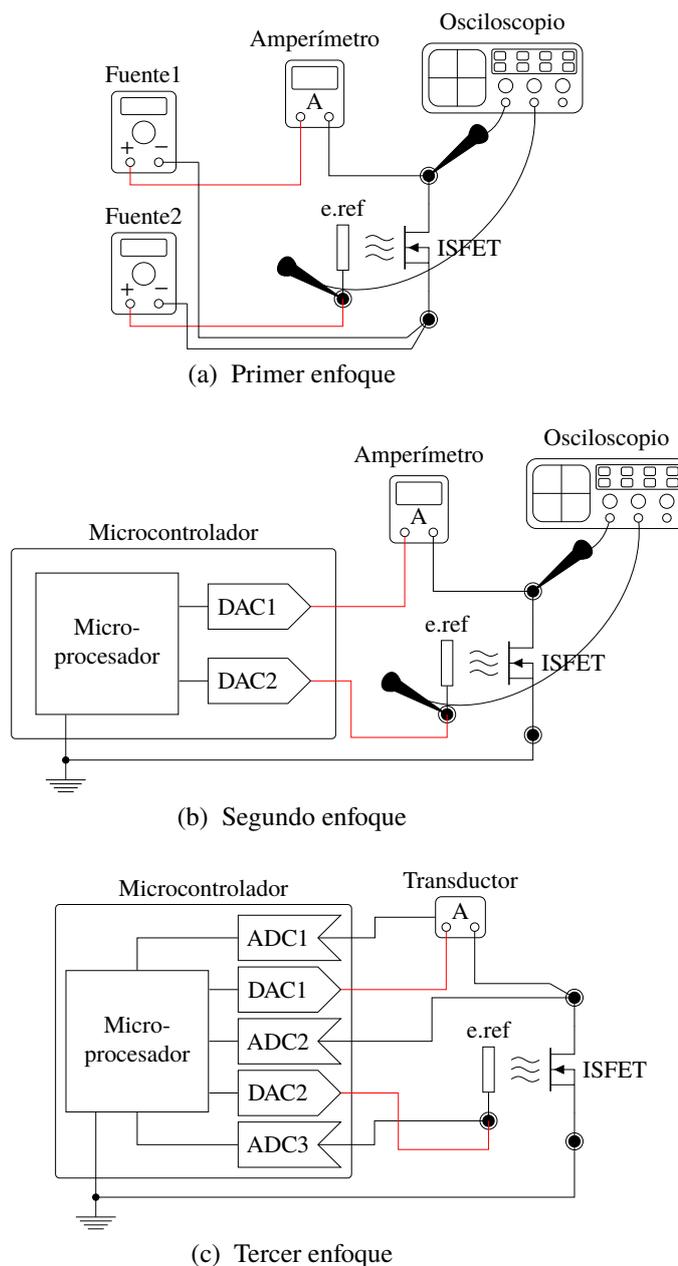


Fig. 3.2: Enfoques de diseño para el sistema de medición

Un segundo enfoque consiste en emplear un microcontrolador para generar estas variaciones de tensión. Además, dado el bajo nivel de corriente proporcionado por éste, ya no haría falta la inclusión de circuitos de protección adicionales. El esquema de esta solución se puede visualizar en Fig. 3.2b. En este caso, ya se tiene cubierta la parte correspondiente a la generación casi automática de las tensiones, no obstante, aún mantiene el empleo de los instrumentos de medición para la obtención de las variables de tensión y corriente.

La tercera alternativa consiste en un control de los estímulos basado en microcontroladores empleado también en la obtención de la respuesta del dispositivo. Esta información

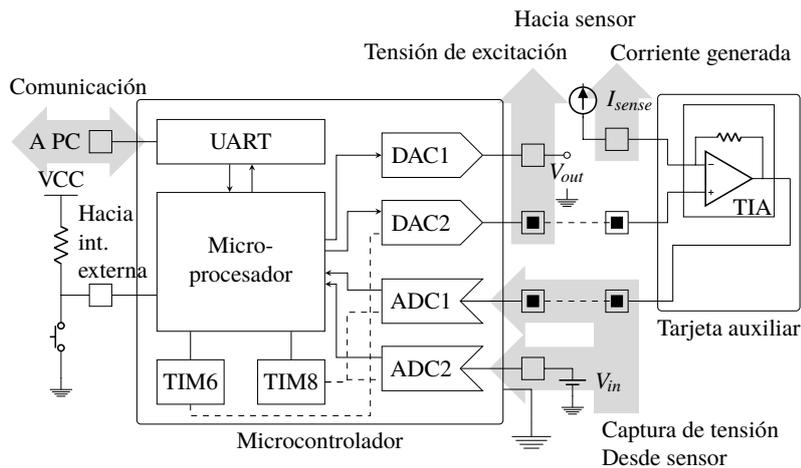


Fig. 3.3: Diagrama del hardware empleado en el sistema.

posteriormente puede ser mostrada en alguna aplicación que permita almacenarla para su futuro uso. El esquema conceptual del sistema se puede apreciar en Fig. 3.2c. En este caso, se tiene un pleno control y acceso a la información obtenida del estímulo del sensor. No obstante, el sistema no cuenta con un elemento de procesamiento tal como lo posee un equipo de medición convencional, por lo que los datos recopilados requieren de una etapa de depuración, a fin de obtener la información final producto de la caracterización.

El sistema planteado en el presente trabajo está basado en este último enfoque, donde tanto la generación como adquisición de las señales se realiza mediante un microcontrolador, el cual se conecta con la computadora para visualizar y almacenar los datos obtenidos. El sistema cuenta con tres componentes principales: el hardware, el software para el microcontrolador y el software para la PC, este último con dos componentes, uno para la adquisición y otro para el procesamiento.

3.1.2 Diseño del hardware

El sistema de caracterización debe contar con elementos con la capacidad de proporcionar niveles de tensión estables a la vez que puedan adquirir las corrientes y tensiones producidas en el dispositivo. Adicionalmente, estos datos deben transmitirse a una computadora desde donde se recupere la información de estas magnitudes.

Puesto que la solución está pensada para ser implementada en microcontroladores, se debe emplear los elementos disponibles en éstos para lograr este objetivo. En la mayoría de dispositivos de este tipo se cuenta con convertidores digital-analógicos (DACs) y convertidores analógico-digitales (ADCs), los cuales se pueden utilizar para generar las señales de tensión, a la vez de capturar dicha información para posteriormente procesarla en el microcontro-

lador. Adicionalmente, con la finalidad de controlar cuándo se producen estos cambios, es necesario incluir un dispositivo de disparo (*trigger*) que accione los periféricos bajo ciertas circunstancias. Para esto se pueden emplear los temporizadores (*timers*) del microprocesador.

En cuanto a la comunicación, existen múltiples periféricos que pueden servir para este propósito, no obstante, el que ofrece mayor flexibilidad es el módulo transmisor-receptor asíncrono universal (UART). Este dispositivo se encarga de enviar la información desde el microcontrolador a la PC.

Adicionalmente, puesto que también se contempla un control manual para el dispositivo, se debe habilitar un bloque de entrada/salida, accesible a través de un pulsador o interruptor externo.

Por otra parte, para la medición de la corriente se emplea un amplificador de transimpedancia (TIA), el cual recibe la corriente desde el sensor y, a través de una resistencia en realimentación, la convierte en una señal de tensión.

Finalmente, la solución se diseña tal como se muestra en la Fig. 3.3. Allí se puede apreciar el empleo de un microcontrolador y una tarjeta auxiliar para implementar el sistema de medición. El microcontrolador cuenta con los ADCs, DACs, temporizadores (TIM6 y TIM8), el módulo UART y la entrada para el pulsador externo, mientras que en la tarjeta auxiliar se sitúa el TIA para la medición de corriente.

3.1.3 Diseño del software para el microcontrolador

Una vez determinados los elementos que forman parte de la solución, se debe especificar su funcionamiento. Esto se logra a través del programa ejecutado en el microcontrolador, el cual se describe a continuación.

Tan pronto como el microcontrolador se enciende, todos los componentes requieren ser inicializados. El proceso comienza con la activación del reloj principal del sistema, luego la rutina de inicialización del microprocesador para finalizar con la activación de los periféricos.

Una vez inicializados los ADCs así como el temporizador que los controla, éstos empiezan a operar, a la par que comienza el proceso de adquisición de información. Posteriormente, se produce un ciclo infinito que controla el comportamiento del microcontrolador, así como los modos de comunicación mediante el UART. Este ciclo se puede apreciar en la Fig. 3.4. En el proceso se emplean 4 banderas para llevar a cabo dicha tarea: *ubKeyPressed*, el cual indica cuando se ha producido una interrupción externa; *manual*, empleado para almacenar el modo de operación; *UARTReady*, indicador de la finalización de recepción de una cadena

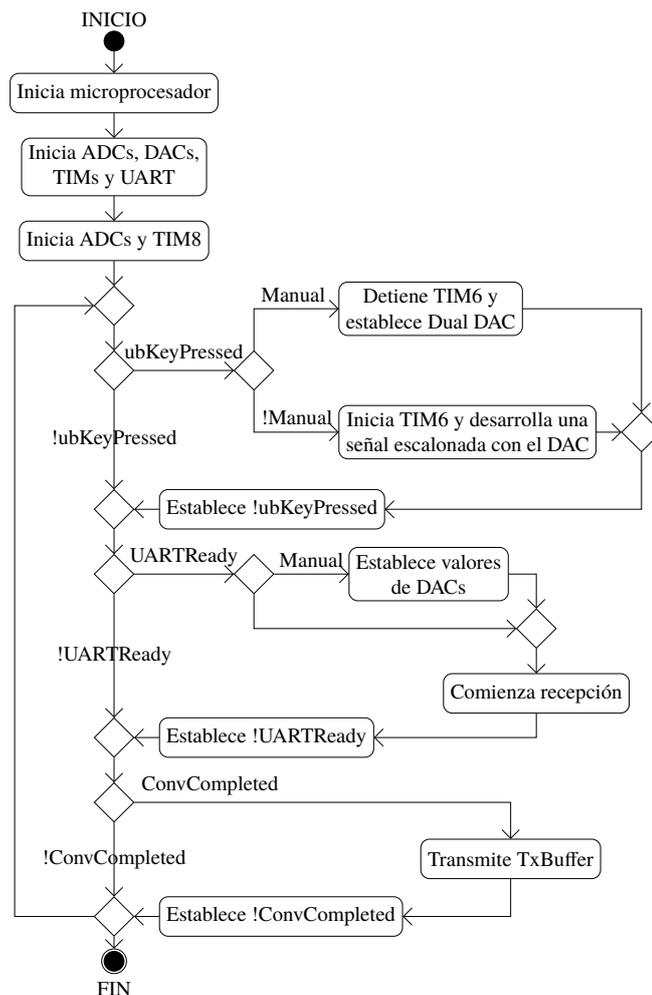


Fig. 3.4: Diagrama general del software para el microcontrolador.

desde el UART y *ConvCompleted*, que indica el término de la conversión del ADC.

Las banderas mencionadas anteriormente están desactivadas por defecto y se activan durante los procesos de interrupción. Para el presente caso, se han considerado tres interrupciones, las cuales se describen en la TABLA N° 3.1.

Otro importante punto a resaltar es la trama enviada por el UART. En este caso, se encuentra en formato ASCII, de tal forma que sea fácil de depurar en cualquier terminal serial. Por el otro lado, puesto que el UART demanda un tiempo prolongado de operación del microcontrolador, es conveniente enviar más datos en cada mensaje en lugar de unos pocos bits, dado que, de lo contrario, implicaría una pérdida de información.

La estructura del dato enviado tiene la forma presentada en (3.1):

$$\underbrace{\overbrace{\text{ADC1, ADC2} : \text{ADC1, ADC2} : \dots : \text{ADC1, ADC2} \# \text{DAC} \# \text{TIM} \# \text{Ctrl} \backslash n}^{\text{muestra}}}_{10 \text{ muestras}} \quad (3.1)$$

TABLA N° 3.1: Interrupciones del software para el microcontrolador.

Interrupción	Descripción
<i>Interrupción externa</i>	Permite al usuario cambiar los modos de operación entre manual y automático.
<i>Conversión del ADC completada</i>	Se emplea para obtener la información de los ADCs y almacenarlos en el buffer de transmisión. Se activa cuando el ADC termina una conversión satisfactoria. Su diagrama se muestra en Fig. 3.5.
<i>Recepción del UART completada.</i>	Basado en el valor recibido, determina cual será la operación a ejecutar. Se produce cuando se termina de recibir satisfactoriamente una trama a través del UART. Su diagrama se muestra en Fig. 3.6.

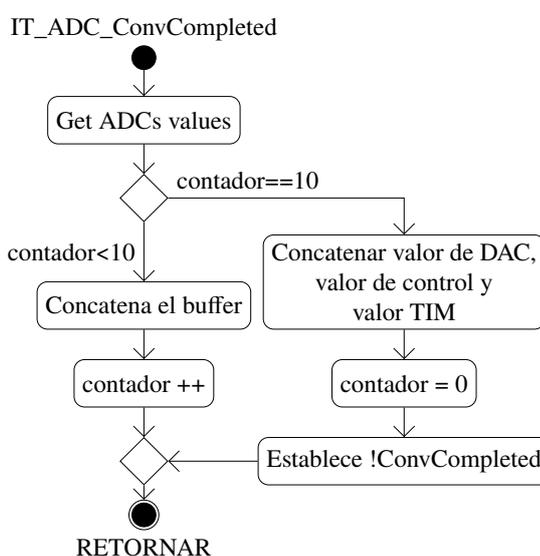


Fig. 3.5: Diagrama de interrupción del software para el microcontrolador. Secuencia de pasos cuando se produce interrupción debido a la finalización de la conversión del DAC.

Los valores "ADC1" y "ADC2" corresponden a las conversiones adquiridas de los ADC1 y ADC2 del módulo. El valor "DAC" representa el valor establecido en el DAC2, el "TIM" es el registro del temporizador y "Ctrl" corresponde a la bandera empleada como información de realimentación para el software en la PC.

3.1.4 Diseño del software de adquisición

Los datos transmitidos a través de UART llegan al puerto serial de la computadora, la cual tiene que almacenarlos y adecuarlos en un formato para su visualización. El software para la PC, denominado simplemente como software, es el programa encargado de realizar esta tarea. Este componente cuenta con dos partes, una de adquisición y otra de procesamiento. La parte de adquisición es la que se discute en esta sección.

El software de adquisición está constituido por dos módulos principales los cuales son:

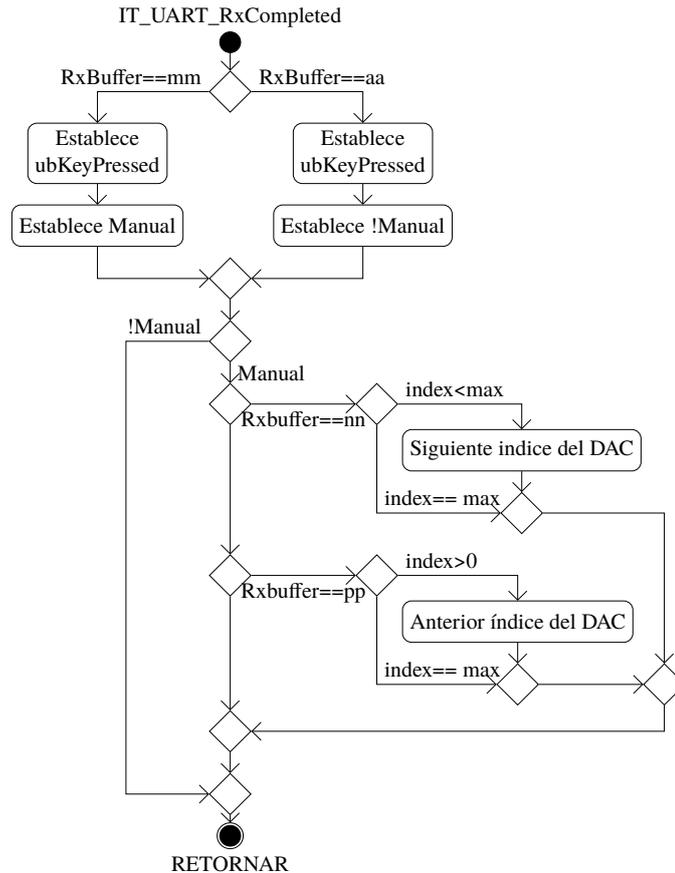


Fig. 3.6: Diagrama de interrupción del software para el microcontrolador. Secuencia de pasos cuando se produce interrupción debido a la finalización de la recepción desde el UART.

el módulo serial *SerialModule* y la interfaz de usuario *GUI*. El primero de éstos se ejecuta en segundo plano y efectúa las operaciones de comunicación con el hardware, mientras que el segundo es el segmento de programa ejecutado en primer plano y constituye la plataforma de interacción con los usuarios. Las clases que constituyen a ambos módulos, para generar el software, se muestran en Fig. 3.7.

La interfaz de usuario *GUI* hereda sus características del módulo *MeasureMainContents*, el cual a su vez hereda de *MainWindow*. El último de éstos provee los elementos gráficos generales de la ventana de aplicación tales como la barra de menús, la barra de estado, la barra de herramientas y el menú contextual, mientras *MeasureMainContents* provee los componentes propios para la comunicación con el módulo serial, así como para la visualización de la información. Estos componentes se describen en la TABLA N° 3.2

Por otro lado, el módulo *SerialModule* tiene la capacidad de comunicarse con el hardware y dotarle de las funciones para recibir información desde el módulo *GUI*, a la vez que establece los valores en este último. Las funciones principales son: *Settings*, que establece los valores en

TABLA N° 3.2: Módulos de interfaz gráfica.

Módulo	Descripción
<i>ConnectionModule</i>	Contiene las funciones para ingresar los parámetros de la comunicación serial tales como la velocidad y la cantidad de bits.
<i>ControlDACModule</i>	Provee los botones para intercambiar entre los modos de operación: manual o automático.
<i>ControlButtons</i>	Permite al usuario adquirir la información del módulo serial e interactuar con el hardware.
<i>LivePlotting</i>	Muestra la información obtenida desde el módulo serial.
<i>RecordingModule</i>	Permite a los usuarios almacenar la información adquirida desde el puerto serial guardándolo en un fichero de texto plano.

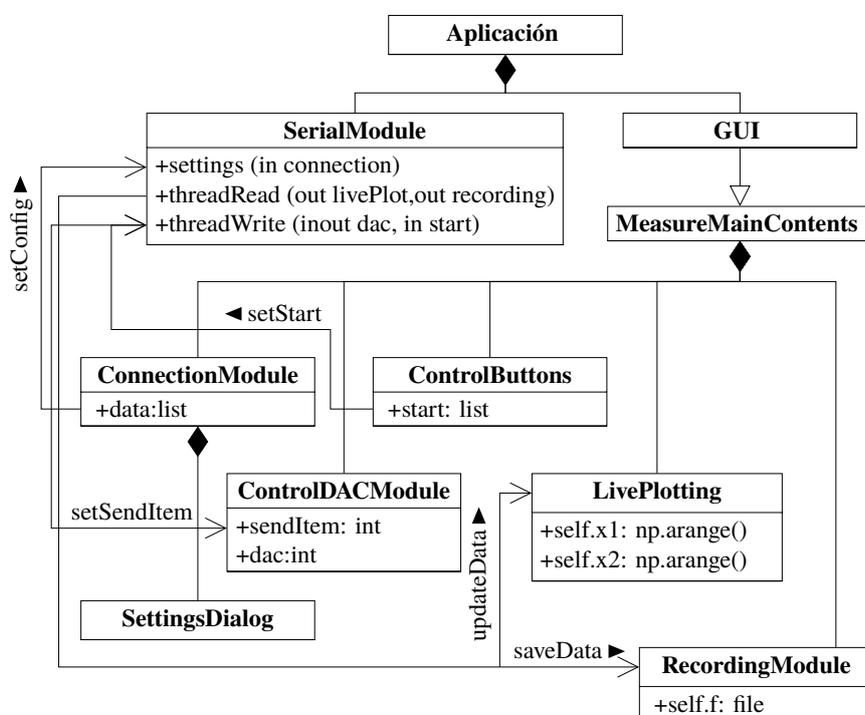


Fig. 3.7: Diagrama de clases del software. Módulos constituyentes del programa ejecutado en la computadora.

el puerto serial, tales como la velocidad y la longitud de bits, *ThreadRead*, que implementa un hilo independiente para adquirir la información del puerto serial y *ThreadWrite*, que implementa también un hilo independiente para enviar las instrucciones al hardware desde la interfaz de usuario.

La secuencia de operación es mostrada por la Fig. 3.8. En este caso, cuando el usuario abre el programa, inmediatamente crea los módulos de la interfaz y de la comunicación serial. Luego, a través de *ConnectionModule*, el usuario abre la ventana de diálogo para la

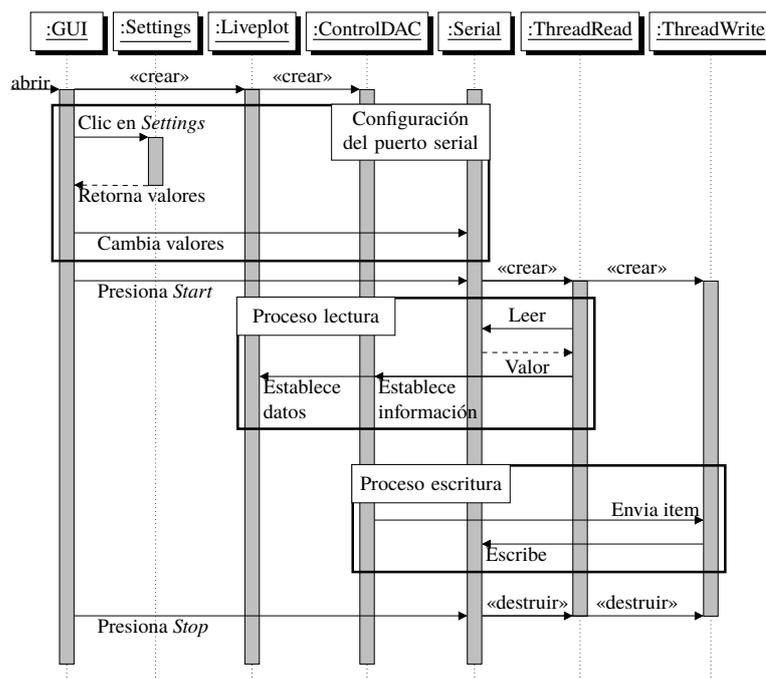


Fig. 3.8: Diagrama de secuencia del software. Muestra el flujo de las principales acciones ejecutadas en el software.

configuración del módulo serial. Una vez realizado este paso, se puede comenzar la recepción de información en la ventana de la interfaz al presionar el botón "Start". Esta acción crea dos hilos correspondientes a *ThreadRead* y *ThreadWrite* para recibir y enviar información a través del puerto serial, habilitando los procesos de lectura y escritura.

El proceso de lectura consiste en solicitudes continuas al puerto serial. A partir de ello se obtiene el valor de lectura, basado en el cual se actualizan las variables en *LivePlotting* y *ControlDACModule*. Por su parte, el proceso de escritura resulta de la interacción con el usuario quien, al realizar una acción en el módulo *ControlDACModule* a través de activar alguno de sus botones de control, envía a través del puerto serial las instrucciones para cambiar el valor del DAC.

Finalmente, una vez presionado el botón *Stop*, se efectúa la rutina de detención, la cual destruye los hilos creados y detiene la visualización de la gráfica, la adquisición de los datos, así como la interacción con los módulos del hardware.

3.1.5 Diseño del software de procesamiento

Los datos obtenidos con el software de adquisición corresponden a las tensiones capturadas por los ADCs del sistema de medición. Durante el proceso de adquisición, las señales sufren alteraciones producto del ruido procedente de diferentes fuentes, tales como la tempe-

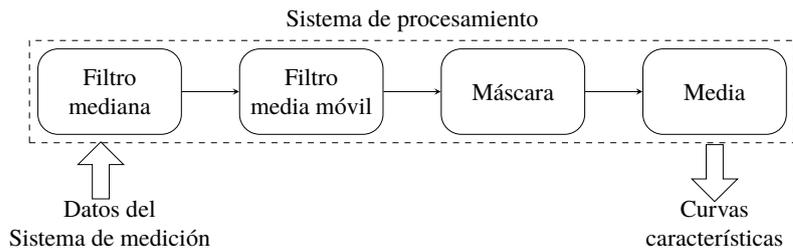


Fig. 3.9: Diagrama de las etapas del sistema de procesamiento para los datos obtenidos con el sistema de medición.

TABLA N° 3.3: Etapas de procesamiento

Etapa	Características
<i>Filtro de mediana</i>	<p>Empleado para eliminar los saltos abruptos de tensión y corriente. Este filtro es no lineal y se puede expresar tal como se muestra en la ecuación (3.2) [104]:</p> $y[n] = \text{med}[x[n - k], x[n - k + 1], \dots, x[n], \dots, x[n + k - 1], x[n + k]] \quad (3.2)$ <p>Donde $2k + 1$ es la dimensión del subconjunto donde se va a calcular la mediana.</p>
<i>Filtro media móvil</i>	<p>Empleado para reducir las alteraciones registradas debido a las diferentes fuentes de ruido que producen irregularidades en la salida, cuando idealmente deberían ser estables. Este filtro puede ser expresado como se muestra en (3.3) [105]:</p> $y[i] = \frac{1}{M} \sum_{j=0}^{M-1} x[i + j] \quad (3.3)$ <p>Donde M es la cantidad de puntos empleados para el promedio.</p>
<i>Máscara</i>	<p>Sirve para separar los grupos de muestras, entre los que corresponden a las rampas y los correspondientes a los tramos relativamente constantes, resultado de aplicar el filtro media móvil.</p>
<i>Promedio</i>	<p>Empleado para obtener el valor representativo de cada uno de los grupos clasificados por la máscara.</p>

ratura, la radiación electromagnética, el proceso de cuantización, entre otros. Dado esto, la señal resultado no es pura, por lo que se requiere una etapa de procesamiento a fin de rescatar los datos útiles que constituirán la información de entrada del proceso de caracterización.

Las partes contempladas en el procesamiento de la información se pueden apreciar en Fig. 3.9, y consta de las etapas descritas en TABLA N° 3.3.

Una vez efectuadas estas etapas se logra obtener la información de las curvas características, las cuales pueden ser utilizadas en el procedimiento de extracción de parámetros.

TABLA N° 3.4: Principales módulos de software de procesamiento

Módulo	Descripción
<i>AnalyzerDock</i>	Consistente en un panel que se puede independizar de la ventana principal y es en donde visualiza las formas de onda obtenidas desde el sistema de medición. Allí también se presentan los datos filtrados, a la vez que se pueden seleccionar cada uno de los pasos durante los cuales la tensión y la corriente se mantienen constantes.
<i>EditorDialog</i>	Es el cuadro de diálogo a través del cual se analiza la información referente a cada uno de los pasos, obteniéndose valores de tensión y corriente definidos.
<i>VisualizerPlot</i>	Es donde se presentan los datos depurados resultado del procesamiento efectuado por <i>DataProcessor</i> y actualizados mediante <i>EditorPlot</i> , a través del cual se puede apreciar la curva final resultado del proceso de caracterización.
<i>ExportDialog</i>	Extrae la información resultado del tratamiento de información en un archivo de texto plano.

3.2.1 Selección de estructura de acondicionamiento

Nicolas Moser *et al.* [59], de la universidad Imperial College London, ha recopilado las principales configuraciones para el acondicionamiento de señal del sensor, separándolas en dos grupos: las que se realizan empleando un solo ISFET y las que requieren de una estructura diferencial. En el primer grupo destacan dos estructuras principales: Seguidor de fuente y drenador, también conocido como circuito de tensión-constante y corriente-constante (CVCC) y el circuito de acondicionamiento con medición de corriente.

El circuito CVCC es el circuito clásico para el acondicionamiento del ISFET, el cual a su vez tiene diversas variantes. Desde los primeros años de creación del ISFET, Bergveld [106] ya había propuesto algunas configuraciones los cuales funcionan con el mismo principio que el CVCC. La primera propuesta consiste en un puente resistivo equilibrado tal como se puede apreciar en la Fig. 3.11. En esta propuesta, el ISFET ocupa uno de las cuatro posiciones del puente, mientras que en el otro lado se tiene dos resistencias encargadas de ajustar el valor de tensión de referencia.

La segunda propuesta de Bergveld es un amplificador de instrumentación realimentado mostrado en la Fig. 3.12. En este caso, la estructura cuenta con un amplificador de instrumentación, cuyas terminales de entrada se encuentran conectadas a una resistencia (R_1) que le provee de una tensión constante ($R_1 \times I$), mientras que en el lugar de la resistencia de

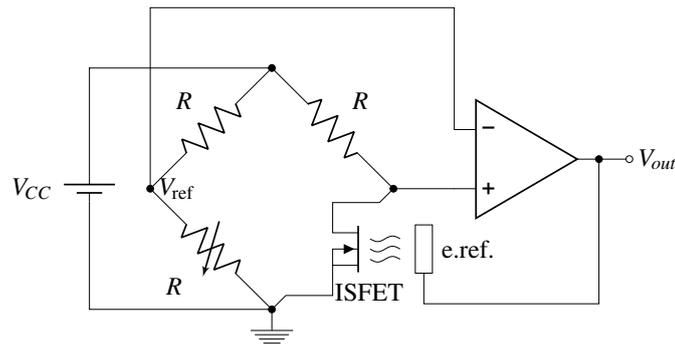


Fig. 3.11: Puente resistivo para ISFET

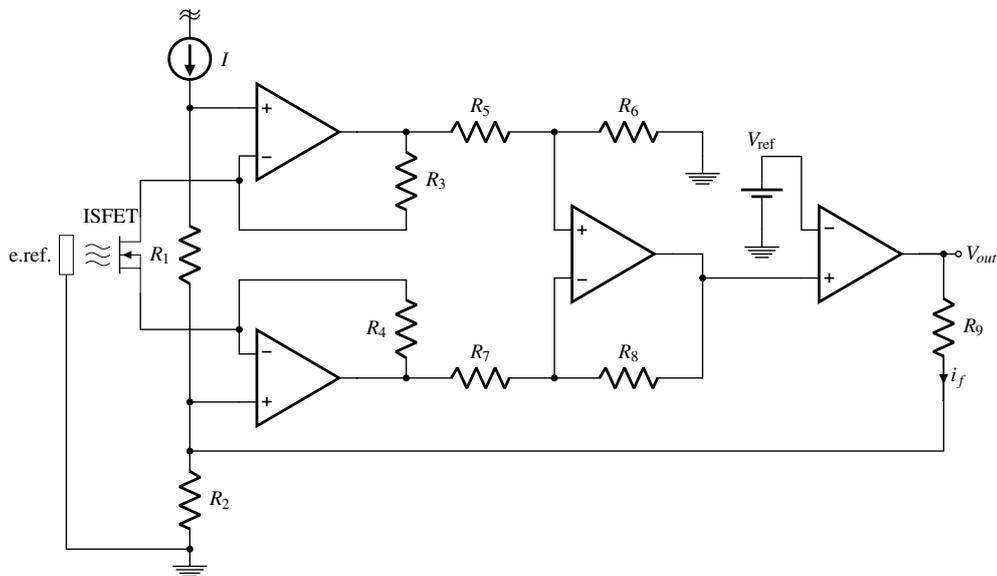


Fig. 3.12: Circuito de amplificador de instrumentación realimentado.

regulación de amplificación, se encuentra el ISFET conectado en drenador y fuente.

Abe *et al.* [48], por su parte, propone un circuito de medición correspondiente a la estructura clásica del CVCC, con realimentación en el electrodo de referencia, el cual se puede apreciar en la Fig. 3.13. En este caso, se emplea un amplificador operacional (OPAMP) en configuración de seguidor de tensión para proporcionar una tensión constante (V_D) en la terminal de drenador del ISFET. Luego, otro OPAMP se encarga de dar la tensión de referencia (V_{ref}) a partir de una configuración inversora. Esta tensión de referencia se relaciona con la resistencia conectada a la terminal de fuente del ISFET (R_3) para proporcionar la corriente constante. Así mismo, esta terminal está conectada con un OPAMP en realimentación, el cual ocasiona que este punto se encuentre en tierra virtual, mientras que su salida está conectada con la terminal del electrodo de referencia proporcional al valor de pH.

De la ecuación de la corriente del ISFET mostrada en (2.13), se puede apreciar que,

TABLA N° 3.5: Comparación de microprocesadores con bloques analógicos

Característica	PIC16F1777	MSP430FR2311	CY8C58LP
MICROPROCESADOR	RISC 8 bits	RISC 16bits	ARM Cortex-M3 32bits
FRECUENCIA	32MHz	16MHz	48MHz
MEMORIA	28Kb RAM programa 2Kb RAM datos	3.75Kb FRAM	256Kb Flash programa 64Kb RAM 32Kb (ECC) 2Kb EEPROM
CONSUMO	Activo: 32uA/MHz a 1.8V Dormido: 50nA a 1.8V	Activo: 126uA/MHz Reposo: 1uA Apagado: 25nA	Activo: 3.8mA a 6MHz, 15.4mA a 48MHz Dormido: 2uA Hibernado: 300nA
ADC	1 ADC de 10 bits con 28 canales	1 ADC de 10 bits con 8 canales	1 ADC Sigma-Delta de 20 bits 2 ADC SAR de 12 bits
DAC	4 DAC de 10bits	1 DAC de 6 bits	4 DAC de 8 bits
BLOQUES ANALÓGICOS	4 OPA 8 comparadores -	Smart Analog Combo (SAC-L1) OA de propósito general - TIA	4 Opamp 4 comparadores 4 bloques configurables (PGA, TIA, Mixer, circuito de muestreo)
TENSIONES DE REFERENCIA	1.024V, 2.048V, 4.096V	1.5V	1.024V

En el fundamento teórico se han identificado tres microcontroladores, los cuales disponen de bloques analógicos para la implementación del sistema de acondicionamiento. Estos tres dispositivos son el PIC16F777/8/9 [109] de Microchip, el MSP430FR2310/1 [110] de Texas Instruments y la familia PSoC de Cypress. De esta última se ha escogido el PSoC 5LP (CY8C58LP [111]), puesto que cuenta con la mayor cantidad de dispositivos analógicos para la implementación del sistema de acondicionamiento. En la TABLA N° 3.5, se muestra una comparación de las características de los dispositivos elegidos.

Como se puede apreciar en esta tabla, los microcontroladores tienen procesadores de 8, 16 y 32 bits respectivamente. Aunque un procesador de 8 bits podría resultar más favorable para sistemas de bajo consumo, no tiene tanta flexibilidad para realizar operaciones más complejas, como en el caso de uno de 16 o 32 bits. Respecto a la frecuencia de operación, el PSoC es el que cuenta con el valor más alto, igual a 48MHz, no obstante, una alta frecuencia

implica también un mayor consumo de potencia. Afortunadamente, dicha frecuencia se puede escoger mediante configuración del dispositivo. En cuanto a la memoria el integrado con mayor capacidad nuevamente es el PSoC5. No obstante, en el consumo, los dispositivos de Microchip y Texas Instruments presentan una notable reducción, siendo el menor para el PIC. Aun así, se puede manejar el consumo de potencia mediante otras estrategias de reducción abordadas desde el punto de vista del sistema global. Respecto a los ADCs, el PSoC es el que cuenta con un ADC Sigma-Delta diferencial, además de dos ADCs tipo SAR (ADC de aproximaciones sucesivas), mientras que los otros dos sólo cuentan con un ADC. En cuanto al DAC, el PIC es el que cuenta con una mayor resolución. No obstante, el PSoC tiene la ventaja de poder configurar los DACs para que tengan el comportamiento de una fuente de tensión o de una fuente de corriente. En cuanto a los bloques analógicos, el que cuenta con mayor variedad nuevamente es el PSoC.

Al tratarse de un prototipo en el cual se debe contar con la mayor disponibilidad de módulos analógicos a fin de facilitar el diseño del sistema de acondicionamiento, el dispositivo elegido para este propósito, basado en la comparación anterior, es el PSoC 5LP.

De esta manera, la electrónica de adaptación del ISFET se efectúa empleando los bloques analógicos internos del PSoC, realizando para ello una modificación de la estructura convencional del circuito CVCC.

Esta modificación se puede apreciar en Fig. 3.14. La diferencia principal respecto a la implementación convencional de la estructura CVCC es que en este caso la corriente de polarización para el ISFET no es proporcionado por la tensión de un amplificador en una resistencia, sino directamente por una fuente de corriente constante (aprovechando la posibilidad de implementar fuentes de corriente regulables a través de los bloques del PSoC). Por otra parte, la tensión constante que se le debe aplicar a las terminales de drenador y fuente es proporcionada, por un lado, a través de un conversor DAC seguido de un amplificador en configuración de seguidor de tensión, mientras que la otra tensión constante la proporciona una terminal de tensión de referencia generada por el mismo microcontrolador. El bloque de realimentación, el cual tiene como salida el valor de tensión proporcional al pH, está constituido por un amplificador operacional cuya terminal de salida se conecta al electrodo de referencia del ISFET.

La ecuación de salida del circuito mostrado por Fig. 3.14 está basada en la ecuación de corriente del ISFET. Puesto que la corriente y la tensión en las terminales son constantes, se

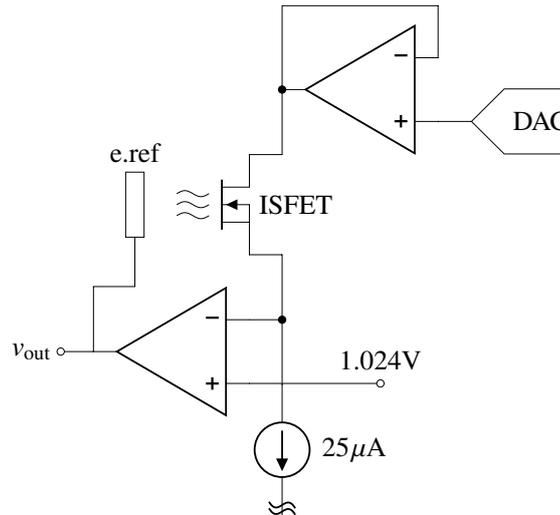


Fig. 3.14: Esquema del sistema de acondicionamiento propuesto para el ISFET

puede despejar el valor de la tensión entre el electrodo de referencia y la fuente v_{GS} , solo que en este caso la corriente y la tensión están dadas por el IDAC y la diferencia de tensión entre el VDAC y la tensión de referencia de 1.024V, como se muestra en (3.5).

$$v_{GS} = V_{TH} + \frac{v_{VDAC} - 1.024}{2} + \underbrace{i_{IDAC} \frac{L}{W\mu C_{ox}(v_{VDAC} - 1.024)}}_{\text{constante}} \quad (3.5)$$

Luego, se puede hallar la tensión en la salida, tal como se aprecia en la ecuación (3.6):

$$v_{out} = 1.024 + v_{GS} = 1.024 + V_{TH} + \frac{v_{VDAC} - 1.024}{2} + \underbrace{i_{IDAC} \frac{L}{W\mu C_{ox}(v_{VDAC} - 1.024)}}_{\text{constante}} \quad (3.6)$$

Respecto a los valores empleados para la operación del dispositivo, se ha establecido un valor para la corriente equivalente a $25\mu A$, el cual corresponde al rango de operación del dispositivo señalado en su hoja de datos técnicos ($25-30\mu A$) [68].

Por otra parte, se debe establecer la tensión constante aplicada a las terminales entre drenador y fuente. Para ello se analiza los rangos de tensión permisibles para la operación del ISFET [68]. Puesto que debe operar en la zona lineal, la tensión no debe superar la diferencia entre el valor de tensión del electrodo de referencia respecto a fuente, menos la tensión umbral. Para ello, es de bastante utilidad las curvas de caracterización, dado que dan una idea de los rangos en los cuales la corriente llega a saturarse y abandonar la zona lineal bajo un determinado valor de tensión umbral. Así mismo, tomando como referencia el circuito de acondicionamiento empleado por el fabricante y mostrado en [68], se puede identificar un valor de tensión recomendado de 300mV, el cual se empleará también para el

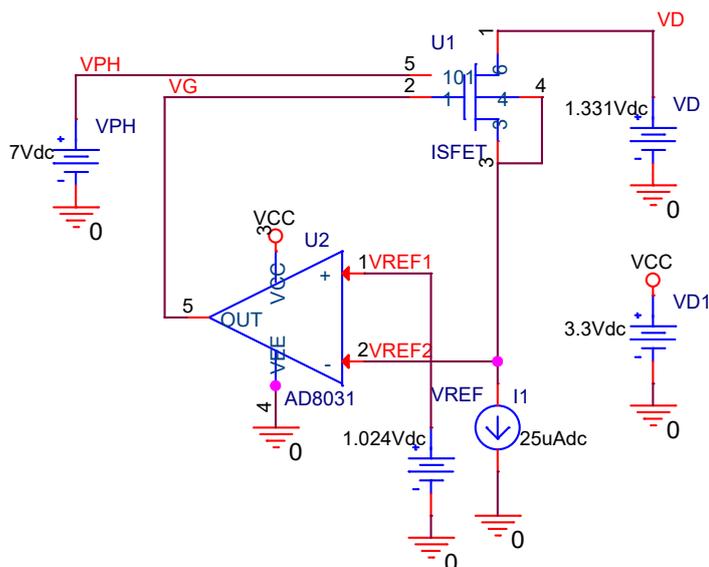


Fig. 3.15: Simulación del circuito de acondicionamiento para el ISFET. Esquema elaborado con Capture CIS Lite.

este diseño. Luego, una vez escogida la tensión en fuente de 1.024V, la tensión en drenador debería ser 1.324V.

Finalmente, el OPAMP conectado en realimentación con el electrodo de referencia tiene una de sus terminales conectada con la fuente de corriente y la terminal de fuente del ISFET, mientras que la otra se encuentra conectada al bloque de tensión de referencia. La salida de tensión de este OPAMP, empleada para la lectura de pH, tiene un valor de desplazamiento equivalente a la tensión de referencia, por lo que eleva las lecturas a valores por encima de los 2V. Es importante considerar este dato, puesto que, de ser muy alto el valor de referencia, podría llegar a saturar la tensión de salida al nivel superior de la fuente de alimentación.

3.2.3 Simulación de circuito de acondicionamiento para el ISFET

Posterior al diseño efectuado del circuito de acondicionamiento, se realiza su simulación para verificar su comportamiento. Puesto que no se tiene el modelo de los dispositivos internos del PSoC se emplean componentes genéricos para los OPAMPs y las fuentes de corriente y de tensión. El esquema del circuito es el presentado en la Fig. 3.15. Para efectos de la simulación se ha empleado el software OrCAD Capture CIS Lite, versión 17.2.

El modelo del ISFET es el obtenido como resultado de la extracción de parámetros, adaptando el modelo original a partir de los datos del fabricante y el proceso de caracterización del sensor. El resultado de la simulación se puede apreciar en la Fig. 3.16.

Allí se muestra que la respuesta exhibe un comportamiento lineal que relaciona la tensión

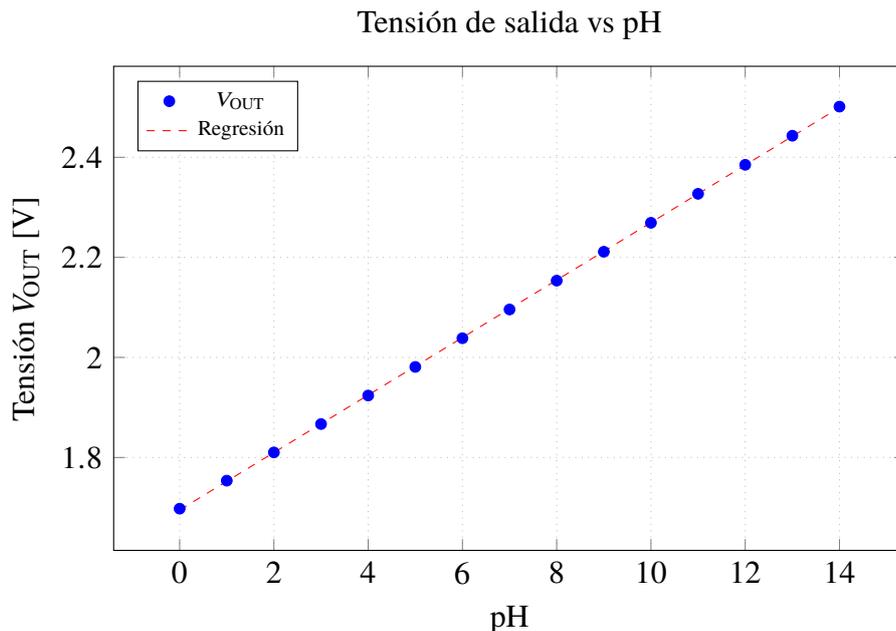


Fig. 3.16: Gráfica de datos obtenidos de simulación del circuito de acondicionamiento para diferentes valores de pH.

de salida del amplificador en realimentación con el valor del pH de la solución donde se encuentra sumergido. Con los datos obtenidos de las simulación se puede obtener una curva de regresión, cuya ecuación se muestra en (3.7):

$$V_{OUT} = 0.0574 \times \text{pH} + 1.695 \quad (3.7)$$

De acuerdo a ello, se puede identificar una sensibilidad de 57.4mV/pH, valor adecuado para el dispositivo y bastante cercano al de máxima sensibilidad dada por la relación de Nernst (59mV/pH), confirmando la elección apropiada de las tensiones y corrientes empleadas para la polarización del ISFET. No obstante, a pesar de los resultados prometedores de las simulación, es necesario considerar que se trata de un modelo aproximado, lo cual puede presentar variaciones en el resultado final de la implementación. Así mismo, se debe tener en cuenta que el rango de operación del dispositivo es de 2 a 12pH de acuerdo a sus especificaciones técnicas [68] y, aunque la simulación muestra una linealidad en esta zona, lo más probable es que el dispositivo real no presente este comportamiento para valores fuera del rango especificado.

3.2.4 Diseño del circuito de acondicionamiento para el sensor de temperatura

De acuerdo a las ecuaciones obtenidas de la relación de la tensión de salida respecto al pH de la solución (2.38), se evidencia una fuerte dependencia respecto a la temperatura. Por tanto,

TABLA N° 3.6: Principales tecnologías de sensores de temperatura [112], [113]

Característica	Termocupla	Termoresistencia	Termistor
RANGO DE TEMPERATURA	Excelente -210°C a 1760°C	Grande -240°C a 650°C	Buena -40°C a 250°C
LINEALIDAD	Adecuada	Buena	Pobre
SENSIBILIDAD	Baja 10 a 50µV/°C	Baja 0.2 a 10Ω/°C	Alta 100 a 1000Ω/°C
TIEMPO DE RESPUESTA	Medio a rápido	Medio	Medio a rápido
ESTABILIDAD	Adecuada	Buena	Pobre
PRECISIÓN	Media	Alta	Media
SUSCEPTIBILIDAD A AUTOCALENTAMIENTO	No	Sí, mínima	Sí, elevada
DURABILIDAD	Excelente	Buena	Pobre
COSTO	Bajo	Alto	Bajo
REQUISITOS DE ACONDICIONAMIENTO	Compensación de junta fría, amplificación, detección de termocupla abierta y escalamiento	Excitación, corrección de resistencias de contactos y escalamiento	Excitación y escalamiento

es necesario realizar una compensación tomando como referencia el valor de temperatura a la que se encuentra la solución medida.

De los sensores de temperatura, tres son los más conocidos: la termocupla, a termoresistencia y el termistor. Una comparación de estas tres tecnologías se puede apreciar en la TABLA N° 3.6. De los tres dispositivos mencionados, el termistor es el que cuenta con la mayor sensibilidad y un rápido tiempo de respuesta, además de contar con los requisitos más sencillos para su acondicionamiento. Adicionalmente, este dispositivo tiene la ventaja de disminuir los efectos de las resistencias de los contactos al tener un elevado rango de resistencias de operación. La desventaja es su pobre linealidad, no obstante, al tener una ecuación característica, es posible calcular matemáticamente el valor correspondiente a la temperatura. Dicha ecuación se presenta en (3.8) [114]:

$$R = R_0 e^{\beta(1/T - 1/T_0)} \quad (3.8)$$

Donde $\beta = 3950\text{K}$ es la constante del material y $R_0 = 10\text{k}\Omega$ es la resistencia a la temperatura de $T_0 = 25^\circ\text{C}$.

El circuito de acondicionamiento clásico para el termistor y la termoresistencia consiste

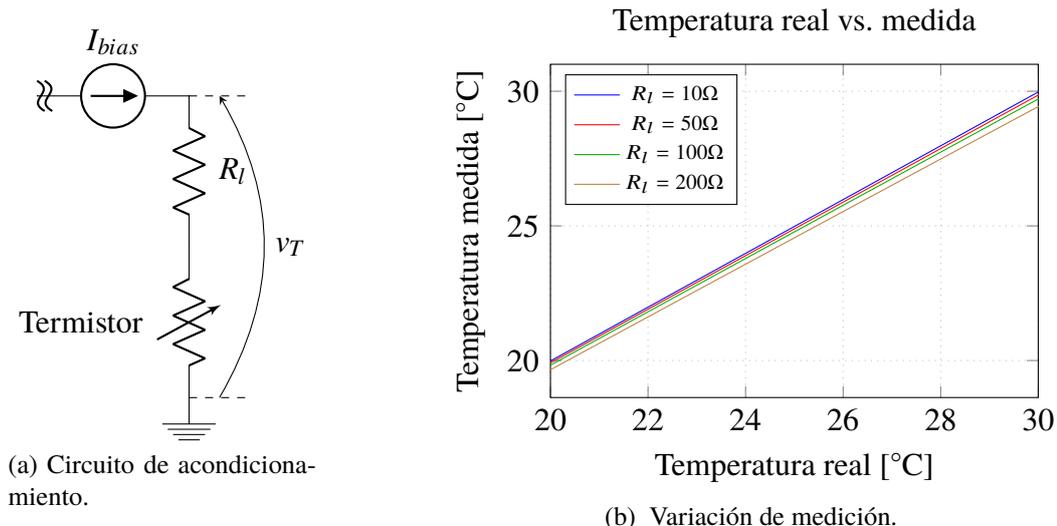


Fig. 3.17: Influencia de las resistencias en la medición de la temperatura.

en una fuente de corriente, la cual proporciona una corriente fija a través del dispositivo con un nivel de tensión en sus terminales, proporcional a la resistencia del sensor [115]. El esquema de este sistema de acondicionamiento se puede apreciar en la Fig. 3.17(a).

En este esquema se presenta una variación en la medición debido a las resistencias en los contactos R_l , cuya influencia se manifiesta en una variación de la lectura respecto a la temperatura original, tal como se muestra en Fig. 3.17(b). Dado que no es posible realizar una medición directa de esta magnitud, se requiere de un proceso de calibración para obtener las medidas de temperatura reales.

Por otra parte, se debe definir el nivel del valor de la corriente de polarización I_{bias} de acuerdo al rango de temperaturas que se requiere medir. Puesto que el sensor exhibe una característica decreciente de resistencia respecto a la temperatura, la tensión máxima se presentará cuando ésta sea mínima, donde el valor debe ser inferior al de la fuente de alimentación. De acuerdo a la aplicación, el rango de temperaturas se encuentran alrededor de la temperatura ambiente, por lo que se establece una temperatura mínima de 10°C . Luego, considerando una alimentación de 3.3V , la corriente sería como se muestra en (3.9):

$$I_{bias} = \frac{3.3\text{V}}{R_{10^{\circ}\text{C}}} = \frac{3.3}{20189} \approx 163\mu\text{A} \quad (3.9)$$

Con la finalidad de dar un margen para temperaturas inferiores, se opta por emplear un valor de corriente $I_{bias} = 100\mu\text{A}$. Con este valor es posible determinar la temperatura máxima medible, la cual dependerá también de la resolución del ADC empleado para convertirlo al dominio digital.

3.2.5 Diseño del sistema de conversión analógico-digital

Para el presente trabajo, los ADCs empleados forman parte de los periféricos del microcontrolador PSoC 5LP, por lo cual el diseño estará sujeto a la disponibilidad de éstos dentro del dispositivo. De acuerdo a las especificaciones técnicas mostradas en TABLA N° 3.5, éste cuenta con dos ADCs tipo SAR de 12 bits y un ADC Sigma-Delta de 20 bits. Puesto que la aplicación se encuentra dentro del ámbito de instrumentación, el primer criterio a tener en cuenta, es la obtención de una elevada resolución, por lo que idealmente la mejor alternativa para ambos sensores (ISFET y NTC) serían ADCs Sigma-Delta. No obstante, al disponer sólo con uno de este tipo, se prioriza su empleo en el ISFET, al ser ésta la medida principal del sistema, empleado un ADC SAR para el sensor NTC. Otro motivo por el cual se emplea el ADC Sigma-Delta con el ISFET, es su entrada diferencial, de gran utilidad al posibilitar su conexión directa a las terminales del ISFET, evitando el ingreso de la tensión de bias de 1.024 empleada para polarizarlo, elevando a su vez la resolución de medida.

Otro criterio a tomar en cuenta es la cantidad de bits de los ADCs, lo cual influye en la resolución y también en la potencia disipada. Lograr un valor óptimo de la cantidad de bits también depende de la frecuencia de muestreo, puesto que a un mismo valor de frecuencia, una variación de algunos bits de resolución puede significar un aumento radical de la corriente consumida. De acuerdo a la información del fabricante [116] y las recomendaciones para un bajo consumo de energía [117], se logra un balance adecuado para el ADC Sigma-Delta con 12 bits de resolución a una frecuencia de muestreo de 10ksps.

Tomando como referencia este valor, es posible determinar la resolución máxima de las mediciones de pH. Si se considera los 12 bits de resolución con un rango máximo de -2.048 a 2.048, entonces la resolución a nivel de tensión sería tal como se muestra en (3.10):

$$\Delta V = \frac{4.096}{10^{12}} = 0.001V \quad (3.10)$$

Luego, si se considera la sensibilidad del sensor hallada experimentalmente en (5.4), igual a 42mV/pH, la resolución en unidades de pH se expresaría tal como en (3.11):

$$\Delta pH = \frac{0.001}{0.042} = 0.0238pH \quad (3.11)$$

Para el caso del ADC SAR, considerando su resolución máxima de 12 bits, se puede calcular la resolución de la medición de temperatura. Aunque la relación no es lineal como en el caso anterior, se puede ofrecer una aproximación de este valor. En primer lugar la

resolución en volts sería obtenida de la ecuación (3.12):

$$\Delta V = \frac{2.048}{10^{12}} = 0.0005V \quad (3.12)$$

Luego, a nivel de resistencia, la resolución se expresaría como en (3.13):

$$\Delta R = \frac{0.0005V}{100\mu A} = 5\Omega \quad (3.13)$$

De acuerdo a la relación presentada en (3.8), se puede obtener la variación instantánea de la resistencia respecto a la temperatura, tal como se muestra en (3.14):

$$\frac{dR}{dT} = -R_0 \frac{\beta}{T^2} e^{\beta(1/T-1/T_0)} = -10000 \times \frac{3950}{T^2} e^{-3950/(298)} e^{3950/T} = -69.184 \frac{e^{3950/T}}{T^2} \frac{\Omega}{^\circ C} \quad (3.14)$$

Luego, la resolución para una temperatura T sería expresada por la ecuación (3.15):

$$\Delta T = 0.072 e^{-3950/T} T^{2^\circ C} \quad (3.15)$$

Así por ejemplo, para 10°C la resolución es de 0.005, para 25°C de 0.011, mientras que para 70°C es de 0.08.

El otro parámetro a tener en consideración es la frecuencia de muestreo. Puesto que las señales medidas son de tipo DC, entonces no se exige una elevada tasa de muestreo a los conversores. Por el lado del ADC Sigma-Delta, éste tiene una frecuencia máxima de conversión de 384ksps [116], no obstante, para la resolución planteada, la velocidad con menor consumo se presenta en 10ksps, por lo que se elige esta última. Por el lado del ADC tipo SAR, tiene una tasa de muestro máxima de 1Msps [118], no obstante, puesto que la potencia disipada aumenta con la frecuencia, se escoge un valor intermedio entre la máxima y el valor de muestreo del Delta-Sigma, llegando a un valor de 100ksps. De acuerdo a las especificaciones técnicas, esta frecuencia requiere de 18 ciclos de reloj para lograr una conversión, por lo que la frecuencia de entrada del ADC es de 1.8MHz.

Adicionalmente a los conversores encargados de trasladar las señales del dominio analógico al digital, se debe incluir dentro del diseño un bloque de comunicaciones para transmitir esta información hacia el exterior del microcontrolador. Existen varios protocolos para realizar esta transmisión, los cuales se encuentran disponibles en forma de interfaces de comunicación digital en la mayoría de microcontroladores. No obstante, uno de los más sencillos y ampliamente usados es el UART [119], existiendo también conversores para su comunicación con otros dispositivos que no cuentan con este protocolo (tales como las PCs a través de un conversor USB-UART), razón por lo que será empleado en la presente implementación.

El parámetro a tener en consideración corresponde a la velocidad en bauds/s, la cual se escoge en un valor intermedio de 115200. De esta manera, se logra una velocidad de transmisión de $8.68\mu\text{s}$ por bit, con un tiempo por byte mostrado en la ecuación (3.16):

$$\begin{aligned}\text{Tiempo por byte} &= 8.68 \times (1 \text{ bit start} + 8 \text{ bits datos} + 1 \text{ bit paridad} + 2 \text{ bits de stop}) \\ &= 8.68 \times 12 = 104.16\mu\text{s}\end{aligned}\quad (3.16)$$

3.3 Diseño del sistema de procesamiento de señal, compensación y control

En el fundamento teórico, en la sección 2.3.5, se presentaron los principales desafíos al emplear el ISFET como un sensor de pH. Éstos fueron recopilados por Moser *et al.* en [59] y son los siguientes: el efecto de cargas atrapadas, la desviación de corriente, la atenuación capacitiva, la sensibilidad a la temperatura y la respuesta al ruido. Moser también plantea las principales técnicas para la reducción de estos efectos no deseados en la respuesta del ISFET.

De acuerdo a ello, la tarea de compensación de dichos efectos se puede abordar desde dos frentes de diseño: a través de un circuito de adaptación o de un sistema de compensación. En el primer escenario, se trata de circuitos que se integran con el acondicionamiento del sensor corrigiendo directamente la lectura del ISFET, siendo los más eficientes en términos de durabilidad y fiabilidad. Una de las técnicas más prometedoras para el efecto de cargas atrapadas, consiste en la estructura de ISFET con compuerta programable PG-ISFET [120], la cual consiste en aplicar un nivel de tensión a la compuerta flotante del ISFET a través de una capacitancia para cancelar el efecto de las cargas en la superficie, la cual también puede ser empleada para mitigar el efecto de desviación de corriente, a través de la variar esta tensión de tal forma que se adapte a la desviación. Otras técnicas empleadas para compensar la desviación de corriente se centran en esquemas de reinicio, siendo uno de ellos el propuesto por Shah *et al.* en [121], donde se establece un comportamiento cíclico de la tensión en el electrodo de referencia con la finalidad de efectuar un reinicio efectivo del sensor.

Por otro lado, en cuanto a sistemas de compensación, la mayoría están basados en técnicas extraídas del ámbito de procesamiento de imágenes. Este es el caso del muestreo doble correlacionado (*correlated double sampling*, CDS), que ha mostrado tener buenos resultados para la compensación de la desviación de corriente [122]. Así mismo, para las cargas atrapadas, se aplica un método de calibración de línea base que emplea el promedio de 99 muestras en modo de reposo, las cuales se sustraen de la medición [123]. Para la compensación de las variaciones con la temperatura se emplea un coeficiente de compensación (TC), mientras que

para la cancelación de ruido se recurre al método de promedio.

En el presente caso, no es posible añadir un sistema de adaptación del ISFET debido a que no se tiene acceso directo a la terminal de compuerta del dispositivo, además que este tipo de técnicas están pensadas para ser implementadas a nivel del dispositivo integrado. De ahí que, de los dos frentes de diseño, la alternativa más adecuada recae en los sistemas de compensación.

Considerando que la aplicación en cuestión no requiere de mediciones continuas, se puede desactivar los sensores periódicamente para reducir el efecto de las desviación de corriente. Luego, respecto a la compensación a las variaciones de la temperatura, si bien las estrategias a nivel de sistema no son las más robustas, pueden ofrecer un grado de conformidad aceptable para mediciones que no requieren de un nivel muy elevado de precisión. Como resultado, se necesita diseñar el sistema de compensación para las muestras obtenidas, además del sistema de control para regular los tiempos de activación del sensor, todo en conjunto después de un procesamiento previo de los datos a fin de reducir el ruido introducido en las muestras.

3.3.1 Diseño del sistema de procesamiento

Las técnicas de procesamiento a emplear dependen del tipo de señal a las que son aplicadas y del tipo de información que se requiere extraer de ellas. En el presente caso se trata de señales DC, puesto que el nivel de tensión obtenida en la salida del sistema de adaptación corresponde a un valor fijo cuando la concentración es estable.

La medida por excelencia para obtener el valor de DC es la media de las muestras. No obstante, debido al ruido, pueden producirse algunas alteraciones no deseadas que originan una desviación de esta medida representativa. Una de estas fuentes es el ruido impulsivo, el cual se presenta con valores muy diferentes a los de su vecindad. Ante ello se escoge emplear en la primera etapa un filtro de mediana para cancelar este tipo particular de ruido. Luego, como una implementación de la media se escoge el filtro media móvil, debido a su simplicidad y eficacia para reducir el ruido blanco en la señal. El esquema del sistema se puede apreciar en Fig. 3.18. El diagrama es bastante similar al empleado al sistema de caracterización del ISFET mostrado en Fig. 3.9 con la diferencia que, en este caso, no se emplea una máscara para clasificar los grupos de datos.

Otro criterio de diseño para los filtros en este apartado corresponden a su dimensión. Para tener una idea del valor de este parámetro es preciso tener la información de la señal procedente del sistema de adaptación, diseñado en la sección 3.2. Con estos valores se ensaya

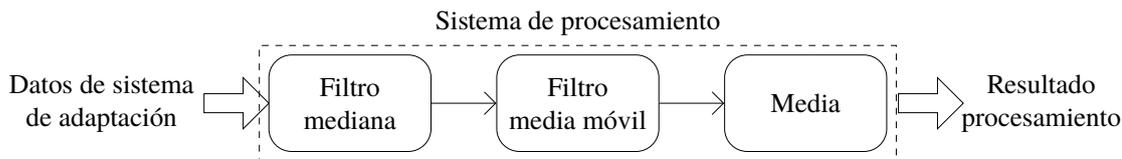


Fig. 3.18: Esquema del sistema de procesamiento

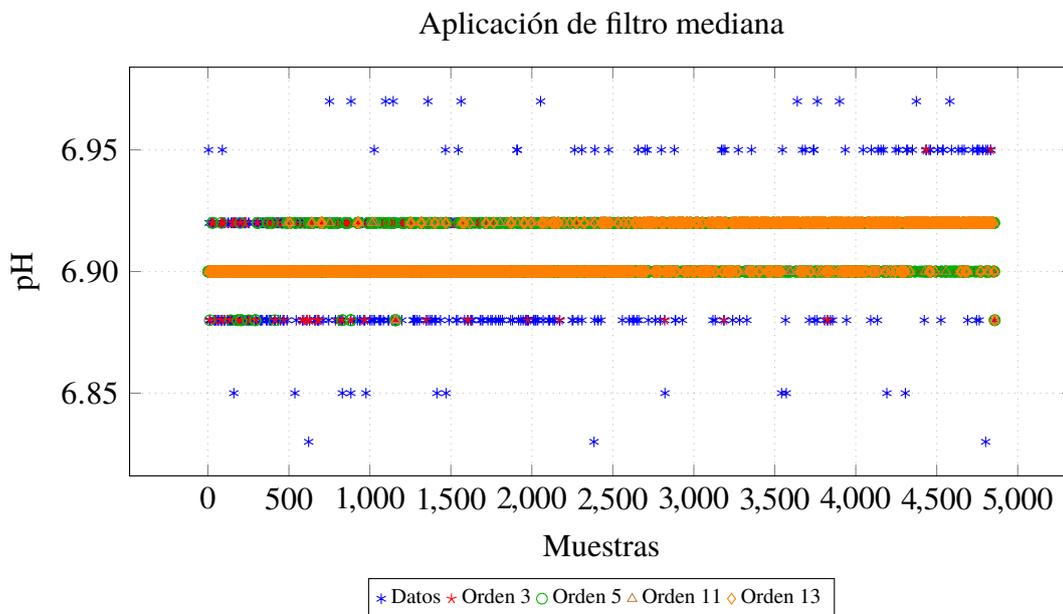


Fig. 3.19: Comparación de resultados de filtros de mediana de diferentes órdenes.

un conjunto de parámetros de mediana con la finalidad que la medición no tenga puntos aislados entre muestras.

Para el presente caso, se realiza una prueba con un buffer de pH 7 a temperatura ambiente, a partir del cual se obtienen las lecturas del ISFET hasta obtener un promedio de 5000 muestras. Luego, se aplican filtros mediana de diferentes órdenes para identificar el orden más apropiado para el filtro. El resultado de estos ensayos se puede visualizar en 3.19. En la figura se distingue que los valores de pH se encuentran variando entre 7 niveles, no obstante, luego de la aplicación del primer filtro (incluso con el orden mínimo igual a 3), ya se obtiene la reducción a una variación entre 3 o 4 niveles. Para poder distinguir con mayor claridad la fluctuación de los valores, se realiza una tabulación de la cantidad de datos por cada valor estable. La gráfica para esta tabulación se puede apreciar en Fig. 3.20.

De acuerdo a los resultados mostrados, se identifica que conforme aumenta el orden del filtro, disminuyen los niveles de variación de la medida de pH, llegando a solo tres cuando se éste es de 13 (con sólo un valor en uno de ellos). Basado en esto, se escoge este valor como el orden del filtro para los datos procedentes del pH.

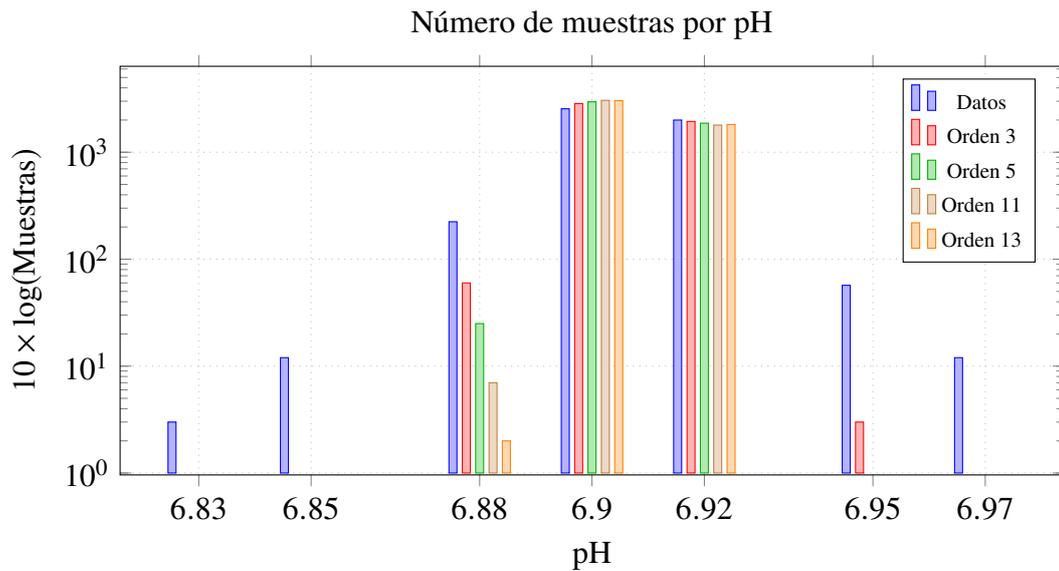


Fig. 3.20: Cantidad de muestras por valor de pH al aplicar filtro mediana.

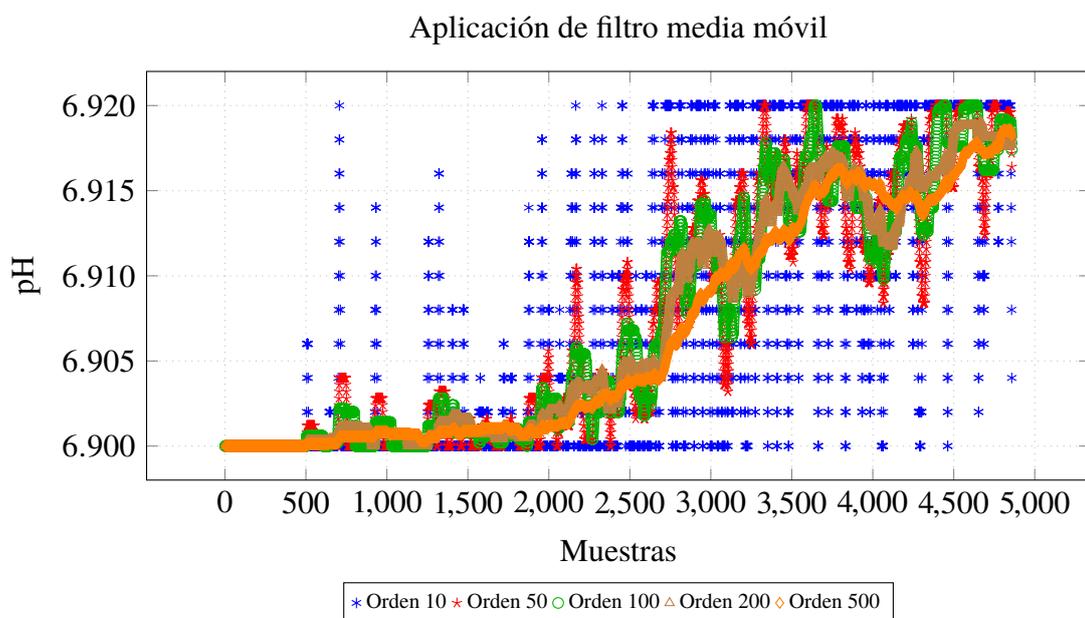


Fig. 3.21: Comparación de resultados de filtros de media móvil de diferentes órdenes.

Posteriormente, una vez escogido el orden del filtro de mediana, se procede a escoger el orden del filtro de media móvil. Para ello se realizan varias pruebas con los datos resultado de la mediana.

De acuerdo a los resultados, se puede apreciar que conforme aumenta el orden del filtro se obtiene un señal a la salida con menos perturbaciones. No obstante, debido a la capacidad computacional del microcontrolador, se requiere limitar su dimensión. Un valor apropiado, de acuerdo a los resultados presentados en Fig. 3.21 y también tomando como referencia la

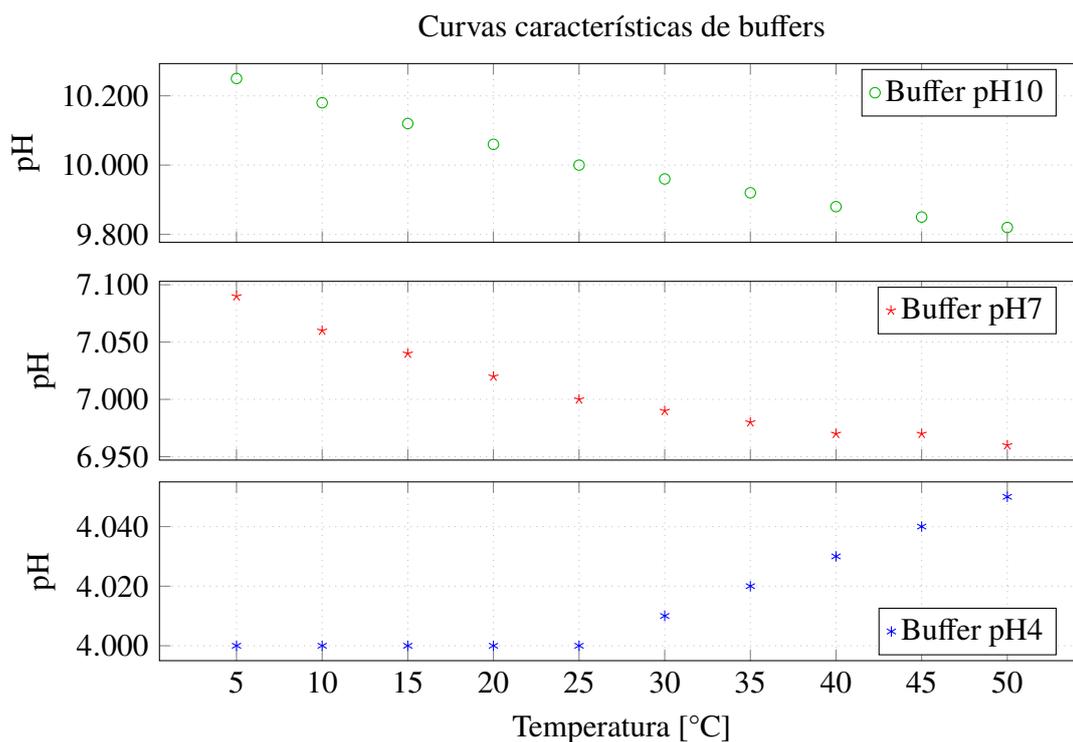


Fig. 3.22: Variación de pH de buffers con temperatura.

técnica de promedio para cargas atrapadas [123], es el de 100.

Por otra parte, también se debe identificar cuántas muestras deben tomarse como mínimo para tener un valor aceptable a la salida. De acuerdo a la gráfica en Fig. 3.21, se aprecia que durante las primeras 500 muestras el valor es estable, por lo cual se considera tomar este valor como referencia para la aplicación de los filtros de procesamiento.

3.3.2 Diseño del sistema de compensación

El pH tiene una variación respecto a la temperatura. Esta variación se debe a que conforme aumenta la temperatura incrementa la ionización de las moléculas de agua, lo cual provoca un incremento de los iones de hidronio. No obstante, eso no quiere decir que su característica básica o ácida cambie, solo que el valor de la concentración cambia pero de forma proporcional, de acuerdo al potencial de autoprotólisis (K_w) del agua [124], tal como se muestra en (2.3).

Este efecto se manifiesta también en las curvas características de los buffers de pH, los cuales, de acuerdo a las especificaciones técnicas, cuentan con una variación del valor de pH con la temperatura. Para el caso de los buffers empleados para las pruebas (Fig. 5.2) la dependencia se puede apreciar en la Fig. 3.22.

No obstante, aparte de este fenómeno, los instrumentos de medición basados en sensores

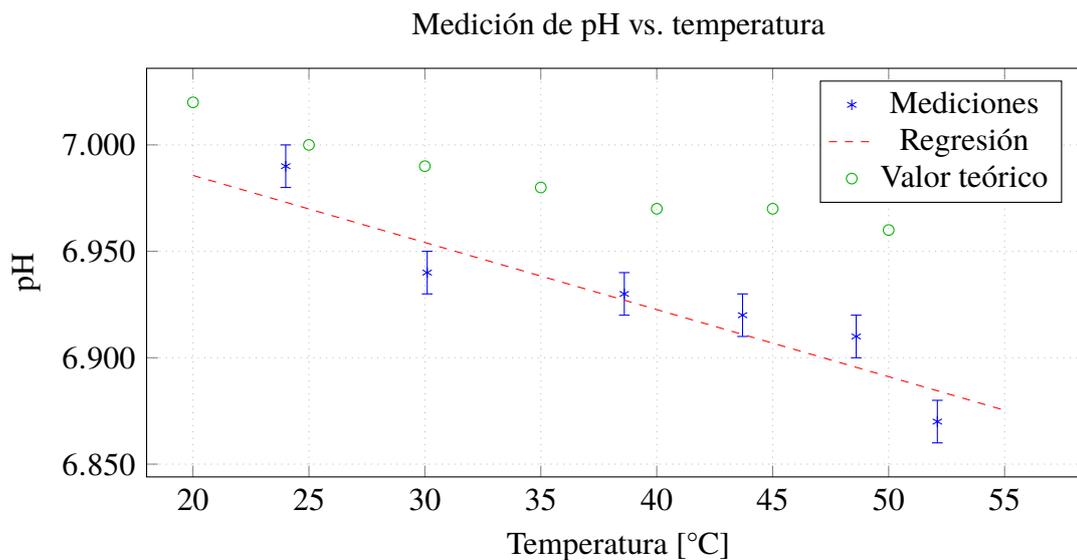


Fig. 3.23: Respuesta del sensor de pH comercial a cambios de temperatura.

potenciométricos también exhiben una dependencia de la temperatura, lo cual es independiente del aumento de concentración de iones. Así por el ejemplo, en el caso del electrodo de bulbo de vidrio, en su ecuación (2.11), se tiene un término dependiente de la temperatura que afecta la pendiente de sensibilidad del dispositivo [125]. No obstante, los sensores comerciales usualmente cuentan con un algoritmo de autocompensación de temperatura (ATC) que corrige esta relación respecto a los valores obtenidos por la calibración. Una muestra del funcionamiento de este algoritmo se puede apreciar en la Fig. 3.23, donde se ha empleado el pH-metro multiparamétrico C5010 de Consort [126] en una evaluación de temperatura desde 24°C hasta 50°C. Tal como se puede apreciar en la imagen, la corrección no es muy lineal, no obstante, se puede hacer una regresión con los datos para obtener un valor aproximado de la tendencia de variación del pH. De acuerdo a ello se obtiene la ecuación (3.17):

$$\text{pH} = -0.0031 \times T + 7.0487 \quad (3.17)$$

De acuerdo a ello se identifica una variación de 0.003 pH/°C. Así mismo, se verifica que a pesar de la compensación, la respuesta presenta variaciones respecto a los valores teóricos del sensor.

Por otra parte, el ISFET, como dispositivo, no es ajeno a esta dependencia por lo que también requiere de un sistema de compensación, a fin de disminuir los efectos derivados de ésta, logrando mediciones más fidedignas.

Existe, por tanto, una dependencia de la tensión obtenida en la salida del circuito de

acondicionamiento respecto a la temperatura y el pH. Para obtener esta relación, se procede a realizar unas pruebas con soluciones buffer de diferentes pH a diferentes temperaturas con la finalidad de identificar la variación de las curvas del sistema de adaptación.

La primera prueba se efectúa con un buffer de pH 4 con el cual se obtiene los gráficos de temperatura y tensión mostrados en Fig. 3.24(a) y (b) respectivamente. Durante el experimento, se sitúa el prototipo en un ambiente controlado de temperatura (dentro de una incubadora [127]) y se realizan variaciones progresivas de la temperatura de establecimiento, desde 25.0°C hasta 40.0°C con pasos de 5°C, los cuales representan las regiones con menor variación en el gráfico de Fig. 3.24(a). El experimento se realizó durante un tiempo aproximado 4 horas, con una tasa de 50 muestras en un promedio de 6 segundos, llegando a un total de 120000 muestras. En Fig. 3.24(b) se identifican también las zonas de establecimiento para las cuales la temperatura se mantiene estable, así como una tendencia decreciente del valor de la tensión de salida con el aumento de ésta. Al graficar esta tendencia, mostrada en la Fig. 3.24(c), se puede verificar una relación relativamente lineal, la cual se puede ajustar mediante una regresión a la ecuación (3.18):

$$V(\text{pH}, T) = -1.227 \times T + 1000.549 \quad (3.18)$$

que, considerando una referencia de 25°C, se puede reescribir como en (3.19):

$$V(\text{pH}, T) = \underbrace{-1.227}_{m(\text{pH}4)} \times (T - 25) + \underbrace{969.874}_{V_{\text{cal}}(\text{pH}4)} \quad (3.19)$$

La segunda prueba se efectúa con un buffer de pH 7, obteniéndose los gráficos mostrados en Fig. 3.25(a) y (b) para la temperatura y la tensión, respectivamente. Igual que en el caso anterior, se sitúa el prototipo en una incubadora y luego se procede a efectuar cambios progresivos de temperatura, en este caso desde 25°C hasta 42.5°C con pasos de 2.5°C. La prueba tuvo una duración aproximada de 8 horas con un total de 200000 muestras. En la gráfica de la tensión de salida mostrada por Fig. 3.25(b), se aprecia también una tendencia decreciente respecto al aumento de la temperatura, lo cual es visualizado con mayor precisión en Fig. 3.25(c), obteniéndose una curva de regresión lineal con la ecuación mostrada en (3.20):

$$V(\text{pH}, T) = -0.868 \times T + 1144.904 \quad (3.20)$$

que reescrita respecto a la temperatura ambiente, se convierte en la ecuación (3.21):

$$V(\text{pH}, T) = \underbrace{-0.868}_{m(\text{pH}7)} \times (T - 25) + \underbrace{1123.204}_{V_{\text{cal}}(\text{pH}7)} \quad (3.21)$$

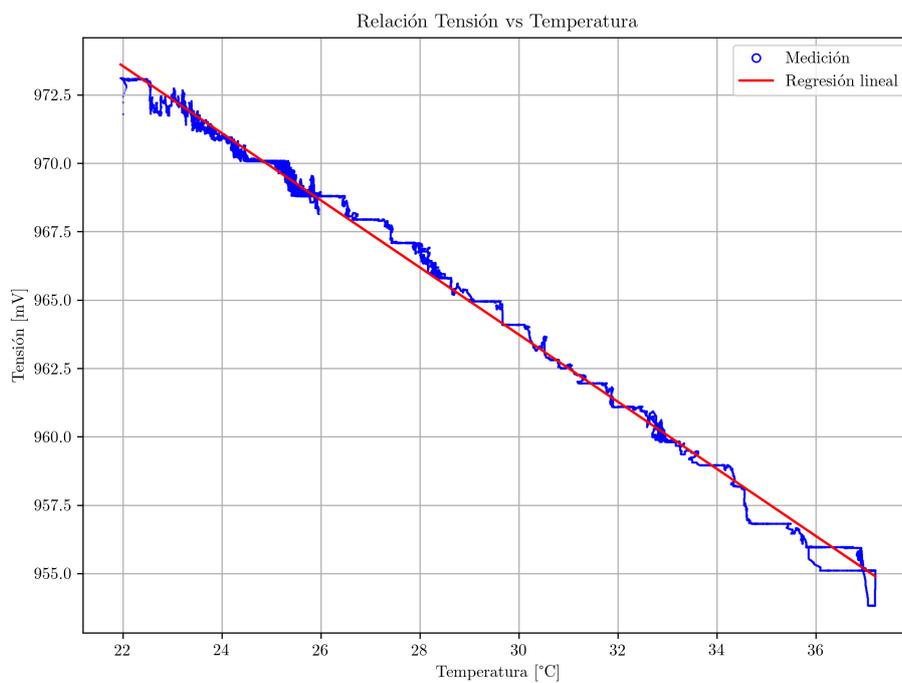
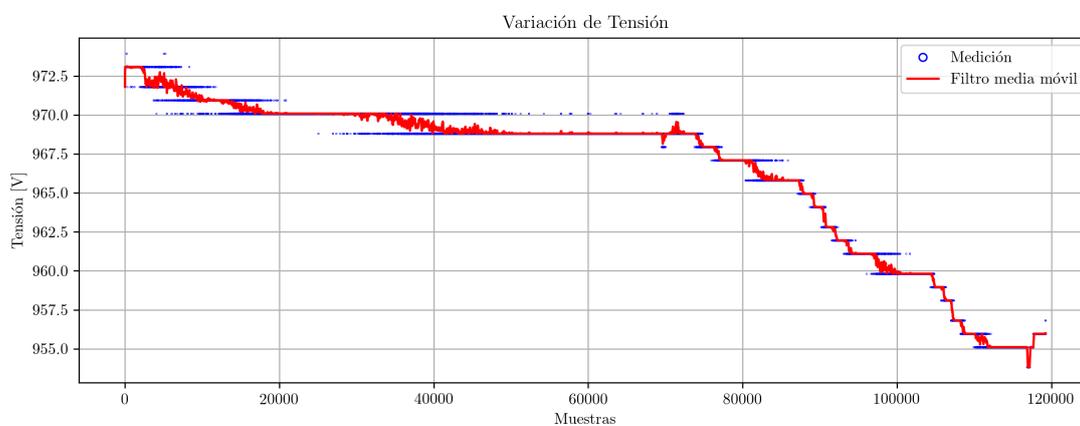
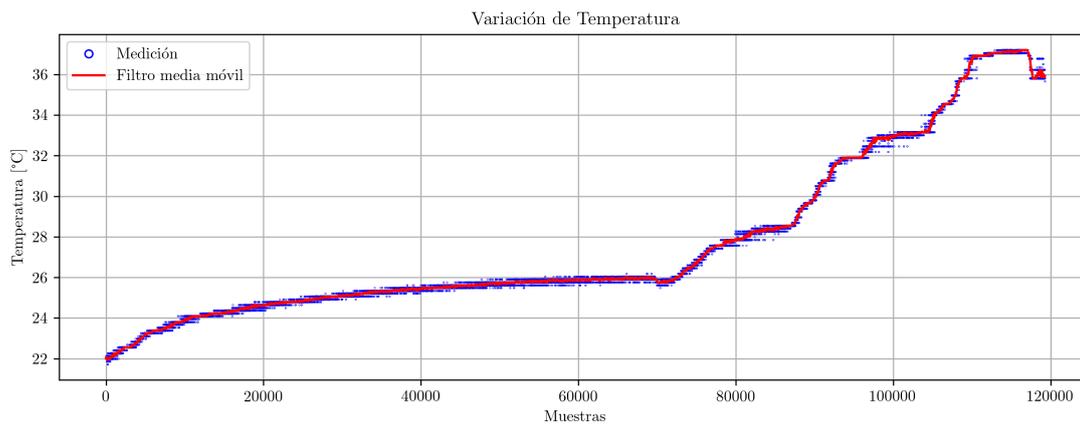


Fig. 3.24: Prueba de temperatura con buffer de pH 4.

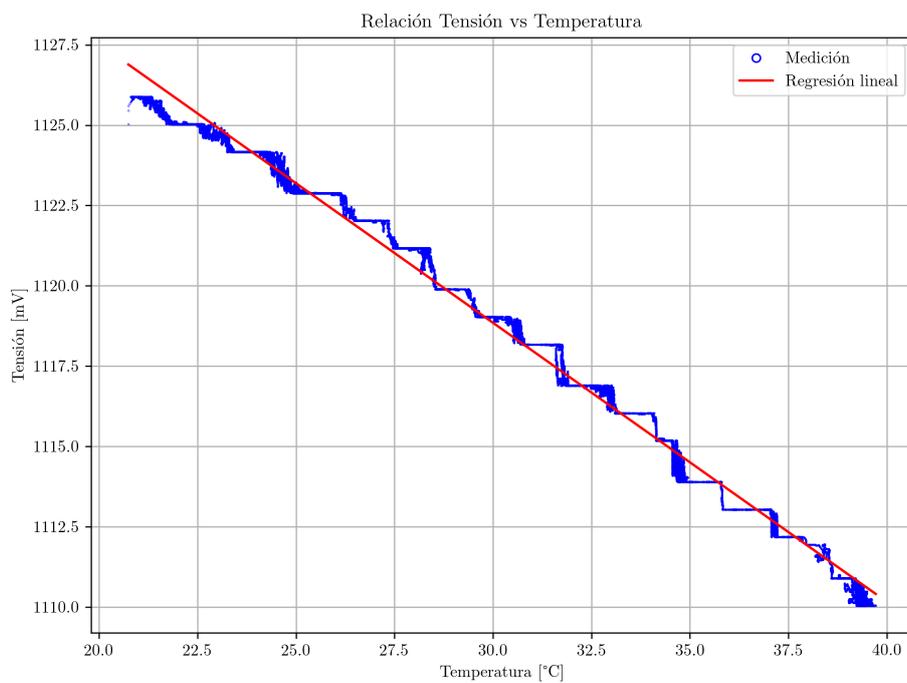
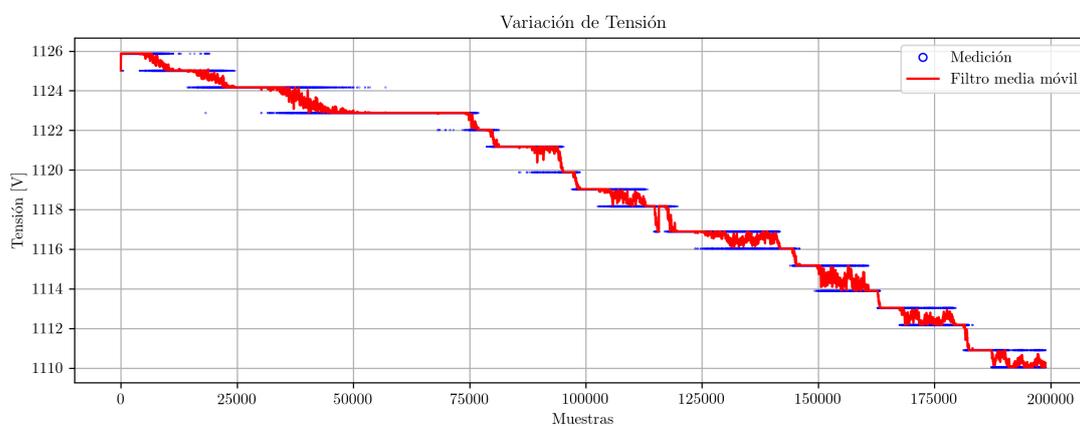
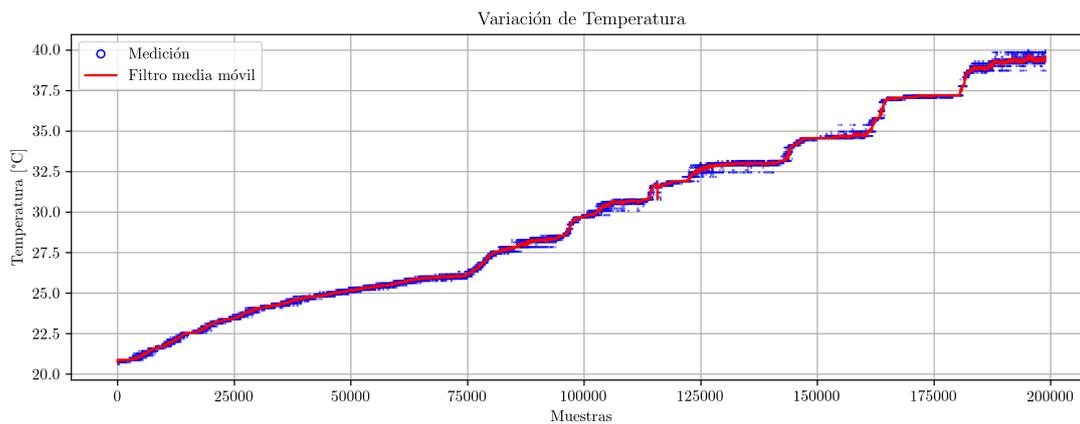


Fig. 3.25: Prueba de temperatura con buffer de pH 7.

En la ecuaciones (3.19) y (3.21) se pueden identificar los términos correspondientes a las tensiones de calibración V_{cal} y las pendientes m , los cuales tienen una dependencia respecto al pH. Entre estos valores se puede realizar una interpolación para obtener los valores intermedios a través de las ecuaciones (3.22) y (3.23):

$$\begin{aligned} V_{cal}(pH) &= V_{cal}(pH4) + \frac{V_{cal}(pH7) - V_{cal}(pH4)}{7 - 4}(pH - 4) \\ &= 969.874 + \frac{1123.204 - 969.874}{7 - 4}(pH - 4) \\ &= 765.474 + 51.1 \times pH \end{aligned} \quad (3.22)$$

$$\begin{aligned} m(pH) &= m(pH4) + \frac{m(pH7) - m(pH4)}{7 - 4}(pH - 4) \\ &= -1.227 + \frac{-0.868 + 1.227}{7 - 4}(pH - 4) \\ &= -1.707 + 0.12 \times pH \end{aligned} \quad (3.23)$$

De la ecuación (3.22), se identifica que el sensor tiene una sensibilidad, a temperatura ambiente, de 51.1mV/pH, lo cual es bastante cercano a los datos técnicos del sensor [68]. Así mismo se identifica de la ecuación (3.23) una tasa de cambio del coeficiente de temperatura a razón de 0.12 mV/(°C pH).

Por otra parte, los resultados obtenidos en (3.22) y (3.23) se pueden combinar en una sola ecuación para obtener la relación de la tensión del circuito de acondicionamiento respecto al pH y la temperatura, tal como se muestra en (3.24).

$$\begin{aligned} V(pH, T) &= m(pH) \times (T - 25) + V_{cal}(pH) \\ &= (-1.707 + 0.12 \times pH) \times (T - 25) + (765.474 + 51.1 \times pH) \end{aligned} \quad (3.24)$$

Despejando el término correspondiente al pH, se obtiene la ecuación (3.25):

$$pH = \frac{V(pH, T) - 765.474 + \overbrace{1.707(T - 25)}^{\text{Compensación tensión}}}{51.1 + \underbrace{0.12(T - 25)}_{\text{Compensación sensibilidad}}} \quad (3.25)$$

Esta es la ecuación que debe emplearse para calcular el valor de pH que tendría la solución a temperatura ambiente (25°C). De (3.25) también se obtiene que el resultado del pH sin compensar, mostrado en (3.26):

$$pH = \frac{V(pH) - 765.474}{51.1} \quad (3.26)$$

3.3.3 Diseño del sistema de control

Uno de los objetivos fundamentales del sistema de sensado de pH es la reducción del consumo de energía. En el fundamento teórico se habían abordado diferentes técnicas para el diseño de bajo consumo. No obstante, se llegó a la conclusión que la estrategia aplicable al presente caso se encuentra en el nivel de sistema, y está orientada hacia la activación y desactivación periódica de los componentes de acuerdo al tiempo en que son utilizados, con la finalidad de optimizar el uso de la energía empleada.

El sistema cuenta con la unidad de adaptación de los sensores, la cual es implementada a través de los periféricos del microcontrolador, y la unidad de procesamiento, implementada por el microprocesador. Los periféricos pueden ser activados solo en el momento en que se requiera obtener una lectura de los sensores, no obstante, estos tiempos están directamente relacionados con sus tiempos de respuesta. De los sensores presentados, el que determina el tiempo de respuesta es el sensor de pH, que para el caso del ISFET tiene un tiempo de respuesta de 2 segundos [128]. Luego, se debe tener en cuenta la cantidad de muestras mínima para obtener un dato válido. De acuerdo a los resultados del procesamiento, esta cantidad de muestras es aproximadamente 100, por lo que se debe adquirir esta información antes de desactivarlo. Una vez desactivado, se inicia la fase de procesamiento, la cual da como resultado un dato válido que será transmitido posteriormente por el módulo de comunicaciones.

De acuerdo a este esquema se puede distinguir las siguientes fases:

1. *Inicio*. Corresponde a la fase de inicialización del CPU y periféricos, los cuales implementan el sistema de adaptación del sensor. Así mismo, en esta fase se ingresan las instrucciones previas al inicio de operación del sistema.
2. *Activación de sensores*. Esta fase corresponde a la primera del ciclo de operación del sensor. Se activan los sensores y se espera durante un periodo correspondiente a su tiempo de respuesta [128]. El sensor que determina este lapso es el de pH, por lo que se escoge un tiempo de 2 segundos.
3. *Lectura de los sensores*. En esta fase, mientras se mantiene activos los sensores, se realiza la lectura de los ADCs con la finalidad de obtener la información del pH y la temperatura. El tiempo durante el cuál se mantienen activos está en función de la cantidad de muestras necesarias para el procesamiento. De acuerdo al diseño anterior, la dimensión de muestras aceptado es de 500. Luego, tomando en cuenta la tasa de

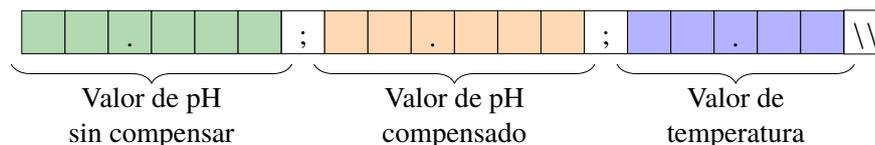


Fig. 3.26: Estructura de la trama

muestreo de los ADCs, se puede hacer una aproximación del tiempo requerido para estas muestras. Puesto que se realiza de forma secuencial, el tiempo sería calculado tal como se muestra en (3.27).

$$\begin{aligned} \text{Tiempo}_{\text{total}} &= \text{Muestras} \times (\text{Periodo ADC Sigma Delta} + \text{Periodo ADC SAR}) \\ &= 500 \times (1/10\text{k} + 1/100\text{k}) = 55\text{ms} \end{aligned} \quad (3.27)$$

4. *Procesamiento de datos.* En esta etapa se realiza el procesamiento de los datos de los ADCs. Primero se realiza la desactivación de los sensores y luego se aplican los filtros correspondientes a los datos obtenidos. El tiempo de procesamiento depende de la cantidad de instrucciones por la frecuencia del CPU. El resultado de esta etapa corresponde a los datos representativos de pH y la temperatura para el conjunto de muestras adquiridas. En esta etapa se incluye también el algoritmo de compensación, cuyo resultado ingresa como un valor adicional a los dos obtenidos del procesamiento.
5. *Transmisión.* En esta etapa se realiza el envío de datos obtenidos de la etapa anterior, a través del protocolo UART. El tiempo de envío se puede calcular a partir de la velocidad del UART, así como de la cantidad de datos. Empleando el valor calculado en (3.16) para un byte, igual a 104.16 μs , se puede estimar el total de acuerdo al número de bytes enviados. Considerando para el pH una precisión de tres dígitos decimales y para la temperatura de dos dígitos decimales, tal como se muestra en la Fig. 3.26, el tiempo empleado se expresa en los términos mostrados por (3.28):

$$\text{Tiempo} = 20 \times 104.16\mu\text{s} = 2.083\text{ms} \quad (3.28)$$

6. *Reposo.* Una vez que se ha transmitido toda la trama se establece el dispositivo en régimen de bajo consumo. El tiempo durante el cual se mantiene en este estado se puede variar dependiendo de la frecuencia con la que se requiera recibir estos datos. Como mínimo, el tiempo debe ser mayor al empleado por las etapas anteriores, tal

como lo muestra la ecuación (3.29):

$$\begin{aligned}
 \text{Tiempo ciclo} &> \text{Activación} + \text{Lectura} + \text{Procesamiento} + \text{Transmisión} \\
 &> 2s + 0.055s + 0.002083s + \text{Procesamiento} \\
 &> 2.057s + \text{Procesamiento}
 \end{aligned}
 \tag{3.29}$$

3.4 Diseño del sistema de comunicación inalámbrica

Dentro del diseño del sistema de comunicaciones se debe seleccionar el protocolo a emplear, el dispositivo donde se implemente el protocolo, el sistema operativo empleado para su implementación y la plataforma para enviar la información adquirida a Internet. A continuación se abordará la selección de cada una de estos componentes.

3.4.1 Selección del protocolo para Internet de las Cosas (IoT)

Los protocolos se pueden clasificar a partir del rango de operación, como se muestra en la Fig. 3.27, dividiendo las redes en: redes de proximidad, redes inalámbricas de área personal (WPAN), de área doméstica (WHAN), de área industrial (WFAN), de área local (WLAN), de área de campo (WNAN), de rango amplio (WWAN) y de baja potencia (LPWAN) [129], [130] ¹.

En este caso, la aplicación está orientada a redes de sensores, donde el rango de operación se encuentra dentro del WPAN, siendo las tecnologías inalámbricas más empleadas en este ámbito el Bluetooth, Zigbee, 6LoWPAN y Wi-Fi.

Por otra parte, los protocolos se encuentran relacionados entre sí de tal forma que, cuando se elige a uno de éstos, se deben emplear necesariamente otros para completar las capas de comunicación. Esto se puede visualizar para los protocolos escogidos a través de la Fig. 3.28. Allí se identifica que en la capa física y de enlace de datos, tanto el protocolo ZigBee como el

¹ Para el conjunto de protocolos formados por (WPAN,WHAN,WFN) y (WWAN,LPWAN) no existe una separación bien definida con relación al rango de operación, por lo cual se no se ha incluido en el gráfico los límites para estas redes.

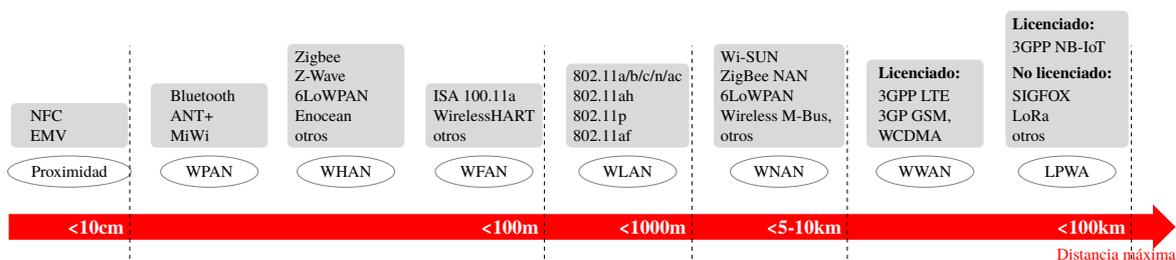


Fig. 3.27: Clasificación de protocolos de acuerdo al rango de operación [129], [130]

	Bluetooth	ZigBee	6LoWPAN	Wi-Fi
Aplicación	Aplicación/RFCOMM/ L2CAP/HCI	Perfil de aplicación	HTTP/CoAP	Capa de aplicación
Transporte	Administrador de enlace	Soporte de aplicación	UDP - ICMP	TCP - UDP
Red	Controlador de enlace	Capa de red ZigBee	IPv6/RPL/6LoWPAN	IP
Enlace de datos	Banda base	802.15.4 MAC	802.15.4 MAC	802.11ah MAC
Física	Radio	802.15.4 PHY	802.15.4 PHY	802.11ah PHY

Fig. 3.28: Estructura de red para los protocolos para IoT Bluetooth, Zigbee, 6LoWPAN y Wi-Fi [131], [132].

6LoWPAN están basados en el estándar IEEE 802.15.4, mientras que el Wi-Fi en uno de los estándares IEEE 802.11 y el Bluetooth en un propio estándar de radio y banda base. En la capa de red, Bluetooth y Zigbee cuentan con sus propios protocolos, los cuales son propietarios, mientras que 6LoWPAN y Wi-Fi emplean estándares desarrollados por la IETF (Grupo de Trabajo de Ingeniería de Internet), que también desarrolla los estándares para Internet.

Así mismo, se puede realizar una comparación entre los protocolos mencionados tal como se presenta en TABLA N° 3.7. De acuerdo a la tabla, se puede identificar algunas características que están directamente vinculadas con el protocolo de la capa física, tal como la banda de frecuencia, el ancho de banda, el tipo de modulación, velocidad de datos y longitud de paquete. Por otra parte, existen otras, las cuales dependen de los protocolos de capas superiores, tal como el rango teórico, la potencia nominal, carga útil y seguridad.

Evidentemente, existen diferencias entre los protocolos mencionados, con ventajas y desventajas en cada una de sus características. No obstante, es necesario realizar la elección basado en la aplicación y condiciones específicas en las cuales se va a implementar la solución. Uno de los criterios con mayor relevancia es el consumo de potencia, pudiendo sacrificar la capacidad de datos a ser transmitidos o su velocidad, puesto que la cantidad de información obtenida del nodo sensor es reducida. Entre las alternativas mostradas en TABLA N° 3.7, los protocolos basados en el estándar 802.15.4 ofrecen el mejor balance entre velocidad de transmisión y capacidad, además de contar con el menor consumo de potencia. Entre ellos, el que posee mejores características de flexibilidad y escalabilidad es 6LoWPAN. Esto debido a que, al estar basado en estándares de Internet, permite comunicarse con toda su infraestructura sin necesidad de emplear un *gateway* como sí lo requiere ZigBee, además que permite interoperabilidad con otros protocolos comunes a Internet como el caso

TABLA N° 3.7: Comparación de principales características de protocolos de comunicación inalámbrica [129], [131], [133], [134].

Tecnología	Bluetooth LE	ZigBee	6LoWPAN	Wi-Fi
ESTÁNDAR	Capa física			
CAPA FÍSICA	Bluetooth LE	IEEE 802.15.4	IEEE 802.15.4	IEEE 802.11ah ^a
BANDAS DE FRECUENCIA	2.4GHz	900MHz y 2.4GHz	900MHz y 2.4GHz	Sub 1GHz
ANCHO DE BANDA	2MHz	2 y 5MHz	2 y 5MHz	1,2,4,8,16 MHz
TIPO DE MODULACIÓN	FHSS con GFSK	DSSS con BPSK/QPSK	DSSS con BPSK/QPSK	QAM-OFDM con MIMO
VELOCIDAD DE DATOS	1Mbps	250kbps	250kbps	150kbps
LONGITUD DE PAQUETE	10 - 256 bytes	127 bytes	127 bytes	Hasta 7 991 bytes
RANGO TEÓRICO	100m	10 a 100 m ^b	25 a 50 m ^c	Hasta 1000m
POTENCIA DE TRANSMISIÓN	0.01 a 0.5W	52.2mW	2.23mW	19.95mW
CARGA ÚTIL (PAYLOAD)	20 bytes	102 bytes	102 bytes	2312 bytes
SEGURIDAD	128-bits AES con contador de modo CBC-MAC y capa de aplicación definida por el usuario	TLS 2.1 AES-128-CCM certificado X.509v3 y cifrado ECC-256	Capa de enlace AES TLS/SSL activado	Capa de seguridad de aplicación similar a 802.11

^aWi-Fi comprende un conjunto de estándares de la familia 802.11. En esta ocasión se abordará la variante 802.11ah la cual corresponde al protocolo para la banda de 900MHz [135].

^bDistancia para la banda de 2.4GHz.

^cDistancia para la banda de 2.4GHz. Otra referencia bibliográfica menciona un rango de 10 a 200m [136].

de IPv6, UDP o HTML. Por otra parte, es necesario mencionar que 6LoWPAN es la base para muchos otros protocolos derivados tales como Thread [137], empleado para automatización de viviendas; ISA100.11a [138], diseñado para automatización industrial; Wi-SUN [139], para comunicaciones de largas distancias e incluso el mismo ZigBee-NAN, con el mismo propósito. De esta forma, al emplear el protocolo 6LoWPAN también podría comunicarse con dispositivos de otras redes basadas en él y, por tanto, formar parte de una solución más grande.

3.4.2 Selección del dispositivo para comunicación inalámbrica

Una comparación de dispositivos de diferentes manufacturas empleadas para Zigbee y 6LoWPAN se puede apreciar en TABLA N° 3.8. De la tabla, se puede identificar que el dispositivo con menor consumo de corriente es el CC2630 fabricado por Texas Instruments,

TABLA N° 3.8: Comparación de dispositivos para implementación de protocolos basados en el estándar IEEE 802.15.4 [129].

Manufactura	Módulo	Protocolo de red	Frecuencia	V _{DD} [V]	I _{TX} [mA]	I _{RX} [mA]	I _{sleep} [μA]
ANS [140]	@ANY2400 SC	ZigBee, 6LoWPAN	2.4GHz	1.8 a 3.6	<15	<13	<1
MICROCHIP [141]	MRF24J40MA	ZigBee	2.4GHz	2.4 a 3.6	23	19	2
TEXAS INSTRUMENTS [142]	CC2630	ZigBee, 6LoWPAN	2.4GHz	1.8 a 3.8	9.1	5.9	1
DRESDEN ELEKTRONIK [143]	deRFmega256-23M00	ZigBee, 6LoWPAN	2.4GHz	1.8 a 3.6	19	16	<1
ST MICROELECTRONICS [144]	S2-LP	6LoWPAN, Sigfox, M-Bus	Sub 1GHz	1.8 a 3.6	10	7	0.6
RADIOCRAFTS[145]	RC2400	ZigBee	2.4GHz	2.0 - 3.6	34	24	0.4

el cual se posiciona como la alternativa más apropiada para la implementación de la comunicación inalámbrica. Así mismo, se prefiere emplear los dispositivos de Texas Instruments debido a su soporte oficial al protocolo 6LoWPAN, además su amplia documentación.

El integrado CC2630 forma parte de una familia de dispositivos inalámbricos denominada *SimpleLink™ ultra-low power MCU*. La plataforma SimpleLink™ se puede dividir en dos subfamilias de acuerdo a la frecuencia de operación: 26xx para 2.4GHz y 13xx para frecuencias por debajo de 1GHz [146].

En cada subfamilia existe un dispositivo que tiene la capacidad de implementar diferentes protocolos en el mismo integrado, lo cual facilita la prueba de las tecnologías inalámbricas, a la vez que cuenta con una plataforma de desarrollo para efectuar una evaluación de su funcionamiento, agilizando el proceso de implementación del sistema de transmisión inalámbrica. En este caso, se opta por emplear el dispositivo multiprotocolo de la familia de 2.4GHz (CC2650), cuyas características se muestran en la TABLA N° 3.9.

El dispositivo requiere de una alimentación entre 1.8 a 3.0V, así como de un circuito de polarización y una antena conectada a la terminal de RF. Afortunadamente, Texas Instruments provee de un módulo de evaluación (CC2650EM [148]) para este integrado, el cual reúne los componentes necesarios para su adecuado funcionamiento. Adicionalmente, también cuenta con una tarjeta de pruebas denominada SmartRF06EB [149], la cual se conecta con el módulo de evaluación y permite su comunicación directa con la PC, facilitando las tareas de verificación del firmware programado en el integrado. Tanto el módulo de evaluación como la tarjeta de evaluación forman parte del kit de desarrollo del integrado CC2650 [150], el cual posee dos unidades de cada uno (para transmisión y recepción), siendo lo mínimo necesario para efectuar la evaluación de la tecnología inalámbrica a implementar en el presente trabajo.

3.4.3 Selección del sistema operativo para implementación del protocolo

Existe una gran variedad de alternativas de sistemas operativos, los cuales están orientados a implementar la funcionalidad requerida por un dispositivo para IoT, considerando

TABLA N° 3.9: Características del circuito integrado para comunicaciones inalámbricas CC2650 [147].

MICROCONTROLADOR	Procesador Frecuencia Memoria	ARM® Cortex®-M3 48MHz 128kB de Flash 8kB de SRAM para cache 20kB de SRAM
CONTROLADOR DE SENSORES	Módulos Analógicos Interfaces	ADC de 12 bits y 8 canales a 200kS/s 2 comparadores de baja potencia Fuente de corriente programable Capacidad para 8 sensores capacitivos Interfaz digital SPI-I ² C
PERIFÉRICOS GENÉRICOS	Comunicaciones Temporizadores Pines entrada y salida	UART, I ₂ S 2 ISS (SPI, μ W, TI) 4 de 32bits 31 en 7x7mm QFN 15 en 5x5mm QFN 10 en 4x4mm QFN
RADIOFRECUENCIA	Capa física Frecuencia Sensibilidad Potencia	IEEE 802.15.4 2.4GHz -97dBm para BLE -100dBm para 802.15.4 +5dBm
ENERGÍA	Tensión Activo Standby Dormido	Normal: 1.8 a 3.8V Regulador: 1.7 a 1.95V RX: 5.9mA TX: 6.1mA (0dBm) 9.1mA (+5dBm) MCU: 61 μ A/MHz 1 μ A 100nA

las restringidas características que éstos poseen. Entre los principales requisitos que deben cumplir [151] se encuentran: una memoria reducida, capacidad de hardware heterogéneo, conexión de red, eficiencia de energía, capacidad de tiempo real y seguridad.

Una vez identificados los requisitos, es necesario comparar las alternativas disponibles basados en sus características. Las más relevantes para dicha comparación, de acuerdo a Gaur [152], son la arquitectura, el modelo de programación, la planificación (*scheduling*), la comunicación de red, la administración de memoria y protección y la portabilidad.

TABLA N° 3.10: Comparación de sistemas operativos para Internet de las Cosas [151], [152], [157].

Característica	Contiki OS	TinyOS	RIOT	Mantis OS ^a
ARQUITECTURA	Modular	Monolítica	Microkernel	Por capas
MODELO DE PROGRAMACIÓN	<i>event-driven prototheards</i>	<i>event-driven</i>	<i>multi-threading</i>	<i>multi-threading</i>
PLANIFICACIÓN	cooperativo	cooperativo	<i>preemptive sin ticks</i>	<i>preemptive</i>
COMUNICACIÓN DE RED	uIP, RIME	BLIP	gnrc, OpenWSN, ccn-lite	Capa COMM a nivel de kernel, capa de red a nivel de usuario
ADMINISTRACIÓN DE MEMORIA	Administración dinámica sin protección	Administración estática con protección	Administración dinámica	Administración dinámica sin protección
LENGUAJE DE PROGRAMACIÓN	Subconjunto de C	nesC	C, C++	C
CLASE DE DISPOSITIVO ^b	Clases 0 y 1	Clase 0	Clases 1 y 2	Clases 0, 1 y 2
FAMILIAS DE MCU PERMITIDOS	AVR, MSP430, ARM7, ARM Cortex-M, PIC32, 6502	AVR, MSP430, px27ax	AVR, MSP430, ARM Cortex-M, ARM7, x86	AVR, MSP430

^aLa última actualización de este sistema fue en 2013 y actualmente no se encuentra disponible en su página de descargas.

^bDentro de los dispositivos de gama baja, se establecen la clases: C0 para RAM \ll 10kB y ROM \ll 100kB, C1 para RAM \sim 10kB y ROM \sim 100kB, y C2 para RAM \sim 50KB y ROM \sim 250KB [158].

Entre los principales sistemas operativos existentes, se escoge un representante por cada arquitectura para efectuar la comparación de sus características, las cuales se pueden apreciar en TABLA N° 3.10. Se ha escogido a Contiki OS [153], como representante de la arquitectura modular; TinyOS [154], para la arquitectura monolítica; RIOT [155], para la arquitectura de microkernel; y Mantis OS [156], para la arquitectura por capas.

En la sección de protocolos y dispositivos, ya se había definido la plataforma de hardware sobre la cual se implementará el protocolo de comunicación inalámbrica, la cual es el CC2650EM con el protocolo 6LoWPAN. Este microcontrolador cuenta con un procesador ARM Cortex-M3. De los sistemas operativos mostrados en la TABLA N° 3.10, solo Contiki OS y RIOT soportan dicha arquitectura, por lo cual constituyen las dos alternativas viables. Por otra parte, considerando el consumo de energía, el sistema operativo con planificador cooperativo hace un uso mas eficiente de energía al permitir establecer el dispositivo en bajo

consumo a diferencia del esquema *preemptive*. Por otra parte, la propuesta de *protothreads* constituye la alternativa más apropiada en el uso de memoria, a la vez que facilita el desarrollo al emular el estilo de programación por hilos. Adicionalmente, Contiki OS es el sistema que provee mayor soporte para tecnologías de comunicación de bajo consumo. Por tanto, desde el punto de vista técnico, Contiki OS constituye la mejor opción para la implementación del protocolo. Por otra parte, desde el punto de vista no técnico, Contiki es uno de los sistemas operativos más maduros que se emplean en la industria [159], a la vez que es recomendado por el fabricante del dispositivo (Texas Instruments) [160].

3.4.4 Selección de la plataforma para comunicación con Internet

Las plataformas de IoT cuentan con un conjunto de características, entre las cuales destaca su arquitectura que puede estar basada en la nube o en un sistema local. En este contexto, la nube o computación en la nube, se refiere a un modelo que permite el acceso a un conjunto de recursos computacionales compartidos en red (servidores, redes, almacenamiento, aplicaciones y servicios), utilizados con facilidad sin una intervención directa del proveedor de servicios, por lo que una plataforma de IoT basada en la nube es aquella ofrecida por un proveedor de servicios, la cual facilita la implementación de aplicaciones de software sin el costo y complejidad de adquirir y administrar el hardware y software necesarios en capas inferiores de la infraestructura de red [161]. La mayoría de plataformas de IoT existentes son ofrecidas a través de servicios en la nube, por lo cual, en este caso, se centrará la atención en las plataformas de este tipo.

Las plataformas de IoT basadas en la nube son diversas, por lo cual escoger una en particular resulta complicado. Ray en [161] propone una forma de abordar su estudio agrupándolas en dominios de aplicación, los cuales son: desarrollo de aplicaciones, administración de dispositivos, administración del sistema, sistemas heterogéneos, administración de datos, análisis, administración del desarrollo, administración del monitoreo, visualización e investigación.

Las plataformas disponibles pueden aplicarse en varios de los escenarios mencionados anteriormente, no obstante, solo algunas de ellas son las más adecuadas para cada ítem. Basado en esto, se pueden encontrar algunas alternativas para cada categoría tal como lo muestra Fig. 3.29. De éstas, adquieren mayor relevancia las correspondientes a visualización y administración de monitoreo, destacando entre ellos Plotly [162] y ThingSpeak [163].

En efecto, existe un amplia variedad de plataformas para IoT, algunas comerciales que se encuentran fuertemente posicionadas en el mercado y son ofrecidas por grandes corpora-

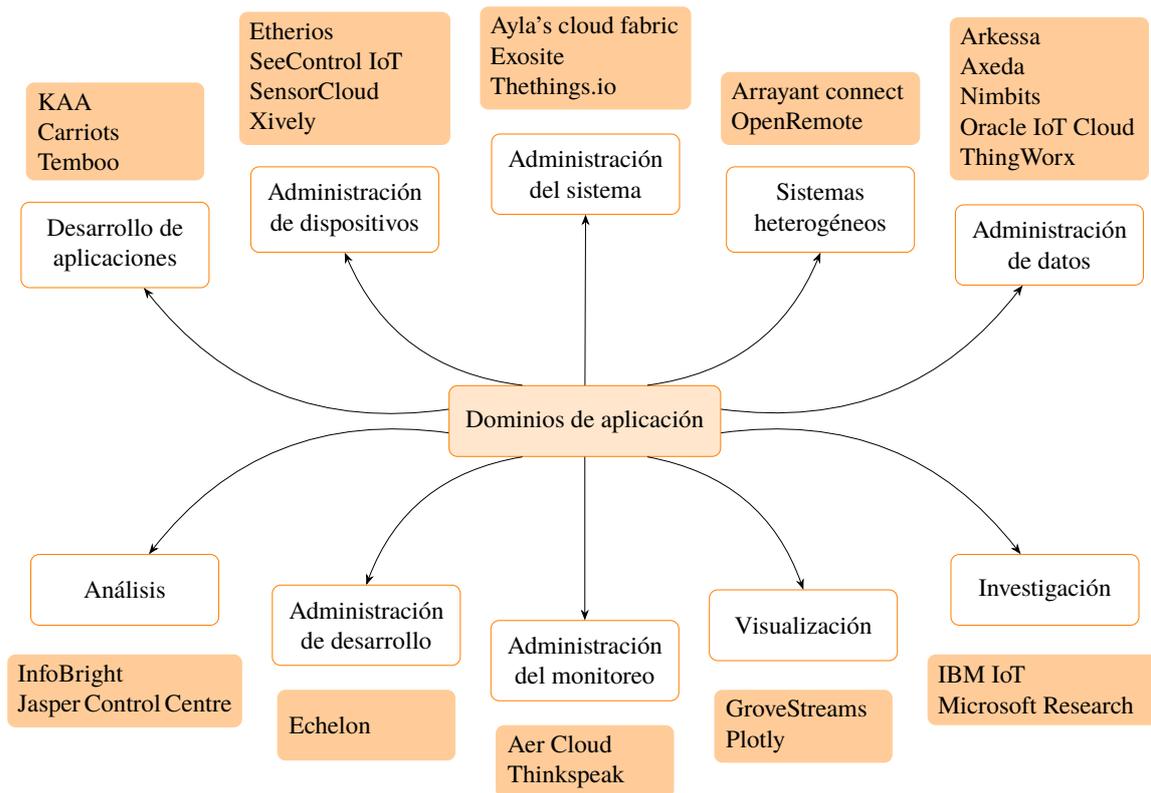


Fig. 3.29: Plataformas para IoT de acuerdo al dominio de aplicación [161].

ciones tales como Microsoft Azure for IoT, Amazon Web Services (AWS) for IoT, Google Cloud for IoT; otras también comerciales conocidas en el ámbito de desarrolladores tales como AdaFruit.io, Bluemix (IBM), Ubidots, entre otros [164]; y también algunas alternativas de código abierto y gratuitas. Se hace un especial énfasis en estas últimas debido a que se consideran más prometedoras que las alternativas propietarias, dada su integración más rápida con nuevas tecnologías de IoT, su continuo mantenimiento y rentabilidad [165]. Entre las principales plataformas de código abierto se encuentran Kaa, Lelylan, ThinkSpeak y The ThingBox [166].

Por tanto, para la implementación de la conexión con Internet existen varias opciones disponibles, las cuales son equiparables en las características ofrecidas. Puesto que el único objetivo del empleo de alguna de estas plataformas es la prueba de la conexión a Internet del sistema diseñado, se puede utilizar cualquiera de las alternativas presentadas, siendo las más convenientes las de código abierto o los planes de prueba de algunas de las soluciones comerciales.

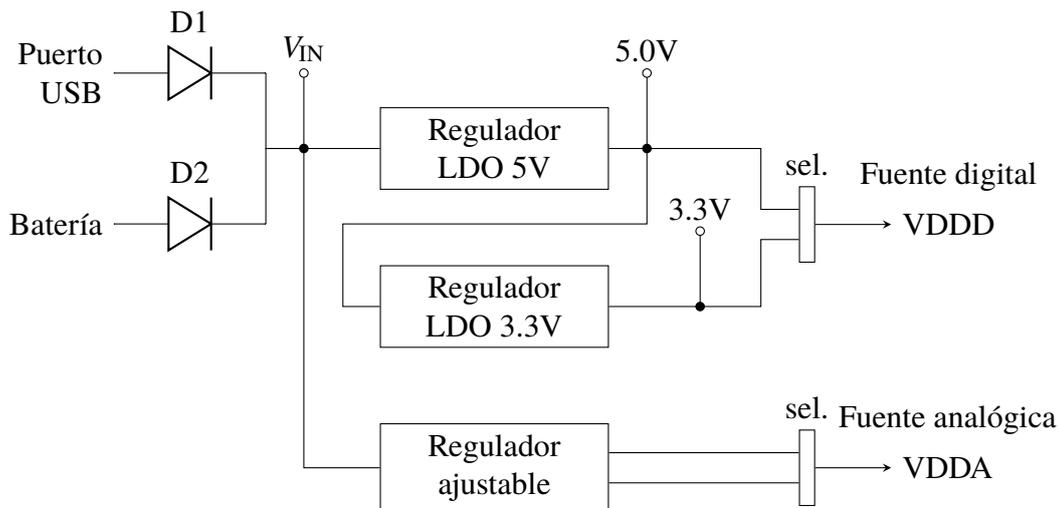


Fig. 3.30: Sistema de energía para el PSoC.

3.5 Diseño del sistema de energía

El diseño del sistema de energía del microcontrolador PSoC 5LP se basa en el sistema de energía empleado para la tarjeta de desarrollo CY8CKIT-050 [167]. Para el sistema se plantea emplear dos alternativas de fuente de alimentación, una mediante una batería y otra mediante un conector tipo USB.

Estas dos fuentes pueden estar conectadas al mismo tiempo, por lo que se emplean diodos a la entrada de ambas, habilitando solo la de mayor tensión. De esta manera, solo se selecciona una de las fuentes, la cual constituye la entrada de tensión para el sistema de energía V_{IN} . Puesto que se desea proporcionar dos dominios de tensión para el microcontrolador, uno para el subsistema analógico y otro para el subsistema digital, el circuito de alimentación cuenta con dos salidas: VDDA y VDDD, para cada uno de los dos subsistemas, respectivamente. Así mismo, basado también en el diseño de la tarjeta de desarrollo empleada en las pruebas del sistema de acondicionamiento del sensor, se incluye la posibilidad de selección de dos tensiones diferentes de alimentación para VDDD, una de 5V y otra de 3.3V. Estas tensiones son proporcionadas por dos reguladores lineales LDO con tensión de salida fija. Por otra parte, para la tensión analógica VDDA también puede elegirse entre dos opciones, solo que, en este caso, se hace mediante un regulador de tensión ajustable. El esquema del sistema de energía planteado se puede apreciar en Fig. 3.30. Para alimentar el microcontrolador que efectúa la comunicación inalámbrica, simplemente se conecta las terminales de la tensión de entrada a la salida del subsistema digital del sistema de energía empleado para el PSoC 5LP.

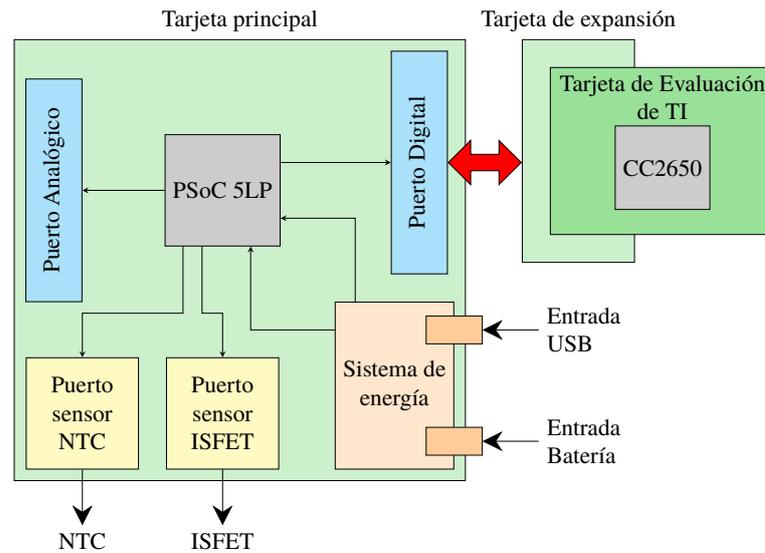


Fig. 3.31: Diseño del sistema global

3.6 Diseño del sistema global

Para el sistema global se plantea el diseño de una tarjeta en circuito impreso (PCB), la cual interconecte los sensores al microcontrolador, además de implementar el sistema de energía encargado de alimentar al prototipo. El sistema de energía cuenta con dos entradas, tal como se había diseñado previamente, las cuales corresponden a una entrada tipo USB y una entrada para batería. Por otra parte, el microcontrolador cuenta con dos puertos, uno analógico y otro digital, para sus pines de entrada y salida analógicos y digitales, respectivamente.

Para la comunicación inalámbrica se plantea emplear la tarjeta de evaluación del microcontrolador CC2650, de Texas Instruments, la cual cuenta con dos puertos de conexión para su alimentación y comunicación con otros dispositivos. A fin de poder comunicar esta tarjeta con la tarjeta principal se plantea diseñar una tarjeta de expansión, la cual se monta sobre el puerto digital de la tarjeta principal y en cuya superficie superior se cuenta con los conectores para adherir la tarjeta de comunicaciones.

El esquema del sistema completo se puede apreciar en Fig. 3.31. Tal como se muestra en la figura, tanto el sistema de energía como los componentes analógicos y digitales formarán parte de la misma tarjeta. No obstante, aún siendo parte del mismo conjunto, se pueden diseñar por bloques, con las consideraciones necesarias para que se complementen en su funcionamiento. Estos bloques se distinguen en cada una de las etapas de diseño de la tarjeta, desde la especificación de los componentes hasta la implementación del PCB.

3.7 Resumen del capítulo

El sistema de medición empleado para la caracterización del sensor basado en ISFET, consta de tres componentes: hardware, software para el microcontrolador y software para la PC, el cual, a su vez, comprende una etapa de adquisición y una etapa de procesamiento. Luego de caracterizar el ISFET se diseña el sistema de sensado, el cual está conformado, en su primera etapa, por el sistema de acondicionamiento, basado en la estructura clásica CVCC y diseñado empleando los bloques internos del PSoC 5LP. Así mismo, se incorpora al sistema un sensor de temperatura y se utiliza un ADC Sigma-Delta para la conversión al dominio digital del pH y un ADC SAR para la temperatura. El sistema de procesamiento está conformado por dos filtros, uno de mediana de orden 13 y un filtro media móvil de orden 100. Así mismo, se logra un sistema de compensación a partir de curvas halladas luego de caracterizar el sensor respecto a los cambios temperatura. Por otro lado, el sistema de control comprende etapas de activación y de reposo para la optimización de la energía empleada en las lecturas de los sensores. El sistema de comunicación inalámbrica se diseña a partir de seleccionar cada uno de sus componentes. De esta manera, se escoge como protocolo de comunicación al 6LoWPAN, el cual se implementa en el dispositivo CC2650. Así mismo, se escoge el sistema operativo Contiki OS para la implementación del protocolo, y se opta por soluciones libres y de evaluación para la plataforma de IoT encargada de enviar los datos a Internet. Finalmente, en el sistema de energía, se emplean reguladores para generar los dos dominios de tensión que alimentarán al subsistema digital y analógico del microcontrolador.

En el presente capítulo se ha abordado el diseño del sistema de caracterización del ISFET, el sistema de acondicionamiento, de procesamiento, de transmisión inalámbrica y de energía para el prototipo de sensado de pH. El diseño efectuado sirve de base para la implementación de los respectivos componentes del sistema propuesto, proceso que se detallará en el siguiente capítulo.

CAPÍTULO 4

IMPLEMENTACIÓN DEL SISTEMA DE SENSADO DE PH

Luego de efectuar el diseño del sistema de sensado de pH, se procede a la implementación de cada una de las partes que lo constituyen, comenzando por el sistema de caracterización del sensor, luego, el sistema de acondicionamiento, de procesamiento, comunicación inalámbrica y, finalmente, la implementación del prototipo en una tarjeta de circuito impreso.

4.1 Implementación del sistema para caracterización del ISFET

El sistema de caracterización está constituido por tres componentes: el hardware, el software para el microcontrolador y el software para la PC, conformado, a su vez, por una componente de adquisición y otra de procesamiento. No obstante, tanto el hardware, el software para el microcontrolador y el software de adquisición se pueden considerar con una subunidad denominada sistema de adquisición, mientras que el software de procesamiento se considera como otra subunidad, en función a lo cual se realiza la implementación.

4.1.1 Implementación del sistema de adquisición

El componente de hardware empleado para la implementación está conformado por el microcontrolador STM32F407VGT6 [168] parte de la tarjeta *STM32 F4 Discovery* [169].

La herramienta de software empleada para programar este microcontrolador es el *System Workbench for STM32* [170] de AC6. Se han empleado también para este propósito los controladores HAL (capa de abstracción de hardware) [171] y sus ejemplos [172], parte del paquete STM32F4Cube [173].

Adicionalmente, se ha diseñado una tarjeta auxiliar para convertir la corriente, producida por la interacción entre la solución y el ISFET, en una señal de tensión. El circuito empleado está basado en un TIA, el cual está conformado por un amplificador operacional LM324 y una resistencia en realimentación (R_f), la cual es seleccionada en función de las características del ISFET tales como su corriente máxima (80 μ A) y tensión máxima (2.0V). Así mismo, considerando que el sistema va a ser operado por una batería de 3.6V, la salida de tensión del

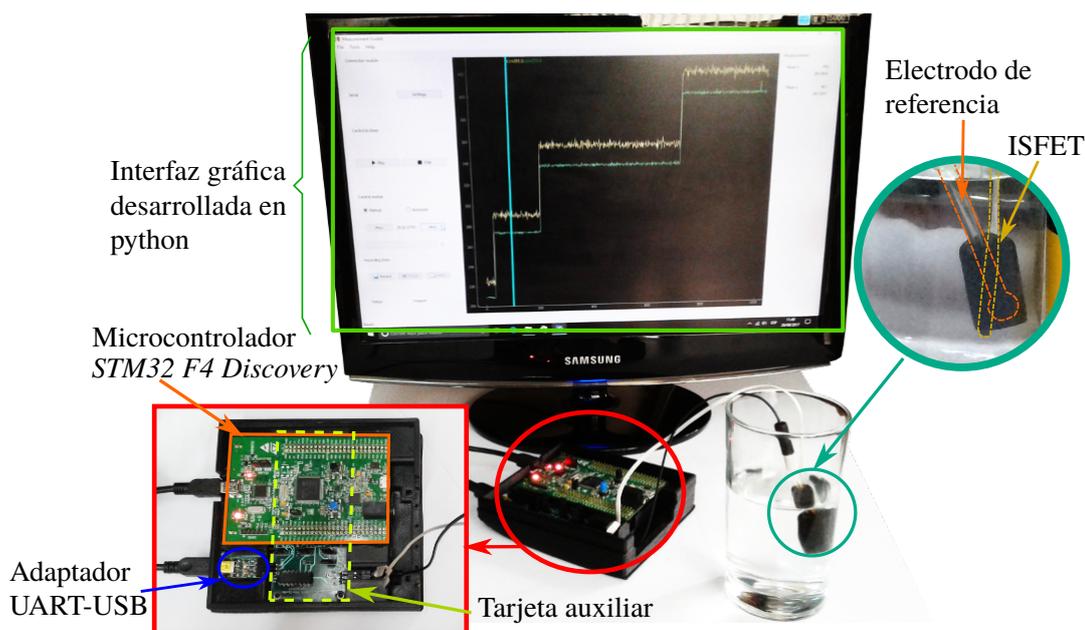


Fig. 4.1: Implementación del sistema de medición para la caracterización del ISFET.

TIA debe ser inferior a este valor. En el peor de los casos la tensión sería $2.0 + 80\mu\text{A} \times R_f$ obteniéndose un valor aproximado de $20\text{k}\Omega$.

La implementación del software se realizó a través de la distribución Python(x,y), la cual incluye las librerías necesarias para cálculos numéricos tales como numpy. La interfaz fue implementado a través de las librerías PyQt4. Adicionalmente, se ha empleado el módulo pyqtgraph para la ventana de monitoreo.

El test del sistema se efectúa a partir de conectar las terminales de los DACs en las terminales de compuerta y drenador del ISFET, mientras que los ADCs se conectan a los terminales de la compuerta o drenador (dependiendo del tipo de prueba) y en la terminal de salida de tensión del transductor de corriente.

En la Fig. 4.1 se puede apreciar la implementación del sistema en donde se muestran las tarjetas empleadas, tanto de STMicroelectronics como la tarjeta auxiliar diseñada, la interconexión entre éstas y su conexión con el sensor ISFET y electrodo de referencia, los cuales se encuentran en una solución de pH conocido. Así mismo, dado que la comunicación es mediante UART, se emplea un adaptador a puerto de bus universal en serie (USB) para poder conectarlo a la computadora. El software mostrado en la pantalla se encarga de reconocer todos los puertos seriales y, luego de seleccionar el puerto correcto y realizar una sencilla configuración por parte del usuario, se puede dar comienzo a la lectura del sensor.

El software implementado se puede apreciar en la Fig. 4.2, donde se muestran cada una

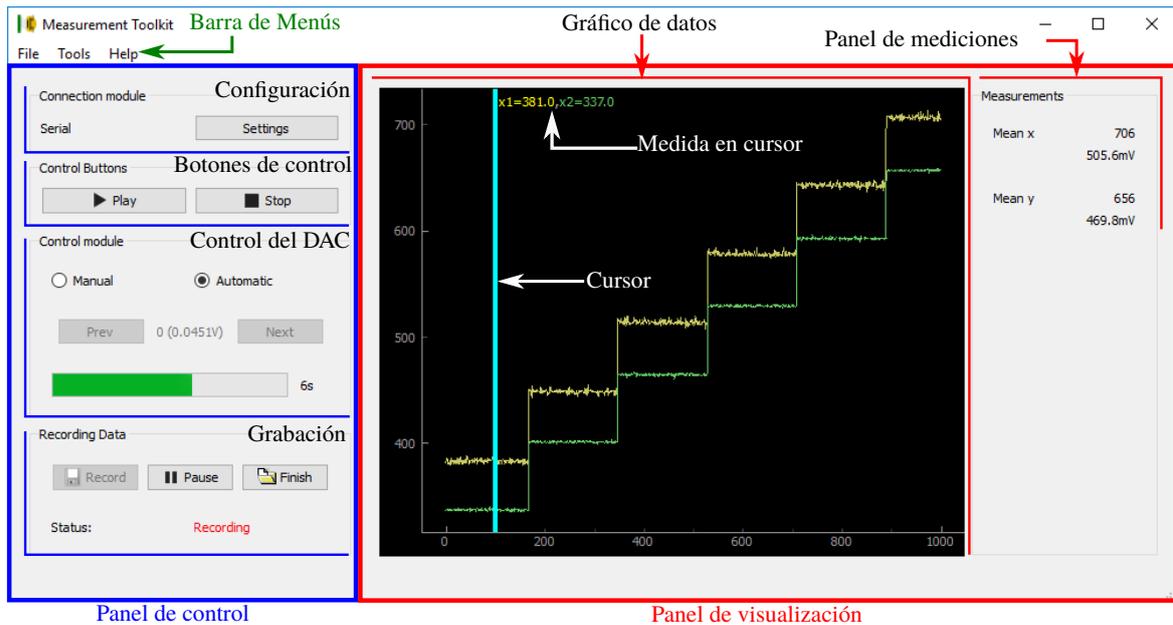


Fig. 4.2: Interfaz gráfica implementada en Python.

de las partes que se han considerado en su diseño. La pantalla principal está dividida en dos regiones: una correspondiente al panel de control y otra, al de visualización. En el caso del panel de control, éste cuenta con cuatro bloques separados donde el primero corresponde a la configuración del puerto serial; el segundo al control de inicio y detención del proceso de lectura; el tercer bloque son los controles del DAC en donde se aprecian los modos manual y automático así como los controles para el cambio de valor al previo o al siguiente; y el cuarto bloque comprende los controles de grabación a través de los cuales se dan las órdenes para iniciar la grabación, suspenderla y finalizarla, abriendo un cuadro de diálogo donde se escoge el nombre con el cual se guardarán los datos obtenidos, almacenados en formato de texto plano separado por comas. En el panel de visualización se encuentran dos bloques: uno correspondiente al gráfico de los datos, donde se muestra casi en tiempo real la información en forma gráfica de las mediciones efectuadas; y otro correspondiente al panel de mediciones, donde se aprecia un promedio de las últimas 100 muestras tomadas para cada una de las señales de entrada. En el bloque del gráfico también se cuenta con un cursor, el cual puede situarse en cualquier región del lienzo y muestra el valor de las señales en la posición donde éste se sitúe. La barra de menús es bastante sencilla y cuenta con solo tres elementos, el primero correspondiente al menú *File*, a través del cual se puede salir de la aplicación; el menú *Tools*, a través del cual se puede activar o desactivar los paneles; y el menú *Help*, a través del cual se accede a la información del software.

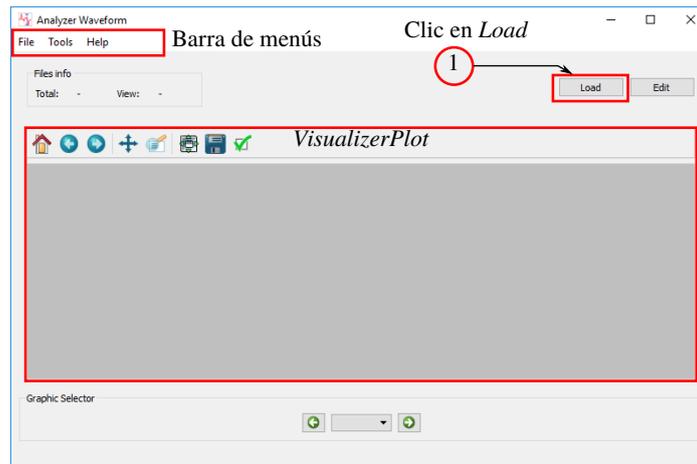


Fig. 4.3: Ventana de inicio de software de procesamiento

4.1.2 Implementación del software de procesamiento

El software diseñado en el capítulo anterior, empleado para el tratamiento de la información, ha sido implementado en Python, empleando para ello las librerías *numpy* y *scipy* para el módulo *DataProcessor*, y las librerías *PyQt4* para el GUI, a la vez de la librería *matplotlib* para generar las gráficas en los diferentes módulos de la interfaz.

La ventana de inicio del software es tal como se muestra en Fig. 4.3. Allí se puede apreciar la región principal de la aplicación, donde se muestran la barra de menús, el panel de visualización (*VisualizerPlot*) y de selección de gráficos (*GraphicSelector*), deshabilitados hasta que se cargue la información a través del botón *Load*.

Una vez cargados los datos de las curvas características, la información correspondiente al resultado del procesamiento se presenta en el panel de visualización mostrado en la Fig. 4.4. Luego, la información que ha sido tratada para obtener cada uno de los puntos de la curva se puede apreciar en el panel de análisis (*AnalyzerDock*) el cual se abre automáticamente y se sitúa a la derecha del panel principal. En este panel es donde se aprecia la información de un archivo en particular, a partir del cual se genera una de las familias de curvas.

En *AnalyzerDock* se puede seleccionar qué gráficas del proceso de tratamiento de información se desean mostrar, ya sea la información original a través de *RawData*, los datos filtrados a través de *FilteredData* o la máscara a través de *Mask*, los cuales se puede acceder en el panel de opciones de visualización en la parte superior de la gráfica (*AnalyzerPlot*). Adicionalmente, al trasladarse con el puntero a través de la gráfica se sombrea la región correspondiente al paso, cuyo valor en el panel de visualización, aparece resaltado mediante un indicador amarillo.

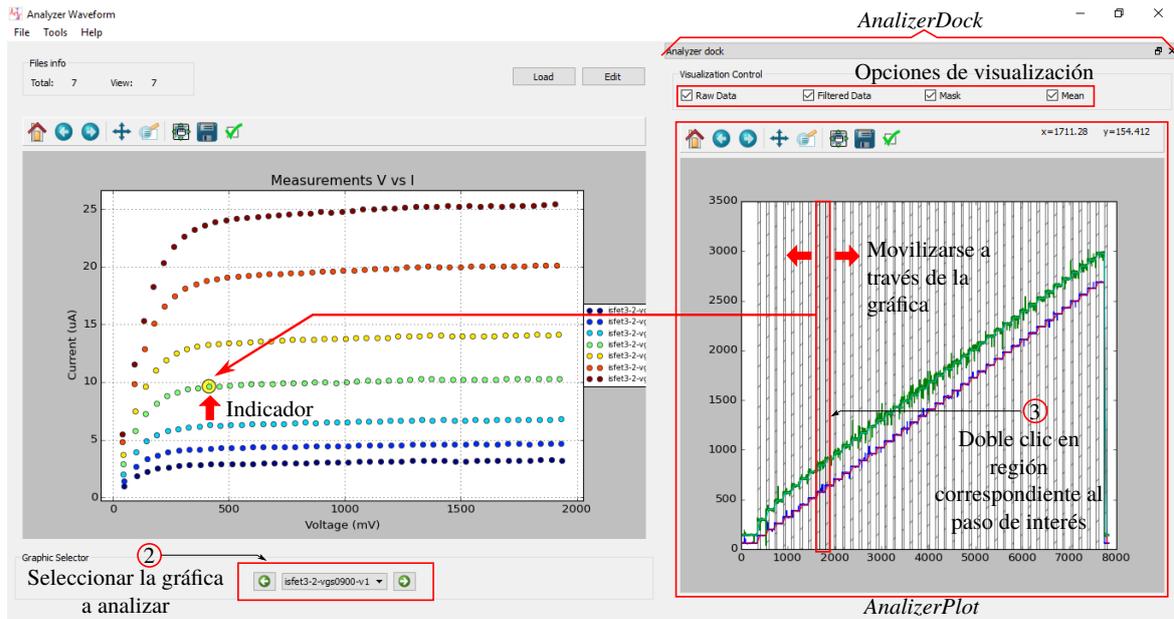


Fig. 4.4: Ventana del software luego de cargar los archivos, los cuales al ser procesados generan las curvas características.

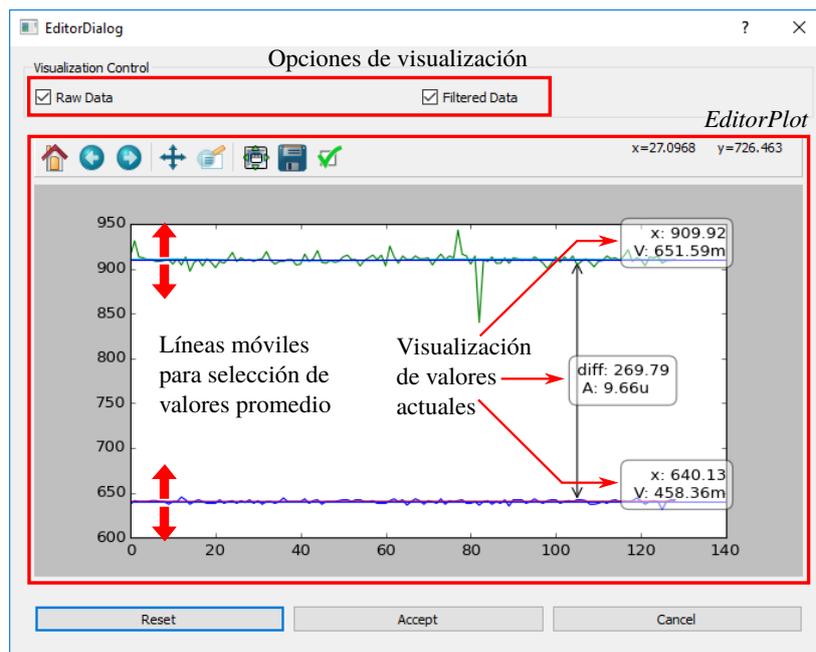


Fig. 4.5: Cuadro de diálogo para la edición de los valores representativos de uno de los puntos de la curva característica seleccionada.

Las regiones correspondientes a los puntos de la curva característica, se muestran en el cuadro de diálogo (*EditorDialog*), a través del cual se visualiza la región de la gráfica comprendida en el paso de interés, y cuyos elementos se describen en la Fig. 4.5. A través de este cuadro de diálogo se puede visualizar y ajustar manualmente el valor de los valores representativos para cada segmento de la curva.

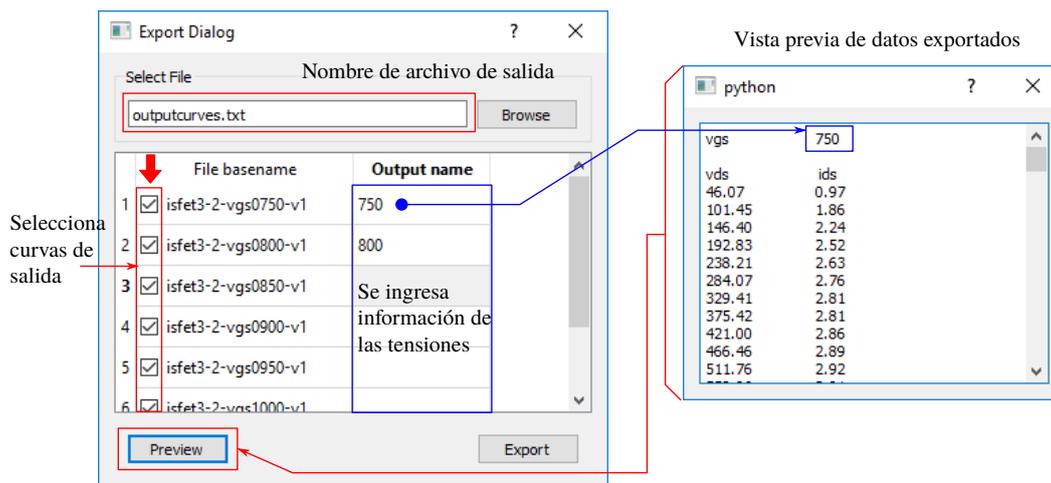


Fig. 4.6: Ventana de inicio de software de procesamiento

Los valores modificados en esta ventana se actualizan en el panel de visualización principal *VisualizerPlot*. Una vez terminado el ajuste de los valores representativos de las curvas, la información obtenida se puede exportar como un archivo de texto para su posterior utilización. El procedimiento se realiza a través del cuadro de diálogo *ExportDialog*, el cual se muestra en la Fig. 4.6.

4.2 Implementación del sistema de acondicionamiento de señal

El circuito diseñado en la sección 3.2.2 y validado con la simulación en 3.2.3, ahora se encuentra listo para su implementación en el microcontrolador. Para los microcontroladores de Cypress de la familia PSoC, se cuenta con una herramienta de desarrollo denominada PSoC Creator[174], también de Cypress, a través de la cual se efectúa este proceso.

Para el sistema de acondicionamiento se ha realizado la implementación en dos fases, una primera, efectuada en la tarjeta de desarrollo del microcontrolador, implementando únicamente el circuito de acondicionamiento del ISFET, y una segunda fase, donde se implementa todo el sistema completo, incluyendo al sensor de temperatura y los conversores ADCs, en el prototipo final.

4.2.1 Implementación del circuito de acondicionamiento del ISFET

La forma como se aborda este proceso en el software citado es como sigue:

- *Diseño de esquema del circuito mostrando las conexiones de sus bloques analógicos y periféricos.* Para el circuito planteado del sistema de adaptación del ISFET, el diseño de tope de jerarquía se puede apreciar en Fig. 4.7. Cada uno de los bloques de imple-

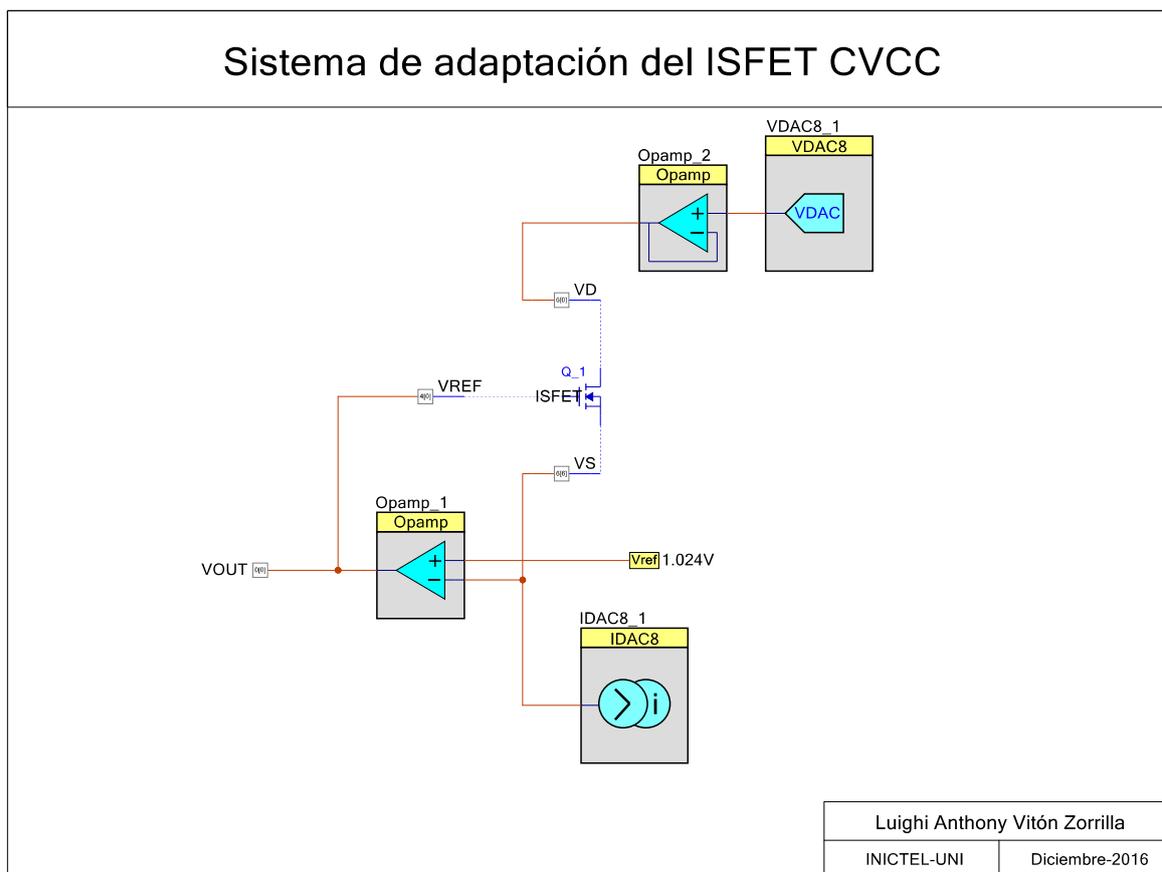


Fig. 4.7: Esquema de tope de jerarquía de la implementación del sistema de adaptación para el ISFET.

mentación tienen unos parámetros de configuración a través de los cuales se especifica sus características. En el caso de los OPAMPs se puede apreciar en Fig. 4.8. El primer OPAMP se encuentra en el lazo de realimentación y está configurado como Opamp y en modo de baja potencia. Esto debido a que su terminal no va a alimentar un ninguna carga . El segundo OPAMP, en cambio, se encuentra en el modo de seguidor de tensión y en mediana potencia puesto que este amplificador tiene que proporcionar la corriente necesaria para polarizar al ISFET. Por otra parte, las configuraciones del IDAC y del VDAC se pueden apreciar en Fig. 4.9. En el IDAC se escoge el tipo de fuente negativa debido a que la corriente está ingresando en el bloque, luego se escoge el rango en su escala original (x1), para finalmente ingresar el valor numérico equivalente a $25\mu\text{A}$. Así mismo, para el VDAC se selecciona el rango, en este caso, de 0-4.080V, y el valor de la tensión requerida en la terminal. Finalmente, para la tensión de referencia, se escoge el valor de 1.024V del conjunto de opciones disponibles en su ventana de configuración.

- *Asignación de pines en el microcontrolador PSoC.* En el presente caso, se cuenta

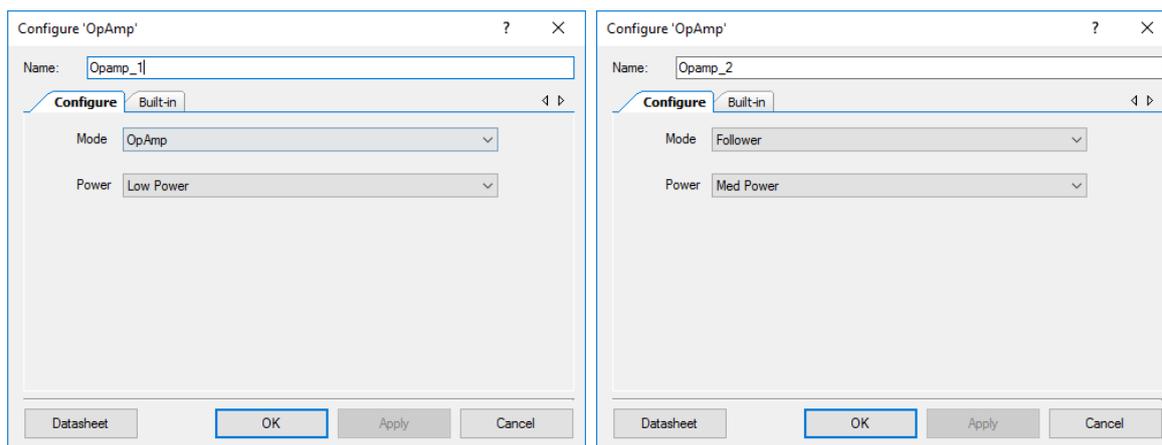


Fig. 4.8: Configuración de los opamps empleados en la implementación. Captura de pantalla obtenida del software PSoC Creator.

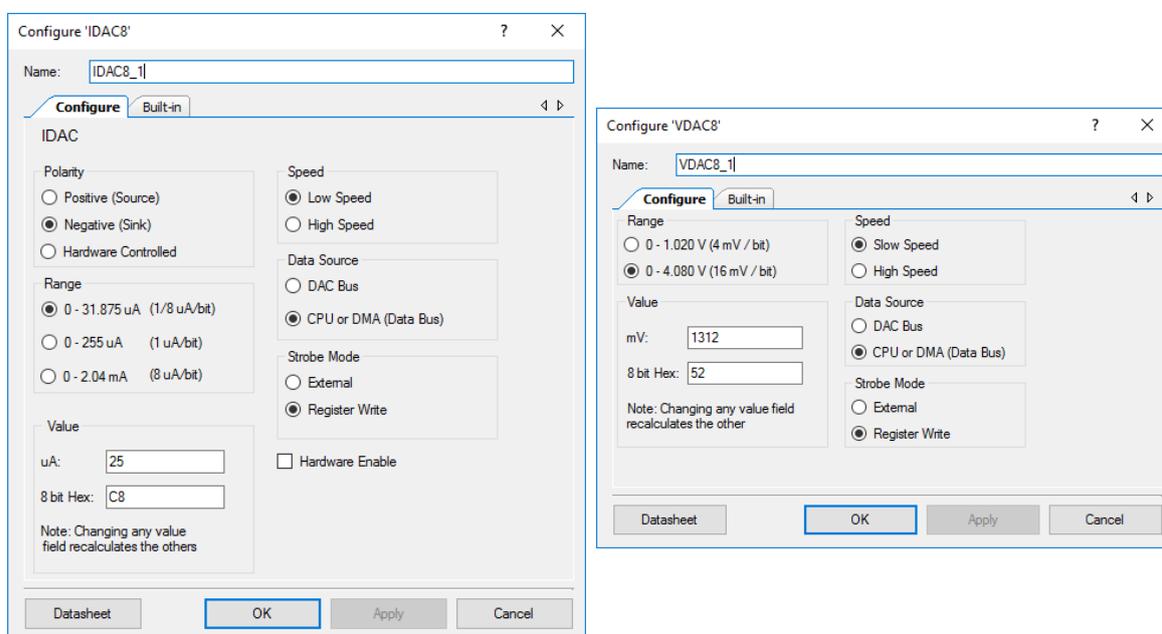


Fig. 4.9: Configuración del IDAC y VDAC empleados en la implementación. Captura de pantalla obtenida del software PSoC Creator.

con cuatro pines correspondientes a las terminales de drenador y fuente del ISFET, del electrodo de referencia y la salida de tensión del sistema de adaptación. Para la asignación se tiene en cuenta los puertos de entrada y salida que están orientados a conexiones analógicas [175]. Dado esto, se han empleado los puertos P6 y P0 para los pines del diseño. La asignación se puede apreciar en Fig. 4.10.

- *Archivo de programación del microcontrolador.* El esquema del diseño de tope de jerarquía (Fig. 4.7) requiere de instrucciones de inicialización para que los bloques puedan activarse. Estas instrucciones están relacionadas con los nombres de los bloques

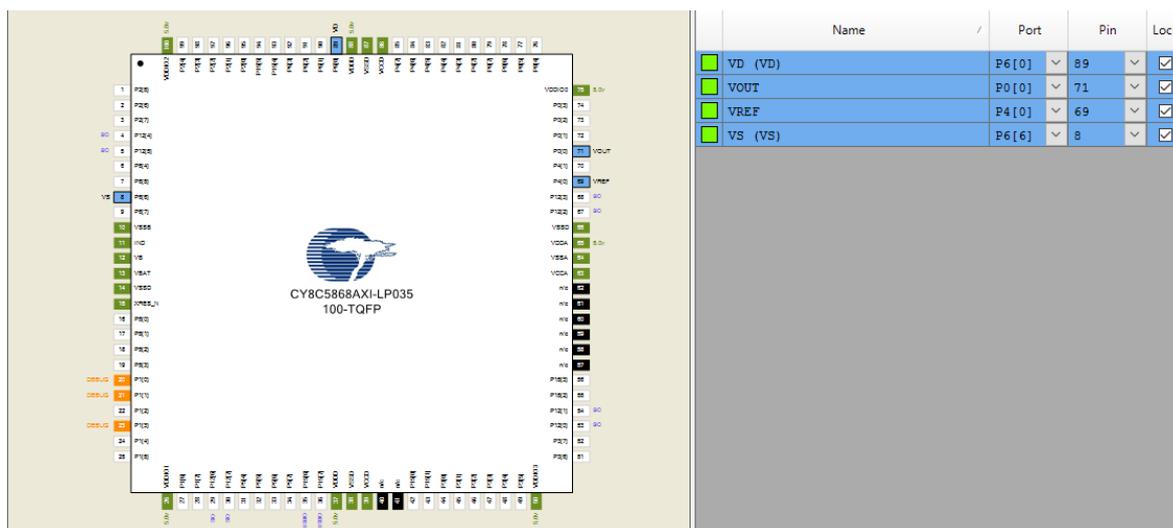


Fig. 4.10: Asignación de pines del microcontrolador para la implementación del sistema de adaptación. Captura de pantalla obtenida del software PSoC Creator.

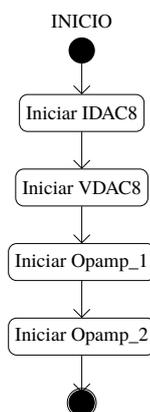


Fig. 4.11: Secuencia de inicialización de los bloques del sistema de adaptación.

empleados en el esquema. El orden de las instrucciones de inicialización se puede apreciar en Fig. 4.11.

Una vez realizados los pasos anteriores, lo único que resta es realizar la compilación del proyecto y descargarlo en la memoria del microcontrolador para su ejecución.

4.2.2 Implementación del sistema de acondicionamiento completo

La interconexión de los periféricos se puede apreciar en la Fig. 4.12, la cual constituye una extensión del circuito presentado en Fig. 4.7, donde se adiciona el circuito de acondicionamiento para el sensor de temperatura, los ADCs para la obtención de la información en formato digital de las lecturas del ISFET y el termistor, y finalmente el bloque de comunicaciones que implementa el protocolo UART para la transmisión de dicha información.

La configuración de los módulos ADCs se muestra en 4.13. Para el caso del ADC Sigma-

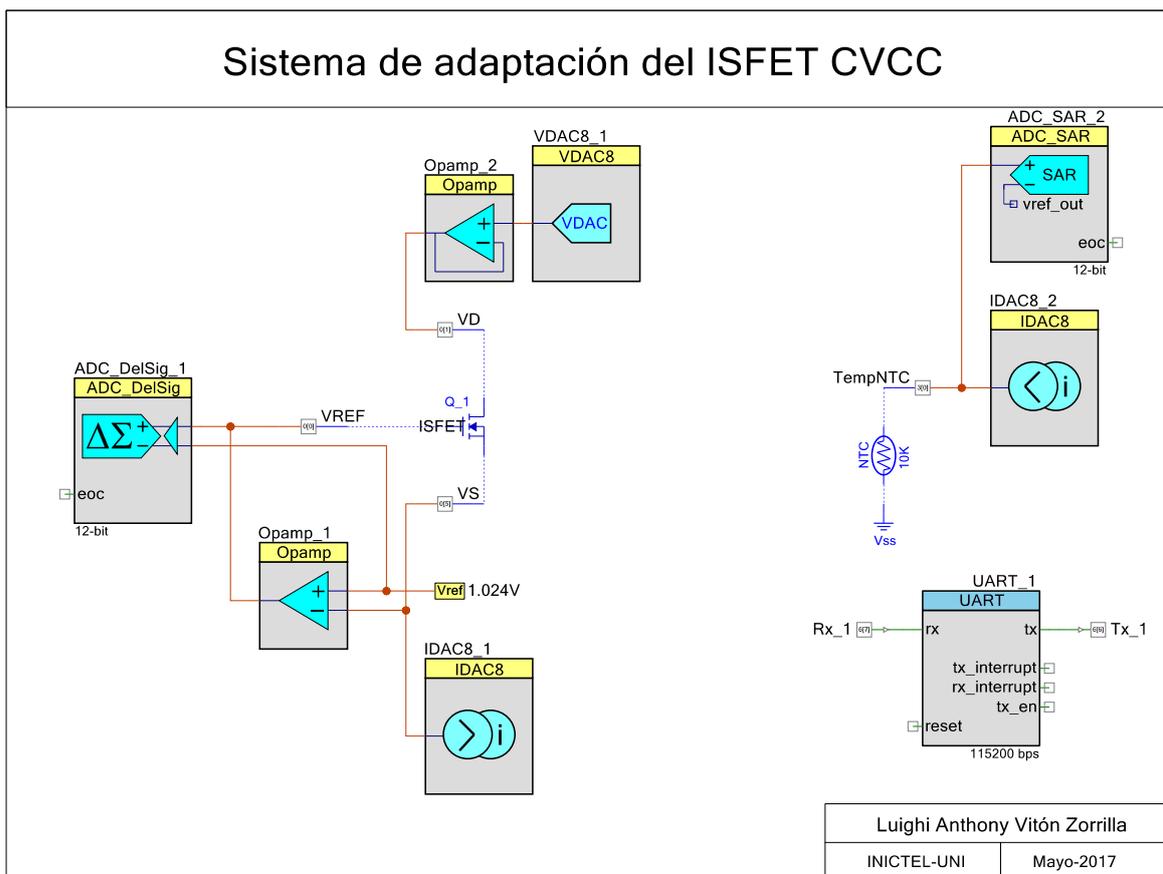
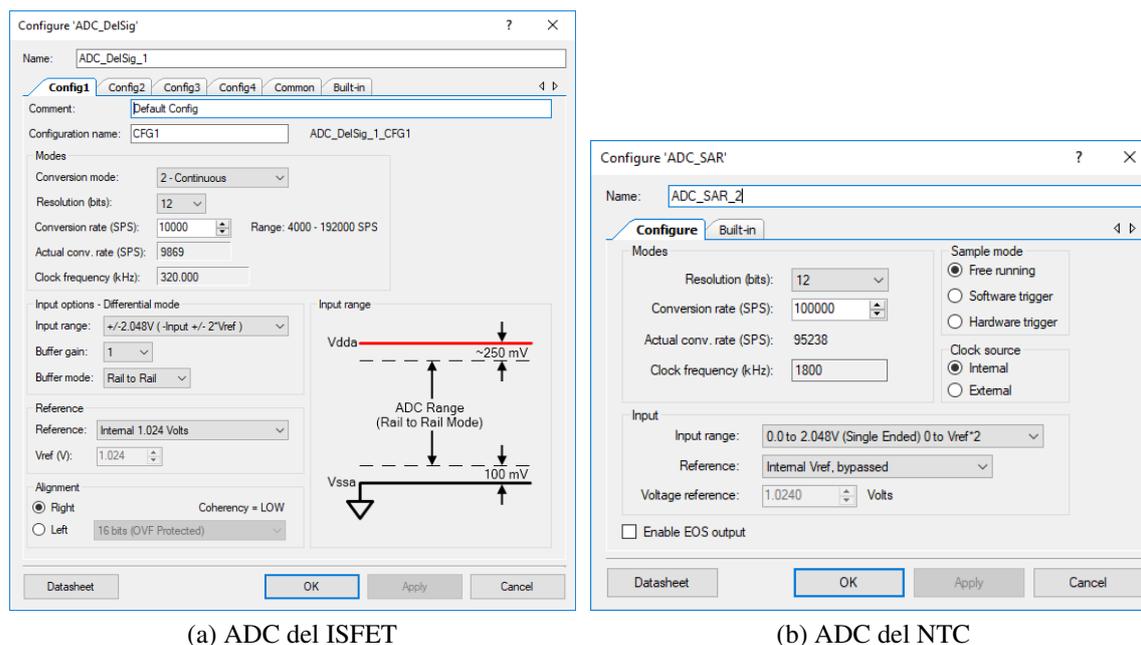


Fig. 4.12: Sistema de adaptación para el sensor de ISFET y de temperatura implementado en PSoc Creator.

Delta conectado al ISFET, se establecen los parámetros definidos anteriormente respecto a la resolución y la frecuencia de muestreo, correspondientes a 12 bits y 10ksps, respectivamente. Por otra parte, también se establece el rango de tensión de entrada entre 2.048 a -2.048, lo cual cubre completamente la salida de tensión del circuito de acondicionamiento que, de acuerdo a (5.4), estaría comprendido entre 0.826V y 1.4252V.

En el caso del ADC SAR empleado para el NTC, también se establecen los parámetros de resolución (12 bits) y frecuencia de muestreo (100ksps). En este caso, se establece automáticamente la frecuencia de reloj requerida en 1.8MHz. Por otra parte, respecto a las tensiones de referencia, el rango se establece de 0 a 2.048, con la finalidad de cubrir el máximo rango de temperatura en el sensor. En ambos casos, las referencias se han establecido respecto a la tensión de referencia interna del microcontrolador de 1.024V, en lugar de las tensiones de alimentación, con la finalidad de evitar posibles variaciones a causa de la disminución de la tensión en la batería.

Respecto a la configuración de la fuente de corriente en el circuito de acondicionamiento



(a) ADC del ISFET

(b) ADC del NTC

Fig. 4.13: Configuración de los ADCs.

del NTC, se configura a través de un bloque IDAC con una corriente de $100\mu\text{A}$, correspondiente a un valor digital de 64H, de acuerdo al diseño planteado en Fig. 3.17(b). Esta configuración se puede apreciar en Fig. 4.14.

Finalmente, se configura el bloque de comunicaciones tal como se muestra Fig. 4.15, con la velocidad de 115200 y con un modo de comunicación bidireccional.

Posteriormente, se debe asignar los pines de salida para las terminales de los sensores, así como para el bloque de comunicaciones. Los pines del circuito de acondicionamiento se escogen a razón de encontrarse lo más cercanos a los módulos de tal forma que tengan una conexión directa con éstos, asegurando a la vez la integridad de la señal, tomando como referencia la estructura del enrutamiento analógico del dispositivo [176]. Para el bloque de comunicaciones se escoge dos pines dentro del conjunto de pines digitales. En este caso, la elección depende de su disposición en la implementación final del prototipo, a fin de que se encuentren lo más cercanos al puerto de conexión.

Otra configuración importante en el microcontrolador corresponde al sistema de relojes encargados de generar las señales de sincronización del CPU y los periféricos. De preferencia, se trata de reducir la frecuencia de operación a fin de disminuir el consumo de energía del CPU, desactivando, a su vez, los módulos adicionales tales como el PLL [117]. No obstante, existen determinados bloques que requieren de frecuencias específicas, las cuales no pueden ser logradas con los niveles proporcionados por el reloj interno del microcontrolador (oscilador

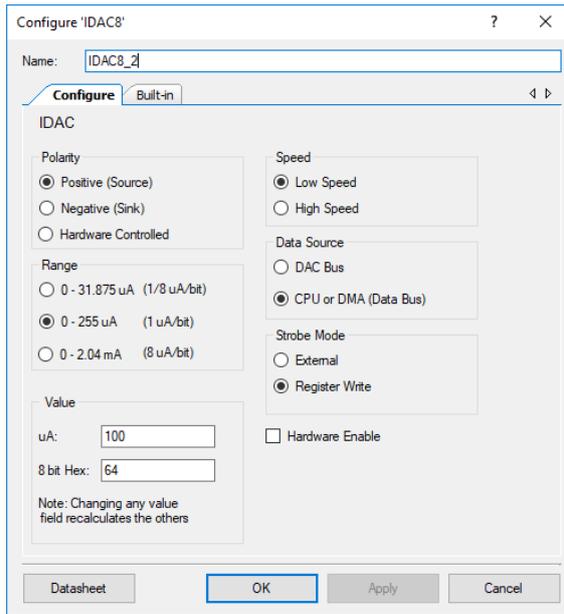


Fig. 4.14: Configuración del IDAC para NTC.

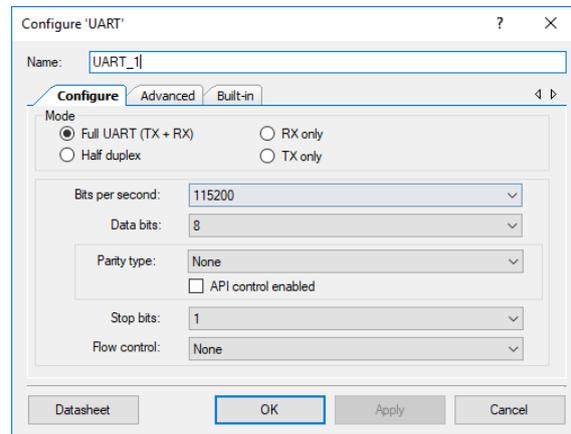


Fig. 4.15: Configuración del UART

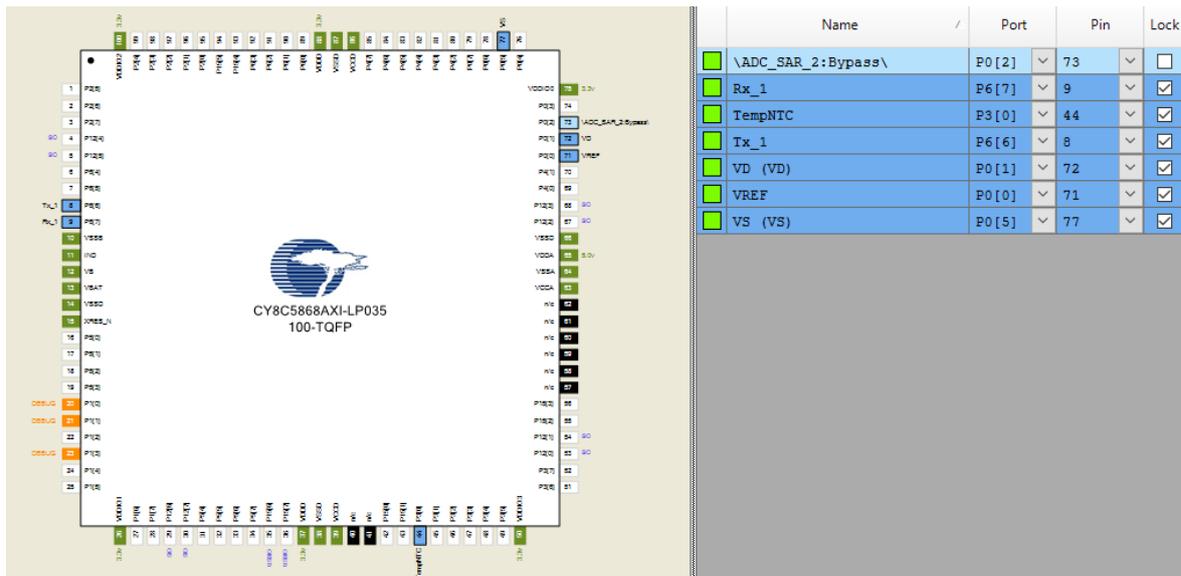


Fig. 4.16: Distribución de pines para el sistema de adaptación del ISFET.

principal interno, IMO). Este es el caso del UART, el cual exige una frecuencia de al menos 8 veces la velocidad de operación (921.6 kHz) que, si bien se encuentra por debajo de la frecuencia mínima del IMO (3MHz), no se logra obtener, a través de los divisores, un resultado dentro del rango de tolerancia exigido por el módulo. Esto obliga a tener una frecuencia mayor, en este caso 12MHz, que a través de un divisor de 13 logra una frecuencia de 923kHz. Los demás periféricos (ADCs) no ofrecen mayor inconveniente al encontrarse sus frecuencias por debajo del valor de frecuencia del IMO, siendo seleccionadas automáticamente por el

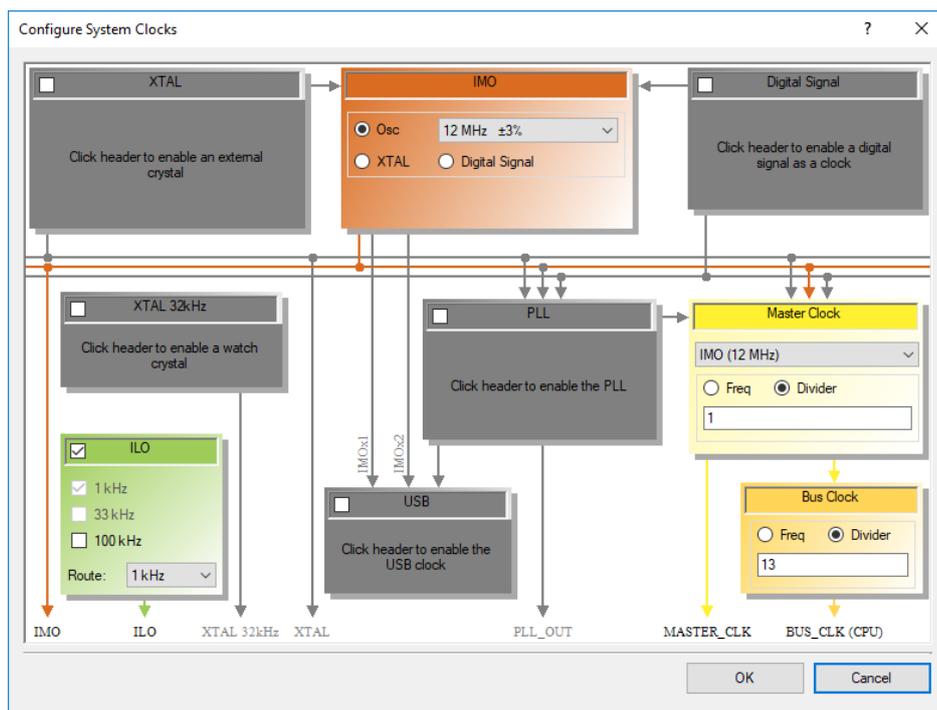


Fig. 4.17: Configuración del sistema de relojes en el microcontrolador.

algoritmo del software de programación. Esta configuración se puede apreciar en Fig. 4.17.

Una vez realizadas las configuraciones de los periféricos y del sistema de relojes del microcontrolador, se procede a implementar el firmware de su CPU. La primera instrucción en el CPU, luego de su inicialización, corresponde a la inicialización de todos sus periféricos (ADCs, DACs, OPAMP y UART). Luego, se envía la orden para que los ADCs comiencen su proceso de conversión. Posterior a ello, comienza el ciclo infinito de operación del microcontrolador. Durante este ciclo es cuando se adquieren las muestras de los ADCs generando una trama con valores de pH y temperatura, la cual, luego de una determinada cantidad de muestras, es enviada mediante el protocolo UART. El proceso comienza con la lectura de los ADCs, luego se realiza la conversión de los valores obtenidos a valores de pH y temperatura mediante las ecuaciones (4.1) y (4.2), derivadas de (5.4) y (3.8):

$$\text{pH} = \frac{V_{out} - (1.850 - 1024)}{0.0428} = \frac{V_{out}(\text{mV}) - 826.71}{42.8} \quad (4.1)$$

$$T = \frac{\beta}{\ln R/R_C} = \frac{3950}{\ln R/0.0175}; \quad R_C = R_0 e^{-\beta/T_0} \quad (4.2)$$

donde, para el caso de la temperatura (4.2), se ha realizado una simplificación matemática a través de un factor R_C , derivado de los términos constantes de la ecuación (3.8).

Una vez efectuada la conversión se almacenan los valores en un buffer, incrementando el contador en 1. Si este contador es menor que la cantidad de muestras requerido, entonces se repite el proceso anterior añadiendo una muestra más al buffer. Cuando el contador iguala el número de muestras, se reinicia y se envía la trama completa por UART. Luego, se reinicia el buffer y se espera un tiempo para comenzar nuevamente con el proceso. Este flujo constituye una versión preliminar del sistema de control a implementar en la siguiente sección.

4.3 Implementación del sistema de procesamiento y control

La implementación del sistema de procesamiento y control se realiza a nivel del software para el microcontrolador. El flujo de control se ejecuta a través de una máquina de estados, la cual, dependiendo del estado en que se encuentre, efectúa una serie de tareas.

El esquema de la implementación del software para el microcontrolador se puede apreciar en la Fig. 4.18. El primer paso consiste en la inicialización de los componentes que forman parte del sistema. Luego se inicia el bucle infinito de operación en el microcontrolador. Dentro de este bucle se implementa la máquina de estados del sistema de control. El primer estado corresponde a la fase *idle* y allí se efectúa la desactivación de todos los periféricos inicializados en el paso anterior, dándose la instrucción para continuar con el siguiente estado. El segundo estado corresponde a la fase de activación de sensores, donde se activan los periféricos analógicos empleados en el circuito de acondicionamiento (IDACs, VDACs y OPAMPs), luego, se efectúa un retardo de 2 segundos de acuerdo al tiempo de respuesta del ISFET, posterior a ello, se activan los ADCs y se envía la instrucción para pasar al siguiente estado. El tercer estado corresponde a la lectura de datos de los ADCs. En este estado se emplea una instrucción de control para seleccionar el momento en que debe pasarse al siguiente estado. Esta señal es determinada por una variable que contabiliza la cantidad de muestras que se adquieren y se compara con el máximo establecido. Si esta cantidad es menor, se obtienen y guardan los datos en un arreglo incrementando la cuenta, mientras que si llega al máximo se desactivan los ADCs y se ingresa la instrucción para pasar al siguiente estado. El cuarto estado corresponde al procesamiento de la información recibida. Nuevamente se emplea una variable que contabiliza el número de veces que se aplican los filtros diseñados en la etapa de procesamiento, tanto de mediana como de media móvil, de tal manera que si es inferior a la dimensión de los datos almacenados, se continúa en el mismo estado aplicándose los filtros, y de ser igual, se calcula la media total y se aplica la ecuación de compensación

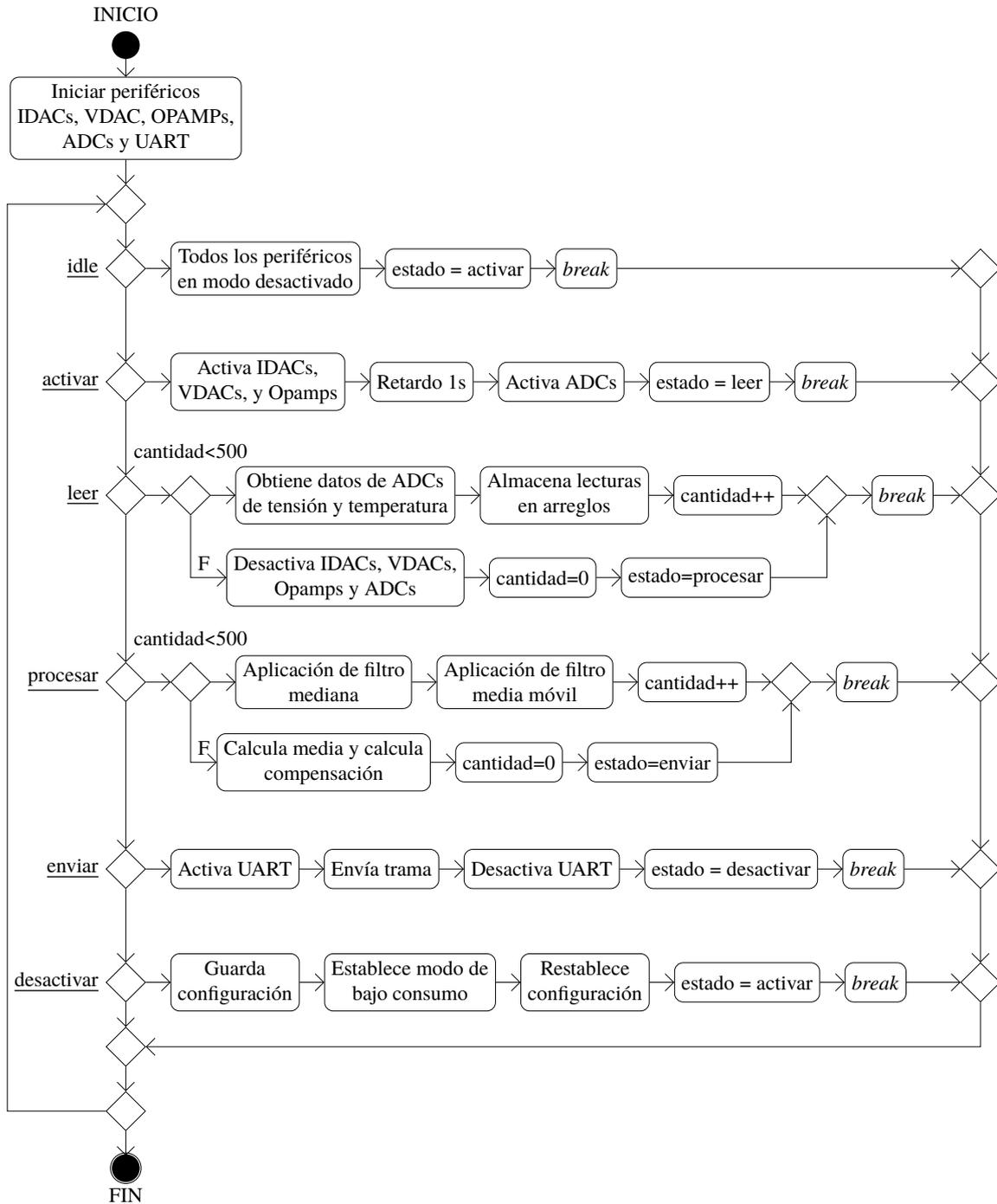


Fig. 4.18: Esquema del control y procesamiento implementado en el microcontrolador.

para luego pasar al siguiente estado. En el penúltimo estado, correspondiente al envío de los datos, se activa el UART y se envía la trama a través de este protocolo, luego se desactiva este componente y se pasa al siguiente estado. El último estado corresponde a la fase de reposo, en la cual se establece el microcontrolador en modo de bajo consumo de energía, durante un tiempo especificado, para luego regresar al estado de activación de sensores y repetir el ciclo de operación.

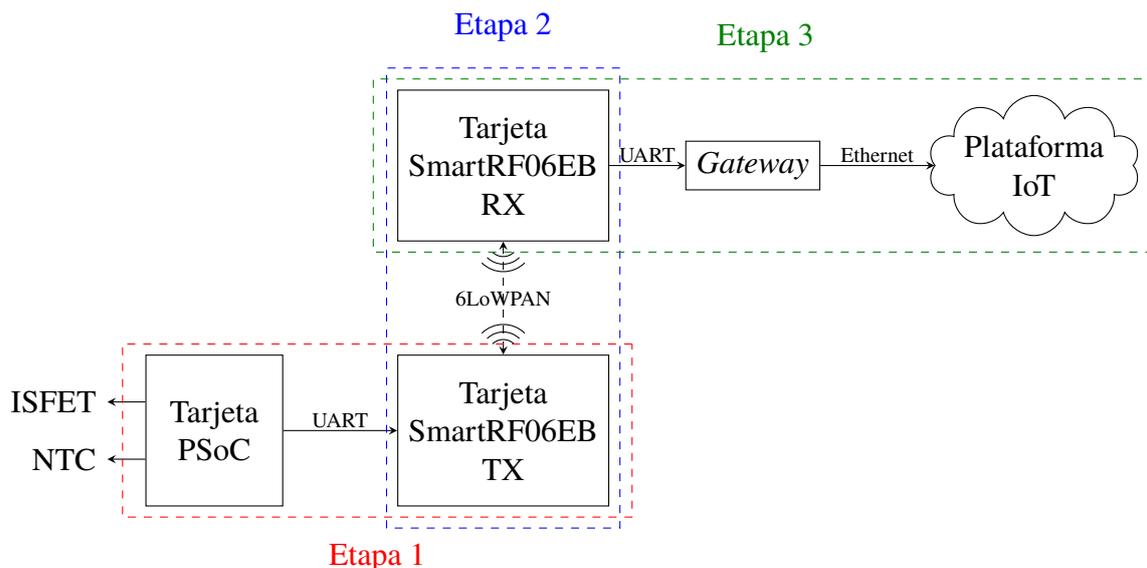


Fig. 4.19: Esquema de implementación de la comunicación inalámbrica.

4.4 Implementación del sistema de comunicaciones inalámbrica

La implementación de la comunicación inalámbrica engloba la conexión de las tarjetas de evaluación que se han empleado hasta el momento tanto para el sistema de acondicionamiento y procesamiento de señal (PSoC), así como para la transmisión inalámbrica (CC2650EM-7ID), la cual incorpora el integrado CC2650 escogido para la implementación del protocolo 6LoWPAN. Adicionalmente, contempla la conexión de la tarjeta de comunicaciones inalámbricas con una tarjeta receptora y el envío de esta información a una plataforma de IoT.

El esquema general de la implementación se muestra en la Fig. 4.19. Allí se puede apreciar que la solución está formada por tres etapas, desde que el dato se encuentra listo para enviar en la tarjeta PSoC, hasta que se visualiza en la plataforma de IoT. La primera etapa corresponde a la comunicación entre las tarjetas PSoC y SmartRF06, la cual se realiza mediante un protocolo UART. La segunda etapa contempla desde la transmisión del dato a través del protocolo 6LoWPAN por el módulo en la tarjeta de evaluación conectada al PSoC hasta su recepción por la tarjeta conectada con el *gateway* para su envío al Internet. Finalmente, la tercera etapa corresponde al proceso desde la recepción por la tarjeta hasta el envío del *gateway* a la plataforma de IoT.

Estas etapas requieren de ciertas características de software en los respectivos microcontroladores encargados de completar ciertas tareas para lograr el objetivo designado. Por tanto, se abordará cada una de las etapas, el programa empleado para cada elemento involucrado, así como las consideraciones que se tuvieron al momento de diseñarlo.

4.4.1 Comunicación entre tarjetas PSoC y SmartRF06

El sensor de temperatura y el ISFET se encuentran conectados a la tarjeta de desarrollo PSoC 5LP CY8CKIT-50, donde se ha implementado el sistema de acondicionamiento de señal y de control. Puesto que el objetivo en esta etapa consiste en lograr la comunicación con Internet, se plantea enviar una trama de prueba donde se agrupen varios valores de pH y temperatura, los cuales serían mostrados posteriormente en la plataforma de IoT.

Tal como se indica en la sección 3.4.2, Texas Instruments provee de una plataforma de evaluación que se conecta con la tarjeta de evaluación SmartRF06EB, expandiendo sus conectores con diferentes sensores y puertos de comunicación a fin de comprobar su funcionalidad. Dentro de estos puertos de comunicación se encuentra un conector para UART, habilitado a través del cambio de la posición del *jumper* J5 en la tarjeta de evaluación. En este caso, se debe retirar este jumper a fin de deshabilitar la comunicación con el integrado de programación XDS100v3, dejándolo sólo conectado a los pines de comunicación externos de la tarjeta [149].

El puerto UART disponible en la tarjeta se conecta a los pines escogidos en la tarjeta de PSoC para la comunicación a través de UART, de tal forma que el pin de transmisión de la tarjeta PSoC se conecte con el de recepción de SmartRF06, y análogamente, el de recepción con el de transmisión.

Una vez identificada la conexión que se debe realizar entre las tarjetas, se aborda el diseño del software para el PSoC. Para este caso, se realiza un programa de prueba donde una vez adquirida la información de los ADCs, se almacenan los datos en un buffer hasta alcanzar una cantidad de muestras determinada, luego esta información se envía a través de UART y se establece un tiempo de pausa, para posteriormente reiniciar el proceso. El proceso se puede apreciar en Fig. 4.20¹. La estructura de datos planteada para la prueba está compuesta por una lista de pares de datos. En este caso, se han considerado 10 pares y tiene la estructura mostrada en (4.3):

$$\text{Temp1 : pH1; Temp2 : pH2; Temp3 : pH3; Temp4 : pH4; \dots Tempn : pHn;} \quad (4.3)$$

El envío se realiza de forma continua a la tarjeta de comunicaciones, con lapsos de 10 segundos entre envíos consecutivos a través de UART.

¹Una vez evaluada la comunicación inalámbrica el software de prueba es reemplazado por el implementado en la sección de procesamiento y control.

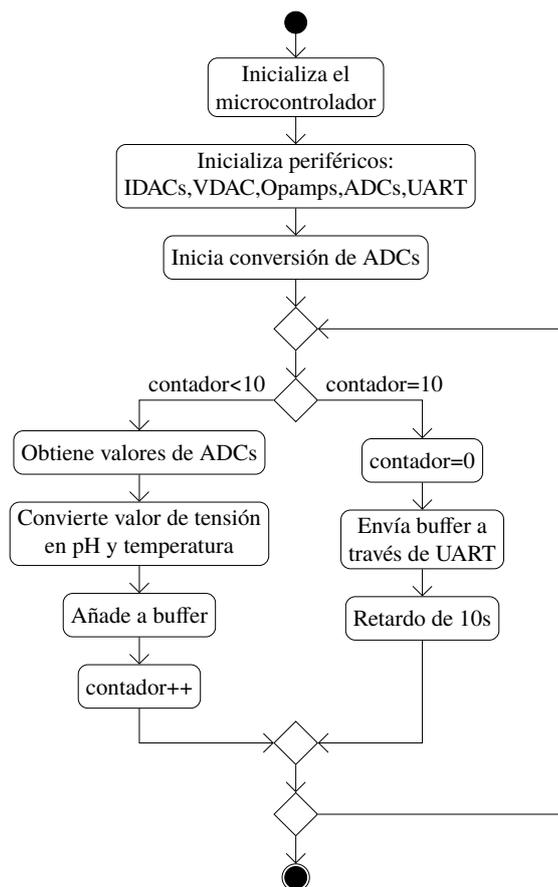


Fig. 4.20: Diagrama de software en el PSoC para comunicación mediante UART.

4.4.2 Transmisión de datos mediante protocolo 6LoWPAN

La comunicación se realiza a través del protocolo de IoT 6LoWPAN. La implementación se efectúa en dos módulos de evaluación para comunicaciones inalámbricas CC2650EM-7ID, uno para transmisión y otra para recepción, situadas en dos tarjetas de pruebas SmartRF06EB. Para efectuar dicha implementación se hace uso del sistema operativo Contiki OS, el cual está diseñado para sistemas embebidos con funciones de transmisión/recepción inalámbrica, ofreciendo la posibilidad de configurar una serie de protocolos estandarizados por el IETF, incluidos entre ellos el 6LoWPAN [177].

La mayoría de características de comunicación inalámbrica son configuradas por defecto en el sistema operativo, mientras que las tareas de ejecución se establecen a través de la definición de procesos dentro de su archivo principal. Si bien es cierto, el sistema es bastante conocido, la documentación se encuentra distribuida dentro del árbol de ficheros empleados para su implementación, siendo algunas funciones un poco difíciles de seguir el rastro en cuanto a su definición y declaración. Afortunadamente, Contiki cuenta con una serie de ejem-

plos los cuales son de bastante utilidad al momento de emplear el sistema operativo, puesto que, con algunas modificaciones, ya se puede tener el sistema ejecutándose adecuadamente. Por otra parte, existe una serie de libros que pueden ser de utilidad para entender su forma de operación, entre los que se encuentran el presentado por Liñán Colina, Vives, Bagula y col. en [178], donde se realiza una revisión acerca de Contiki OS y se provee de una serie de ejemplos y ejercicios paso a paso para la conexión de la plataforma Zolertia Z1 a través de 6LoWPAN. Si bien la tarjeta empleada en este caso es diferente, a nivel global el flujo de operación se mantiene.

Para la implementación de una comunicación inalámbrica entre las dos tarjetas se debe configurar una de ellas como transmisora mientras la otra se establece como receptora. Para ello se emplea el ejemplo contenido dentro de la instalación del sistema operativo denominado `simple-udp-rpl` [179], a través del cual se implementa una comunicación donde el nodo puede asumir el modo de operación *broadcast*² o *unicast*³ para la recepción y transmisión de información. En el presente caso, se empleará la configuración de *broadcast* para el nodo transmisor, mientras el otro recibe en modo *unicast*.

Para el transmisor se implementa un flujo de operación, como se muestra en Fig. 4.21(a). Allí se describe el proceso ejecutado por el sistema operativo para efectuar las tareas contempladas para el transmisor. En primer lugar, se inicia el proceso, luego se enlaza la entrada del módulo UART con la entrada serial del microcontrolador, donde los caracteres que recibe el puerto se almacenan en un buffer y se traspasan al módulo serial. Posteriormente, se realiza la configuración del módulo para comunicación mediante UDP, estableciendo el puerto y la estructura para el receptor. Luego de ello, inicia el bucle infinito del programa, donde se establece en modo de espera hasta que se produzca algún evento. Si este evento proviene del puerto serial indicando que se ha enviado un carácter de fin de línea, entonces se copia la información del buffer en una cadena de texto, la cual posteriormente es enviada a través de UDP, en modo *broadcast*, a todos los nodos que se encuentren dentro de la misma red local. Finalmente, regresa nuevamente al estado de espera de un nuevo evento en el microcontrolador.

Para el receptor, se emplea el ejemplo denominado `unicast-receiver` en el cual se establecen el conjunto de rutinas a ejecutar en la tarjeta de recepción. El flujo de operación se

²En todas las direcciones.

³En una sola dirección.

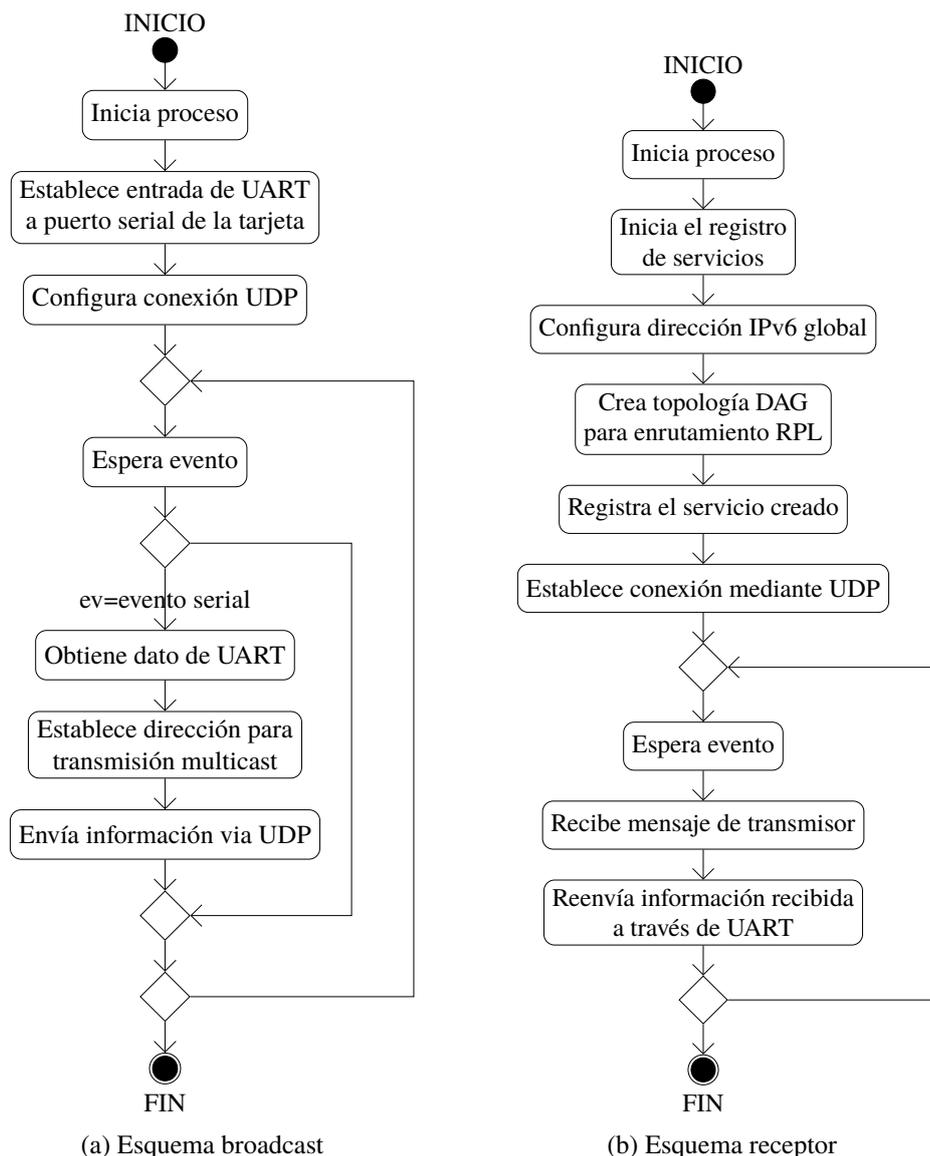


Fig. 4.21: Esquema de los procesos de transmisión y recepción implementados en Contiki OS.

puede apreciar en Fig. 4.21(b). El flujo comienza con la inicialización del registro de servicios. En este caso, se emplea una librería adicional de Contiki OS denominada *servreg-hack*, la cual permite registrar los servicios dentro de un nodo de tal forma que pueda comunicar a los vecinos acerca del servicio proporcionado. Luego, se procede a asignarle una dirección IPv6 al nodo receptor. Posteriormente, se genera la tabla para el protocolo de enrutamiento RPL[180]⁴ empleado para redes con pérdidas y de baja potencia (*Low-Power and Lossy Networks*, LLN), tales como en redes de sensores. Luego, se registra el servicio creado, asociándolo a la dirección del nodo. A continuación, se establece la conexión mediante UDP

⁴Protocolo de Enrutamiento para Redes de baja potencia con pérdidas.

para, luego, dar paso al bucle infinito del proceso en el cual se espera la ocurrencia de un evento, en este caso la recepción de la trama, para luego reenviar el mensaje al *gateway* a través de UART.

4.4.3 Envío de datos a la plataforma de Internet de las Cosas (IoT)

De acuerdo al gráfico presentado en Fig. 4.19, una vez recibida la información en la tarjeta receptora, ésta envía dicha información a Internet. En el presente caso, se ha optado por emplear un *gateway*, el cual se encarga de recibir la información a través de UART de la tarjeta receptora y retransmitirla a algún servicio o plataforma de IoT a través de Ethernet.

La tarjeta receptora se encuentra igualmente montada sobre la tarjeta SmartRF06EB, en la cual se hace uso del canal de UART a través de USB, proporcionado por el integrado de programación XDS100v3 de la misma tarjeta. Para el *gateway* se opta por emplear una computadora con GNU/Linux, la cual puede ser fácilmente reemplazada por un computador reducido tal como Raspberry o BeagleBone. Luego, mediante los recursos de comunicación del *gateway*, se envía la información a la plataforma de IoT elegida, en este caso Ubidots [181].

El flujo de operación del *gateway* se puede apreciar en Fig. 4.22. Los primeros pasos consisten la configuración del puerto serial, así como de los parámetros para el envío a la plataforma de IoT. Luego comienza el bucle infinito de operación dentro del cual se abre el puerto serial y se espera a la recepción del mensaje. Este mensaje se almacena en un buffer temporal. Si este buffer no es nulo, entonces se procesa la información para obtener los datos de la temperatura y el pH. Una vez obtenidos estos datos, se envían a la plataforma de Ubidots, empleando los parámetros de acceso configurados anteriormente.

Para la implementación del software se ha empleado el lenguaje python, por una parte, debido a su facilidad de desarrollo y, por otra, dado que Ubidots proporciona una API (interfaz de programación de aplicaciones) para este lenguaje [182], la cual es empleada para enviar los datos a su plataforma.

4.5 Implementación del hardware para el prototipo de sensado de pH

Para la implementación del hardware se debe diseñar las tarjetas de circuito impreso en las cuales se ensamblarán los componentes que constituyen el prototipo final. Los componentes elegidos se basan en la tarjeta de desarrollo proporcionada por el fabricante, mientras que el diseño del circuito impreso se basa en las recomendaciones proporcionadas por la literatura

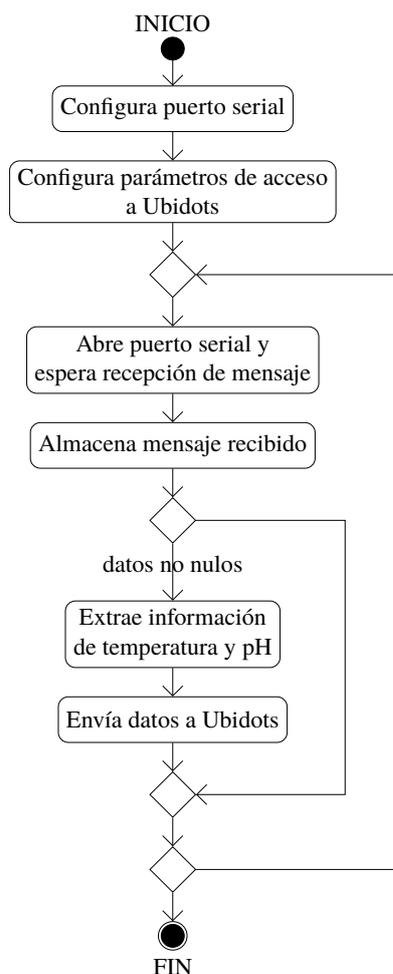


Fig. 4.22: Esquema de programa ejecutado en el *gateway*.

con la finalidad de garantizar la integridad de señal. Finalmente, el diseño es enviado para su fabricación y es ensamblado por etapas, verificando su correcto funcionamiento.

4.5.1 Especificación de componentes y esquema de implementación

El diseño está orientado a las dos tarjetas que se van a emplear para construir el prototipo: la tarjeta principal y la tarjeta de expansión. La primera tarjeta es la que contiene la mayor cantidad de componentes y es donde se realiza la distinción de los bloques constitutivos relacionados con el sistema de energía, el subsistema digital y el subsistema analógico.

El diseño de la tarjeta principal comienza con el sistema de energía. De acuerdo a ello, el sistema tiene dos entradas de alimentación una para USB y otra para batería. Por otra parte, cuenta con tres reguladores de tensión a partir de los cuales se proporciona la alimentación al subsistema analógico y al digital. El esquema del sistema de energía a implementar se puede apreciar en la Fig. 4.23. Allí se distinguen tres bloques, uno correspondiente a los puertos de entrada, otro a los reguladores de tensión y otro a la conexión entre tierras.

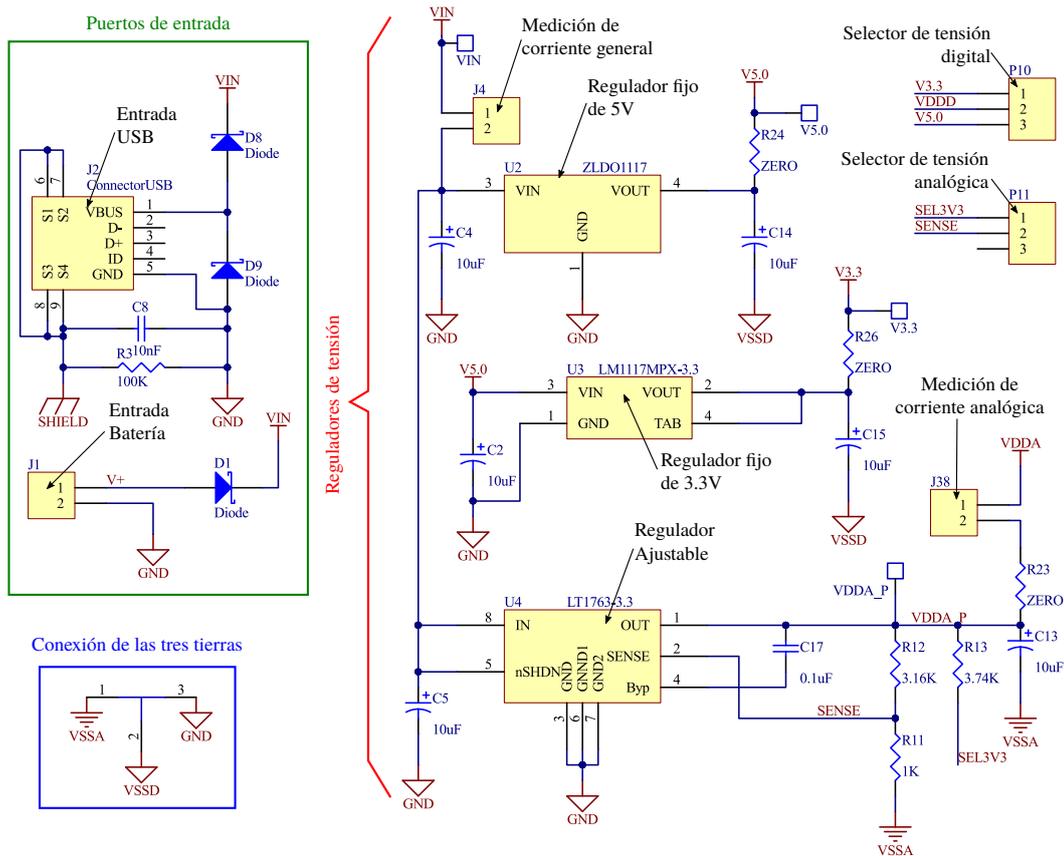


Fig. 4.23: Esquemático del sistema de energía. Diseñado en Altium Designer 17.

El bloque de reguladores está conformado por tres reguladores, dos fijos de 5V y 3.3V y uno ajustable. Éstos se eligen con las mismas especificaciones que en la tarjeta de desarrollo CY8CKIT-050 [167]. En los reguladores se hace una distinción entre la tierra de la alimentación de entrada y el regulador (GND), y la tierra de la salida, correspondiente a la tierra del subsistema analógico VSSA y digital VSSD. Las tres tierras diferenciadas para la alimentación, el subsistema digital y el subsistema analógico, se conectan en un único punto, el cual es efectuado por un elemento especial creado en la herramienta de diseño para el PCB.

El segundo bloque lo conforma el circuito de polarización del microcontrolador PSoc 5LP, el cual se puede apreciar en Fig. 4.24. El integrado empleado corresponde al CY8C5868AXI-LP035. En esta etapa es donde se diferencian los subsistemas digital y analógico.

El tercer bloque lo conforman los puertos de comunicación con dispositivos externos cuya conexión se puede apreciar en Fig. 4.25. De acuerdo a ello, se puede distinguir tres grupos: el puerto de programación, los puertos de expansión analógico y digital, y los puertos para sensores. Para los puertos de conexión de los sensores se utilizan dos tipos de conectores diferentes: para el sensor de temperatura, un conector TRS (punta-anillo-funda) y para el

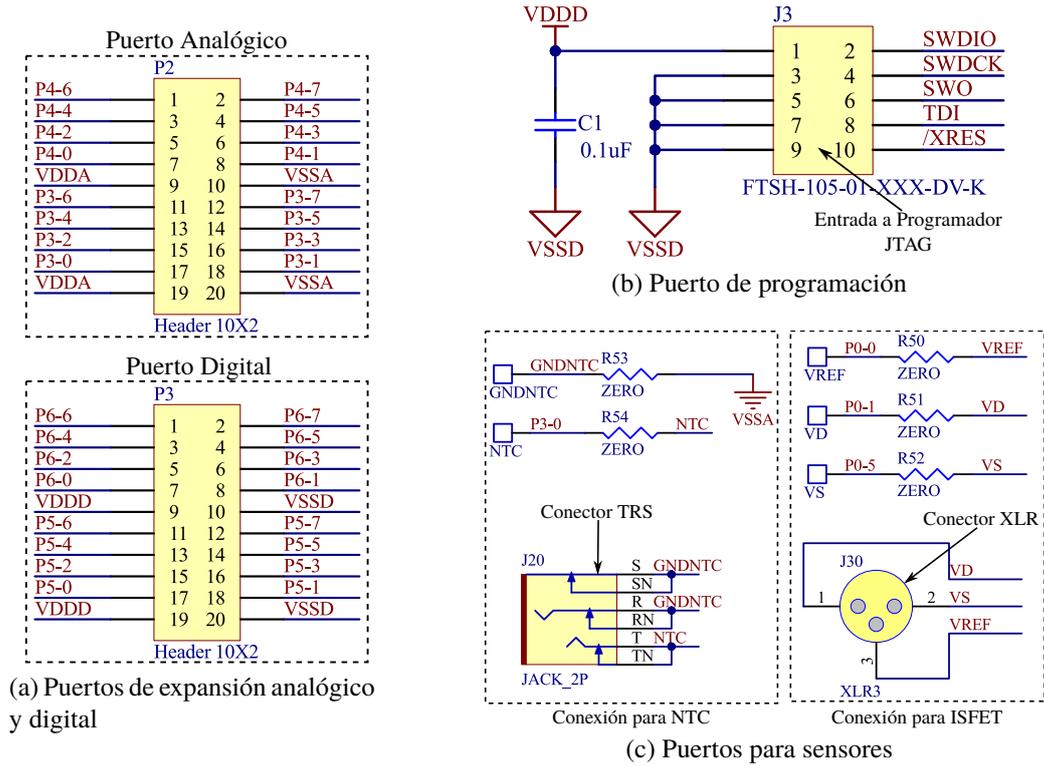


Fig. 4.25: Puertos de conexión con dispositivos externos.

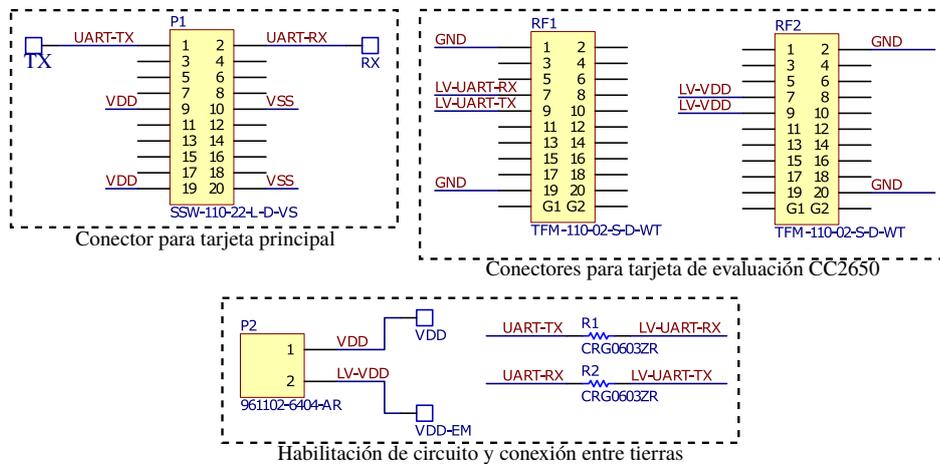


Fig. 4.26: Esquemático del circuito de conexión con otras tarjetas.

la configuración por defecto del programa, se encuentra el aumento de la distancia entre las rutas y el plano de tierra (0.762mm), así como la posibilidad de cortocircuito entre las tierras GND, VSSD y VSSA. El área sobre la cual se dibuja el circuito está dividida en tres regiones, correspondientes a la separación de los dominios de tensión, y siguiendo la recomendación de ubicación de componentes en un diseño de señal mixta. Una región corresponde al sistema de energía; otra región, al subsistema analógico; y la última, al subsistema digital. El resultado del diseño efectuado de la tarjeta se puede verificar en la Fig. 4.27 para la capa superior y

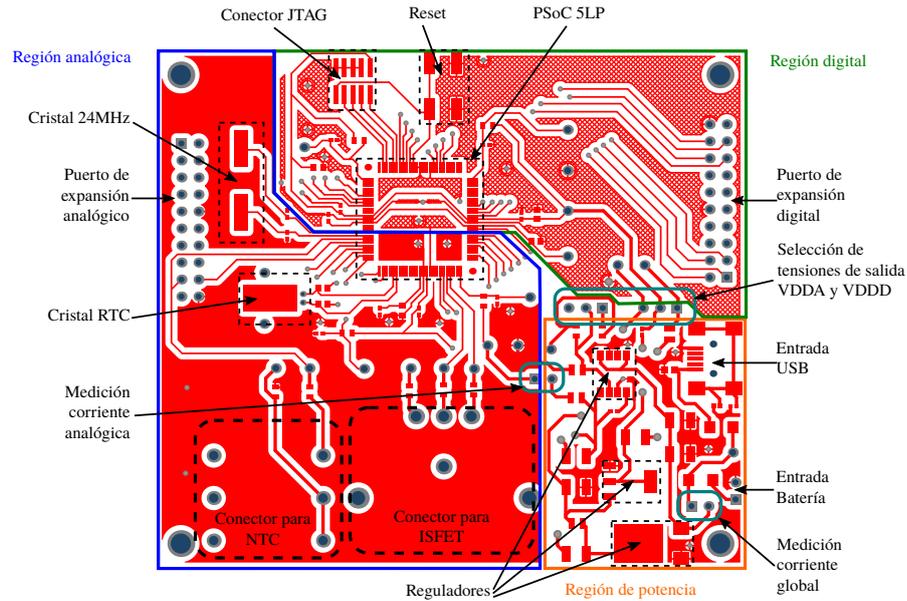


Fig. 4.27: Capa superior del PCB

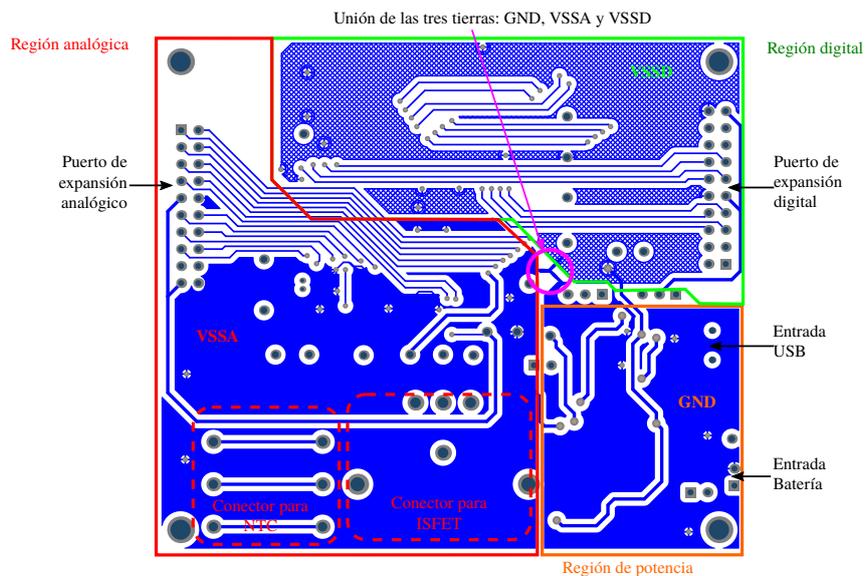


Fig. 4.28: Capa inferior del PCB

Fig. 4.28 para la capa inferior.

El diseño de la tarjeta de expansión es similar a la tarjeta principal, puesto que también se emplea un diseño a dos capas, teniendo una dimensión mínima de pistas de 0.254mm (10mil) y con un área de 4.5 x 4 cm. También en este caso se ha añadido la regla del aumento de distancia entre las rutas y el plano de tierra (0.762mm), así como la posibilidad de cortocircuito entre las tierras GND y VSS.

El resultado del diseño se puede apreciar en la Fig. 4.29, tanto para la capa superior como para la capa inferior de la tarjeta.

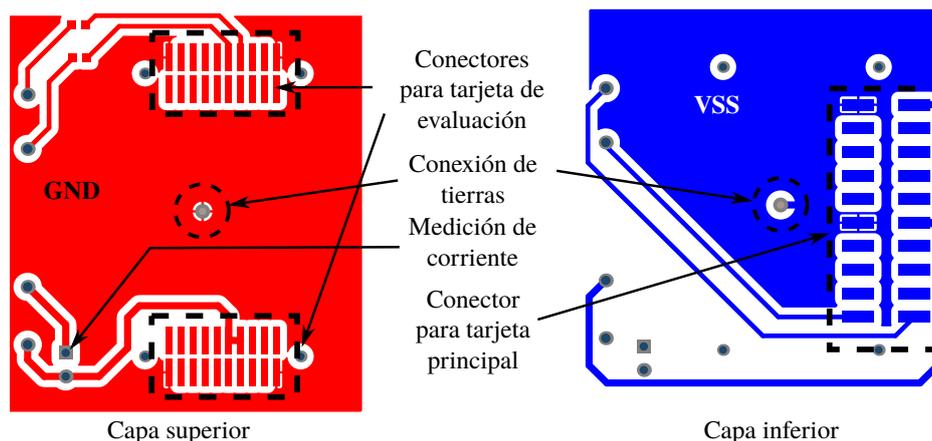


Fig. 4.29: Capas de tarjeta de expansión

4.5.3 Ensamblado de componentes y resultados preliminares

La tarjeta fue fabricada por una empresa local de nombre JOBARECI [183], en un material de fibra de vidrio (FR4), a doble capa y con proceso de metalizado para las vías. El ensamblado de los componentes se realizó manualmente, lo cual dio la oportunidad de realizar pruebas de cada uno de los bloques que constituyen la placa. El procedimiento que se siguió fue el siguiente:

1. Se efectúa la verificación de la continuidad de vías y pistas en la placa fabricada. Esto con la finalidad de detectar posibles fallos de fabricación.
2. Se realiza el ensamblado del primer bloque de la tarjeta principal correspondiente al sistema de energía. Así mismo, se efectúa la verificación de funcionamiento a partir de la medición de las tensiones de salida VDDA y VDDD.
3. Se procede a ensamblar los componentes del circuito de polarización del integrado. Posteriormente, se efectúa la programación del dispositivo empleando para ello el dispositivo de programación PSoC® MiniProg3 [184] también de Cypress, el cual se conecta mediante el puerto JTAG ensamblado en la tarjeta. Para garantizar la correcta comunicación con el programador, se deben tener en cuenta lo siguiente:
 - a) El microcontrolador debe estar energizado externamente.
 - b) La tensión analógica debe ser mayor o igual a la tensión digital.

En la Fig. 4.30 se puede apreciar la conexión que se ha realizado para efectuar la programación del dispositivo. Se muestra el ensamblado parcial de la tarjeta donde se

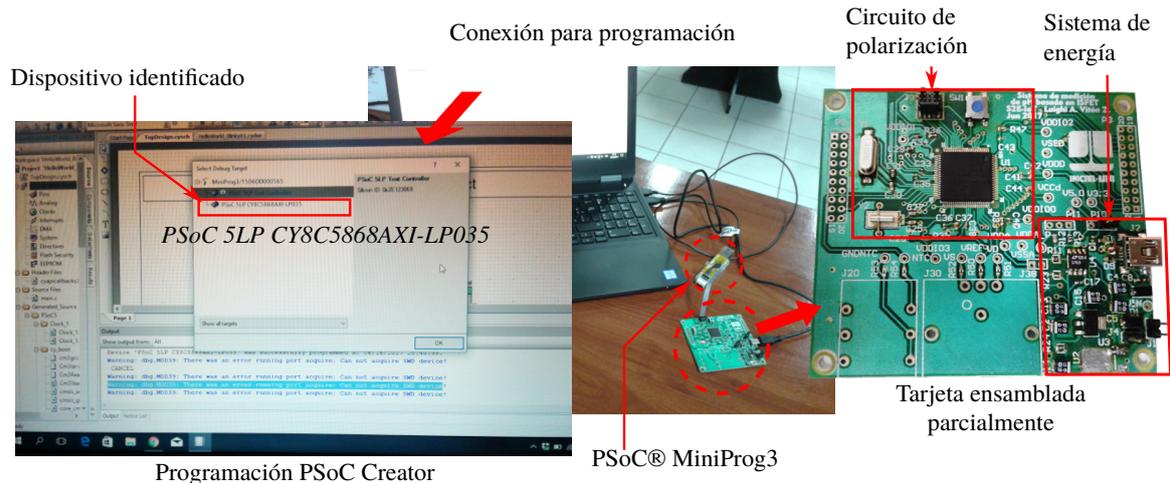


Fig. 4.30: Ensamblado de circuito de polarización y primera programación del microcontrolador.

contempla el sistema de energía, verificado en el punto anterior, además del circuito de polarización del microcontrolador. El software empleado para la programación es el PSoC Creator, y se puede apreciar que, al conectar la tarjeta a través del PSoC® MiniProg, se logra identificar el integrado y, por tanto, se puede efectuar su programación.

4. Posteriormente, se carga el firmware implementado en la tarjeta de desarrollo y se procede a verificar el funcionamiento de los periféricos del microcontrolador, a través de la lectura del sensor de temperatura y envío de información a la PC.

En la Fig. 4.31 se puede apreciar el ensamblado de la tarjeta hasta esta etapa, además de la conexión que se realiza con el sensor de temperatura y el puerto UART, el cual se comunica con la computadora a través de un adaptador UART-USB, mostrando los valores medidos con el sensor de temperatura.

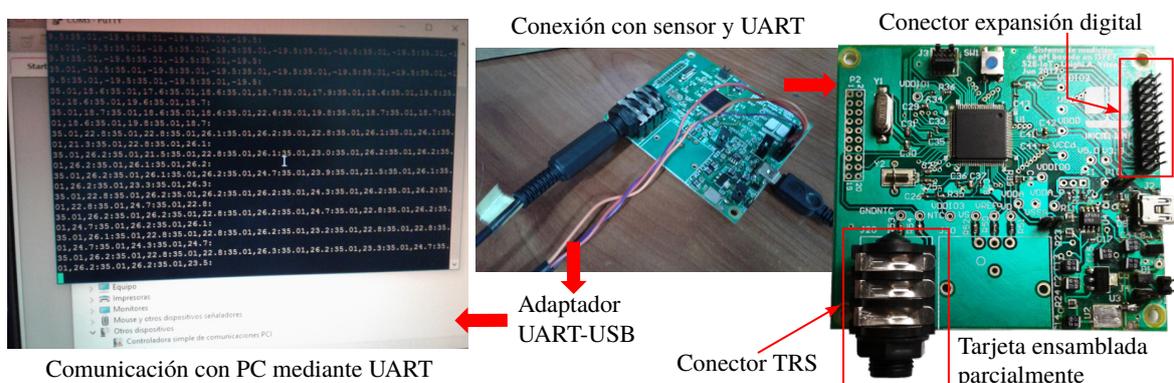
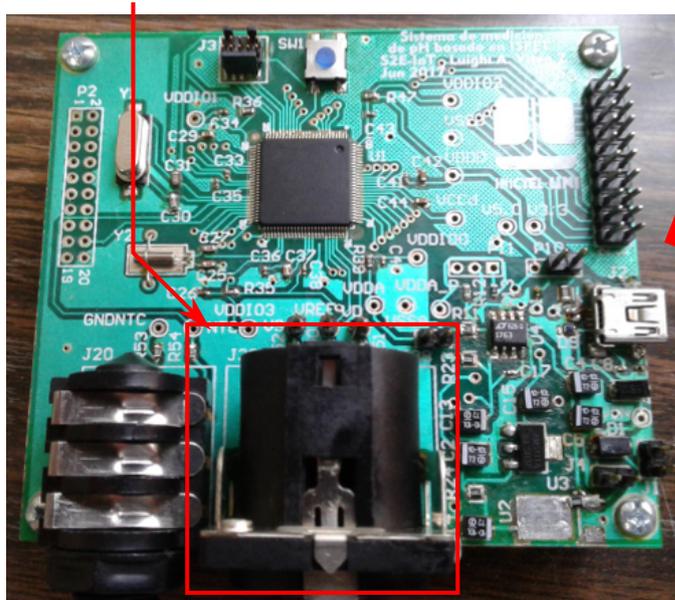


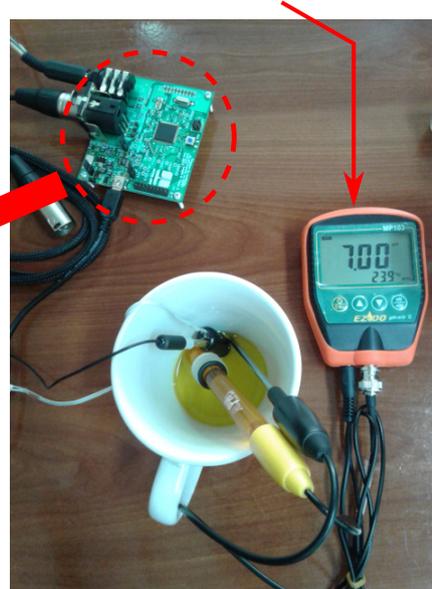
Fig. 4.31: Ensamblado de conector de sensor de temperatura y lectura de valores obtenidos mediante UART.

Conector XLR para ISFET



Ensamblado completo de tarjeta principal

pH-metro para validación



Prueba de tarjeta con ISFET

Fig. 4.32: Ensamblado de conector XLR y verificación del funcionamiento de tarjeta cuando se encuentra conectado al ISFET.

5. Luego, se realiza la verificación del ISFET. En este proceso se debe identificar adecuadamente cuáles son las terminales que le corresponde al drenador, fuente y electrodo de referencia, de acuerdo al diseño empleado en el PCB, y luego comprobar las tensiones de salida del dispositivo, una vez conectado a la tarjeta.

En la Fig. 4.32 se puede apreciar el ensamblado completo de la tarjeta principal, lo cual culmina con el ensamblado del conector XLR, empleado para conectarlo con el ISFET. Así mismo, se aprecia el funcionamiento de la tarjeta cuando se conecta con el sensor situado en un buffer de pH 7, para lo cual se emplea también un pH-metro comercial (MP-103, [185]) a fin de validar las mediciones obtenidas.

6. Finalmente, se realiza el montaje con la tarjeta de comunicaciones. Previo a ello, se comprueba el adecuado funcionamiento del módulo CC2650 con una tensión de 3.3V, a través de la tarjeta de evaluación SmartRF06. Una vez ensamblados los conectores y demás componentes en la tarjeta de expansión, se realiza la energización del sistema y se verifican los niveles de tensión proporcionados. Finalmente, se comprueba la recepción de la información de la tarjeta a través de un nodo receptor conectado a una computadora mediante puerto serial, donde se aprecia los mensajes recibidos. En la Fig. 4.33 se puede apreciar el ensamblado del sistema completo.

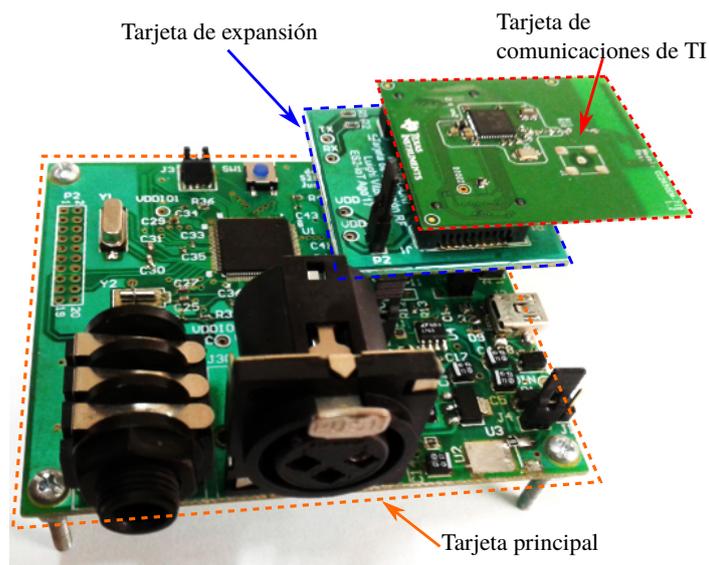


Fig. 4.33: Montaje de módulo de comunicaciones CC2650 sobre la tarjeta principal a través de la tarjeta de expansión.

4.6 Resumen del capítulo

La implementación del sistema de caracterización se realiza en un microcontrolador y una tarjeta de expansión con un convertor de corriente a tensión. Los componentes de software de este sistema se efectúan en Python, dotando de las características necesarias para la adquisición, así como el seguimiento del procesamiento para obtener las curvas características. El sistema de acondicionamiento de señal se implementa en el PSoC 5LP, estableciendo las características de los componentes, así como una adecuada asignación de pines. El sistema de procesamiento y de control se implementa en un flujo de software, también en el PSoC 5LP. El sistema de comunicación inalámbrica se implementa empleando los módulos de evaluación de Texas Instruments, a través del sistema operativo Contiki OS. El gateway empleado para el envío a Internet se implementa a través de un script en Python ejecutándose en una computadora. Los componentes del prototipo final se especifican empleando como referencia la tarjeta de desarrollo del PSoC 5LP. El prototipo se implementa en un PCB, diseñado de acuerdo a recomendaciones para un sistema de señal mixta, el cual es fabricado y luego ensamblado realizando pruebas a cada una de sus partes.

En este capítulo se ha detallado la implementación del sistema de caracterización para el ISFET y del sistema de sensado de pH tanto a nivel de software como de hardware. El prototipo construido es empleado para la obtención de los resultados que se presentarán en el siguiente capítulo.

CAPÍTULO 5

RESULTADOS EXPERIMENTALES

En el presente capítulo se expone el conjunto de pruebas efectuadas con el sistema implementado para el sensado de pH basado en ISFET.

En primer lugar se realiza una prueba de calibración con el sensor ISFET. Luego, se presentan los resultados de caracterización del dispositivo y se efectúa una primera prueba solo del sistema de acondicionamiento de señal. Posteriormente, se realizan pruebas al prototipo construido con mediciones del consumo de corriente de cada una de las etapas, la calibración de los sensores empleados y, finalmente, la transmisión inalámbrica y envío de datos a Internet.

5.1 Verificación de información del fabricante

El sensor de ISFET empleado en este trabajo ha sido fabricado por una startup Tailandesa denominada WINSENSE. Ellos proporcionan junto con el sensor y su electrodo de referencia, un manual con los datos técnicos y procedimiento de calibración, así como un circuito de acondicionamiento de prueba para poder verificar su funcionamiento. Estos materiales se exponen en Fig. 5.1.

Una de las primeras tareas que se tienen, es la comprobación de las mediciones del sensor. Para ello, se realiza el proceso de calibración del sensor con soluciones buffer de pH 4, 7 y 10, los cuales se pueden apreciar en la Fig. 5.2. Para efectuarlo se requiere de un multímetro,



Fig. 5.1: Materiales proporcionados por WINSENSE: sensores ISFET (izquierda), electrodos de referencia (centro) y circuito de lectura (derecha).



Fig. 5.2: Soluciones de calibración: buffer de pH 4 (izquierda), buffer de pH 7 (centro) y buffer de pH 10 (derecha).



Fig. 5.3: Circuito para la calibración de los sensores.

para la medición de la tensión, un termómetro para la medición de la temperatura (en este caso se emplea la termocupla de un multímetro para obtener esta medida), el circuito de acondicionamiento para la adaptación del valor del pH a un valor de tensión, un recipiente para colocar los líquidos de la muestra y, finalmente, el ISFET con su electrodo de referencia. La disposición de estos materiales se puede apreciar en la Fig. 5.3.

Luego de efectuar los pasos mencionados en su manual, se obtienen tres valores de tensión correspondientes a cada uno de los buffers empleados para el proceso. La curva generada con estos datos es una recta la cual se puede apreciar en la Fig. 5.4 y cuya ecuación es la (5.1):

$$V_{\text{ISFET}} = 0.0525 \times \text{pH} + 0.80883333 \quad (5.1)$$

De esta ecuación se puede obtener la sensibilidad del sensor, la cual es equivalente a 0.052V por pH, correspondiendo con los valores mostrados en el manual del dispositivo.

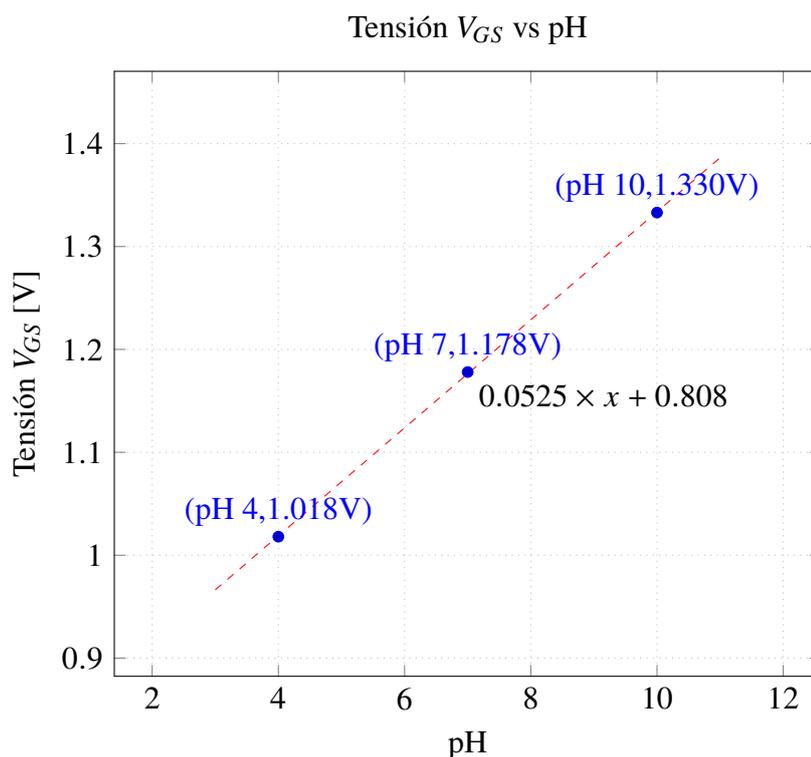


Fig. 5.4: Resultado de la calibración del sensor ISFET.

5.2 Caracterización del dispositivo

Para caracterizar el dispositivo se realizan un conjunto de pasos, los cuales comprenden la obtención de curvas características, la extracción de parámetros a partir de éstas y la adaptación del modelo con los parámetros obtenidos. Para hallar las curvas características, se emplea el sistema de medición diseñado en la sección 3.1 e implementado en la sección 4.1. Luego, para la extracción, se aplica la metodología señalada en la sección 2.10.

5.2.1 Resultados del sistema de medición para caracterización del ISFET

Los resultados obtenidos con el sistema implementado se pueden visualizar en Fig. 5.5 y Fig. 5.6, las cuales corresponden, respectivamente, a las dos pruebas realizadas para la caracterización del dispositivo: la prueba de i_{DS} vs. v_{DS} y de i_{DS} vs. v_{GS} .

En la Fig. 5.5, se aprecian los datos obtenidos del ADC, correspondiente a la tensión en v_{DS} , y del ADC, correspondiente a la tensión en la salida del transductor de corriente. Al ser la corriente positiva, este valor se encuentra sobre el de v_{DS} , lo cual deriva en una gráfica por encima de la anterior, cuya diferencia es proporcional al valor de la corriente multiplicado por la resistencia del TIA. Se aprecia, asimismo, que la diferencia para valores pequeños de v_{DS} es insignificante, aumentando progresivamente hasta mantenerse casi constante.

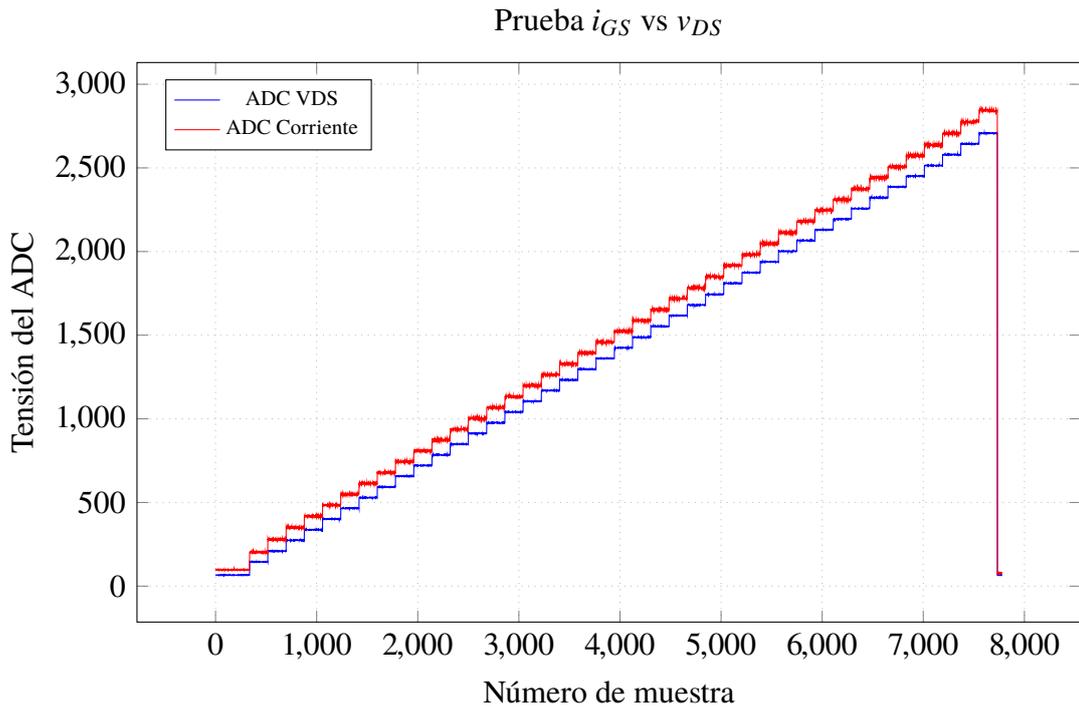


Fig. 5.5: Valores obtenidos de la prueba i_{DS} vs. v_{DS} para una tensión de $V_{GS} = 905\text{mA}$

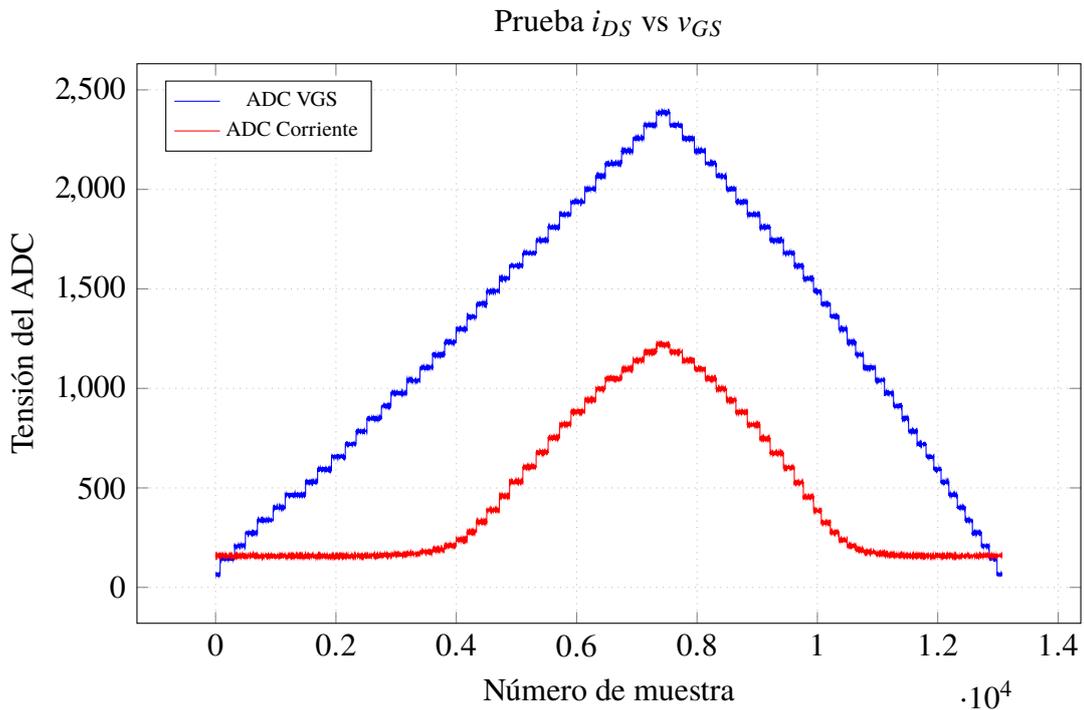


Fig. 5.6: Valores obtenidos de la prueba i_{DS} vs. v_{GS} para una tensión de $V_{DS} = 109\text{mA}$

En la Fig. 5.6, de igual modo, se aprecian las tensiones obtenidas de los ADCs correspondientes a v_{GS} y el transductor de corriente, respectivamente. En este caso, el valor de la tensión a la salida del TIA se encuentra por encima de un valor constante, el cual es igual a la

tensión V_{DS} empleada para la prueba, cuyo comportamiento se refleja también en la gráfica presentada, donde la curva se mantiene constante para valores pequeños y relativamente iguales a la tensión V_{DS} hasta llegar a un punto de inflexión a partir del cual aumenta esta diferencia, significando con ello un aumento en la corriente de drenador.

Una diferencia en los ensayos realizados se manifiesta en que el primer ensayo emplea la función automática de control del DAC, mientras que el segundo lo hace mediante la función manual. Esto se evidencia en los anchos de los escalones, puesto que, mientras en el primero los anchos son relativamente iguales, en el segundo caso, estos varían en cada nivel de tensión. Si bien la función automática facilita las mediciones en el sistema, tiene el inconveniente de no regular adecuadamente el tiempo de establecimiento, mientras que en el caso del control manual, es posible mantener los niveles de tensión hasta lograr una adecuada respuesta del sensor. Afortunadamente, dado que el ISFET cuenta con un tiempo de respuesta relativamente corto, la función automática es suficiente para caracterizarlo.

Estas pruebas se ejecutaron varias veces para diferentes valores de tensión a fin de obtener la familia de curvas para cada uno de los ensayos. Sin embargo, como es posible apreciar en las gráficas, es necesario procesar estos datos a fin de extraer la información útil para construir las mencionadas curvas características del sensor.

Las curvas resultantes del procesamiento efectuado a los datos adquiridos corresponden a las dos familias de curvas i_{DS} vs. v_{GS} y i_{DS} vs. v_{DS} , tal como se menciona en 2.9.

Para el caso de la familia i_{DS} vs. v_{GS} , estas curvas se pueden apreciar en Fig. 5.7. Aquí se puede observar que el resultado no es del todo lineal, puesto que cuenta con efectos de la variación de la movilidad efectiva, adicionalmente a la longitud de canal efectivo y las resistencias parásitas en drenador y fuente.

Por otra parte, la familia de curvas i_{DS} vs. v_{DS} se aprecia en Fig. 5.8. Allí se puede identificar las dos zonas de operación del dispositivo, además del incremento progresivo de la corriente de saturación conforme aumenta la tensión en V_{GS} . Así mismo, se puede apreciar el efecto de modulación de la corriente en la zona de saturación, lo cual se presenta como rectas para los valores de tensión superiores a la tensión de estrangulamiento.

5.2.2 Extracción de parámetros aplicado a curvas obtenidas del proceso de caracterización del ISFET

En este caso se aplicará el método a los datos obtenidos en la caracterización del dispositivo (Fig. 5.7 y Fig. 5.8). De acuerdo a lo revisado en la sección 2.10 para la extracción

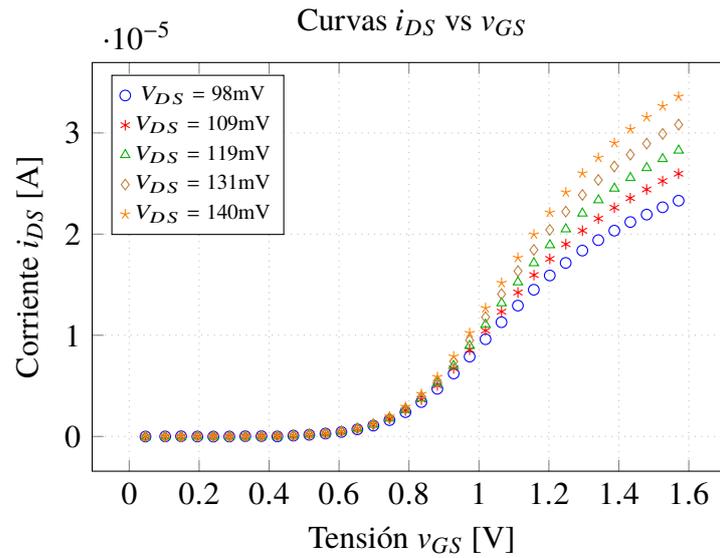


Fig. 5.7: Curvas características obtenidas de la medición de i_{DS} vs. v_{GS} para diferentes valores de V_{DS} a un $\text{pH}=7$,

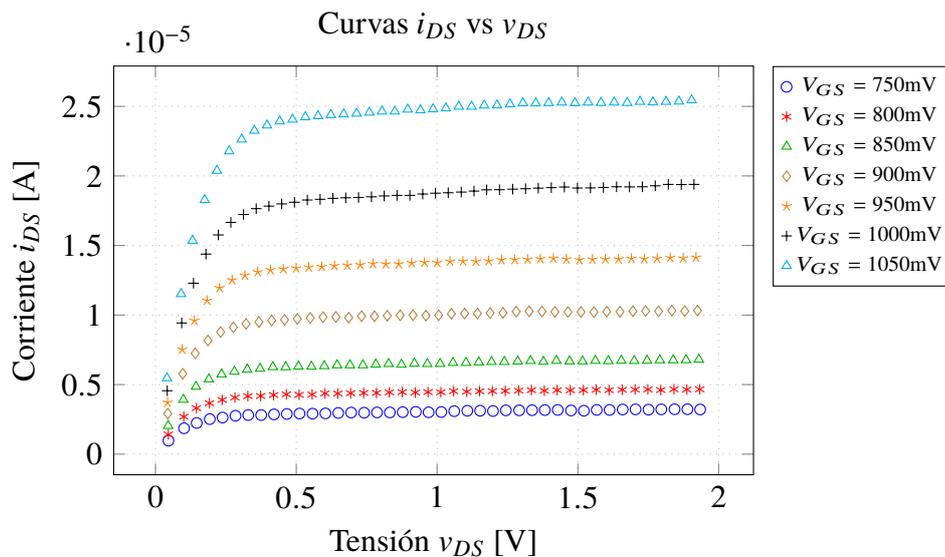


Fig. 5.8: Curvas características obtenidas de la medición de i_{DS} vs. v_{DS} para valores de V_{GS} desde 750mV hasta 1.050V a un $\text{pH}=7$.

se emplea un método basado en el ELR, el cual requiere de las curvas de i_{DS} vs. v_{GS} , para obtener el valor de la tensión umbral V_{T0} .

Este procedimiento se puede apreciar en la Fig. 5.9. Allí se puede visualizar la gráfica de las curvas características i_{DS} vs. v_{GS} , representadas mediante líneas sólidas; luego, la curvas de transconductancia g_m , representadas con curvas punteadas; y las líneas tangentes, mediante trazos discontinuos. De los puntos en los cuales interseca a la recta de corriente $i_{DS} = 0$, se obtiene el valor de la tensión umbral aumentada en $v_{DS}/2$.

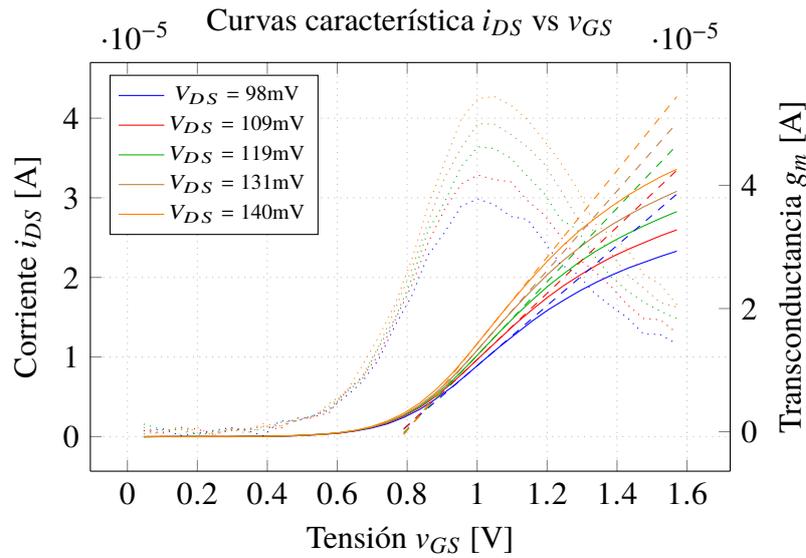


Fig. 5.9: Aplicación del método ELR a la familia de curvas i_{DS} vs. v_{GS} del ISFET medidas para valores de V_{DS} : 0.098, 0.109, 0.119, 0.131, 0.140V con un valor de $\text{pH} = 7$

La puntos obtenidos para cada uno de los valores de V_{DS} empleados en la medición se muestran en la Fig. 5.10. Luego, al efectuar la regresión lineal, se obtiene la ecuación mostrada en (5.2):

$$V_{TH} = 0.0427365V_{DS} + 0.71135035 \quad (5.2)$$

Se puede verificar de la ecuación que el valor de V_{TO} es igual a 711mV. En este caso, se trata de la tensión del ISFET, luego para obtener la tensión del MOSFET, la cual debe incluirse en el modelo, se adiciona la constante calculada de la simulación. Por lo que el valor de $V_{TO(\text{MOS})}$ es igual al mostrado en (5.3):

$$V_{TO(\text{MOS})} = V_{TO(\text{ISFET})} + 0.225V = 0.936V \quad (5.3)$$

Luego, se puede aplicar el método de la transconductancia cruzada, analizado previamente, para calcular el valor de U_0 en el modelo. Para ello se grafica nuevamente la transconductancia pero solo para tensiones mayores al de máxima transconductancia, lo cual se puede apreciar en Fig. 5.11. De los valores obtenidos se calcula un promedio de $152\text{cm}^2/\text{V s}$, correspondiente al valor de U_0 .

5.2.3 Verificación de la adaptación del modelo del ISFET

Como se mencionó al inicio de la sección, el procedimiento seguido hasta ahora tiene la finalidad de adaptar el modelo del ISFET al dispositivo real. Para ello, se han efectuado los procesos de caracterización y de extracción de parámetros. De este último, se han obtenido los

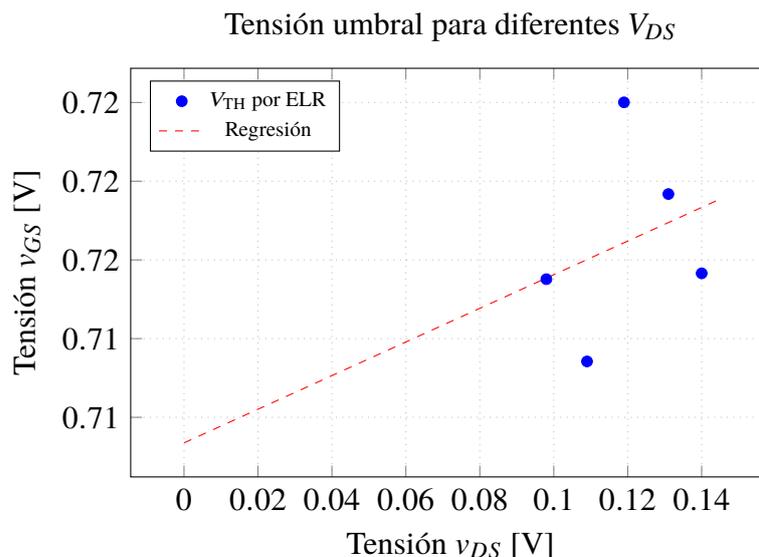


Fig. 5.10: Gráfica de la regresión lineal de los valores de tensión umbral obtenidos mediante el método ELR aplicado a datos experimentales del ISFET para un valor de $\text{pH} = 7$.

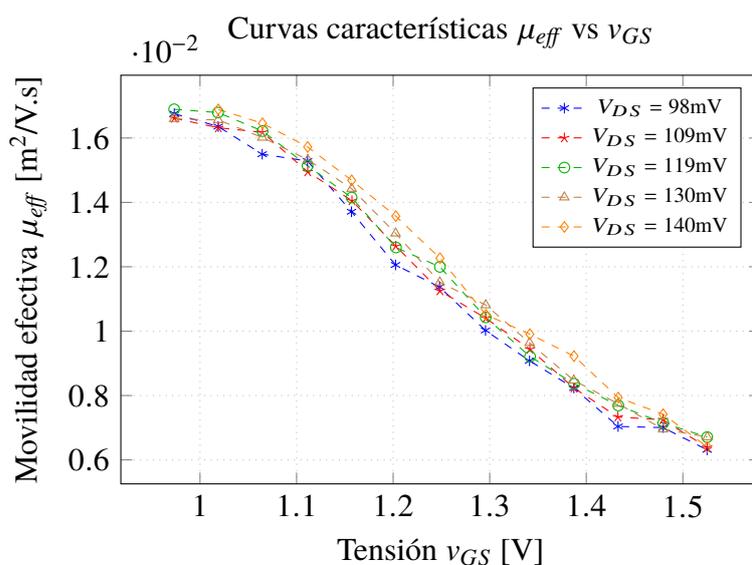


Fig. 5.11: Curva de movilidad efectiva calculada a partir de las mediciones de la corriente i_{DS} vs. v_{GS} para varios valores de V_{DS} .

valores de la tensión umbral V_{T0} y la movilidad U_0 . Adicionalmente, dentro del modelo es necesario cambiar también los parámetros correspondientes a las características geométricas como es el caso de la longitud del canal L , el ancho del canal W y el grosor del óxido TOX , propias del proceso de fabricación del dispositivo. De acuerdo a la información proporcionada por los representantes de WINSENSE, los valores de estos parámetros son los mostrados en la TABLA N° 5.1.

Luego de ingresar estos valores, junto con los encontrados mediante el proceso de extrac-

TABLA N° 5.1: Parámetros proporcionados por WINSENSE

Parámetro	Valor
Longitud de canal (L)	100 μ m
Ancho de canal (W)	2000 μ m
Grosor de óxido (TOX)	30nm
Grosor de capa de Si ₃ N ₄	10nm

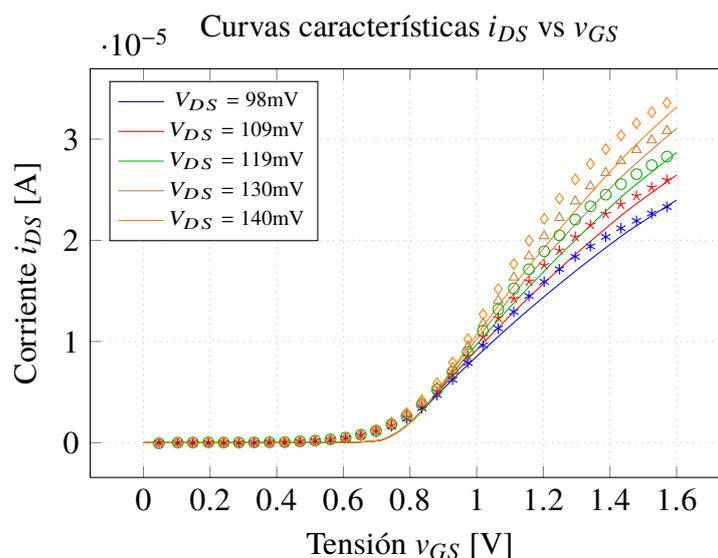


Fig. 5.12: Comparación de las mediciones de la familia de curvas i_{DS} vs. v_{GS} , respecto a la simulación del modelo adaptado.

ción de parámetros, y hacer un ajuste posterior de acuerdo a los datos de las mediciones, se obtiene como resultado el modelo del dispositivo real, cuya simulación se puede apreciar en la Fig. 5.12. En esta imagen es posible identificar las diferencias respecto a las mediciones efectuadas.

En este caso, las regiones que se asemejan más se encuentran en la región cercana a la tensión umbral, sobretodo en la zona de la máxima transconductancia, puesto que para valores superiores a ésta se empiezan a distanciar más. Puesto que el valor de las resistencias de los contactos es calculado a partir de aproximaciones, esto también introduce incertidumbre al momento de obtener las curvas características.

Así mismo, es posible ver el resultado de la adaptación del modelo al simular las curvas i_{DS} vs. v_{DS} y compararlas con los datos obtenidos en las mediciones, tal como se puede apreciar en la Fig. 5.13. En este caso, las corrientes de saturación distan más de los valores experimentales. No obstante, en la zona óhmica todavía se mantiene una correspondencia entre ambos valores. Puesto que ésta es la zona de interés para las mediciones, la adaptación

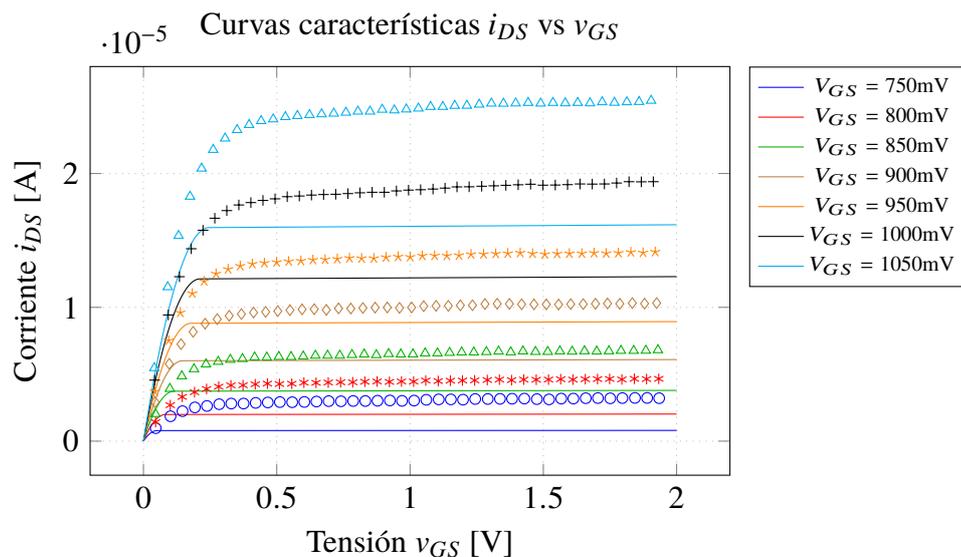


Fig. 5.13: Comparación de las mediciones de la familia de curvas i_{DS} vs. v_{DS} , respecto a la simulación del modelo adaptado.

del modelo se limitará hasta este punto.

5.3 Resultados del sensado de pH con el sistema de acondicionamiento del ISFET

Una vez caracterizado el dispositivo se realiza las pruebas con el sistema de sensado de pH, cuya primera etapa corresponde al sistema de acondicionamiento. En esta sección se realiza una prueba considerando solo el circuito de acondicionamiento de señal con la finalidad de evaluar su comportamiento ante variaciones de pH. Las conexiones de los pines asignados en la tarjeta con el dispositivo se realizan de acuerdo a la implementación mostrada en la sección 4.2.

El ISFET se sitúa en un recipiente donde se encuentra la solución de prueba y cuyo pH se desea medir. Normalmente esta solución es un buffer con un pH conocido. Luego, se realizan las pruebas con diferentes soluciones de distintos valores de pH para identificar las variaciones en la tensión de salida, las cuales deben ser proporcionales al valor de pH.

Adicionalmente, se emplea un instrumento de medición MP-103 [185], el cual cuenta con una sonda de temperatura y un electrodo de pH, con la finalidad de validar los resultados obtenidos con las mediciones del ISFET.

Los resultados obtenidos son los mostrados en TABLA N° 5.2. En esta tabla, la columnas correspondientes al pH y la temperatura son las mediciones obtenidas con el instrumento MP-103, mientras que la tensión es la salida del sistema de adaptación, medida con el voltímetro, y la corriente, es la corriente de drenador, medida con el amperímetro.

TABLA N° 5.2: Resultados del sistema de adaptación

pH	Tensión	Corriente	Temperatura
-	[V]	[μ A]	[$^{\circ}$ C]
4.08	2.032	24.66	25.7
5.03	2.070	24.63	25.4
5.93	2.087	24.60	25.1
7.02	2.145	24.53	25.9
8.08	2.215	24.48	24.8
9.04	2.234	24.50	24.9
9.32	2.248	24.49	26.2

Con los datos apreciados en la tabla, es posible notar que se presenta un cambio de temperatura durante el proceso de medición. Esto se debe a que no se cuenta con un sistema de adecuación de temperatura para mantenerla en un valor estable durante las mediciones. Por otra parte, se detecta también algunas variaciones en la corriente de drenador, aunque son relativamente despreciables al ser inferiores a 0.2μ A.

Así mismo, con estos datos se puede graficar la tensión respecto al valor de pH, a fin de obtener la curva de respuesta del sistema. El resultado se puede apreciar en Fig. 5.14. La regresión es una recta cuya ecuación es la presentada por (5.4):

$$V_{\text{out}} = 1.850 + 0.0428\text{pH} \quad (5.4)$$

Allí se identifica una sensibilidad del dispositivo de 42mV/pH . Este valor es inferior a los valores obtenidos con el sistema de medición proporcionado por el fabricante, e inferior al resultado obtenido con la simulación, lo cual puede deberse a la corriente empleada para la polarización del ISFET, así como a una degradación del electrodo de referencia.

En la gráfica se puede identificar dos puntos correspondientes al pH 5.93 y 8.08, los cuales distan bastante de la recta de regresión. También en la tabla se puede identificar que para estos valores de pH se han efectuado las mediciones con los menores valores de temperatura, lo cual, aunque no es determinante, ofrece una primera muestra de la influencia que puede tener este parámetro en la medición del pH.

Puesto que la influencia de la temperatura es un factor importante en la medición de pH con el ISFET, se procede a verificar el algoritmo empleado para la compensación respecto a los cambios de temperatura, el cual se incorpora en el software del dispositivo.

Con las relaciones obtenidas en (3.25) y (3.26) es posible obtener los valores de pH compensados y sin compensar, a partir de las mediciones mostradas en Fig. 3.24 y Fig. 3.25,

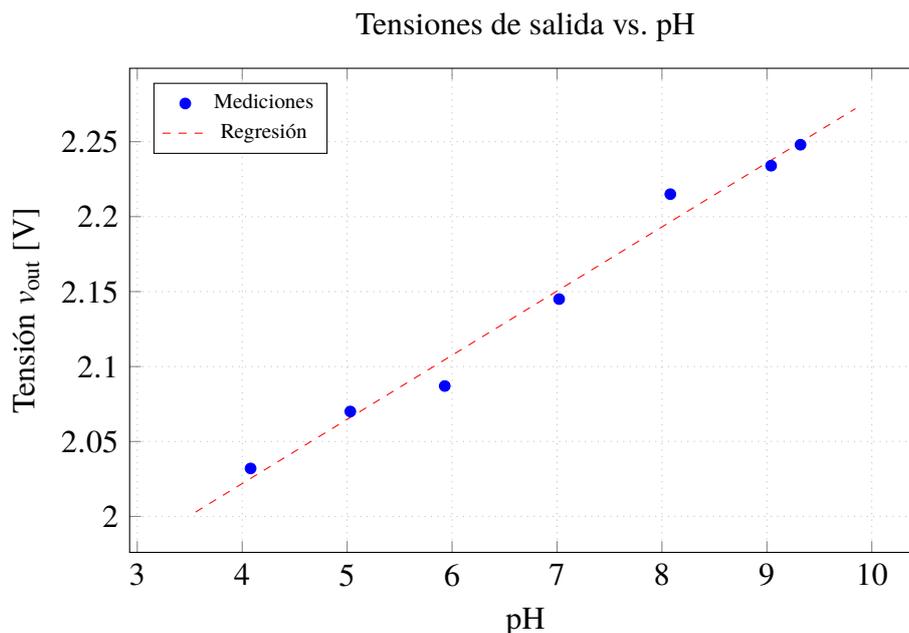


Fig. 5.14: Gráfica de los valores de tensión respecto a las variaciones de pH, empleando los valores de TABLA N° 5.2.

lo cual deriva en los resultados mostrados en Fig. 5.15 y Fig. 5.16, correspondientes a los buffers de pH 4 y pH 7 respectivamente.

Como es posible apreciar, en ambos casos se identifica una variación del valor de pH entre ± 0.02 su valor a 25°C . Esta variación representa el margen de tolerancia dentro del cual se encuentra el valor de pH compensado. Este valor se considera aceptable, teniendo en cuenta que la resolución del dispositivo también se encuentra en el mismo orden (0.02).

Por otra parte, se puede verificar en las gráficas que a una temperatura de 25°C los valores del pH compensado y pH sin compensar coinciden, mientras que para valores inferiores de temperatura, el valor de pH sin compensar se encuentra por encima del compensado, y para valores superiores de temperatura, el valor sin compensar es inferior.

5.4 Consumo de energía

Una vez evaluado el funcionamiento del sistema de acondicionamiento, se realizan las pruebas con el prototipo construido para este fin, considerando para ello los objetivos trazados en la tesis. Uno de los objetivos principales es la reducción del consumo de energía del sistema de adaptación del sensor, tomando como referencia el electrodo de bulbo de vidrio, el cual, de acuerdo a mediciones efectuadas, tiene un consumo de 13mA, mientras que en referencias bibliográficas se afirma un valor de 8mA.

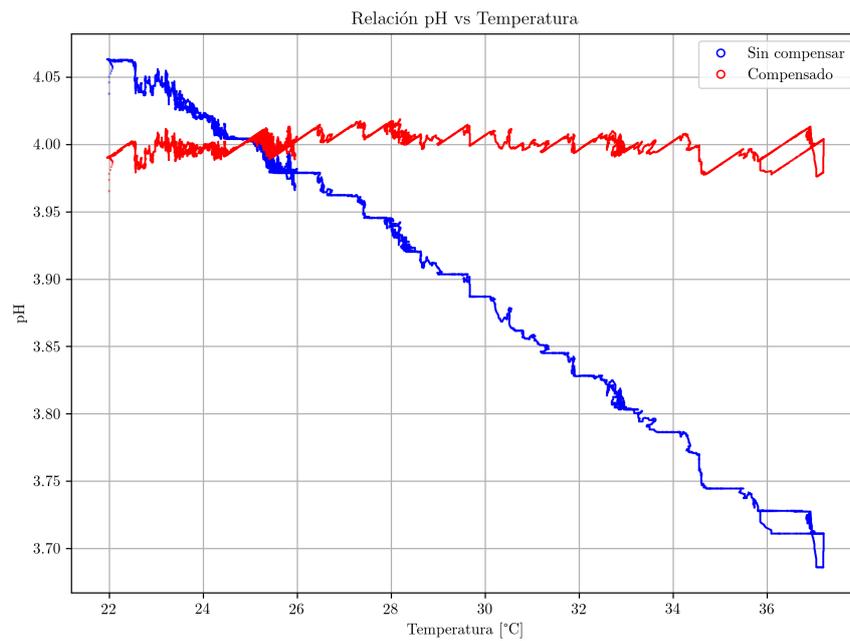


Fig. 5.15: Resultado de la compensación de la temperatura para buffer con pH 4.

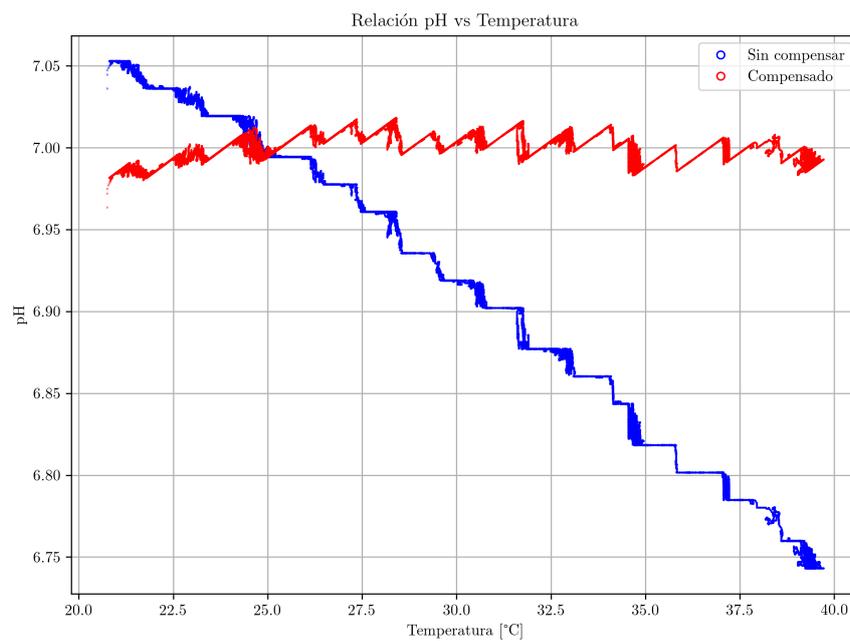


Fig. 5.16: Resultado de la compensación de la temperatura para buffer con pH 7.

Puesto que el sistema está conformado por varias etapas, se tomará la medición de la corriente de operación de cada una de éstas. En primer lugar, se toma la medida sólo del sistema de alimentación, cuando los conectores que lo comunican con el resto del prototipo se encuentran desconectados. Posteriormente, se realiza la conexión de las fuentes para el dominio digital y analógico, y se efectúa la medición del subsistema analógico sin sensores

TABLA N° 5.3: Mediciones efectuadas al sistema de monitoreo.

Bloque del sistema	Tensión [V]	Corriente [mA]	Potencia [mW]
Sistema de alimentación con VDDA y VDDD desconectados	4.137	4.530	18.741
Sistema global sin sensores ni tarjeta inalámbrica	4.137	13.013	53.835
	4.137	4.657	19.266
Subsistema analógico sin sensores	3.863	2.025	7.823
	3.863	0.002	0.008
Subsistema analógico con sensor de temperatura	3.863	2.124	8.205
	3.863	0.002	0.008
Subsistema analógico con sensores de temperatura y de pH	3.863	2.667	10.303
	3.863	0.002	0.008
Subsistema digital sin tarjeta de comunicaciones	2.903	6.532	18.962
	2.903	0.127	0.369
Tarjeta de comunicaciones	2.867	9.821	28.157
	2.867	1.119	3.208
Sistema global	4.137	21.199	87.700
	4.137	5.429	22.460

y luego con los sensores. Luego, se realiza la medición del subsistema digital sin tarjeta inalámbrica y, después, se toma la medición con la tarjeta inalámbrica. Finalmente, se realiza la medición solo de la tarjeta inalámbrica. Estas mediciones se pueden verificar en la TABLA N° 5.3, mostrando los valores máximos y mínimos en cada etapa.

De acuerdo a la tabla, existe un costo fijo de energía debido a los reguladores, el cual representa una corriente de 4.53mA y una potencia de 18.741mW. Luego, también se identifica que el costo de energía representado por el sistema de adaptación, considerando al ISFET y al sensor de temperatura, corresponde a 2.667mA con una potencia de 10.303mW, del cual el sensor de temperatura representa un consumo de 100 μ A (correspondiente a la fuente de corriente empleado para su circuito de acondicionamiento), mientras que el ISFET representa un consumo de 543 μ A (correspondiente a los OPAMPs y las fuentes de tensión y corriente). Respecto al subsistema digital, se obtiene un consumo máximo de 6.532mA, producido cuando se efectúa la transmisión mediante UART. Por otra parte, se obtiene también que el costo de energía del proceso de transmisión es uno de los más importantes del sistema dado que emplea una corriente de 9.8mA con una potencia de 87.7mW.

Durante los periodos de reposo, el sistema se establece en modo de bajo consumo de

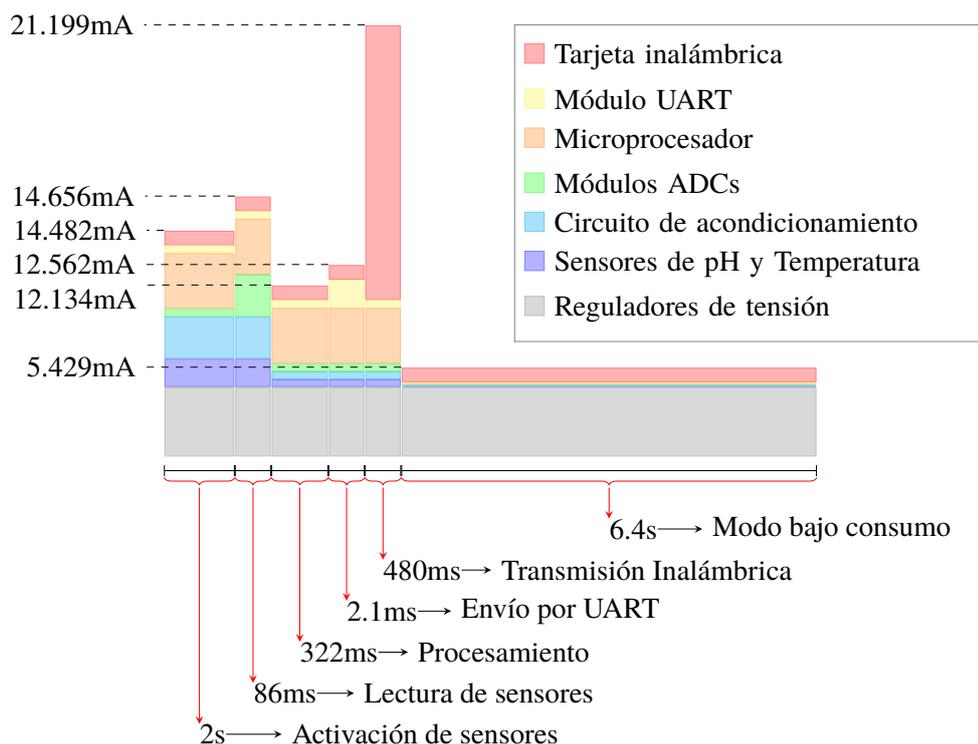


Fig. 5.17: Diagrama de consumo de energía del sistema de monitoreo.

energía obteniéndose los menores valores de corriente. De esta manera, se identifica que el subsistema analógico no tiene un consumo representativo en esta etapa (alrededor de $2\mu\text{A}$), mientras que el subsistema digital tiene una mayor influencia ($127\mu\text{A}$). No obstante, el bloque que tiene un mayor impacto en el consumo es la corriente de la tarjeta de comunicaciones con 1.119mA . Esto se debe a que, de acuerdo a la programación, si bien el sistema operativo establece el integrado en modo de bajo consumo cuando no se realiza la transmisión, el sistema se encuentra en continua espera de la recepción de datos en su puerto de UART, lo cual implica mantener activo el CPU durante esta etapa.

Así mismo, como se aprecia en la tabla, los costos de energía no son constantes durante el tiempo, sino que tiene valores máximos y mínimos de acuerdo a la etapa de operación en la que se encuentre. Estas etapas están reguladas por el sistema de control del dispositivo, mostrado en Fig. 4.18. Considerando estas fases se puede plantear un diagrama de consumo, tal como el presentado en la Fig. 5.17.

5.5 Calibración del sensor de temperatura y de pH

Previo al procedimiento de compensación de temperatura efectuado en la sección 3.3, se realiza un proceso de calibración del sensor de temperatura con la finalidad de obtener

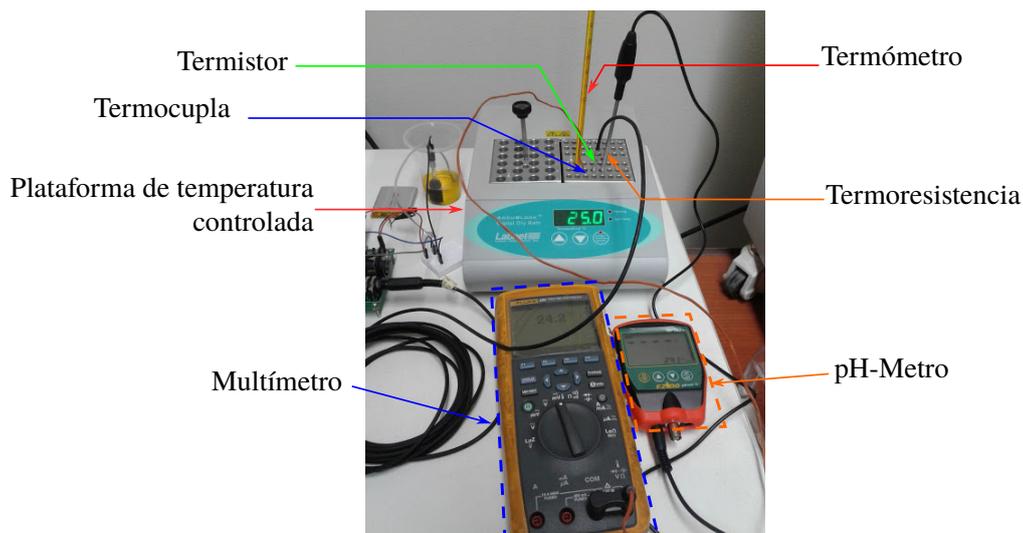


Fig. 5.18: Calibración del sensor de temperatura.

los valores más cercanos a los reales, comparándolo con las mediciones obtenidas con otros sensores. Las otras tecnologías de sensado comprenden: un termómetro de alcohol, una termoresistencia (incluida en el pH-metro MP-103[185]) y una termocupla (serie 80BK-A incluida en el multímetro FLUKE 289 [186]). Todos estos instrumentos de medición se sitúan sobre una plataforma que cuenta con un control de temperatura [187]. En la Fig. 5.18, se puede apreciar la disposición de los sensores para dos valores diferentes de temperatura.

Con dicha disposición, se realiza una serie de pruebas para varios valores de temperatura, obteniéndose los resultados mostrados en Fig. 5.19. Como instrumento patrón se considera, en este caso, al termómetro, por lo que constituye la referencia para determinar la temperatura real de la plataforma. Luego es necesario calibrar el valor del termistor respecto al valor del termómetro. Realizando una regresión lineal para estos se obtiene la ecuación (5.5):

$$T = -7.993 + 1.395 \times T_{\text{NTC}} \quad (5.5)$$

La relación presentada en (5.5) se ingresa dentro de la ecuación característica del termistor, con la finalidad de obtener los valores más cercanos a la temperatura real del líquido.

Luego de efectuar la calibración del sensor de temperatura, se efectúa la calibración del ISFET. En la sección de procesamiento 3.3, ya se había empleado una ecuación a partir de una calibración previa para obtener los valores de pH. No obstante, una vez implementado todo el sistema, es necesario realizar de nuevo este proceso.

Para efectuar dicha calibración es necesario emplear buffers de pH, que en este caso son de pH 4 y pH 7. Con estos buffers se realiza la medición del pH en una plataforma con

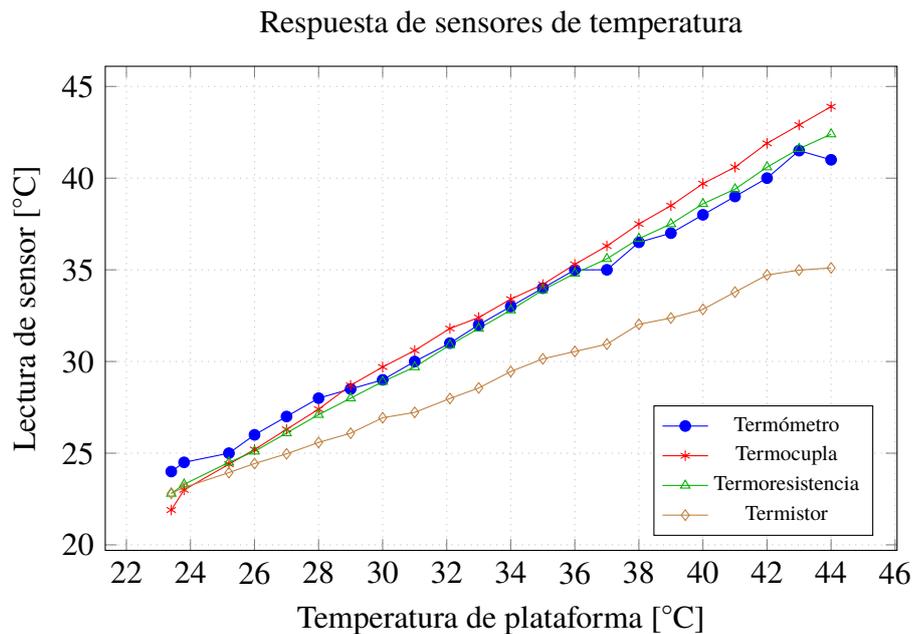


Fig. 5.19: Valores obtenidos con diferentes sensores de temperatura.

temperatura controlada tal como se empleó en la calibración del sensor de temperatura. Los resultados se pueden apreciar en Fig. 5.20 y Fig. 5.21. Como es posible apreciar, los valores no coinciden con los niveles de pH de las soluciones por lo que se requiere efectuar una calibración a partir de los valores obtenidos. De esta manera, a una temperatura cercana a 25°C que constituye la temperatura de referencia para las mediciones, se obtiene el valor promedio de éstas.

En el primer caso presentado en Fig. 5.20, se obtiene un valor promedio de pH compensado 2.803 y de pH sin compensar 2.794, a una temperatura promedio de 25.33°C. Como medida representativa se opta por tomar el promedio de los dos valores, y luego retornar al valor de tensión, igual a 908.527mV. En el segundo caso, presentado en Fig. 5.20, los valores promedio de pH compensado y sin compensar son 5.88 y 5.866, respectivamente, a una temperatura media de 25.70. Con estos valores se obtiene una tensión promedio de 1065.602mV. Finalmente, con los dos valores de tensión se puede obtener la nueva ecuación de calibración del sensor de pH, mostrada en (5.6):

$$V_{out} = 699.094 + 52.358 \times \text{pH} \quad (5.6)$$

Despejando el término de pH de esta ecuación se obtiene la relación para ingresarla en el cálculo de pH, a partir de la tensión medida.

Una vez ingresados estos valores se realiza una nueva prueba con los buffers de pH para

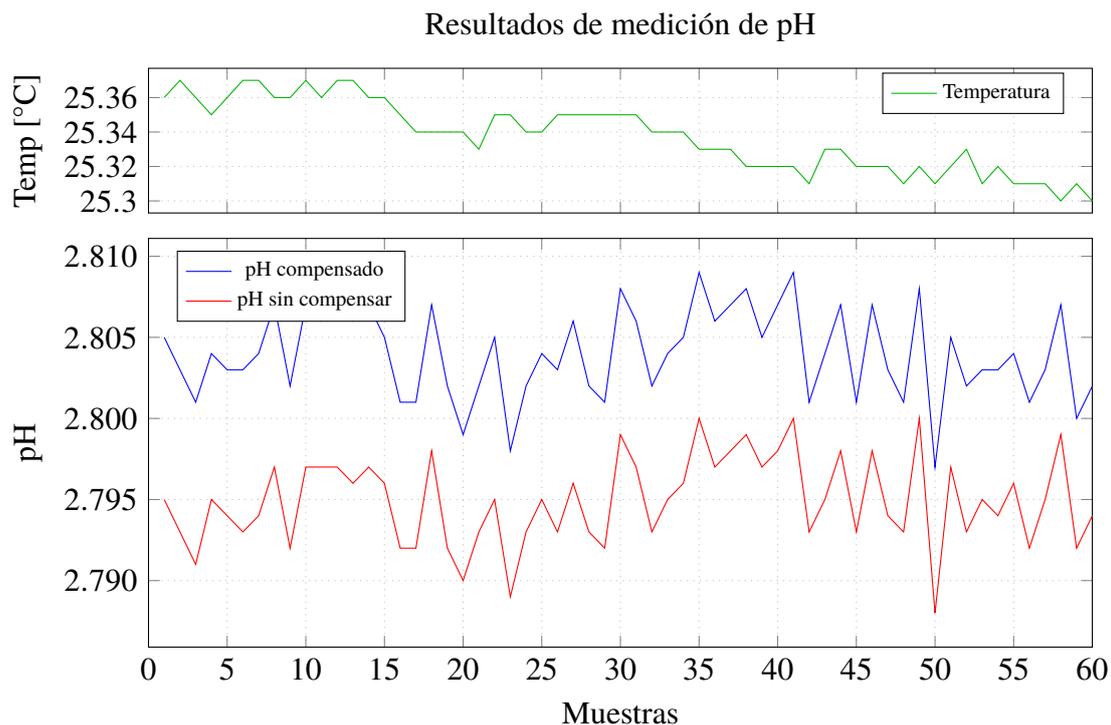


Fig. 5.20: Valores para calibración con buffer pH4

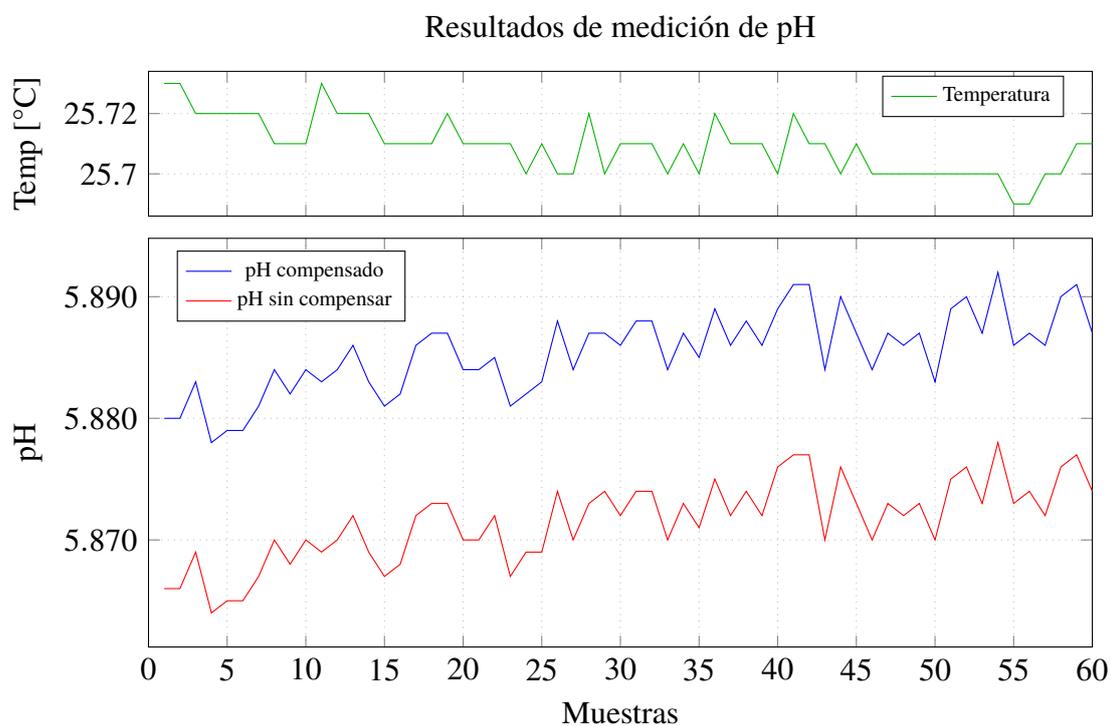


Fig. 5.21: Valores para calibración con buffer pH7

verificar el correcto funcionamiento del dispositivo. El resultado se puede apreciar en la Fig. 5.22 y la Fig. 5.23. En el primer caso se aprecia que el valor del pH es bastante cercano a 4, con un promedio de 4.007, y una variación de ± 0.01 a una temperatura promedio de

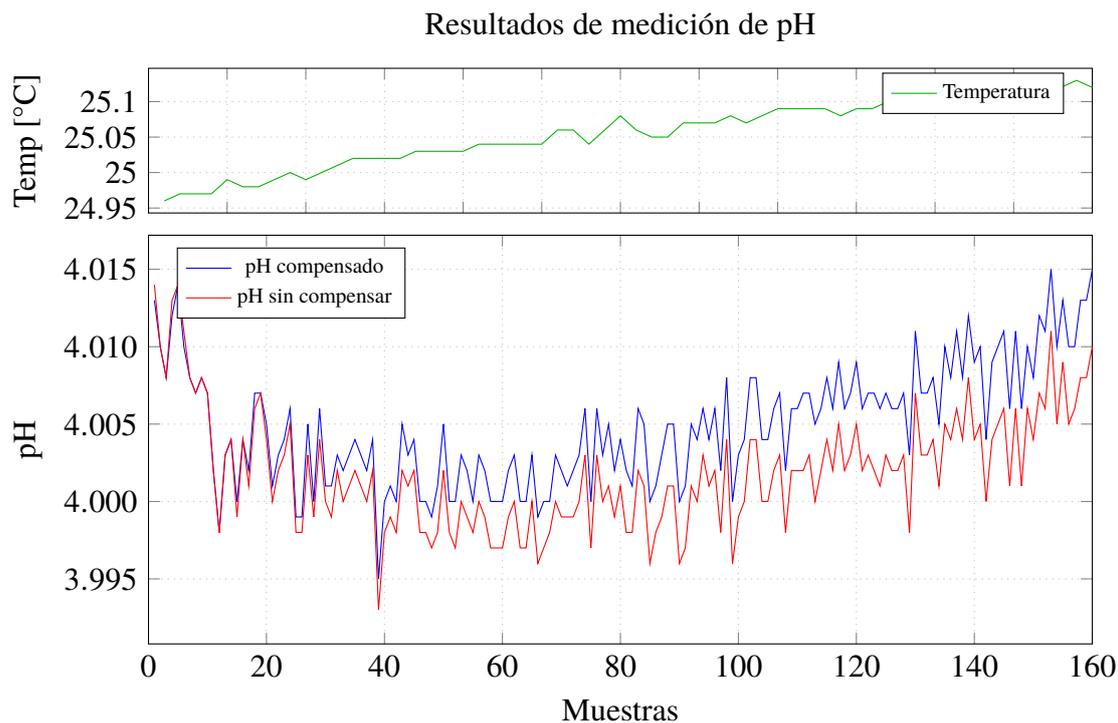


Fig. 5.22: Resultados con buffer pH4

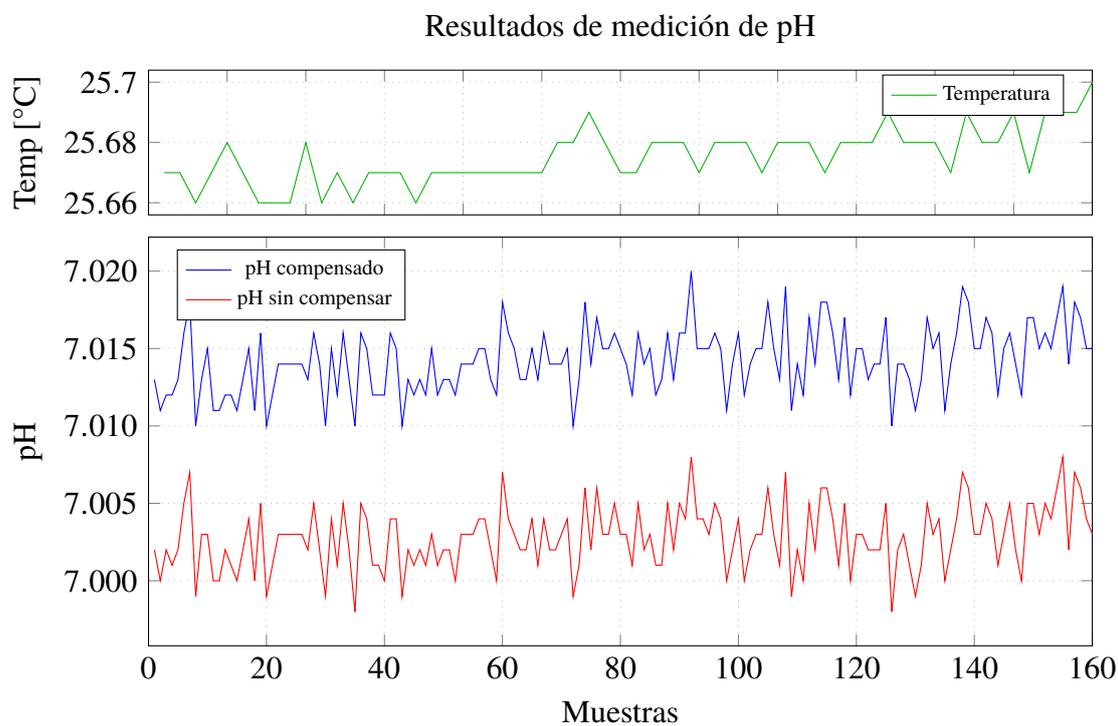


Fig. 5.23: Resultados con buffer pH7

25.149°C. Algo similar se obtiene en el segundo caso, con un valor de 7.015 y una variación de ± 0.005 , a una temperatura de 25.694°C. En ambos casos, la variación se encuentra dentro de la tolerancia esperada de acuerdo al diseño planteado ($\pm 0.02\text{pH}$).

5.6 Transmisión inalámbrica y envío de datos a Internet

Una vez calibrado el sistema se puede efectuar la prueba de envío de datos a Internet, lo cual constituye el tercer objetivo de la tesis. Para realizar dicha prueba, se emplean plataformas de IoT gratuitas escogidas de la Fig. 3.29, orientadas hacia el dominio de visualización, monitoreo y desarrollo de aplicaciones. De esta manera se eligen: Ubidots [181], GroveStreams [188], ThingSpeak [163] y Carriots [189].

La prueba se efectúa situando el sensor en una muestra de buffer de pH 7, registrando los cambios de temperatura y variaciones en la medición de pH durante aproximadamente 24 horas.

El resultado de Ubidots se puede apreciar en la Fig. 5.24. La implementación del script se realiza a través de una librería creada para esta plataforma en python, la cual internamente implementa la comunicación mediante HTTP, a través de una interfaz REST. Las limitaciones de esta plataforma están orientadas a la capacidad de envío de información, la cual disminuye conforme se utiliza la plataforma. Así mismo, presenta retardos de conexión, lo cual implica que algunas de las muestras no pueden ser enviadas a tiempo, perdiéndose algunos datos.

El resultado de GroveStreams se aprecia en Fig. 5.25. En este caso, no existe una librería para comunicarse con la plataforma, por lo que se emplean librerías proporcionadas por python para el envío de datos mediante HTTP, empleando también la estructura REST. Esta plataforma tiene ciertas limitaciones referidas a su visualización, la cual no se actualiza automáticamente, a la vez que no permite más de un envío cada 10 segundos.

Respecto a la plataforma ThingSpeak, los resultados se muestran en Fig. 5.26. Para el envío de la información a la plataforma se emplea el protocolo MQTT, diseñado para envío de datos en aplicaciones de IoT, utilizando una librería en python que implementa dicho protocolo. La plataforma tiene la ventaja de tener una integración con funciones de MatLab, al ser elaborado por el mismo fabricante. Así mismo, tiene la opción de exportar la información a una tabla en excel o un archivo separado por comas. No obstante, tiene la desventaja de que no se puede editar individualmente la información de los canales.

Por otra parte, respecto a Carriots, el resultado se muestra en Fig. 5.27. Para el envío de información se emplea también el protocolo MQTT, con una estructura REST para los datos, utilizando un código en python similar al anterior. Carriots permite el envío de información a su plataforma, sin embargo, tiene varias limitaciones en su cuenta gratuita, tales como la cantidad de *streams* que se pueden enviar por día (500) y por minuto (50).

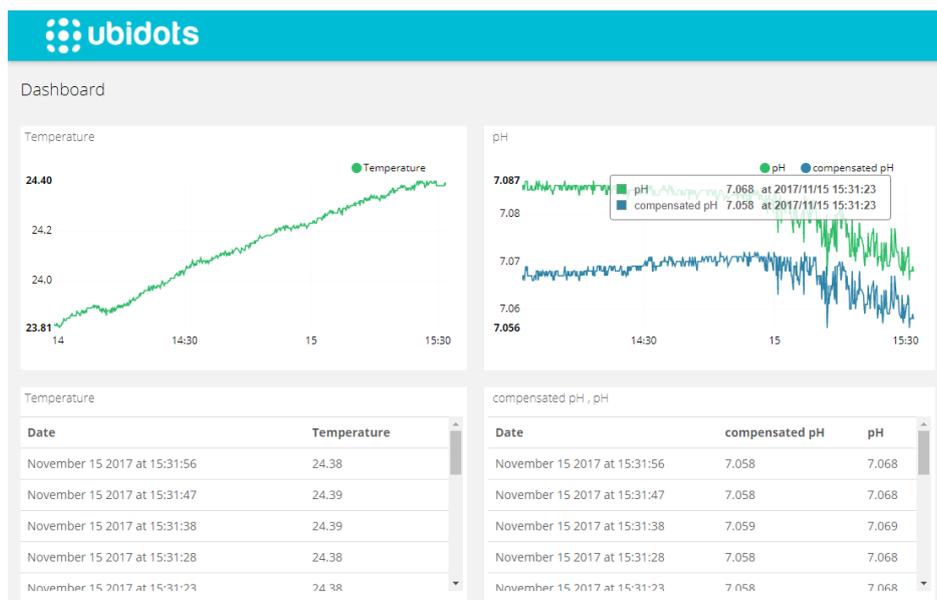


Fig. 5.24: Captura de datos enviados a Ubidots



Fig. 5.25: Captura de datos enviados a GroveStreams

Desde cualquiera de las cuatro plataformas es posible descargar la información enviada por el sistema de medición. Tomando el caso de ThingSpeak, al descargar los datos y graficarlos, se puede obtener el resultado mostrado en Fig.5.28. Allí se aprecia la variación de la temperatura y del pH durante 24 horas de operación. La prueba se ha efectuado un día después de la calibración presentada en Fig.5.23, obteniéndose un valor de pH promedio de 7.074 a una temperatura media de 23.8°C.

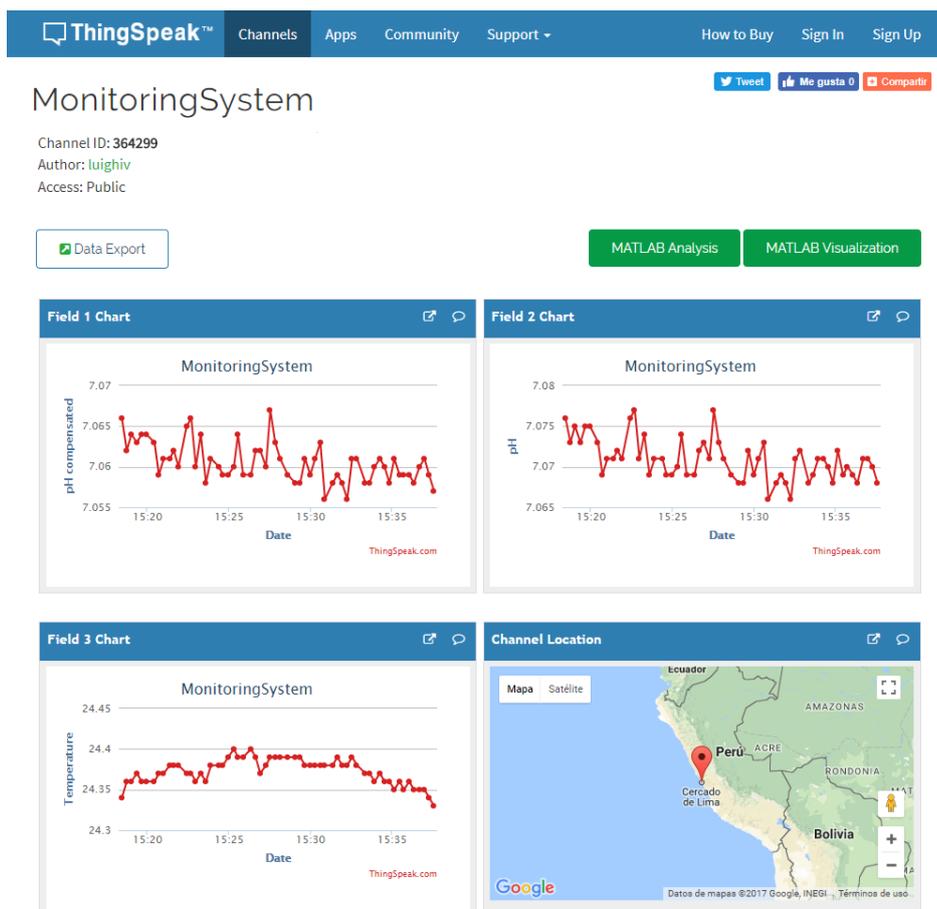


Fig. 5.26: Captura de datos enviados a ThingSpeak

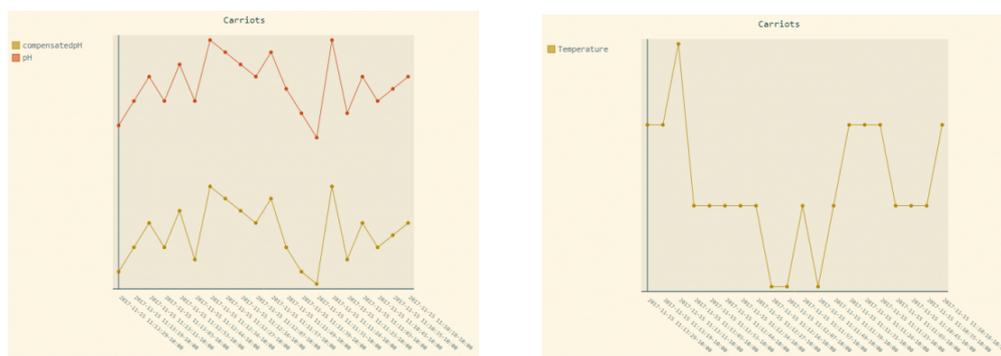


Fig. 5.27: Captura de datos enviados a Carriots

5.7 Resumen del capítulo

Se ha realizado un proceso de calibración del sensor mediante un dispositivo proporcionado por el fabricante, obteniendo resultados similares a las hoja de datos técnicos. Luego, se ha efectuado el proceso de caracterización, obteniéndose las curvas características y los parámetros de ajuste, a partir de los cuales se adapta el modelo del ISFET al dispositivo real, con resultados similares en la zona lineal. Posteriormente, se realiza una evaluación del

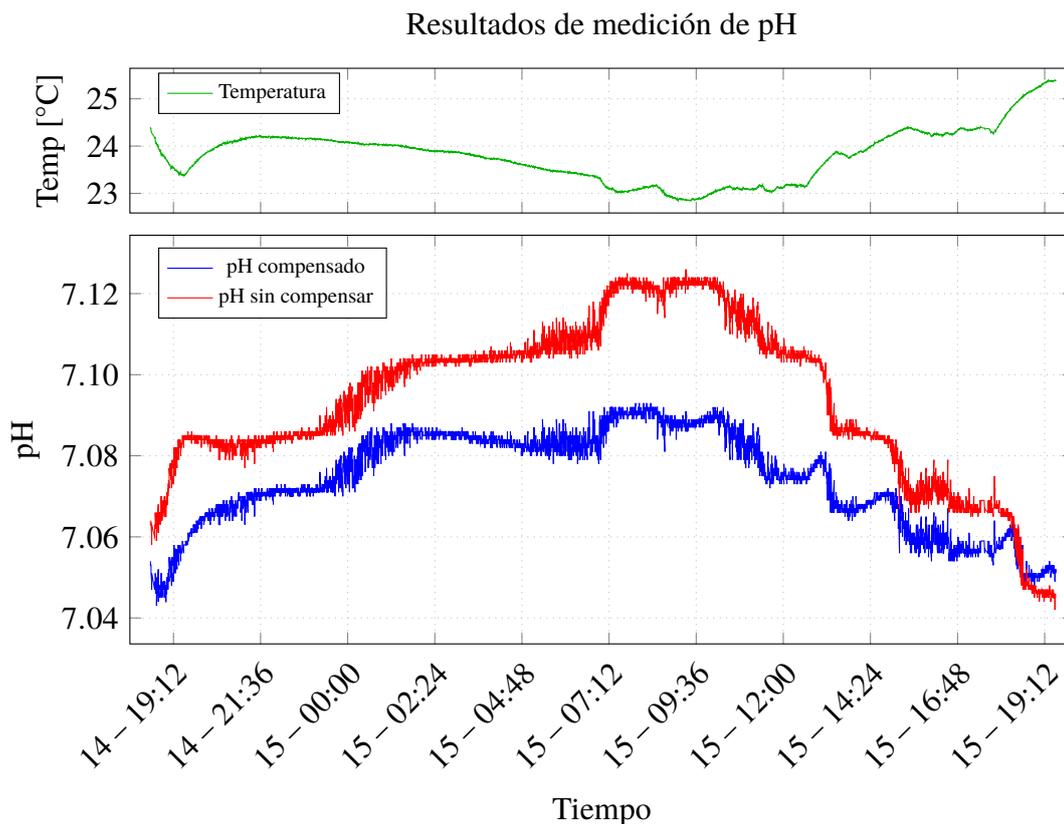


Fig. 5.28: Datos enviados a la plataforma durante 24 horas de operación.

funcionamiento del ISFET considerando únicamente el circuito de acondicionamiento.

Como un objetivo principal de la tesis consiste en la reducción del consumo de energía respecto a sensores existentes, se realizan varias mediciones de las diferentes etapas que forman parte del sistema de monitoreo, obteniendo como resultado un consumo del sensor y su componente de adaptación equivalente a 2.667mA, con una potencia de 10.303mW. Así mismo, se identifica un costo fijo de energía correspondiente a los reguladores, así como un elevado costo de transmisión de 9.8mA.

Luego, se efectúan los procesos de calibración de los sensores para obtener las mediciones más precisas en cada caso. En primer lugar, se realiza la calibración del sensor de temperatura, empleando para ello una plataforma con control de temperatura y otros 4 sensores de diferentes tecnologías, con los cuales se efectúa una comparación y se obtiene la ecuación característica. Posteriormente, se efectúa la calibración con el sensor ISFET, empleando para ello dos buffers de pH 4 y 7. A partir de los valores de tensión obtenidos, se genera una nueva ecuación de calibración que se ingresa en el microcontrolador, obteniéndose finalmente las mediciones esperadas de pH 4.00 y 7.00 con variaciones de ± 0.01 a temperaturas cercanas a la ambiente.

Finalmente, se efectúa el envío de los datos a Internet, empleando para ello plataformas de IoT. Para los ensayos se emplean las plataformas Ubidots, ThingSpeak, GroveStreams y Carriots, la comunicación con éstas se efectúa a través de programas escritos en python que implementan los protocolos HTTP o MQTT, dependiendo del caso. En tres de estas plataformas se puede visualizar los datos de forma pública. La prueba de envío tuvo una duración de 24 horas, obteniéndose como resultado un pH promedio de 7.07 a una temperatura media de 23.8°C.

En este capítulo se presentan los resultados experimentales del sistema de sensado de pH implementado, verificando la respuesta a las variaciones de pH, así como el consumo de energía y su envío a la plataforma de IoT. Basados en estos resultados, se obtienen las conclusiones de la tesis que se enunciarán en el siguiente capítulo.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

1. A través del desarrollo efectuado se ha confirmado la veracidad de la hipótesis al lograr que el sensor y su electrónica de acondicionamiento tengan un consumo inferior al registrado por otros sistemas (2.661mA). Esto gracias a que se ha empleado una tecnología de sensor de bajo consumo, además de implementar la electrónica de adaptación en una tecnología embebida, la cual facilita su integración y, también, la reducción de energía empleada por sus componentes, facilitando las tareas de control de habilitación para establecer el sistema en modo de bajo consumo.
2. A fin de emplear el sensor de pH apropiadamente y diseñar su electrónica de acondicionamiento, fue necesario modelarlo y caracterizarlo, obteniendo como resultado los parámetros característicos del sensor, los cuales son utilizados dentro del modelo para efectuar su simulación. Esto se logró gracias al macromodelo de comportamiento del dispositivo, descrito en el estado del arte, así como del sistema de caracterización diseñado para este propósito, que permitió obtener las curvas características del sensor.
3. Una vez identificado su comportamiento, se diseña su electrónica de acondicionamiento, la cual constituye una adaptación del circuito clásico de tensión constante y corriente constante (CVCC), simulado a través de componentes genéricos y, empleando para el ISFET, el modelo adaptado con los parámetros obtenidos en el paso anterior.
4. Con la simulación se confirma la propiedad de los parámetros empleados en el diseño, con lo cual se realiza la implementación en un microcontrolador de señal mixta (PSoC 5LP), empleando para ello sus componentes analógicos internos tales como OPAMPs, IDACs, VDACs; con los cuales se polariza el ISFET y se obtiene como resultado la tensión umbral dependiente del pH. En el mismo integrado también se diseña el circuito de conversión al dominio digital constituido por ADCs, escogiendo los parámetros más adecuados para obtener una alta resolución y un consumo reducido de corriente. Al

tratarse de un integrado de señal mixta, permite que tanto el sistema de adaptación analógico como el de procesamiento digital se encuentren en el mismo dispositivo, implementándose también el sistema de tratamiento de datos que tiene como resultado el valor representativo de pH compensado, pH sin compensar y temperatura, los cuales pueden ser enviados posteriormente al dispositivo de transmisión inalámbrica.

5. Finalmente, como parte de una solución de Internet de las Cosas, el sistema cuenta con comunicación inalámbrica con capacidad de insertarse en una red de nodos que empleen el mismo protocolo (6LoWPAN), lo cual también brinda la opción de capturar la información a través de un receptor conectado a Internet, donde los datos son publicados empleando plataformas de IoT.
6. El sistema planteado cuenta con varias ventajas respecto a soluciones comerciales, las cuales incluyen una resolución relativamente elevada (0.02 pH), una sensibilidad adecuada de 51mV/pH y un rápido tiempo de respuesta de 3s, gracias a la tecnología empleada en el sensor de pH (ISFET). Adicionalmente, cuenta con la posibilidad de escalamiento y una alta integración con la infraestructura de Internet, gracias al protocolo empleado para las comunicaciones (6LoWPAN). Así mismo, cuenta con una alta reproducibilidad al estar basado en componentes comerciales, tanto a nivel del sensor como de los componentes empleados para la adaptación, procesamiento y comunicación, con un inferior consumo de energía en su sección analógica (sensor y sistema de adaptación), que contribuye a una reducción del consumo global, gracias al sistema de control implementado en su microprocesador.

Recomendaciones

1. Para un adecuado funcionamiento del sistema de sensado de pH, se recomienda un ensamblado apropiado de los componentes de la tarjeta, así como la calibración respectiva de los sensores antes de ponerlos en operación. Para el sensor de temperatura, se recomienda seguir el mismo procedimiento presentado en el capítulo de resultados experimentales, tomando como referencia el termómetro. Para el sensor ISFET, la calibración se efectúa tomando dos puntos de referencia, empleando buffers de pH 4 y 7. Para las comunicaciones inalámbricas, se recomienda configurar adecuadamente las direcciones asignadas para que se encuentren en la misma red y puedan ser reconocidas

por los nodos que la componen. Así mismo, por el lado de la plataforma inalámbrica, para fines de test, se sugiere emplear el servicio ThingSpeak dada su fiabilidad al no presentar pérdidas de datos y ofrecer el servicio gratuitamente, sin tantas restricciones de cantidad de información enviada o tasa de envío. En su defecto, emplear una plataforma que sea construida específicamente para recibir la información de los nodos, basada en alguna alternativa de código abierto. Para las pruebas en campo, el sistema requiere, adicionalmente, de una estructura mecánica, la cual puede estar constituida por un flotador o una boya, y debe contar con un espacio a prueba de agua donde se deposite el sistema electrónico, de preferencia elaborada de un material no conductor, para evitar posible interferencia con la comunicación inalámbrica.

2. El sistema puede mejorarse en algunos aspectos relacionados al consumo, sobre todo en lo referido a los reguladores que impiden la disminución de la energía empleada en la fase de reposo. Esto podría mitigarse a través de la elección de otros reguladores o prescindiendo de ellos, para lo cual tendría que hacerse una evaluación de la tolerancia de los integrados a una alimentación directa con baterías. Por otra parte, por el lado de la tarjeta de comunicaciones, se puede optimizar el modo de operación para reducir su consumo a través de la inclusión de pines de interrupción externos, comandados por el microcontrolador principal, los cuales regulen el modo de bajo consumo de la tarjeta.
3. Si bien el ISFET es un dispositivo empleado a nivel general para la medición de pH, también cuenta con algunas variantes para medir la presencia de otros analitos a través del cambio de membrana sensitiva, llegando a tener diversas aplicaciones sobretodo en el campo de la medicina. Esto brinda un amplio espectro de posibilidades en las cuales se podría aplicar directamente el sistema, solo cambiando el tipo de ISFET y, en donde también, se podría investigar nuevas técnicas para la medición de otros elementos con este dispositivo.
4. Como trabajo futuro derivado el sistema construido y la investigación realizada, se contempla el empleo del prototipo para otras aplicaciones, no solo de agua, si no también para leche, suelo y evaluar la posibilidad de extenderlo hacia una aplicación en biomedicina. Así mismo, se pretende trabajar en una solución no solo para un ISFET individual, sino para un arreglo de sensores, lo cual podría mejorar la sensibilidad y precisión de la medición, a parte de incluir la medición de otros analitos. Para ello se

plantea también la colaboración con otras universidades y centros de investigación con un desarrollo más maduro de la tecnología de sensores basada en ISFET.

BIBLIOGRAFÍA

- [1] I. A. Shiklomanov, *Water in Crisis: A Guide to the World's Fresh Water Resources*, P. H. Gleick, editor. Oxford University Press, 1993, página 13.
- [2] United Nations Educational, Scientific and Cultural Organization, «Water for a Sustainable World», UNESCO, informe técnico, 2015.
- [3] F. D. Owa, «Water pollution: sources, effects, control and management», *International Letters of Natural Sciences*, volumen 3, páginas 65-68, 2014.
- [4] J. Froebrich, «New challenges in lake and river monitoring», en *Freshwater Forum*, volumen 23, 2005.
- [5] United Nations Educational, Scientific and Cultural Organization and World Health Organization, *Water quality assessments: a guide to the use of biota, sediments, and water in environmental monitoring*, D. V. Chapman y col., editores. E & Fm Spon London, 1996.
- [6] C. Borden y S. Roy, *Water Quality Monitoring System Design*. Winnipeg, Manitoba, Canada: International Institute for Sustainable Development, septiembre de 2015.
- [7] United Nations Environmental Programme and the World Health Organization, *Water Quality Monitoring - A practical Guide to the Design and Implementation of Freshwater Quality Studies and Monitoring Programmes*, Segunda, J. Bartam y R. Ballance, editores. UNEP/WHO, 1996.
- [8] World Health Organization y otros, *Guidelines for Drinking-water Quality*. World Health Organization, 2011.
- [9] World Health Organization, «pH in Drinking-water», OMS, informe técnico, 2007.
- [10] Dirección General de Salud, «Parámetros organolépticos», Ministerio de Salud, informe técnico, 2006.
- [11] Dirección General de Salud Ambiental, Ministerio de Salud, *Reglamento de la Calidad del Agua para Consumo Humano*, febrero de 2011.
- [12] Organización Mundial de la Salud, *Guías para la calidad del agua potable*. OMS, 2006.

- [13] Fondriest Environmental, Inc. (noviembre de 2013). pH of Water, dirección: <http://www.fondriest.com/environmental-measurements/parameters/water-quality/ph/>.
- [14] N. Uria, N. Abramova, A. Bratov, F.-X. Muñoz-Pascual y E. Baldrich, «Miniaturized metal oxide pH sensors for bacteria detection», *Talanta*, volumen 147, páginas 364-369, 2016.
- [15] Congreso de la República del Perú, *Ley de Recursos Hídricos, Ley N° 29338*, El Peruano, marzo de 2009.
- [16] Autoridad Nacional del Agua, *Plan Nacional de Recursos Hídricos del Perú*. ANA, 2013.
- [17] Autoridad Nacional del Agua, «Reporte de Gestión 2011-2016», MINAGRI, ANA, informe técnico, 2016.
- [18] SENAMHI, «Monitoreo de la calidad de agua de ríos en el Perú», Servicio Nacional de Meteorología e Hidrología, Dirección General de Hidrología y Recursos Hídricos, informe técnico, 2008.
- [19] Dirección de Gestión de calidad de los recursos hídricos, *Vigilancia de la Calidad de Agua en el Perú*, noviembre de 2012.
- [20] Libelium. (24 de febrero de 2014). Libelium World, dirección: <http://www.libelium.com/smart-water-sensors-to-monitor-water-quality-in-rivers-lakes-and-the-sea/>.
- [21] M. Rios, R. Yauri y C. Camarena, «Sistema de Monitoreo remoto para los parámetros del ambiente de peces amazónicos en cautiverio utilizando la tecnología ZigBee», en *Memorias del XXI Congreso Internacional de Ingeniería Electrónica, Eléctrica y Computación - INTERCON*, 2014.
- [22] (2016). WAPOSAT, Protecting water by monitoring, dirección: <http://waposat.com/>.
- [23] S. Karastogianni, S. Girousi y S. Sotiropoulos, «pH: Principles and Measurement», en *Encyclopedia of Food and Health*, B. Caballero, P. Finglas y F. Toldra, editores. Elsevier, 2016, capítulo pH: Principles and Measurement, páginas 333-337.
- [24] R. Buck, S. Rondinini, A. Covington, F. Baucke, C. Brett, M. Camoes, M. Milton, T. Mussini, R. Naumann, K. Pratt y others., «Measurement of pH. Definition, standards, and procedures (IUPAC Recommendations 2002)», *Pure and applied chemistry*, volumen 74, número 11, páginas 2169-2200, 2002.
- [25] A. K. Covington, R. G. Bates y R. A. Durst, «Definition of pH scales, standard reference values, measurement of pH and related terminology (Recommendations 1984)», *Pure and Applied Chemistry*, volumen 57, número 3, páginas 531-542, 1985.

- [26] G. H. Jeffery, J. Bassett, J. Mendham y R. C. Denney, *Vogel's Textbook of Quantitative Chemical Analysis*. Longman Scientific y Technical, 2006.
- [27] W. Vonau y U. Guth, «pH monitoring: a review», *Journal of Solid State Electrochemistry*, volumen 10, número 9, páginas 746-752, 2006.
- [28] L. M. Moretto y K. Kalcher, *Environmental analysis by electrochemical sensors and biosensors*. Springer, 2014.
- [29] J. Janata, *Principles of Chemical Sensors*. Springer Science & Business Media, 2010.
- [30] P. M. Monk, *Physical chemistry: understanding our chemical world*. John Wiley y Sons, 2008.
- [31] P. Bergveld, «ISFET, theory and practice», en *IEEE Sensor Conference, Toronto*, volumen 328, 2003.
- [32] P. Bergveld, «Development of an ion-sensitive solid-state device for neurophysiological measurements», *IEEE Transactions on Biomedical Engineering*, número 1, páginas 70-71, 1970.
- [33] C.-T. Sah, *Fundamentals of solid state electronics*. World Scientific Publishing Co Inc, 1991.
- [34] C. G. Fonstad, *Microelectronic devices and circuits*. McGraw-Hill, Inc., 1994.
- [35] D. E. Yates, S. Levine y T. W. Healy, «Site-binding model of the electrical double layer at the oxide/water interface», *Journal of the Chemical Society, Faraday Transactions 1: Physical Chemistry in Condensed Phases*, volumen 70, páginas 1807-1818, 1974.
- [36] M. Baylav, «Ion-sensitive field effect transistor (ISFET) for MEMS multisensory chips at RIT», Master, Rochester Institute of Technology, 2010.
- [37] J. C. Dutta, «Ion sensitive field effect transistor for applications in bioelectronic sensors: a research review», en *Computational Intelligence and Signal Processing (CISP), 2012 2nd National Conference on*, IEEE, 2012, páginas 185-191.
- [38] C. D. Fung, P. W. Cheung y W. H. Ko, «A generalized theory of an electrolyte-insulator-semiconductor field-effect transistor», *IEEE Transactions on Electron Devices*, volumen 33, número 1, páginas 8-18, 1986.
- [39] P. Hammond, D. Cumming y D. Ali, «A single-chip pH sensor fabricated by a conventional CMOS process», en *Sensors, 2002. Proceedings of IEEE*, IEEE, volumen 1, 2002, páginas 350-355.
- [40] M. Grattarola, G. Massobrio y S. Martinoia, «Modeling H⁺-sensitive FETs with SPICE», *IEEE Transactions on Electron Devices*, volumen 39, número 4, páginas 813-819, abril de 1992.

- [41] M. Janicki, M. Daniel, M. Szermer y A. Napieralski, «Ion sensitive field effect transistor modelling for multidomain simulation purposes», *Microelectronics Journal*, volumen 35, número 10, páginas 831-840, 2004.
- [42] A. J. Bard, L. R. Faulkner, J. Leddy y C. G. Zoski, *Electrochemical methods: fundamentals and applications*. Wiley New York, 1980, volumen 2.
- [43] R. H. Kingston y S. F. Neustadter, «Calculation of the space charge, electric field, and free carrier concentration at the surface of a semiconductor», *Journal of Applied Physics*, volumen 26, número 6, páginas 718-720, 1955.
- [44] L. Bousse, N. De Rooij y P. Bergveld, «Operation of chemically sensitive field-effect sensors as a function of the insulator-electrolyte interface», *IEEE Transactions on Electron Devices*, volumen 30, número 10, páginas 1263-1270, 1983.
- [45] P. Bergveld, R. Van Hal y J. Eijkel, «The remarkable similarity between the acid-base properties of ISFETs and proteins and the consequences for the design of ISFET biosensors», *Biosensors and Bioelectronics*, volumen 10, número 5, páginas 405-414, 1995.
- [46] D. Troy, J. Remington y P. Beringer, *Remington: The Science and Practice of Pharmacy*. Lippincott Williams & Wilkins, 2006.
- [47] W. Mönch, *Semiconductor surfaces and interfaces*. Springer Science & Business Media, 2001, volumen 26.
- [48] H. Abe, M. Esashi y T. Matsuo, «ISFET's using inorganic gate thin films», *IEEE Transactions on Electron Devices*, volumen 26, número 12, páginas 1939-1944, 1979.
- [49] T. Matsuo y K. D. Wise, «An integrated field-effect electrode for biopotential recording», *IEEE Transactions on Biomedical Engineering*, número 6, páginas 485-487, 1974.
- [50] M. Esashi y T. Matsuo, «Integrated micro multi ion sensor using field effect of semiconductor», *IEEE Transactions on Biomedical Engineering*, número 2, páginas 184-192, 1978.
- [51] T. Matsuo y M. Esashi, «Methods of ISFET fabrication», *Sensors and Actuators*, volumen 1, páginas 77-96, 1981.
- [52] C. Cane, I. Gracia y A. Merlos, «Microtechnologies for pH ISFET chemical sensors», *Microelectronics Journal*, volumen 28, número 4, páginas 389-405, 1997.
- [53] C. Toumazou y P. Georgiou, «Piet Bergveld-40 years of ISFET technology: From neuronal sensing to DNA sequencing», *Electronics Letters*, volumen 47, número 26, S7-S12, 2011.

- [54] P. Bergveld, «Thirty years of ISFETOLOGY: What happened in the past 30 years and what may happen in the next 30 years», *Sensors and Actuators B: Chemical*, volumen 88, número 1, páginas 1-20, 2003.
- [55] F. Xu, G. Yan, Z. Wang y P. Jiang, «Continuous accurate pH measurements of human GI tract using a digital pH-ISFET sensor inside a wireless capsule», *Measurement*, volumen 64, páginas 49-56, 2015.
- [56] M. Futagawa, T. Iwasaki, H. Murata, M. Ishida y K. Sawada, «A miniature integrated multimodal sensor for measuring pH, EC and temperature for precision agriculture», *Sensors*, volumen 12, número 6, páginas 8338-8354, 2012.
- [57] F. Cao, F. Jiang, Z. Liu, B. Chen y Z. Yang, «Application of ISFET microsensors with mobile network to build IOT for water environment monitoring», en *Intelligent Environments (IE), 2014 International Conference on*, IEEE, 2014, páginas 207-210.
- [58] P. Kurzweil, «Metal oxides and ion-exchanging surfaces as pH sensors in liquids: State-of-the-art and outlook», *Sensors*, volumen 9, número 6, páginas 4955-4985, 2009.
- [59] N. Moser, T. S. Lande, C. Toumazou y P. Georgiou, «ISFETs in CMOS and Emergent Trends in Instrumentation: A Review», *IEEE Sensors Journal*, volumen 16, número 17, páginas 6496-6514, 2016.
- [60] N. I. M. Noh, K. A. Yusof, A. Z. Abdullah, S. H. Herman, W. Fazlida y H. Abdullah, «Light effect characterization of ISFET based pH sensor with Si₃N₄ gate insulator», en *Computer Applications and Industrial Electronics (ISCAIE), 2014 IEEE Symposium on*, IEEE, 2014, páginas 82-87.
- [61] Toyohashi University of Technology. (2017). Tech Overtures: The Toyohashi Tech pH image sensor, dirección: http://www.tut.ac.jp/english/newsletter/archive/no6/tech_overtures/index.html.
- [62] Sentron. (2016). Modular pH-kit for development and testing, dirección: http://www.sentron.nl/files/8314/5493/4156/E7500344.02_pH-kit-leaflet.pdf.
- [63] Campbell Scientific. (2015). CS526 ISFET pH Probe, dirección: <https://s.campbellsci.com/documents/us/manuals/cs526.pdf>.
- [64] EMERSON. (2017). Non-Glass pH Sensor for submersion/insertion, dirección: <http://www.emerson.com/resource/blob/69306/19fb8dc44108bcf72701157b9e66a0aa/product-data--tf396-non-glass-ph-sensor-for-submersion-insertion-data.pdf>.
- [65] HACH. (2017). ISFET pH Stainless Steel Micro Probe with Waterproof Connector, dirección: <http://www.hach.com/product-detail-print.version.jsa?id=7640516433>.

- [66] Centre Nacional de Microeletronica. (2017). pH-ISFET CHARACTERISTICS, dirección: http://gtq.imb-cnm.csic.es/docs/pH_ISFET_especifications.pdf.
- [67] Centre Nacional de Microeletronica. (2017). Additional information for ISFET sensor user, dirección: http://gtq.imb-cnm.csic.es/docs/ISFET%20Additional_Info.pdf.
- [68] WINSENSE. (2013). Winsense ISFET pH Sensor (WIPS), dirección: <http://tmc.nectec.or.th/public/uploaded/products/isfet/ISFET2016/ISFET%20pH%20Sensor.pdf>.
- [69] J. M. Rabaey y M. Pedram, *Low power design methodologies*, 1.^a edición. Springer Science & Business Media, 1996.
- [70] A. Deshmukh, *Microcontrollers: Theory and Applications*, época Computer engineering series. Tata McGraw-Hill, 2005.
- [71] STMicroelectronics. (2017). Microcontrollers, dirección: <http://www.st.com/en/microcontrollers.html>.
- [72] Texas Instruments. (2016). MSP430™ ultra-low-power Microcontrollers, dirección: http://www.ti.com/lstds/ti/microcontrollers_16-bit_32-bit/msp/overview.page.
- [73] Microchip. (2017). Products, dirección: <http://www.microchip.com/products/>.
- [74] Ambiq Micro. (2017). Apollo Ultra-Low Power Microcontrollers, dirección: <http://ambiqmicro.com/apollo-ultra-low-power-mcu/>.
- [75] Cypress. (2017). Microcontroller (MCU) and Programmable System-on-Chip (PSoC®) Families, dirección: <http://www.cypress.com/products/microcontroller-mcu-and-programmable-system-chip-psoc-families>.
- [76] Cypress. (2017). 32-bit ARM® Cortex®-M0 PSoC® 4, dirección: <http://www.cypress.com/products/32-bit-arm-cortex-m0-psoc-4>.
- [77] Cypress. (2017). 32-bit ARM® Cortex®-M3 PSoC® 5LP, dirección: <http://www.cypress.com/products/32-bit-arm-cortex-m3-psoc-5lp>.
- [78] M. Kocakulak e I. Butun, «An overview of Wireless Sensor Networks towards internet of things», en *Computing and Communication Workshop and Conference (CCWC), 2017 IEEE 7th Annual*, IEEE, 2017, páginas 1-6.
- [79] F. Karray, W. M. Jmal, M. Abid, D. Houssaini, A. M. Obeid, S. M. Qasim y M. S. BenSaleh, «Architecture of wireless sensor nodes for water monitoring applications: From microcontroller-based system to soc solutions», en *Environmental Instrumen-*

- tation and Measurements (IMEKO), 2014 5th IMEKO TC19 Symposium on*, 2014, páginas 20-24.
- [80] Y.-M. R. Huang, *Sensors: Advancements in Modeling, Design Issues, Fabrication and Practical Applications*. Springer Science & Business Media, 2008, volumen 21.
- [81] J. Olsson, «6LoWPAN demystified», *Texas Instruments*, página 13, 2014.
- [82] S. C. Mukhopadhyay y N. Suryadevara, «Internet of Things: Challenges and opportunities», en *Internet of Things*, Springer, 2014, páginas 1-17.
- [83] F. Maloberti y A. C. Davies, *A Short History of Circuits and Systems*. River Publishers, 2016.
- [84] A. Vladimirescu y S. Liu, «The simulation of MOS integrated circuits using SPICE2», DTIC Document, informe técnico, 1980.
- [85] S. Martinoia y G. Massobrio, «A behavioral macromodel of the ISFET in SPICE», *Sensors and Actuators B: Chemical*, volumen 62, número 3, páginas 182-189, 2000.
- [86] W. M. Siu y R. S. Cobbold, «Basic properties of the electrolyte—SiO₂—Si system: physical and theoretical aspects», *IEEE Transactions on Electron Devices*, volumen 26, número 11, páginas 1805-1815, 1979.
- [87] N. Arora, *Mosfet modeling for VLSI simulation: theory and practice (International Series on Advances in Solid State Electronics and Technology)*. World Scientific, 2007.
- [88] C. Galup-Montoro y M. C. Schneider, *MOSFET modeling for circuit analysis and design*. World scientific, 2007.
- [89] A. Ortiz-Conde, F. J. García-Sánchez, J. Muci, A. T. Barrios, J. J. Liou y C.-S. Ho, «Revisiting MOSFET threshold voltage extraction methods», *Microelectronics Reliability*, volumen 53, número 1, páginas 90-104, 2013.
- [90] D. K. Schroder, *Semiconductor material and device characterization*. John Wiley & Sons, 2006.
- [91] A. Cunha, M. Schneider, C. Galup-Montoro, C. Caetano y M. Machado, «Unambiguous extraction of threshold voltage based on the ACM model», en *Microelectronics Technology and Devices, SBMICRO 2004: Proceedings of the Nineteenth International Symposium*, The Electrochemical Society, volumen 2004, 2004, página 69.
- [92] R. M. da Ponte y F. R. de Sousa, «An Analog Front-End for an ISFET-Based Sensor Using Off-The-Shelf Components», 2016.

- [93] D. Tomaszewski, C.-M. Yang, B. Jaroszewicz, M. Zaborowski, P. Grabiec y D. Pijanowska, «Electrical characterization of ISFETs», *Journal of telecommunications and information technology*, páginas 55-60, 2007.
- [94] Y. Tsvividis y C. McAndrew, *Operation and Modeling of the MOS Transistor*. Oxford Univ. Press, 2011.
- [95] H. Park y B. Choi, «Electrical extractions of 1-D doping profiles and effective mobility in MOSFET», en *Solid Dielectrics (ICSD), 2010 10th IEEE International Conference on*, IEEE, 2010, páginas 1-3.
- [96] W. Zhu, J.-P. Han y T. Ma, «Mobility measurement and degradation mechanisms of MOSFETs made with ultrathin high-k dielectrics», *IEEE Transactions on Electron Devices*, volumen 51, número 1, páginas 98-105, 2004.
- [97] D. Tomaszewski, A. Kociubiński, J. Marczewski, K. Kucharski, K. Romański y P. Grabiec, «A versatile tool for extraction of MOSFETs parameters», *Journal of Telecommunications and information technology*, páginas 129-134, 2005.
- [98] Keithley, A Tektronix Company, *Simplifying FET Testing with 2600B System SourceMeter® SMU Instruments — Application Note*, 2016.
- [99] Keysight Technologies, *Keysight B1500A Semiconductor Device Analyzer*, 2016.
- [100] M. A. Cejer, J. L. Tucker y L. Weng, «Choosing the Optimal Source Measurement Unit Instrument for Your Test and Measurement Application», Keithley Instruments, Inc., informe técnico, 2013.
- [101] V. Ochoa Mayorga, A. Gutiérrez Aldana y P. Hernández Rodríguez, «Microcontroller-based system for multiple is fet characterization using a single reference electrode», en *19th IEEE Conference*, 1997.
- [102] V. Midili, «Realization of a capacitance-voltage measurement system for semiconductor characterization», Tesis de maestría, Aalto University, School of Electrical Engineering, Degree Programme of Micro y Nanotechnology, 2012.
- [103] M. Catelani y L. Ciani, «Design and implementation of an automatic measurement system for the characterization of power MOSFETs», en *Proc. 19th IMEKO TC-4 Symposium on Measurements of Electrical Quantities*, 2013, páginas 405-409.
- [104] D. Stranneby, *Digital Signal Processing: DSP and Applications*. Elsevier Science, 2001.
- [105] S. W. Smith, *Digital Signal Processing A Practical Guide for Engineers and Scientists*, época Demystifying technology series : by engineers, for engineers. Newnes, 2003.
- [106] P. Bergveld, «The operation of an ISFET as an electronic device», *Sensors and Actuators*, volumen 1, páginas 17-29, 1981.

- [107] W.-Y. Chung, C.-L. Chen y J.-b. Chen, «Design and implementation of low power wireless sensor system for water quality monitoring», en *Bioinformatics and Biomedical Engineering, (iCBBE) 2011 5th International Conference on*, IEEE, 2011, páginas 1-4.
- [108] M. A. B. Z. Abidin, M. Zolkapli, N. M. Noh y W. Abdullah, «Data acquisition for ISFET pH sensor system by using Seeeduino Stalker as a controller», en *Control and System Graduate Research Colloquium (ICSGRC), 2013 IEEE 4th*, IEEE, 2013, páginas 127-131.
- [109] Microchip, *PIC16(L)F1777/8/9 28/40/44-Pin, 8-Bit Flash Microcontroller*, 2016.
- [110] Texas Instruments, *MSP430FR231x Mixed-Signal Microcontrollers (Rev. B)*, 2016.
- [111] Cypress, *PSoC® 5LP: CY8C58LP Family Datasheet*, 2016.
- [112] National Instruments. (2016). Overview of Temperature Sensors, dirección: <http://www.ni.com/white-paper/4218/en/>.
- [113] Nanmac Corporation, «Comparison of Temperature Sensors and Response Times», informe técnico, 2010.
- [114] E. Doebelin, *Measurement Systems: Application and Design*, época McGraw-Hill series in mechanical and industrial engineering. McGraw-Hill, 2004.
- [115] G. Reynolds, «Practical Temperature Measurements», Agilent, informe técnico, 2000.
- [116] Cypress, *Delta Sigma Analog to Digital Converter (ADC_DelSig)*, 2016.
- [117] R. Lossio, «PSoC® 3 and PSoC 5LP Low-Power Modes and Power Reduction Techniques», Cypress, informe técnico, 2017.
- [118] Cypress, *ADC Successive Approximation Register (ADC_SAR)*, 2015.
- [119] M. Tahir y K. Javed, *ARM Microprocessor Systems: Cortex-M Architecture, Programming, and Interfacing*. CRC Press, 2017.
- [120] P. Georgiou y C. Toumazou, «CMOS-based programmable gate ISFET», *Electronics Letters*, volumen 44, número 22, páginas 1289-1290, octubre de 2008.
- [121] D. Welch, S. Shah, S. Ozev y J. B. Christen, «Experimental and Simulated Cycling of ISFET Electric Fields for Drift Reset», *IEEE Electron Device Letters*, volumen 34, número 3, páginas 456-458, marzo de 2013.
- [122] B. Premanode, N. Silawan y C. Toumazou, «Drift reduction in ion-sensitive FETs using correlated double sampling», *Electronics Letters*, volumen 43, número 16, páginas 857-859, agosto de 2007.

- [123] P. Shields, B. Nemeth, R. B. Green, M. O. Riehle y D. R. S. Cumming, «High-Speed Imaging of 2-D Ionic Diffusion Using a 16, *times*, 16 Pixel CMOS ISFET Array on the Microfluidic Scale», *IEEE Sensors Journal*, volumen 12, número 9, páginas 2744-2749, septiembre de 2012.
- [124] J. Clark. (2017). Temperature Dependence of the pH of pure Water, dirección: https://chem.libretexts.org/Core/Physical_and_Theoretical_Chemistry/Acids_and_Bases/Acids_and_Bases_in_Aqueous_Solutions/The_pH_Scale/Temperature_Dependence_of_the_pH_of_pure_Water.
- [125] J. J. B. C. Ashton y L. Geary, «The effects of temperature on ph measurement», *Reagecon Delivering the Correct Result, Shannon, Co. Clara, Ireland*, 2011.
- [126] Consort, *Manual C5010, C5020, C5030*, 2012.
- [127] BIOBASE. (2017). CO2 Incubator, dirección: http://www.biobase.cc/sys_manager/Images/proImg/CO2%20Incubator%20BJPX-C.jpg.
- [128] L. A. Vitón Zorrilla y J. G. Lezama Calvo, «Monitoring system for ISFET and glass electrode behavior comparison», en *2017 IEEE XXIV International Conference on Electronics, Electrical Engineering and Computing (INTERCON)*, agosto de 2017, páginas 1-4.
- [129] M. S. Mahmoud y A. A. Mohamad, «A study of efficient power consumption wireless communication techniques/modules for internet of things (IoT) applications», *Advances in Internet of Things*, volumen 6, número 02, página 19, 2016.
- [130] Keysight Technologies, «The Internet of Things: Enabling Technologies and Solutions for Design and Test», informe técnico, 2016.
- [131] Q. Hassan, *Innovative Research and Applications in Next-Generation High Performance Computing*, época *Advances in Systems Analysis, Software Engineering, and High Performance Computing*. IGI Global, 2016.
- [132] U. F. Khan, S. Hameed y T. Macintyre, «TCP/IP Over Bluetooth», en *Advances in Computer and Information Sciences and Engineering*, T. Sobh, editor. Springer Science & Business Media, 2008.
- [133] C. J. Hansen, «Internetworking with Bluetooth Low Energy», *GetMobile: Mobile Computing and Communications*, volumen 19, número 2, páginas 34-38, 2015.
- [134] Y. Seok. (2016). IEEE 802.11AH (Wi-Fi in 900 MHz licence-exempt band) for IoT application, dirección: <https://www.standardsuniversity.org/e-magazine/august-2016-volume-6/ieee-802-11ah-wi-fi-900-mhz-license-exempt-band-iot-application/>.
- [135] W. Sun, M. Choi y S. Choi, «IEEE 802.11 ah: A long range 802.11 WLAN at sub 1 GHz», *Journal of ICT Standardization*, volumen 1, número 1, páginas 83-108, 2013.

- [136] R. Tabish, A. B. Mnaouer, F. Touati y A. M. Ghaleb, «A comparative analysis of BLE and 6LoWPAN for U-HealthCare applications», en *GCC Conference and Exhibition (GCC), 2013 7th IEEE*, IEEE, 2013, páginas 286-291.
- [137] Thread Group. (2017). Thread Technology, dirección: <https://threadgroup.org/technology/ourtechnology>.
- [138] International Society of Automation, *ISA-100.11a-2009 An ISA Standard Wireless systems for industrial automation: Process control and related applications*, 2009.
- [139] B. Heile y col. (2017). Wi-SUN FAN Overview, dirección: <https://tools.ietf.org/id/draft-heile-lpwan-wisun-overview-00.html>.
- [140] Adaptive Network Solutions, *@ANY2400-SC 2.4GHz IEEE802.15.4 RF Module*, 2011.
- [141] Microchip, *MRF24J40MA Data Sheet 2.4 GHz IEEE Std. 802.15.4™ RF Transceiver Module*, 2008.
- [142] Texas Instruments, *CC2630 SimpleLink™ 6LoWPAN, ZigBee® Wireless MCU*, 2016.
- [143] dresden elektronik, *Radio modules deRFmega256 23M00 | 23M10 Datasheet*, 2013.
- [144] STMicroelectronics, *S2-LP Ultra-low power, high performance, sub-1GHz transceiver*, 2016.
- [145] Radiocrafts, *RC2400/RC2400HP-ZNM Transceiver Modules*, 2017.
- [146] K. Tomvmark y J. Copley, «Bringing Wireless Scalability to Intelligent Sensing Applications (Rev. B)», Texas Instruments, informe técnico.
- [147] Texas Instruments, *CC2650 SimpleLink Multistandard Wireless MCU*, 2015.
- [148] Texas Instruments. (2017). SimpleLink™ CC2650 Evaluation Module Kit, dirección: <http://www.ti.com/tool/cc2650emk>.
- [149] Texas Instruments, *SmartRF06 Evaluation Board User's Guide (Rev. B)*, 2017.
- [150] Texas Instruments. (2017). CC2650 Smart RF06 Development Kit, dirección: <http://www.ti.com/tool/cc2650dk>.
- [151] O. Hahm, E. Baccelli, H. Petersen y N. Tsiftes, «Operating systems for low-end devices in the internet of things: a survey», *IEEE Internet of Things Journal*, volumen 3, número 5, páginas 720-734, 2016.
- [152] P. Gaur y M. P. Tahiliani, «Operating Systems for IoT Devices: A Critical Survey», en *Region 10 Symposium (TENSYMP), 2015 IEEE*, IEEE, 2015, páginas 33-36.

- [153] Thingsquare. (2017). Contiki: The Open Source OS for the Internet of Things, dirección: <http://www.contiki-os.org/>.
- [154] J. Sallai, A. Linan, E. Decker, V. Handziski, M. Becker, M. Healy y col. (2017). TinyOS, dirección: <https://github.com/tinyos/tinyos-main>.
- [155] (2017). RIOT: The friendly Operating System for the Internet of Things., dirección: <https://riot-os.org/>.
- [156] S. Bhatti, J. Carlson, H. Dai, J. Deng, J. Rose, A. Sheth, B. Shucker, C. Gruenwald, A. Torgerson y R. Han, «MANTIS OS: An embedded multithreaded operating system for wireless micro sensor platforms», *Mobile Networks and Applications*, volumen 10, número 4, páginas 563-579, 2005.
- [157] M. O. Farooq y T. Kunz, «Operating systems for wireless sensor networks: A survey», *Sensors*, volumen 11, número 6, páginas 5900-5930, 2011.
- [158] C. Bormann, M. Ersue y A. Keranen, «Terminology for constrained-node networks», informe técnico, 2014.
- [159] A. Elvstam y D. Nordahl, «Operating systems for resource constraint Internet of Things devices: An evaluation», Bachelor Thesis, Malmö University, 2016.
- [160] Texas Instruments. (2017). Contiki-6LOWPAN, dirección: <http://processors.wiki.ti.com/index.php/Contiki-6LOWPAN>.
- [161] P. P. Ray, «A survey of IOT cloud platforms», *Future Computing and Informatics Journal*, volumen 1, número 1-2, páginas 35-46, 2016.
- [162] Plotly Team. (2017). Plotly: Visualize Data, Together, dirección: <https://plot.ly/>.
- [163] The MathWorks, Inc. (2017). ThingSpeak™ Understand Your Things, The open IoT platform with MATLAB analytics, dirección: <https://thingspeak.com/>.
- [164] A. Pelaez. (2016). Top IoT Platforms for Makers, dirección: <https://ubidots.com/blog/top-iot-platforms-2016/>.
- [165] J. Mineraud, O. Mazhelis, X. Su y S. Tarkoma, «A gap analysis of Internet-of-Things platforms», *Computer Communications*, volumen 89, páginas 5-16, 2016.
- [166] K. J. Singh y D. S. Kapoor, «Create Your Own Internet of Things: A survey of IoT platforms.», *IEEE Consumer Electronics Magazine*, volumen 6, número 2, páginas 57-68, 2017.
- [167] Cypress, *CY8CKIT-050 PSoC® 5LP Development Kit Guide*, 2013.

- [168] STMicroelectronics, *RM0090 Reference manual: STM32F405/415, STM32F407/417, STM32F427/437 and STM32F429/439 advanced ARM®-based 32-bit MCUs*.
- [169] STMicroelectronics, *UM1472 User manual: Discovery kit with STM32F407VG MCU*, 2016.
- [170] AC6. (2016). System Workbench for STM32, dirección: <http://www.openstm32.org/System+Workbench+for+STM32>.
- [171] STMicroelectronics, *UM1725 User Manual: Description of STM32F4xx HAL drivers*, 2016.
- [172] STMicroelectronics, *AN4739 Application note: STM32Cube firmware examples for STM32F4 Series*, 2016.
- [173] STMicroelectronics, *STM32CubeF4*, 2015.
- [174] Cypress. (2016). PSoC® Creator™ Integrated Design Environment (IDE), dirección: <http://www.cypress.com/products/psoc-creator-integrated-design-environment-ide>.
- [175] M. Hastings, «PSoC® 3 and PSoC 5LP – Pin Selection for Analog Designs - AN58304», Cypress, informe técnico, 2014.
- [176] Cypress, *PSoC® 5LP Architecture TRM*, 2017.
- [177] Contiki community. (2017). Contiki: The Open Source OS for the Internet of Things, dirección: <http://www.contiki-os.org/index.html>.
- [178] A. Liñán Colina, A. Vives, A. Bagula, M. Zennaro y E. Pietrosemoli, *IoT in 5 days*. 2015.
- [179] A. Dunkels, G. Oikonomou, F. Rosendal, R. Léone y S. Duquennoy. (2015). simple-udp-rpl, dirección: <https://github.com/contiki-os/contiki/tree/master/examples/ipv6/simple-udp-rpl>.
- [180] T. Winter y col., «RFC 6550 - RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks», Internet Engineering Task Force (IETF), informe técnico, 2012.
- [181] Ubidots. (2017). Ubidots, An Application Builder for the Internet of Things, dirección: <https://ubidots.com/>.
- [182] D. Da Silva, G. Angulo, A. Pelaez y D. Sánchez. (2017). ubidots-python, dirección: <https://github.com/ubidots/ubidots-python>.
- [183] J. Bazán. (2017). JOBARECI, dirección: <http://www.jobareci.com/>.
- [184] Cypress, *CY8CKIT-002 MiniProg3 User Guide*, 2017.

- [185] EZDO, *User's Guide MP-103 pH/ORP/Temperature Meter*, 2013.
- [186] FLUKE, *287/289 True-rms Digital Multimeters, Manual de uso*, 2009.
- [187] Labnet International, *Digital Dry Bath Models D1100 and D1200*, 2003.
- [188] Grove Streams LLC. (2017). GroveStreams, The platform to build your Internet of Things, dirección: <https://www.grovestreams.com/>.
- [189] Carriots. (2017). Carriots, Let's Make Things Better, An End-to-End Solution for Internet of Things Applications, dirección: <https://www.carriots.com/>.
- [190] WINSENSE. (2013). ISFET pH Sensor, dirección: http://www.winsense.co.th/item/item_2.html.
- [191] WINSENSE. (2013). Reference Electrode Ag/AgCl, dirección: http://www.winsense.co.th/item/item_3.html.
- [192] Texas Instruments, *CC2650EM-7ID Schematic*, 2015.
- [193] Mentor Graphics, «SystemVision® Multi-discipline System Verification», Mentor Graphics, informe técnico, 2014.
- [194] Mentor Graphics, «Higher Education Program», Mentor Graphics, informe técnico, 2017.
- [195] Cadence. (2017). OrCAD Cadence PCB Solutions, dirección: <http://www.orcad.com/>.
- [196] Cadence, «OrCAD Capture: Fast, intuitive PCB schematic design solution», Cadence, informe técnico, 2016.
- [197] Cadence, «OrCAD PSpice Designer Advanced circuit simulation and analysis for analog and mixed-signal circuits», Cadence, informe técnico, 2016.

ANEXO A
HOJA DE DATOS TÉCNICOS ISFET WINSENSE

TABLA N° A.1: Información de sensor ISFET [190].

Nombre:	Sensor ISFET para medición de pH (WIPSK-S)
Fabricante:	WINSENSE
Código:	NO.SO10101
Precio:	39 USD

TABLA N° A.2: Información de electrodo de referencia [191].

Nombre:	Electrodo de Ag/AgCl para medición de pH (WIPSK-RE)
Fabricante:	WINSENSE
Código:	NO.SO10102
Precio:	29 USD

El dispositivo empleado para el sensado de pH está basado en ISFET y es fabricado por la empresa tailandesa WINSENSE. El material empleado para la membrana es nitruro de silicio (Si_3N_4) y cuenta con dos terminales, una para el drenador y otro para la fuente. Este dispositivo funciona en conjunto con su electrodo de referencia de plata/cloruro de plata (Ag/AgCl).

La información del dispositivo se puede apreciar en la TABLA N° A.1. Así mismo, para su electrodo de referencia en la TABLA N° A.2.

A continuación, en las páginas siguientes de este anexo, se presenta la hoja de datos técnicos para el sensor [68].



Product datasheet

Ion Sensitive Field Effect Transistor (ISFET)

Winsense ISFET pH Sensor (WIPS)

Special Features:

- Si_3N_4 (Silicon Nitride) Insulating gate
- Operates as a MOSFET at a constant voltage V_{ds} current I_{ds}
- Quality control by predetermined electrical measurement cycle after packaging
- Single supply, low power, small size

Product Description:

Sensing principle:

The sensitive element is a Field Effect Transistor; whose metal gate is replaced by a Reference Electrode and the solution of interest.

The ISFET devices are realized with microelectronic technology compatible with CMOS processes.

- Si_3N_4 insulating gate ISFET devices measure the pH value in a wide range from basic to acidic solutions

Applications:

- Smart farming
- Water Quality monitoring
- Environment control
- Security, industrial process control

Interface electronics:

- Analog read out circuit with output 1-2 V.

Characteristics

Input/Outputs:

- Bias condition: $V_{ds}=0.3\text{ V}$
 $I_{ds}=25-35\text{ uA}$
- Output: Analog voltage 1-2 V

Base structure

- Sensor base materials: Silicon, Silicon nitride, Silicodioxide
- Technology: 6" planar CMOS process

Selective membrane

- pH-sensitive material: Si_3N_4

Sensor dimensions:

	Width	Length	Thickness	Unit
Sensor chip dimension	1400	3550	650	Um
PCB dimension	2	20	1.6	mm

Fig. A.1: Hoja de datos técnicos ISFET Winsense (WISP), pág.1. Extraído de [68].

pH Sensor Characteristics

DC Specifications:

	Min	Typical	Max
Biased Vds		0.3 V	
Biased Ids		30 uA	
Sensitivity ($\Delta V/\Delta pH$)	45 mV/pH	50 mV/pH	58 mV/pH
Temperature coefficient	1.29 mV/°C (pH 10)	1.84 mV/°C (pH 7)	2.15 mV/°C (pH 4)

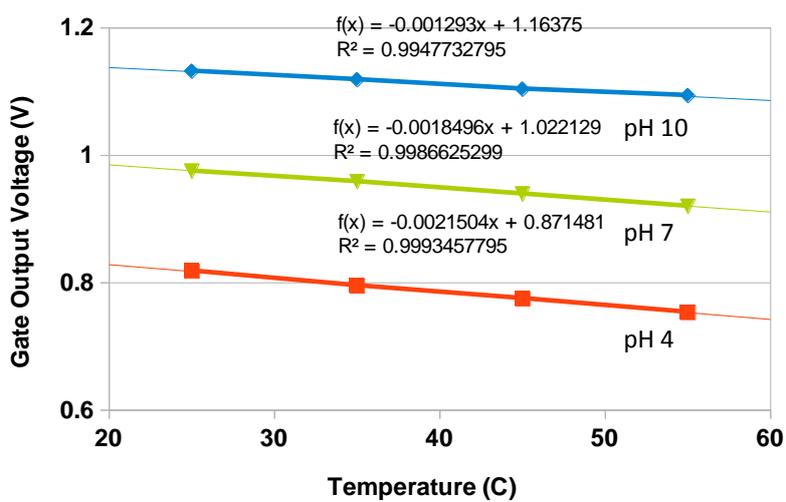
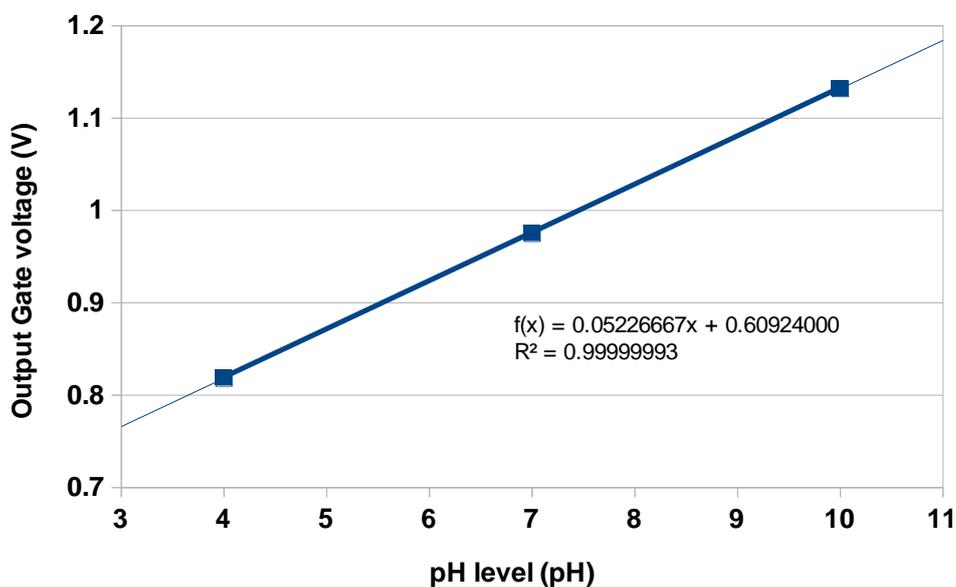


Fig. A.2: Hoja de datos técnicos ISFET Winsense (WISP), pág.2. Extraído de [68].

pH Sensor Specifications

Sensitivity: 50 mV/pH

Range: pH 2 - pH 12

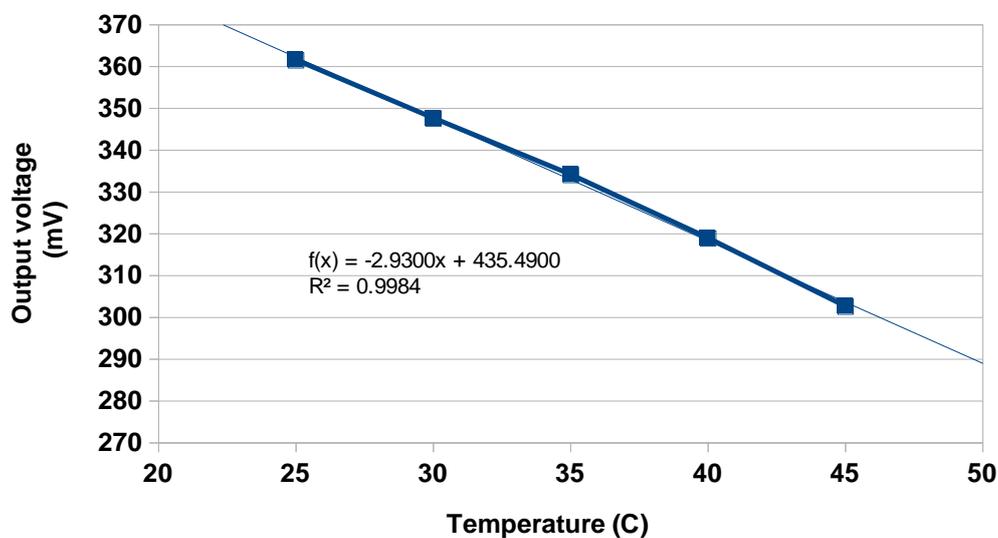
Accuracy: 0.01 pH

Operating temperature: 0°C - 100°C

Response time: 10 s

Temperature Sensor Characteristics

Temperature Sensor Diode:



Temperature Sensor Specifications

Sensitivity: -2.93 mV/degC

Range: 0 - 100°C

Response time: 1 s

Sensor terminals and connections:

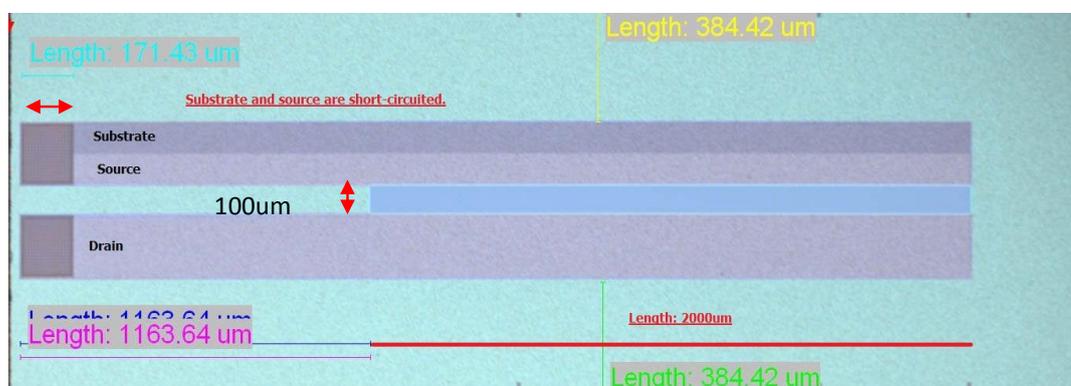
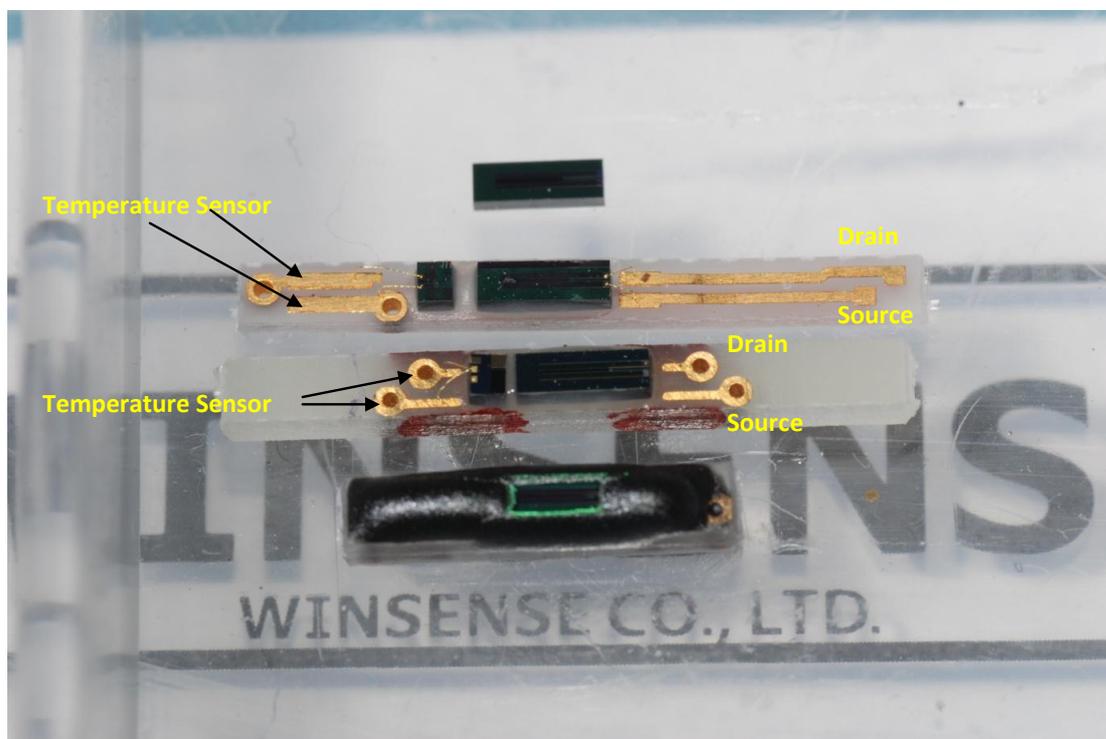


Fig. A.3: Hoja de datos técnicos ISFET Winsense (WISP), pág.3. Extraído de [68].

Chip connections and connections of packaged sensor:



From top to bottom:

1. ISFET chip
2. Temperature (left) and ISFET (right) chips wire bonded on a PCB
3. Temperature (left) and ISFET (right) chips wire bonded on another PCB with shorter wiring
4. ISFET wire bonded to PCB after encapsulation

Reference-electrode

For stable measurements an Ag/AgCl Reference electrode is required. Submerged together with the packaged ISFET chip, it acts as metal gate electrode and provides a stable reference potential.

Fig. A.4: Hoja de datos técnicos ISFET Winsense (WISP), pág.4. Extraído de [68].

ANEXO B
HOJA DE DATOS TÉCNICOS PSOC 5LP

TABLA N° B.1: Información de microcontrolador PSoC 5LP [111].

Nombre:	Microcontrolador PSoC 5 de arreglo de señal mixta
Fabricante:	Cypress
Código:	CY8C5868AXI-LP035
Precio:	18 USD

TABLA N° B.2: Información de tarjeta de desarrollo para el PSoC 5LP [167].

Nombre:	Tarjeta de desarrollo PSoC 5LP
Fabricante:	Cypress
Código:	CY8CKIT-050B
Precio:	94 USD

El microcontrolador empleado para la implementación del sistema de acondicionamiento de señal, de procesamiento y control para el sistema de sensado de pH es el CY8C58LP, perteneciente a la familia PSoC 5LP de la empresa Cypress. Este microcontrolador está basado en un microprocesador ARM Cortex-M3 y cuenta con un conjunto de bloques analógicos y digitales reconfigurables por el usuario, lo que le permite ser empleado para sistemas de señal mixta, como en el presente caso. Cypress también ofrece una plataforma de desarrollo para este microcontrolador, denominada CY8CKIT-050, la cual es empleada para pruebas e implementaciones preliminares.

La información general del dispositivo puede ser ubicada en la TABLA N° B.1. Así mismo, para la plataforma de desarrollo se puede apreciar en la TABLA N° B.2.

A continuación, en las páginas siguientes de este anexo, se presenta la hoja de datos técnicos para el microcontrolador empleado [111].



PSoC[®] 5LP: CY8C58LP Family Datasheet

Programmable System-on-Chip (PSoC[®])

General Description

PSoC[®] 5LP is a true programmable embedded system-on-chip, integrating configurable analog and digital peripherals, memory, and a microcontroller on a single chip. The PSoC 5LP architecture boosts performance through:

- 32-bit ARM Cortex-M3 core plus DMA controller and digital filter processor, at up to 80 MHz
- Ultra low power with industry's widest voltage range
- Programmable digital and analog peripherals enable custom functions
- Flexible routing of any analog or digital peripheral function to any pin

PSoC devices employ a highly configurable system-on-chip architecture for embedded control design. They integrate configurable analog and digital circuits, controlled by an on-chip microcontroller. A single PSoC device can integrate as many as 100 digital and analog peripheral functions, reducing design time, board space, power consumption, and system cost while improving system quality.

Features

- Operating characteristics
 - Voltage range: 1.71 to 5.5 V, up to 6 power domains
 - Temperature range (ambient): -40 to 85 °C ^[1]
Extended temperature parts: -40 to 105 °C
 - DC to 80-MHz operation
 - Power modes
 - Active mode 3.1 mA at 6 MHz, and 15.4 mA at 48 MHz
 - 2- μ A sleep mode
 - 300-nA hibernate mode with RAM retention
 - Boost regulator from 0.5-V input up to 5-V output
- Performance
 - 32-bit ARM Cortex-M3 CPU, 32 interrupt inputs
 - 24-channel direct memory access (DMA) controller
 - 24-bit 64-tap fixed-point digital filter processor (DFB)
- Memories
 - Up to 256 KB program flash, with cache and security features
 - Up to 32 KB additional flash for error correcting code (ECC)
 - Up to 64 KB RAM
 - 2 KB EEPROM
- Digital peripherals
 - Four 16-bit timer, counter, and PWM (TCPWM) blocks
 - I²C, 1 Mbps bus speed
 - USB 2.0 certified Full-Speed (FS) 12 Mbps peripheral interface (TID#10840032) using internal oscillator^[2]
 - Full CAN 2.0b, 16 Rx, 8 Tx buffers
 - 20 to 24 universal digital blocks (UDB), programmable to create any number of functions:
 - 8-, 16-, 24-, and 32-bit timers, counters, and PWMs
 - I²C, UART, SPI, I2S, LIN 2.0 interfaces
 - Cyclic redundancy check (CRC)
 - Pseudo random sequence (PRS) generators
 - Quadrature decoders
 - Gate-level logic functions
- Programmable clocking
 - 3- to 74-MHz internal oscillator, 1% accuracy at 3 MHz
 - 4- to 25-MHz external crystal oscillator
 - Internal PLL clock generation up to 80 MHz
 - Low-power internal oscillator at 1, 33, and 100 kHz
 - 32.768-kHz external watch crystal oscillator
 - 12 clock dividers routable to any peripheral or I/O
- Analog peripherals
 - Configurable 8- to 20-bit delta-sigma ADC
 - Up to two 12-bit SAR ADCs
 - Four 8-bit DACs
 - Four comparators
 - Four opamps
 - Four programmable analog blocks, to create:
 - Programmable gain amplifier (PGA)
 - Transimpedance amplifier (TIA)
 - Mixer
 - Sample and hold circuit
 - CapSense[®] support, up to 62 sensors
 - 1.024 V \pm 0.1% internal voltage reference
- Versatile I/O system
 - 46 to 72 I/O pins – up to 62 general-purpose I/Os (GPIOs)
 - Up to eight performance I/O (SIO) pins
 - 25 mA current sink
 - Programmable input threshold and output high voltages
 - Can act as a general-purpose comparator
 - Hot swap capability and overvoltage tolerance
 - Two USBIO pins that can be used as GPIOs
 - Route any digital or analog peripheral to any GPIO
 - LCD direct drive from any GPIO, up to 46 \times 16 segments
 - CapSense support from any GPIO
 - 1.2-V to 5.5-V interface voltages, up to four power domains
- Programming, debug, and trace
 - JTAG (4-wire), serial wire debug (SWD) (2-wire), single wire viewer (SWV), and Traceport (5-wire) interfaces
 - ARM debug and trace modules embedded in the CPU core
 - Bootloader programming through I²C, SPI, UART, USB, and other interfaces
- Package options: 68-pin QFN, 100-pin TQFP, and 99-pin CSP
- Development support with free PSoC Creator[™] tool
 - Schematic and firmware design support
 - Over 100 PSoC Components[™] integrate multiple ICs and system interfaces into one PSoC. Components are free embedded ICs represented by icons. Drag and drop component icons to design systems in PSoC Creator.
 - Includes free GCC compiler, supports Keil/ARM MDK compiler
 - Supports device programming and debugging

Notes

1. The maximum storage temperature is 150 °C in compliance with JEDEC Standard JESD22-A103, High Temperature Storage Life.
2. This feature on select devices only. See [Ordering Information](#) on page 127 for details.

Fig. B.1: Hoja de datos técnicos PSoC 5LP (CY8C58LP), pág.1. Extraído de [111].



PSoC[™] 5LP: CY8C58LP Family Datasheet

More Information

Cypress provides a wealth of data at www.cypress.com to help you to select the right PSoC device for your design, and to help you to quickly and effectively integrate the device into your design. For a comprehensive list of resources, see the knowledge base article [KBA86521](#), [How to Design with PSoC 3](#), [PSoC 4](#), and [PSoC 5LP](#). Following is an abbreviated list for PSoC 5LP:

- Overview: [PSoC Portfolio](#), [PSoC Roadmap](#)
- Product Selectors: [PSoC 1](#), [PSoC 3](#), [PSoC 4](#), [PSoC 5LP](#)
In addition, PSoC Creator includes a device selection tool.
- Application notes: Cypress offers a large number of PSoC application notes and [code examples](#) covering a broad range of topics, from basic to advanced level. Recommended application notes for getting started with PSoC 5LP are:
 - [AN77759](#): Getting Started With PSoC 5LP
 - [AN77835](#): PSoC 3 to PSoC 5LP Migration Guide
 - [AN61290](#): Hardware Design Considerations
 - [AN57821](#): Mixed Signal Circuit Board Layout
 - [AN58304](#): Pin Selection for Analog Designs
 - [AN81623](#): Digital Design Best Practices
 - [AN73854](#): Introduction To Bootloaders
- Development Kits:
 - [CY8CKIT-059](#) is a low-cost platform for prototyping, with a unique snap-away programmer and debugger on the USB connector.
 - [CY8CKIT-050](#) is designed for analog performance, for developing high-precision analog, low-power, and low-voltage applications.
 - [CY8CKIT-001](#) provides a common development platform for any one of the PSoC 1, PSoC 3, PSoC 4, or PSoC 5LP families of devices.
 - The [MiniProg3](#) device provides an interface for flash programming and debug.
- Technical Reference Manuals (TRM)
 - [Architecture TRM](#)
 - [Registers TRM](#)
- Programming Specification

PSoC Creator

PSoC Creator is a free Windows-based Integrated Design Environment (IDE). It enables concurrent hardware and firmware design of PSoC 3, PSoC 4, and PSoC 5LP based systems. Create designs using classic, familiar schematic capture supported by over 100 pre-verified, production-ready PSoC Components; see the [list of component datasheets](#). With PSoC Creator, you can:

1. Drag and drop component icons to build your hardware system design in the main design workspace
2. Codesign your application firmware with the PSoC hardware, using the PSoC Creator IDE C compiler
3. Configure components using the configuration tools
4. Explore the library of 100+ components
5. Review component datasheets

Figure 1. Multiple-Sensor Example Project in PSoC Creator

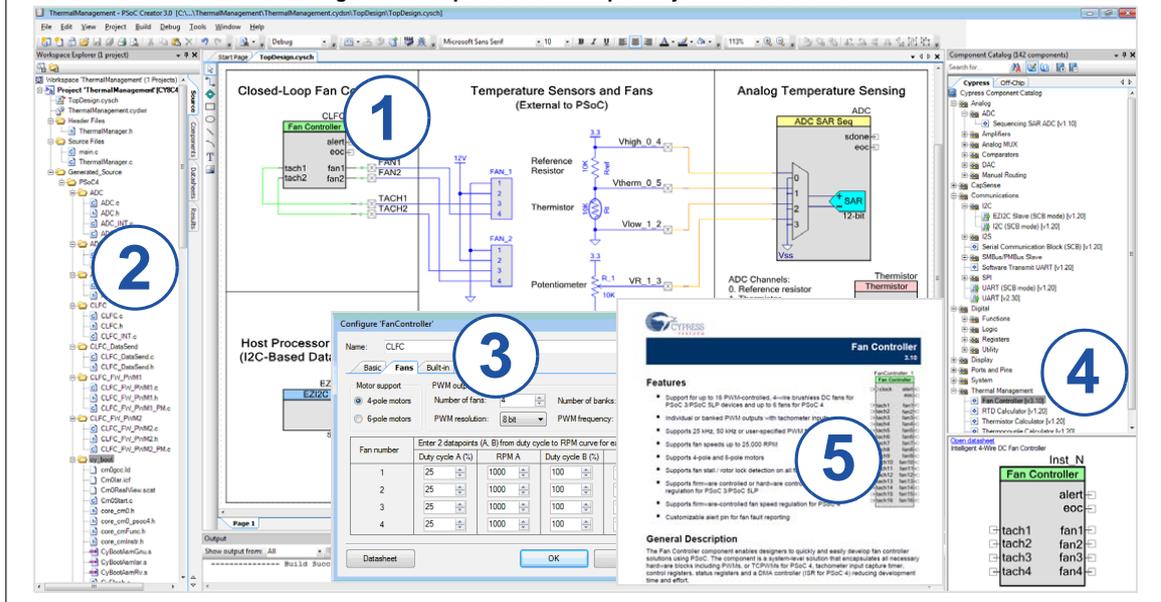


Fig. B.2: Hoja de datos técnicos PSoC 5LP (CY8C58LP), pág.2. Extraído de [111].

ANEXO C
HOJA DE DATOS TÉCNICOS CC2650

TABLA N° C.1: Información de microcontrolador CC2650 [147].

Nombre:	Microcontrolador para comunicaciones inalámbricas CC2650
Fabricante:	Texas Instruments
Código:	CC2650F128RGZR
Precio:	6.31 USD

TABLA N° C.2: Información de módulo de evaluación CC2650EM [192].

Nombre:	Módulo de evaluación CC2650
Fabricante:	Texas Instruments
Código:	CC2650EM-7ID
Precio:	99 USD

TABLA N° C.3: Información de tarjeta de evaluación SmartRF06EB [149].

Nombre:	Tarjeta de evaluación SmartRF06
Fabricante:	Texas Instruments
Código:	SMARTRF06EBK
Precio:	149 USD

El microcontrolador empleado para la implementación de la comunicación inalámbrica es el CC2650 de Texas Instruments. Este dispositivo cuenta con un módulo de evaluación (CC2650EM), el cual es empleado en el prototipo final del sistema de sensado de pH, anexo a la tarjeta principal a través de una tarjeta de expansión. Así mismo, cuenta con una plataforma de evaluación (SmartRF06EB), la cual es empleada como módulo receptor, así como en implementaciones preliminares del sistema de sensado.

La información general del microcontrolador se puede apreciar en TABLA N° C.1, para el módulo de evaluación en TABLA N° C.2 y la tarjeta de evaluación en TABLA N° C.3.

En las páginas siguientes de este anexo, se presentan la hoja de datos técnicos del microcontrolador empleado, extraído de [147].

CC2650 SimpleLink™ Multistandard Wireless MCU

1 Device Overview

1.1 Features

- Microcontroller
 - Powerful ARM® Cortex®-M3
 - EEMBC CoreMark® Score: 142
 - Up to 48-MHz Clock Speed
 - 128KB of In-System Programmable Flash
 - 8KB of SRAM for Cache
 - 20KB of Ultralow-Leakage SRAM
 - 2-Pin cJTAG and JTAG Debugging
 - Supports Over-The-Air Upgrade (OTA)
- Ultralow-Power Sensor Controller
 - Can Run Autonomous From the Rest of the System
 - 16-Bit Architecture
 - 2KB of Ultralow-Leakage SRAM for Code and Data
- Efficient Code Size Architecture, Placing Drivers, Bluetooth® Low Energy Controller, IEEE 802.15.4 MAC, and Bootloader in ROM
- RoHS-Compliant Packages
 - 4-mm × 4-mm RSM VQFN32 (10 GPIOs)
 - 5-mm × 5-mm RHB VQFN32 (15 GPIOs)
 - 7-mm × 7-mm RGZ VQFN48 (31 GPIOs)
- Peripherals
 - All Digital Peripheral Pins Can Be Routed to Any GPIO
 - Four General-Purpose Timer Modules (Eight 16-Bit or Four 32-Bit Timers, PWM Each)
 - 12-Bit ADC, 200-ksamples/s, 8-Channel Analog MUX
 - Continuous Time Comparator
 - Ultralow-Power Analog Comparator
 - Programmable Current Source
 - UART
 - 2× SSI (SPI, MICROWIRE, TI)
 - I2C
 - I2S
 - Real-Time Clock (RTC)
 - AES-128 Security Module
 - True Random Number Generator (TRNG)
 - 10, 15, or 31 GPIOs, Depending on Package Option
 - Support for Eight Capacitive-Sensing Buttons
 - Integrated Temperature Sensor
- External System
 - On-Chip internal DC-DC Converter
 - Very Few External Components
 - Seamless Integration With the SimpleLink™ CC2590 and CC2592 Range Extenders
 - Pin Compatible With the SimpleLink CC13xx in 4-mm × 4-mm and 5-mm × 5-mm VQFN Packages
- Low Power
 - Wide Supply Voltage Range
 - Normal Operation: 1.8 to 3.8 V
 - External Regulator Mode: 1.7 to 1.95 V
 - Active-Mode RX: 5.9 mA
 - Active-Mode TX at 0 dBm: 6.1 mA
 - Active-Mode TX at +5 dBm: 9.1 mA
 - Active-Mode MCU: 61 µA/MHz
 - Active-Mode MCU: 48.5 CoreMark/mA
 - Active-Mode Sensor Controller: 8.2 µA/MHz
 - Standby: 1 µA (RTC Running and RAM/CPU Retention)
 - Shutdown: 100 nA (Wake Up on External Events)
- RF Section
 - 2.4-GHz RF Transceiver Compatible With Bluetooth Low Energy (BLE) 4.2 Specification and IEEE 802.15.4 PHY and MAC
 - Excellent Receiver Sensitivity (–97 dBm for BLE and –100 dBm for 802.15.4), Selectivity, and Blocking Performance
 - Link budget of 102 dB/105 dB (BLE/802.15.4)
 - Programmable Output Power up to +5 dBm
 - Single-Ended or Differential RF Interface
 - Suitable for Systems Targeting Compliance With Worldwide Radio Frequency Regulations
 - ETSI EN 300 328 (Europe)
 - EN 300 440 Class 2 (Europe)
 - FCC CFR47 Part 15 (US)
 - ARIB STD-T66 (Japan)
- Tools and Development Environment
 - Full-Feature and Low-Cost Development Kits
 - Multiple Reference Designs for Different RF Configurations
 - Packet Sniffer PC Software
 - Sensor Controller Studio
 - SmartRF™ Studio
 - SmartRF Flash Programmer 2
 - IAR Embedded Workbench® for ARM
 - Code Composer Studio™

Fig. C.1: Hoja de datos técnicos CC2650, pág.1. Extraído de [147].

1.2 Applications

- Consumer Electronics
- Mobile Phone Accessories
- Sports and Fitness Equipment
- HID Applications
- Home and Building Automation
- Lighting Control
- Alarm and Security
- Electronic Shelf Labeling
- Proximity Tags
- Medical
- Remote Controls
- Wireless Sensor Networks

1.3 Description

The CC2650 device is a wireless MCU targeting *Bluetooth*, ZigBee® and 6LoWPAN, and ZigBee RF4CE remote control applications.

The device is a member of the CC26xx family of cost-effective, ultralow power, 2.4-GHz RF devices. Very low active RF and MCU current and low-power mode current consumption provide excellent battery lifetime and allow for operation on small coin cell batteries and in energy-harvesting applications.

The CC2650 device contains a 32-bit ARM Cortex-M3 processor that runs at 48 MHz as the main processor and a rich peripheral feature set that includes a unique ultralow power sensor controller. This sensor controller is ideal for interfacing external sensors and for collecting analog and digital data autonomously while the rest of the system is in sleep mode. Thus, the CC2650 device is ideal for applications within a whole range of products including industrial, consumer electronics, and medical.

The *Bluetooth* Low Energy controller and the IEEE 802.15.4 MAC are embedded into ROM and are partly running on a separate ARM Cortex-M0 processor. This architecture improves overall system performance and power consumption and frees up flash memory for the application.

The *Bluetooth* and ZigBee stacks are available free of charge from www.ti.com.

Device Information⁽¹⁾

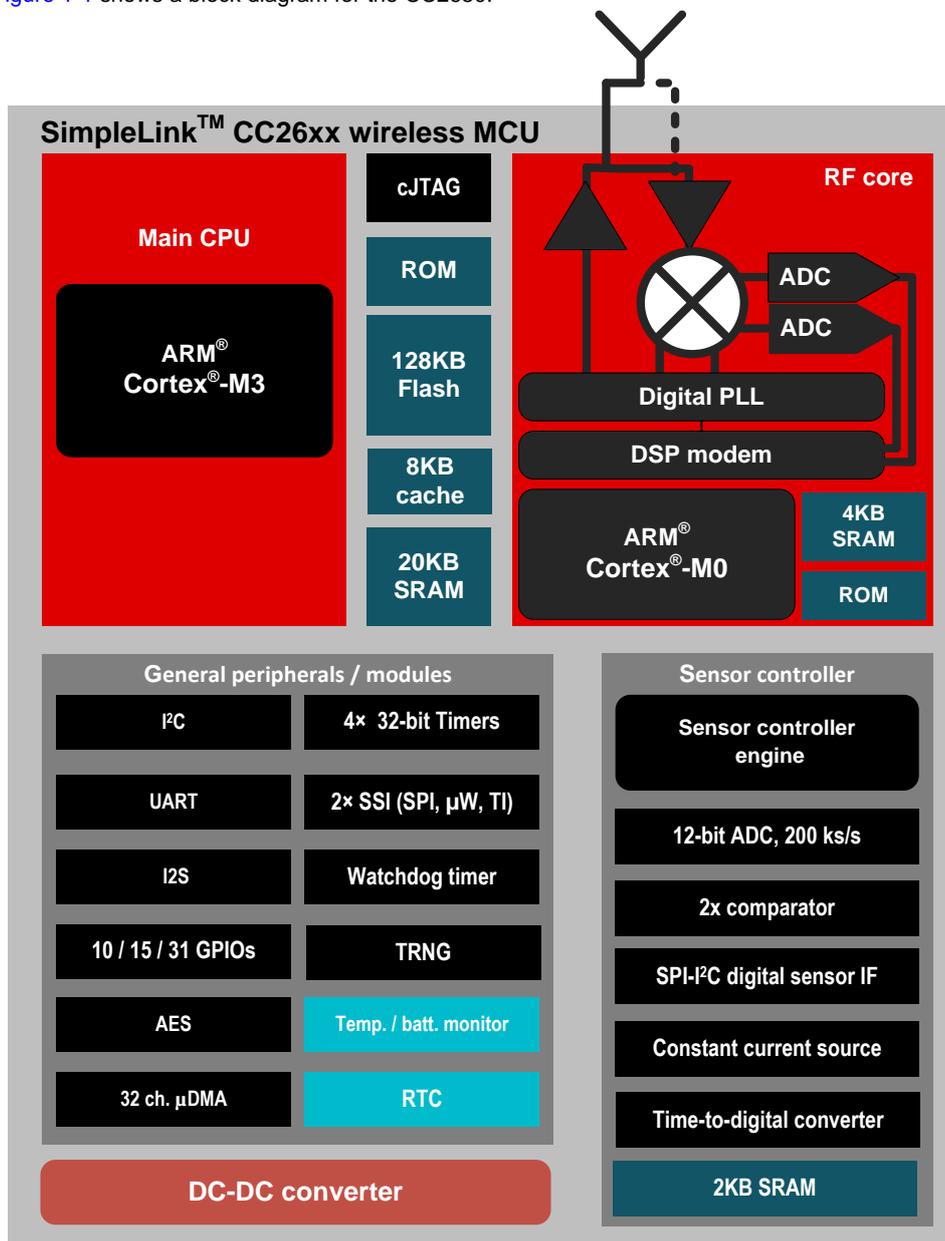
PART NUMBER	PACKAGE	BODY SIZE (NOM)
CC2650F128RGZ	VQFN (48)	7.00 mm × 7.00 mm
CC2650F128RHB	VQFN (32)	5.00 mm × 5.00 mm
CC2650F128RSM	VQFN (32)	4.00 mm × 4.00 mm

(1) For more information, see [Section 9](#), *Mechanical Packaging and Orderable Information*.

Fig. C.2: Hoja de datos técnicos CC2650, pág.2. Extraído de [147].

1.4 Functional Block Diagram

Figure 1-1 shows a block diagram for the CC2650.



Copyright © 2016, Texas Instruments Incorporated

Figure 1-1. Block Diagram

Fig. C.3: Hoja de datos técnicos CC2650, pág.3. Extraído de [147].

ANEXO D
MODELO DEL ISFET EN SPICE

El ISFET empleado para la implementación del sistema de sensado es modelado en SPICE, basado en el macromodelo de comportamiento proporcionado por Martinoia en [85]. Los parámetros en este modelo han sido modificados de acuerdo al proceso de caracterización efectuado, con la finalidad de ajustarlo a las características del dispositivo real.

A continuación en el Código D.1, se presenta el código en SPICE del modelo del dispositivo, el cual se ejecuta en un programa de simulación EDA. En este caso, se emplea el programa PSpice A/D Lite de Cadence.

Código D.1: Modelo del ISFET macromodel.cir

```
* macromodel.cir
*-----
* Macromodelo por comportamiento del ISFET
* Adaptado por:
*   Luighi A. Vitón Zorrilla <luighiavz@gmail.com>
*   Instituto Nacional de Investigación y Capacitación de Telecomunicaciones
*   Universidad Nacional de Ingeniería, PERU, Octubre 2017
* Basado en:
*   Macromodelo de comportamiento para el IFET con dos
*   tipos de sitios activos: sinalol y sitios de amina
*   por Sergio Martinoia y Giuseppe Massobrio
*   Laboratorio de Bioelectrónica, Departamento de Biofísica e Ingeniería Electrónica
*   Via Opera Pia 11A, 16145, Genova, ITALY
*   email: giaser@dibe.unige.it
*   Septiembre 1998
*****
* Lista de parámetros
*-----
* Constantes generales:
* q = carga electrónica [C]
* k = constante de Boltzmann [J/K]
* T = temperatura absoluta [K]
* NAv = constante de Avogadro [1/mole]
*****
* Parámetros geométricos del ISFET:
* dihp = distancia entre el Plano Interno de Helmholtz (IHP) y la superficie del ISFET [m]
* dohp = distancia entre el Plano Externo de Helmholtz (OHP) y la superficie del ISFET [m]
*****
* Parámetros electroquímicos del ISFET:
* Ka = constante de disociación positiva [mol/l]
* Kb = constante de disociación negativa [mol/l]
* Kn = constante de disociación para sitios de aminas [mol/l]
* Nsil = densidad de sitios del silanol [#m2]
* Nnit = densidad de sitios de las aminas [#m2]
* Cbulk = concentración del electrolito [l/mol]
* epsihp = permitividad relativa del Plano Interno de Helmholtz
* epsohp = permitividad relativa del Plano Externo de Helmholtz
* epsw = permitividad relativa del bulk de la solución
*****
* Parámetros electroquímicos del electrodo de referencia:
* Eabs = potencial absoluto del electrodo estándar de hidrógeno [V]
* Erel = potencial del electrodo de (Ag/AgCl) relativo al electrodo de hidrógeno [V]
* Phim = función de trabajo del contacto posterior del metal / carga del electrón [V]
* Philj = diferencia de potencial de la unión líquida entre la solución de referencia y el electrolito [V]
* Chieo = potencial dipolar de la superficie [V]
*****
.PARAM
+ k=1.38e-23 T=300 eps0=8.85e-12
+ Ka=15.8 Kb=63.1e-9 Kn=1e-10
+ Nsil=3.0e18
+ Nnit=1.0e16
+ q=1.6e-19 NAv=6.023e23
+ epsw=78.5 epsihp=32 epsohp=32
+ dihp=0.1n dohp=0.3n Cbulk=0.1
```

```

+ Eabs=4.7
+ Phim=4.7
+ Erel=0.200
+ Chieo=3e-3
+ Philj=1e-3
+ ET = {q/(k*T)}
+ sq = {sqrt(8*eps0*epsw*k*T)}
+ Cb = {NAv*Cbulk}
+ KK = {Ka*Kb}
+ Ch = {(eps0*epsihp*epsohp)/(epsohp*dihp + epsihp*dohp)}
+ Cd = {(sq*ET*0.5)*sqrt(Cb)}
+ Ceq = {(Cd*Ch)/(Cd + Ch)}
*****
* Modelo del ISFET
*-----
.SUBCKT ISFET 6 1 3 4 101
* drenador | electrodo referencia | fuente | pH
Eref 1 10 value = {Eabs - Phim - Erel + Chieo + Philj}
Cref 10 2 {Ceq}
EP1 46 0 value = {log(KK) + 4.6*V(101)}
RP1 46 0 1G
EP2 23 0 value = {log(Ka) + 2.3*V(101)}
RP2 23 0 1G
EPH 2 10 value = {(q/Ceq)*(Nsil*((exp(-2*V(2,10)*ET)-exp(V(46)))/(exp(-2*V(2,10)*ET)+
+ exp(V(23))*exp(-1*V(2,10)*ET)+exp(V(46))))+
+ Nnit*((exp(-1*V(2,10)*ET))/(exp(-1*V(2,10)*ET)+(Kn/Ka)*exp(V(23))))))}
RpH 101 0 1K
MIS 6 2 3 4 MISFET L = 100um W = 2000um NRS = 5 NRD = 5
* Modelo del MOSFET
*-----
.MODEL MISFET NMOS LEVEL=2
+ VTO=7.11E-01 LAMBDA=7.59E-03 RSH=8.0E+01 TOX=30E-9
+ UO=1.52E+02 TPG=0
+ UEXP=7.64E-02 NSUB=3.27E+15 NFS=1.21E+11
+ NEFF=3.88 VMAX=5.35E+04 DELTA=1.47 LD=2.91E-06
+ UCRIT=7.97E+04 XJ=6.01E-09 CJ=4.44E-4 IS=1E-11
+ CJSW=5.15E-10 PHI=5.55E-01 GAMMA=9.95E-01
+ MJ=0.395 MJSW=0.242 PB=0.585
.ENDS ISFET
*****
* Circuito de prueba para el ISFET
XIS 100 1 0 0 200 ISFET
Vbias 1 0 DC 1.5
VpH 200 0 DC 7
Vd 110 0 DC 0.098
Vid 110 100 DC 0
*****
.DC Vbias 0.0 1.6 0.01
.STEP Vd LIST 0.098 0.109 0.119 0.131 0.140
.PROBE/CSDF
.END

```

ANEXO E
PROGRAMA IMPLEMENTADO EN PSOC 5LP

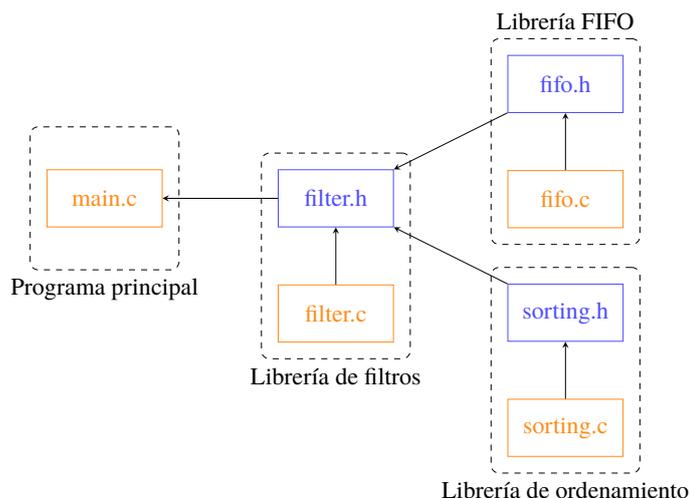


Fig. E.1: Organización de ficheros del programa en el PSoC 5LP

El PSoC 5LP es el dispositivo donde se implementa el sistema de procesamiento y de control para el sistema de sensado de pH. Por tanto, se ejecutan una serie de instrucciones en su microprocesador para llevar a cabo estas tareas.

A continuación, se presenta el programa escrito en C para la implementación de este sistema. Este programa, a su vez, está dividido en un conjunto de ficheros cuya organización se puede apreciar en Fig. E.1. De acuerdo a ello, a parte del programa principal, se cuenta con tres librerías que implementan los filtros, los registros tipo FIFO (primero en entrar, primero en salir) para almacenamiento de los datos y el algoritmo de ordenamiento.

El programa principal se aprecia en Código E.1. Allí se implementan el conjunto de métodos e instrucciones que se ejecutarán en el microcontrolador, y se declara la inclusión de métodos procedentes de la librería de filtros. Esta librería consta de su cabecera, mostrada en Código E.2 y la descripción de métodos en Código E.3, la cual, hace uso de las librerías FIFO y de ordenamiento, cuyas cabeceras se muestran en Código E.4 y Código E.6, y la descripción de sus métodos en Código E.5 y Código E.7, respectivamente.

Código E.1: Programa principal PSoC 5LP main.c

```

/* =====
 *
 * Copyright Luigi Vitón, 2016
 * Todos los derechos reservados
 * NO PUBLICADO, SOFTWARE LICENCIADO.
 *
 * Referencias:
 * Datasheet: ADC_SAR_v3.0
 * Para imprimir datos tipo float:
 * Usar "newlib-nano = true"
 * Usar "nano-newlib Float Formatting = true"
 * Adicionar parámetro a "linker command line -u_printf_float"
 * Cambiar la dimensión de heap a 0x200 en .cydwr file
 *
 */

```

```

* =====
*/
#define LENGTH_STRING 200
#define BETA 3950
#define CURRENT 100
#define VCALZERO 699.1
#define SENSIBILITY 52.35
#define MEDIAN_ORDER 13
#define MOVING_ORDER 100

#define BUFFER_SIZE 200

#include <project.h>
#include <string.h>
#include <math.h>
#include <stdio.h>

#include "filters.h"

#define SLEEPTIMER_INTERVAL_COUNTER (25u)

/* Usado para activar eventos */
uint16 wakeupIntervalCounter;

int16 resultadc1 = 0;
int16 resultadc2 = 0;

uint8 dataReady1 = 0;
uint8 dataReady2 = 0;

float* mvoltvalues;
float* tempvalues;

enum state {
    idle, activate_sense, read_sense, processing, send_data, low_power
};

void ADCs_StartConvert(){
    /* Inicia conversión */
    ADC_DelSig_1_StartConvert();
    ADC_SAR_2_StartConvert();
}

void StopConvert(){
    /* Detiene conversión */
    ADC_DelSig_1_StopConvert();
    ADC_SAR_2_StopConvert();
}

float getResReference(){
    float resreference=0;
    resreference = 10e3*exp((float)-BETA/(25+273));
    return resreference;
}

float ADCTomVolts(uint16 adcvalue){
    int16 mvolts = 0;
    mvolts = ADC_DelSig_1_CountsTo_mVolts(adcvalue) ;
    return mvolts;
}

float mVoltsTopH(float mvolts){
    float pH = 0;
    pH = (mvolts - VCALZERO)/SENSIBILITY;
    return pH;
}

float mVoltsTemptopH(float mvolts, float temp){
    float pH = 0;
    pH = (mvolts - VCALZERO + 1.707*(temp-25.0))/(SENSIBILITY + 0.12*(temp-25.0));
    return pH;
}

```

```

float ADCTopH(uint16 adcvalue){
    int16 mvolts = 0;
    mvolts = ADC_DelSig_1_CountsTo_mVolts(adcvalue) ;
    return mVoltsTopH(mvolts);
}

float ADCToTemperature(uint16 adcvalue, float resreference){
    int32 uvolts = 0;
    float resistance = 0;
    float temperature = 0;

    uvolts = ADC_SAR_2_CountsTo_uVolts(adcvalue);

    resistance = (float)uVolts/CURRENT;
    temperature = BETA/log(resistance/resreference)-273;
    temperature = 1.395*temperature - 7.993;

    return temperature;
}

void Readout_Start(){
    /* Inicia los IDAC */
    IDAC8_1_Start();
    IDAC8_2_Start();

    /* Inicia el VDAC */
    VDAC8_1_Start();

    /* Inicia los OPAMP*/
    Opamp_1_Start();
    Opamp_2_Start();
}

void Readout_Sleep(){
    /* Detiene los IDAC */
    IDAC8_1_Sleep();
    IDAC8_2_Sleep();

    /* Detiene el VDAC */
    VDAC8_1_Sleep();

    /* Detiene los OPAMP*/
    Opamp_1_Sleep();
    Opamp_2_Sleep();
}

void Readout_Wakeup(){
    /* Despierta los IDAC */
    IDAC8_1_Wakeup();
    IDAC8_2_Wakeup();

    /* Despierta el VDAC */
    VDAC8_1_Wakeup();

    /* Despierta los OPAMP*/
    Opamp_1_Wakeup();
    Opamp_2_Wakeup();
}

void ADCs_Start(){
    /* Inicia los ADCs */
    ADC_DelSig_1_Start();
    ADC_SAR_2_Start();
}

void ADCs_Sleep(){
    /* Detiene los ADCs */
    ADC_DelSig_1_Sleep();
    ADC_SAR_2_Sleep();
}

```

```

void ADCs_Wakeup(){
    /* Despierta los ADCs */
    ADC_DelSig_1_Wakeup();
    ADC_SAR_2_Wakeup();
}

void printArray(float* array, char8* buffer, int number){
    int i = 0;

    for(i=0; i <number; i++){
        sprintf(buffer, "%1.2f:", *(array+i));
        UART_1_PutString(buffer);
        UART_1_PutString("\n");
    }
}

/*Método para iniciar la interrupción */
CY_ISR(WakeupIsr){
    /*Requerido después de activarse para reiniciar el estado del bit ctw_int.*/
    SleepTimer_1_GetStatus();
}

int main()
{
    CyGlobalIntEnable; /* Habilita interrupciones globales. */

    enum state current_state;
    current_state = idle;

    Readout_Start();
    ADCs_Start();

    /* Inicia el UART*/
    UART_1_Start();
    UART_1_PutString("Hello!!\n\r");

    /* Inicia sleep timer */
    isr_1_StartEx(WakeupIsr);

    /* Inicia operación del SleepTimer con la configuración establecida*/
    SleepTimer_1_Start();

    /* variables */
    char8 txbuffer[1500];
    int16 data1 = 0;
    int16 data2 = 0;
    float dataf2 = 0;
    float pHVolts = 0;
    char8 tempbuffer[500];
    float resreference = 0;

    mvoltvalues = (float *) malloc(sizeof(float)*BUFFER_SIZE);
    tempvalues = (float *) malloc(sizeof(float)*BUFFER_SIZE);

    float* mvoltvaluesmedian = (float *) malloc(sizeof(float)*BUFFER_SIZE);
    float* tempvaluesmedian = (float *) malloc(sizeof(float)*BUFFER_SIZE);
    float* mvoltvaluesmean = (float *) malloc(sizeof(float)*BUFFER_SIZE);
    float* tempvaluesmean = (float *) malloc(sizeof(float)*BUFFER_SIZE);

    memset(txbuffer, 0, strlen(txbuffer));
    memset(tempbuffer, 0, strlen(tempbuffer));

    resreference = getResReference();

    /*variables de control */
    uint8 quantity = 0;
    uint8 quantityproc = 0;

    UART_1_PutString("Starting the process!!\n\r");

    for(;;)
    {

```



```

        quantityproc=0;
        memset(tempbuffer,0,strlen(tempbuffer));
        sprintf(tempbuffer,"% .3f, % .3f, % .2f\n",pHcomp,the pH,temp);

        current_state = send_data;
    }
}
else if (current_state==send_data){

    UART_1_Wakeup();

    UART_1_PutString(tempbuffer);
    memset(txbuffer,0,strlen(tempbuffer));

    CyDelay(500);
    UART_1_Sleep();

    current_state = low_power;
}
else if (current_state==low_power){

    /* Prepara el reloj del sistema para modo Sleep */
    CyPmSaveClocks();
    do
    {
        /******
        * Cambia al modo Sleep para los otros dispositivos:
        * - PM_SLEEP_TIME_NONE: tiempo de activación es definido por Sleep Timer
        * - PM_SLEEP_SRC_CTW : activación por fuentes de CTW (Central Time Wheel) está permitida
        *****/
        CyPmSleep(PM_SLEEP_TIME_NONE, PM_SLEEP_SRC_CTW);

        /******
        * Luego de la activación del dispositivo se ejecuta el ISR de Sleep Timer.
        * Después de la ejecución de CyPmSleep() la configuración del reloj se restaura.
        *****/
        if(SLEEPTIMER_INTERVAL_COUNTER == wakeupIntervalCounter)
        {
            wakeupIntervalCounter = 0u;
        }
        else
        {
            wakeupIntervalCounter++;
        }

    } while (wakeupIntervalCounter != 0u);

    /* Restaura configuración del reloj */
    CyPmRestoreClocks();
    current_state = activate_sense;
}
}
}

/* [] FIN DEL ARCHIVO */

```

Código E.2: Cabecera de librería de filtros filters.h

```

/*****
*Nombre: filters.h
*Objetivo: Filtros para procesar información
*Autor: Luighi Anthony Vitón Zorrilla
*****/

#ifndef FILTERS_H_INCLUDED
#define FILTERS_H_INCLUDED

#include <stdio.h>
#include <stdlib.h>

```

```

#include "fifo.h"
#include "sorting.h"

float median (float* buffer,int size);
float mean (float* buffer,int size);
float medianFIFO (FIFO* buffer, int size);
float meanFIFO (FIFO* buffer, int size);
float meanBuffer (float* buffer, int currentindex,int size);
float medianBuffer (float* buffer, int currentindex,int size);

#endif // FILTERS_H_INCLUDED

```

Código E.3: Librería de filtros filters.c

```

/*****
*Nombre: filters.c
*Objetivo: Filtros para procesar información
*Autor: Luighi Anthony Vitón Zorrilla
*****/

#include "filters.h"

float median (float* buffer,int size){
    float* tempbuffer = (float *) malloc(sizeof(float)*size);

    int index = 0;
    for (index=0;index<size;index++)
        *(tempbuffer + index)=*(buffer + index);

    float themedian;
    quicksort(tempbuffer,size);
    themedian = tempbuffer[size/2];

    free(tempbuffer);
    return themedian;
}

float mean (float* buffer,int size){
    float themean;
    float sum=0;
    int index=0;
    for (index=0;index<size;index++)
        sum+= *(buffer+index);

    themean=sum/size;
    return themean;
}

float medianFIFO(FIFO* buffer, int size){
    if( (buffer->inQueue - buffer->outQueue + buffer->_size)%buffer->_size< size){
        return *(buffer->QueueRegister+buffer->inQueue-1);
    }

    float* temp = (float *)malloc(sizeof(float)*size);
    int i = 0;
    for (i=0; i<size;i++){
        int newidx = (buffer->inQueue-i-1+buffer->_size)%buffer->_size;
        *(temp+i) = *(buffer->QueueRegister+newidx);
    }

    float themedian;
    quicksort(temp,size);
    themedian = temp[size/2];

    free(temp);
    return themedian;
}

float meanFIFO(FIFO* buffer, int size){
    float themean=0;
    float sum = 0;
    if( (buffer->inQueue - buffer->outQueue + buffer->_size)%buffer->_size< size){

```

```

    return *(buffer->QueueRegister+buffer->inQueue-1);
}

int i= 0;
for (i=0;i<size;i++)
{
    int newidx = (buffer->inQueue-i-1+buffer->_size)%buffer->_size;
    sum+= *(buffer->QueueRegister+newidx);
}

themean = sum/size;
return themean;
}

float meanBuffer(float*buffer, int currentind, int size){
    float themean;
    float sum = 0;

    if( currentind < size-1){
        size = currentind+1;
    }

    int index = 0;
    for (index=0;index<size;index++){
        sum+= *(buffer+currentind-index);
    }

    themean = sum/size;
    return themean;
}

float medianBuffer(float*buffer, int currentind, int size){
    if( currentind < size-1){
        return *(buffer+currentind);
    }

    float* temp = (float *)malloc(sizeof(float)*size);

    int i = 0;
    for (i=0; i<size;i++){
        *(temp+i) = *(buffer + currentind - size + 1 +i);
    }

    float themedian;
    quicksort(temp,size);
    themedian = temp[size/2];

    free(temp);
    return themedian;
}

```

Código E.4: Cabecera de librería FIFO fi fo .h

```

/*****
*Nombre: fifo.h
*Objetivo: crear un registro circular como FIFO
*Autor: Luighi Anthony Vitón Zorrilla
*****/

#ifndef FIFO_H_INCLUDED
#define FIFO_H_INCLUDED

#include <stdio.h>
#include <stdlib.h>

typedef struct Queue{
    int _status;
    int _size;
    int inQueue;
    int outQueue;
    float* QueueRegister;
} FIFO;

```

```

int initQueue (FIFO* myQueue,int _size);
int putQueue(FIFO* myQueue,float value);
int pushQueue(FIFO* myQueue,float value);
float getQueue(FIFO* myQueue);

#endif // FIFO_H_INCLUDED

```

Código E.5: Librería FIFO fi fo .c

```

/*****
*Nombre: fifo.c
*Objetivo: Operar con el registro circular como FIFO
*Autor: Luigi Anthony Vitón Zorrilla
*****/

#include "fifo.h"

int initQueue (FIFO* myQueue,int _size){
    myQueue->_status = 0;
    myQueue->_size = _size;
    myQueue->inQueue=0;
    myQueue->outQueue=0;
    myQueue->QueueRegister = (float *) malloc(sizeof(float)*_size);
    return 0;
}

int putQueue(FIFO* myQueue,float value){

    if(myQueue->inQueue ==(myQueue->outQueue-1 + myQueue->_size)%myQueue->_size){
        return -1;
    }

    *(myQueue->QueueRegister + myQueue->inQueue)= value;
    myQueue->inQueue = (myQueue->inQueue+1 + myQueue->_size)%myQueue->_size;
    return 0;
}

int pushQueue(FIFO* myQueue,float value){

    if(myQueue->inQueue ==(myQueue->outQueue-1 + myQueue->_size)%myQueue->_size){
        myQueue->outQueue = (myQueue->outQueue+1 + myQueue->_size)%myQueue->_size;
    }

    *(myQueue->QueueRegister + myQueue->inQueue)= value;
    myQueue->inQueue = (myQueue->inQueue+1 + myQueue->_size)%myQueue->_size;
    return 0;
}

float getQueue(FIFO* myQueue){

    float value;

    if(myQueue->inQueue == myQueue->outQueue)
    {
        return -1;
    }
    value = *(myQueue->QueueRegister + myQueue->outQueue);
    myQueue->outQueue = (myQueue->outQueue+1 + myQueue->_size)%myQueue->_size;
    return value;
}

```

Código E.6: Cabecera de librería de ordenamiento sorting.h

```

/*****
*Nombre: sorting.h
*Objetivo: Implementar algoritmo de ordenamiento
*Author: Luigi Anthony Vitón Zorrilla
*****/

#ifndef SORTING_H_INCLUDED
#define SORTING_H_INCLUDED

```

```

#include <stdio.h>
#include <stdlib.h>
#include <time.h>

void quicksort (float* buffer,int size);
int swap (float* a, float* b);
void delay(unsigned int mseconds);

#endif // SORTING_H_INCLUDED

```

Código E.7: Librería de ordenamiento sorting.c

```

/*****
*Nombre: sorting.c
*Objetivo: Implementar algoritmo de ordenamiento
*Author: Luighi Anthony Vitón Zorrilla
*****/

#include "sorting.h"

void quicksort (float* buffer,int size){
    if (size<2){
        return;
    }

    float pivot = *(buffer+size/2);
    int i = 0;
    int j = size-1;

    while(i<=j){
        while (*(buffer+i)<pivot) i++;
        while (*(buffer+j)>pivot) j--;

        if (i >= j) break;
        swap(buffer+i,buffer+j);
        i++;
        j--;
    }

    quicksort(buffer,i);
    quicksort(buffer+i,size-i);
    return;
}

int swap (float* a, float* b){
    float temp;
    temp = *a;
    *a = *b;
    *b = temp;
    return 0;
}

void delay(unsigned int mseconds)
{
    clock_t goal = mseconds + clock();
    while (goal > clock());
}

```

ANEXO F
PROGRAMA IMPLEMENTADO EN CC2650

El módulo de evaluación CC2650EM constituye la plataforma donde se implementa la comunicación inalámbrica. Este proceso se lleva a cabo a través del sistema operativo embebido Contiki OS, el cual, a través de realizar algunas adaptaciones, implementa los procesos requeridos para el funcionamiento como transmisor o receptor, según sea el caso.

A continuación, se presentan los archivos que se crearon para la implementación de los procesos mencionados. En primer lugar, se tiene el archivo de configuración del proyecto mostrado en Código F.1, donde se establece la longitud de buffer para el puerto serial. Luego se muestra el archivo del proceso del transmisor en Código F.2, el cual es descargado en el módulo transmisor montado en la tarjeta principal del nodo; así mismo, el proceso del receptor mostrado en Código F.3, descargado en el módulo receptor; y finalmente el archivo de control de compilación Código F.4, a través del cual se obtiene los archivos binarios para cada módulo.

Código F.1: Archivo de configuración del proyecto `project-conf.h`

```

/*
 * project-conf.h
 *
 * Creado: 31 may. 2017
 * Autor: luighi
 */

#ifndef PROJECT_CONF_H_
#define PROJECT_CONF_H_

#define SERIAL_LINE_CONF_BUFSIZE 1024

#endif /* PROJECT_CONF_H_ */

```

Código F.2: Proceso del transmisor `broadcast-serial.c`

```

/*
 *broadcast-serial.c
 *
 *Prueba la transmisión via broadcast
 *
 *Referencias:
 *-----
 *broadcast-example.c
 *unicast-sender.c
 *
 */

/**
 * \file
 * Archivo para comunicar con CC2650
 *
 * \author Luighi Vitón <luighiavz@gmail.com>
 *
 * Envía información via broadcast
 *
 */

#include "contiki.h"
#include "sys/ctimer.h"
#include "sys/etimer.h"
#include "net/ip/uip.h"
#include "net/ipv6/uip-ds6.h"

```

```

#include "dev/leds.h"
#include "button-sensor.h"
#include "batmon-sensor.h"
#include "board-peripherals.h"
#include "dev/serial-line.h"
#include "dev/cc26xx-uart.h"

#include "simple-udp.h"

#include <stdio.h>
#include <string.h>

#define UDP_PORT 1234

#define SEND_INTERVAL (10 * CLOCK_SECOND)

/*Definiciones para sensores*/
#define CC26XX_DEMO_SENSOR_1 &button_left_sensor
#define CC26XX_DEMO_SENSOR_2 &button_right_sensor
#define CC26XX_DEMO_SENSOR_3 &button_up_sensor
#define CC26XX_DEMO_SENSOR_4 &button_down_sensor
#define CC26XX_DEMO_SENSOR_5 &button_select_sensor

#define BUFFER_LENGTH 480

static struct simple_udp_connection broadcast_connection; //Procedente de simple-udp.h

/*-----*/
PROCESS(broadcast_sensors, "Transmission broadcast of sensors");
AUTOSTART_PROCESSES(&broadcast_sensors);
/*-----*/

/*Método para activación de sensores*/
static void activate_sensors(){
    SENSORS_ACTIVATE(batmon_sensor);
}

static void
receiver(struct simple_udp_connection *c,
         const uip_ipaddr_t *sender_addr,
         uint16_t sender_port,
         const uip_ipaddr_t *receiver_addr,
         uint16_t receiver_port,
         const uint8_t *data,
         uint16_t datalen)
{
    printf("Data received on port %d from port %d with length %d\n",
           receiver_port, sender_port, datalen);
}

/*-----*/
PROCESS_THREAD(broadcast_sensors, ev, data)
{
    static struct etimer periodic_transmission; //sys/etimer.h

    uip_ipaddr_t addr; //net/ip/uip.h

    char *rxdata;
    rxdata = (char *) malloc(sizeof(char)*BUFFER_LENGTH);

    PROCESS_BEGIN();

    activate_sensors();

    cc26xx_uart_set_input(serial_line_input_byte);

    printf("Starting the process\n");

    simple_udp_register(&broadcast_connection, UDP_PORT, NULL, UDP_PORT, receiver);

    etimer_set(&periodic_transmission, SEND_INTERVAL);
}

```

```

while (1){
    PROCESS_YIELD();
    if(ev == serial_line_event_message){
        leds_on(1);
        rxdata=data;
        leds_off(1);
        printf("Received line in main: %s\n", rxdata);

        uip_create_linklocal_allnodes_mcast(&addr);
        simple_udp_sendto(&broadcast_connection, rxdata,BUFFER_LENGTH,&addr);
    }
}

PROCESS_END();
}

```

Código F.3: Proceso del receptor unicast-receiver.c. Extraído de Contiki OS.

```

/*
 * Copyright (c) 2011, Swedish Institute of Computer Science.
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. Neither the name of the Institute nor the names of its contributors
 * may be used to endorse or promote products derived from this software
 * without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE INSTITUTE AND CONTRIBUTORS "AS IS" AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE INSTITUTE OR CONTRIBUTORS BE LIABLE
 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 *
 * This file is part of the Contiki operating system.
 */

#include "contiki.h"
#include "lib/random.h"
#include "sys/ctimer.h"
#include "sys/etimer.h"
#include "net/ip/uip.h"
#include "net/ipv6/uip-ds6.h"
#include "net/ip/uip-debug.h"

#include "simple-udp.h"
#include "servreg-hack.h"

#include "net/rpl/rpl.h"

#include <stdio.h>
#include <string.h>

#define UDP_PORT 1234
#define SERVICE_ID 190

#define SEND_INTERVAL (10 * CLOCK_SECOND)
#define SEND_TIME (random_rand() % (SEND_INTERVAL))

```

```

static struct simple_udp_connection unicast_connection;

/*-----*/
PROCESS(unicast_receiver_process, "Unicast receiver example process");
AUTOSTART_PROCESSES(&unicast_receiver_process);
/*-----*/

static void
receiver(struct simple_udp_connection *c,
         const uip_ipaddr_t *sender_addr,
         uint16_t sender_port,
         const uip_ipaddr_t *receiver_addr,
         uint16_t receiver_port,
         const uint8_t *data,
         uint16_t datalen)
{
    printf("Data received from ");
    uip_debug_ipaddr_print(sender_addr);
    printf(" on port %d from port %d with length %d: '%s'\n",
           receiver_port, sender_port, datalen, data);
}

/*-----*/
static uip_ipaddr_t *
set_global_address(void)
{
    static uip_ipaddr_t ipaddr;
    int i;
    uint8_t state;

    uip_ip6addr(&ipaddr, UIP_DS6_DEFAULT_PREFIX, 0, 0, 0, 0, 0, 0);
    uip_ds6_set_addr_iid(&ipaddr, &uip_lladdr);
    uip_ds6_addr_add(&ipaddr, 0, ADDR_AUTOCONF);

    printf("IPv6 addresses: ");
    for(i = 0; i < UIP_DS6_ADDR_NB; i++) {
        state = uip_ds6_if.addr_list[i].state;
        if(uip_ds6_if.addr_list[i].isused &&
           (state == ADDR_TENTATIVE || state == ADDR_PREFERRED)) {
            uip_debug_ipaddr_print(&uip_ds6_if.addr_list[i].ipaddr);
            printf("\n");
        }
    }

    return &ipaddr;
}

/*-----*/
static void
create_rpl_dag(uip_ipaddr_t *ipaddr)
{
    struct uip_ds6_addr *root_if;

    root_if = uip_ds6_addr_lookup(ipaddr);
    if(root_if != NULL) {
        rpl_dag_t *dag;
        uip_ipaddr_t prefix;

        rpl_set_root(RPL_DEFAULT_INSTANCE, ipaddr);
        dag = rpl_get_any_dag();
        uip_ip6addr(&prefix, UIP_DS6_DEFAULT_PREFIX, 0, 0, 0, 0, 0, 0);
        rpl_set_prefix(dag, &prefix, 64);
        PRINTF("created a new RPL dag\n");
    } else {
        PRINTF("failed to create a new RPL DAG\n");
    }
}

/*-----*/
PROCESS_THREAD(unicast_receiver_process, ev, data)
{
    uip_ipaddr_t *ipaddr;

    PROCESS_BEGIN();

```

```
servreg_hack_init();

ipaddr = set_global_address();

create_rpl_dag(ipaddr);

servreg_hack_register(SERVICE_ID, ipaddr);

simple_udp_register(&unicast_connection, UDP_PORT,
                  NULL, UDP_PORT, receiver);

while(1) {
    PROCESS_WAIT_EVENT();
}
PROCESS_END();
}
```

Código F.4: Archivo de control de compilación Makefile.

```
all: broadcast-serial unicast-sender unicast-receiver
APPS=servreg-hack
DEFINES+=PROJECT_CONF_H=\"project-conf.h\"
CONTIKI=../contiki

CONTIKI_WITH_IPV6 = 1
include $(CONTIKI)/Makefile.include
```

ANEXO G
SOFTWARE PARA DISEÑO Y SIMULACIÓN

Para efectuar el modelamiento del ISFET, así como la simulación y el diseño de su sistema de acondicionamiento, se han empleado algunos software de automatización de diseño electrónico (EDA), los cuales facilitan las tareas de diseño a partir de proveer un conjunto de herramientas empleadas para la traducción de esquemas a un lenguaje de simulación y la simulación en sí del diseño.

En el diseño efectuado para la tesis se han empleado dos software EDA, los cuales son: SystemVision y OrCAD Capture CIS Lite.

SystemVision es un software de Mentor Graphics, el cual se encuentra dentro del conjunto de productos orientados hacia la verificación de sistemas multidisciplinarios, que comprende diferentes dominios de ingeniería, incluyendo a sistemas de señal mixta, sistemas multifísicos y sistemas de control de datos [193]. Las características principales de este programa se pueden apreciar en la TABLA N° G.1.

En el presente trabajo se ha empleado SystemVision para la simulación del macromodelo del ISFET, a través de un script en SPICE donde se describe el modelo del dispositivo y se efectúan las simulaciones respectivas a fin de obtener sus curvas características (Capítulo 2: Fundamento Teórico, sección 2.8.3). Así mismo, se emplea este software para la simulación del modelo ajustado y la verificación respecto a los datos experimentales (Capítulo 5: Resultados experimentales, sección 5.2.3).

El software empleado forma parte del programa Higher Education Program, ofrecido por Mentor Graphics [194], parte del paquete de Diseño, Verificación y Test, el cual fue adquirido por INICTEL-UNI por el periodo de un año, desde agosto del 2016 hasta agosto del 2017. La versión empleada fue la 16.1.1.

TABLA N° G.1: Características de SystemVision

<i>SystemVision</i>	<ul style="list-style-type: none"> ■ Soporte para modelamiento multilenguaje incluyendo VHDL-AMS, SPICE y C/C++. ■ Extensiva librería de modelos en VHDL-AMS y SPICE. ■ Herramientas de modelamiento intuitivas de auto-generación a partir de diferentes fuentes de datos. ■ Análisis en el dominio del tiempo y de la frecuencia, así como un análisis paramétrico avanzado. ■ Simulación automatizada a través de scripts que permiten modificar el diseño, ejecutar la simulación y analizar los datos. ■ Colaboración con otros entornos de desarrollo tales como Simulink, LabVIEW, C/C++, Java y SystemC.
---------------------	--

TABLA N° G.2: Características de los principales programas de OrCAD

<i>OrCAD Capture</i> [196]	<ul style="list-style-type: none"> ▪ Extenso conjunto de características para el diseño esquemático. ▪ Permite el diseño jerárquico, reutilizable y variante. ▪ Incorpora la integridad de señal, el diseño analógico y de señal mixta y la simulación de circuitos. ▪ Interfaz para netlists de diferentes fabricantes.
<i>OrCAD PSpice Designer</i> [197]	<ul style="list-style-type: none"> ▪ Extensa librería de modelos. ▪ Interfaz con MATLAB/Simulink para que interfaces de bajo nivel puedan ser emuladas en aplicaciones del mundo real. ▪ Modelos de diferentes fabricantes. ▪ Visualizador de formas de onda con herramientas para un análisis de post-procesamiento. ▪ Generador de códigos de modelos de sistemas de señal mixta escritos en C/C++, SystemC y modelos compactos de Verilog-A. ▪ Arquitectura abierta permite la personalización de algoritmos y post-procesamiento de resultados.

OrCAD [195], por su parte, comprende un conjunto de software EDA proporcionado por Cadence, el cual cubre gran parte de las etapas requeridas en el diseño electrónico, tales como la elaboración de los esquemáticos, la simulación mediante SPICE e incluso el diseño del PCB. De los programas que dispone, destaca OrCAD Capture, al ser la interfaz principal para el inicio del diseño de un sistema electrónico, así como OrCAD PSpice, al ser su núcleo de simulación. Las principales características de éstos se puede apreciar en G.2.

En el presente trabajo se ha empleado OrCAD Capture, para la elaboración del esquemático del sistema de acondicionamiento de señal, desde el cual se crea el perfil de simulación para un análisis paramétrico, invocando para tal efecto a OrCAD PSpice, obteniendo con él la respuesta para diferentes valores de pH (Capítulo 3: Diseño del sistema de sensado de pH, sección 3.2.3).

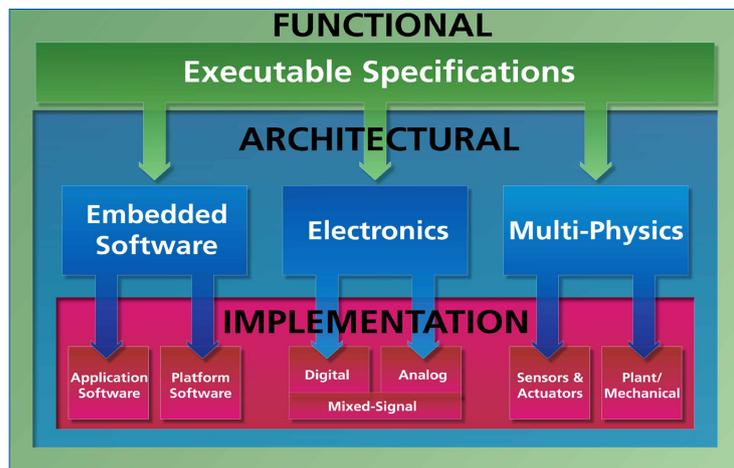
La versión empleada es la Lite 17.2, la cual es distribuida de forma gratuita por Cadence previo registro en su página web.

A continuación se presentan algunos extractos de las hojas de información de los programas citados: SystemVision, OrCAD Capture y OrCAD PSpice.

SystemVision® Multi-discipline System Verification

System Modeling & Analysis

D A T A S H E E T



The SystemVision multi-discipline collaboration environment lets you explore concepts, validate performance specifications, investigate architectural partitions, and integrate implementation-level details, all in an easy-to-use virtual prototyping environment. Focus on a single design domain, or combine multiple domains, for full-system verification.

Powerful System Design & Analysis

SystemVision integrates multiple design and engineering domains, including mixed-signal (analog/digital), multi-discipline (thermal, mechanical, hydraulic, etc.), and continuous and sampled data control systems. Engineering teams across multiple industries use SystemVision to speed development time, simplify HW/SW integration, promote design re-use, and reduce the risk of late stage defects.

The SystemVision environment supports mixed-language system and circuit descriptions, generates simulation data in technology-specific units, and ensures accurate multi-domain and mixed-signal simulation results. It includes graphical design entry, VHDL-AMS model and library development tools, advanced simulation technology, and powerful waveform viewing and data analysis applications.

Flexible System Modeling

SystemVision leverages the power and flexibility of VHDL-AMS, the IEEE standard for modeling electronic and multi-discipline systems at multiple levels of design abstraction. SystemVision also supports ModelSim®-compatible libraries; SPICE format models; and C/C++, SystemC, and Verilog-AMS language descriptions for exceptional modeling flexibility. Model entire systems in a single language, or easily mix languages in a single system simulation.

SystemVision supports any level of component and system definition, from high-level behavioral, signal-flow (s-domain) and sampled-data (z-domain)

FEATURES AND BENEFITS:

- Virtual environment for mechatronic system design and analysis
- Multi-language modeling support including VHDL-AMS, SPICE, and C/C++
- Extensive library of VHDL-AMS and SPICE simulation models for analog and mixed-signal system design
- Intuitive modeling tools automate model generation from multiple data sources
- Standard time- and frequency-domain analyses quantify nominal system performance
- Advanced parametric analyses verify system performance beyond nominal conditions
- Full featured data measurements and waveform calculations quantify performance metrics
- Scripting automation for modifying designs, running simulations, and analyzing simulation data
- Collaborative simulation environment linking multiple tools and processes, including Simulink, LabVIEW, C/C++, Java, and SystemC
- Integration with Mentor Graphics industry-leading PCB flows
- Large library of general purpose and domain-specific simulation tutorials



www.mentor.com/systemvision

**Mentor
Graphics®**

Fig. G.1: Hoja informativa sobre SystemVision, pág.1. Extraído de [193].

descriptions, to detailed component- and device-level implementations. Models at multiple abstraction levels can be mixed together to improve simulation speed while focusing on specific system performance details.

Advanced System Simulations

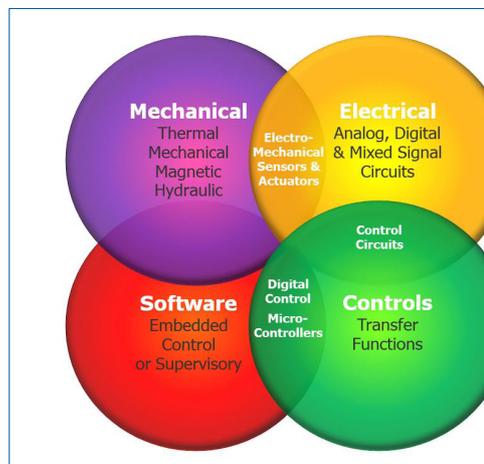
While SystemVision supports the standard operating point, time-domain, and frequency-domain simulations, modern systems often require more detailed analyses. SystemVision enables advanced system performance verification with support for parametric sweep, sensitivity, statistical, and worst case analyses. Easily determine which design parameters most affect system performance, how component tolerances influence design quality and manufacturing yields, and what happens if extremes in component tolerances combine in a worst-case system scenario.

Innovative Data Analysis

SystemVision's powerful waveform analyzer displays complex simulation data in clear, easily understood graphs. Once a waveform is displayed, the analyzer supports a variety of data analysis options. Along with the measurement tool's 45+ detailed performance measurements, the analyzer's waveform calculator supports advanced data calculations and transformations. Measurements and calculations can be saved and later rerun on the same or new data file. Graph windows and templates can be saved and later used to further analyze system performance.

System Experiments Improve Performance

Use SystemVision's Experiment Manager to create, execute, and manage system simulation experiments. Each experiment defines a set of parameters to vary during the experiment, simulation commands to execute for each parameter set, and measurements that evaluate system performance. SystemVision's link to Microsoft Excel® makes creating and running performance-improving simulation experiments quick and easy.



Integration With Leading Industry Tools

The SystemVision connexion technology links industry leading tools, bringing together best-in-class system modeling, analysis, and test capabilities into a single development environment. SystemVision + MathWorks Simulink® toolset combines mechatronic system hardware design and analysis with control algorithm development. SystemVision + National Instruments LabVIEW™ environment enables concurrent system and test program development from the beginning to the end of the design flow. With these and similar integrations, the SystemVision environment is an essential tool for developing and testing complex mechatronic systems.

System Requirements

Operating Systems:
Windows XP Pro (SP1, SP2); Windows 7

Minimum System Requirements:
1.0 GHz clock
512 MB RAM
3 GB disk space

For the latest product information, call us or visit: www.mentor.com/systemvision

©2014 Mentor Graphics Corporation, all rights reserved. This document contains information that is proprietary to Mentor Graphics Corporation and may be duplicated in whole or in part by the original recipient for internal business purposes only, provided that this entire notice appears in all copies. In accepting this document, the recipient agrees to make every reasonable effort to prevent unauthorized use of this information. All trademarks mentioned in this document are the trademarks of their respective owners.

Corporate Headquarters
Mentor Graphics Corporation
8005 SW Boeckman Road
Wilsonville, OR 97070-7777
Phone: 503.685.7000
Fax: 503.685.1204

Sales and Product Information
Phone: 800.547.3000
sales_info@mentor.com

Silicon Valley
Mentor Graphics Corporation
46871 Bayside Parkway
Fremont, CA 94538 USA
Phone: 510.354.7400
Fax: 510.354.7467

North American Support Center
Phone: 800.547.4303

Europe
Mentor Graphics
Deutschland GmbH
Arnulfstrasse 201
80634 Munich
Germany
Phone: +49.89.57096.0
Fax: +49.89.57096.400

Pacific Rim
Mentor Graphics (Taiwan)
11F, No. 120, Section 2,
Gongdao 5th Road
HsinChu City 300,
Taiwan, ROC
Phone: 886.3.513.1000
Fax: 886.3.573.4734

Japan
Mentor Graphics Japan Co., Ltd.
Gotenyama Garden
7-35, Kita-Shinagawa 4-chome
Shinagawa-Ku, Tokyo 140-0001
Japan
Phone: +81.3.5488.3033
Fax: +81.3.5488.3004

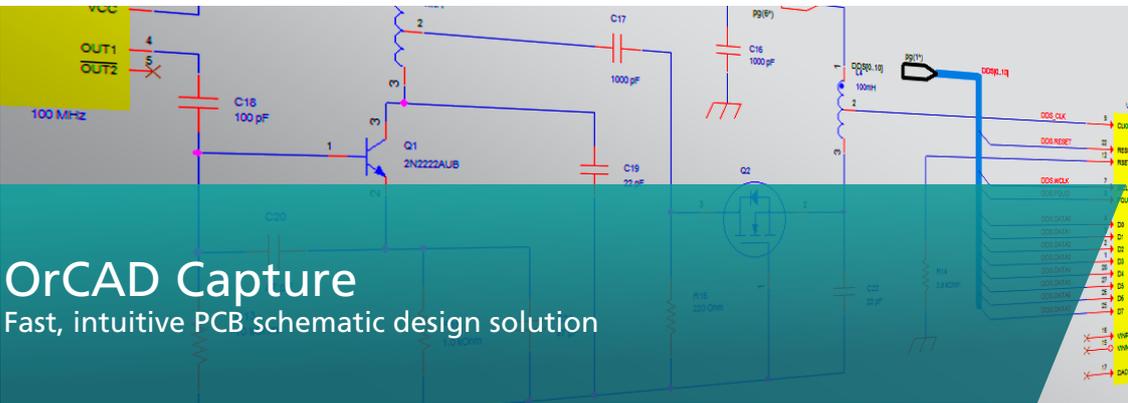
Mentor Graphics

MGC 3-14 1030990

Fig. G.2: Hoja informativa sobre SystemVision, pág.2. Extraído de [193].



OrCADTM
CADENCE PCB SOLUTIONS



OrCAD Capture

Fast, intuitive PCB schematic design solution

OrCAD® Capture is one of the most widely used schematic design solutions for the creation and documentation of electrical circuits. Fast, easy, and intuitive circuit capture, along with highly integrated flows supporting the engineering process, make OrCAD Capture one of the most popular design environments for today's product creation.

Overview

Your design of today's electronic products involves more than simply capturing connectivity, building parts, netlisting to PCB... and hoping for the best. Component information, variant design and circuit reuse, hierarchal schematics, circuit and signal integrity simulation, and integration into corporate data systems all play a significant role in reducing your development time and project cost, improving product reliability and manufacturability, and helping you achieve first-pass success.

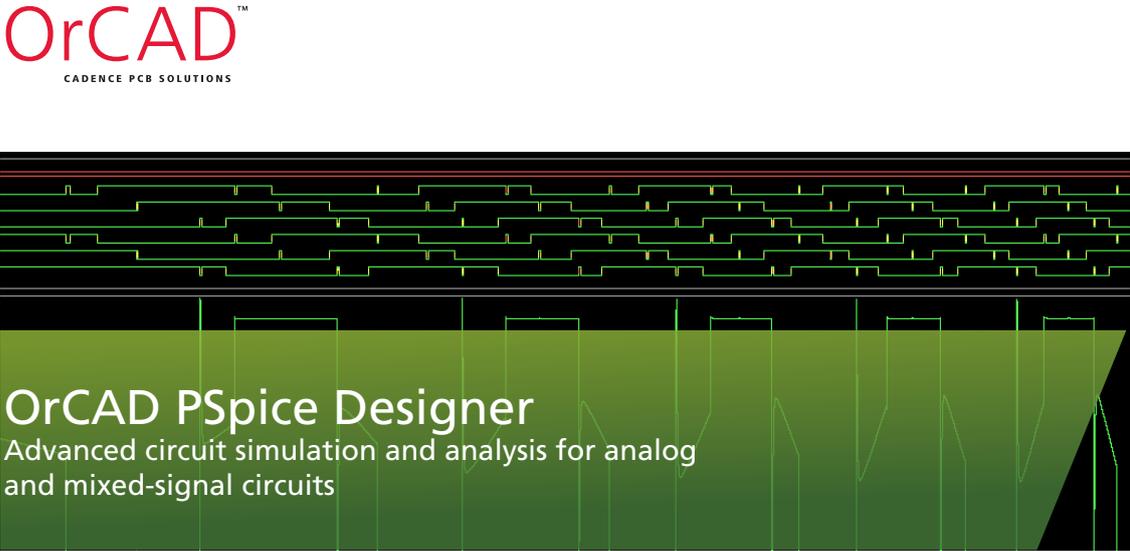
Whether you're designing a new analog circuit, revising digital schematics for an existing PCB, or implementing hierarchical block design, the OrCAD Capture solution and integrated flows provide everything you need for circuit design, analog /mixed-signal simulation, component optimization and selection, and signal integrity planning.

OrCAD Capture provides core schematic editing capabilities, but does not stop there. It is highly integrated with OrCAD PCB Editor for physical PCB design, OrCAD PSpice® for analog / mixed-signal circuit simulation, OrCAD PCB SI for signal integrity analysis and planning, and OrCAD CIS (Component Information System) for component optimization, selection, and variant design, greatly extending the schematic design process.

Highlights

- Extensive schematic entry capabilities and productivity features ensure easy, fast, intuitive design capture
- Hierarchical, reuse, and variant design capabilities streamline the creation of complex designs
- Graphical design difference enables distributed teams to perform design reviews and comparisons visually and interactively
- OrCAD CIS accelerates the design process and lowers project costs by promoting preferred parts and optimizing part selection
- Integrated signal integrity, analog/mixed-signal design, and simulation support enable circuit exploration, constraint development, and verification to help achieve first-pass success
- Bi-directional integration with the OrCAD PCB SI and the OrCAD PCB design products
- CAD vendor netlist interfaces for most PCB packages extends OrCAD Capture into multi-vendor flows
- Interactive learning tutorial provides easily navigable and convenient learning environment for new users, providing access to a large database of models and 100+ demo designs

Fig. G.3: Hoja informativa sobre OrCAD Capture, pág.1. Extraído de [196].



OrCAD™
CADENCE PCB SOLUTIONS

OrCAD PSpice Designer
Advanced circuit simulation and analysis for analog and mixed-signal circuits

OrCAD® PSpice® Designer and OrCAD Capture combine to provide industry-leading, schematic entry, native analog and mixed-signal analysis engines to deliver a complete circuit simulation and verification solution. Whether you're prototyping simple circuits or designing complex systems, the OrCAD PSpice Designer product provides the best circuit-simulation technology to analyze and refine your circuits, components, and parameters before committing to layout and fabrication.

Overview

OrCAD PSpice Designer is a high-performance, industry-proven, mixed-signal simulator and waveform viewer for analog and mixed-signal circuits. As one of the most popular, general-purpose and mixed-mode circuit simulators with extensively available models from component and IC vendors, OrCAD PSpice simulation technology is applicable for product design in numerous industries such as aerospace, medical, power electronics, and automotive. It is also utilized extensively within the research community as a reference implementation. It is capable of simulating your designs from simple circuits, complex electronics, and power supplies to radio-frequency systems and targeted IC designs. With built-in mathematical functions, behavioral modeling, circuit optimization, and electromechanical co-simulation, the OrCAD PSpice environment goes far beyond general circuit simulation.

Included in the OrCAD PSpice Designer product with the OrCAD PSpice solution, OrCAD Capture provides fast, easy, and intuitive design entry, along with highly integrated flows supporting the engineering process. With an upgrade to the OrCAD PSpice Designer Plus product, advanced analysis simulation engines provide you with functional simulation to improve design performance, cost-effectiveness, and reliability.

In addition, integration with MathWorks MATLAB/Simulink technologies provide an analysis flow enabling multi-domain simulation, such as electromechanical co-simulation.

Highlights

- Extensive model library, model association and creation, multi-core support, and full integration with OrCAD Capture improve productivity and data integrity
- MATLAB/Simulink interface allows system-level interfaces to be tested with electrical designs emulating real-world applications
- Ability to determine which components are over-stressed using Smoke analysis or observing the affects of component variations on yield using Monte Carlo analysis helps prevent "in-field" failures
- Multi-vendor models, built-in mathematical functions, and behavioral modeling techniques enable highly tailored simulations
- Powerful waveform viewing and post-processing expression support speed review and analysis without having to rerun simulations
- Virtual prototyping leveraging GUI-based code generation of mixed-signal system models written in C/C++, SystemC®, and compact models from VerilogA using Automatic Device Model Synthesizer can be easily used in the PSpice environment
- Open architecture and program platform allows easy customization of algorithms and post-processing of results

Fig. G.4: Hoja informativa sobre OrCAD PSpice, pág.1. Extraído de [197].