



UNIVERSIDAD TECNICA FEDERICO SANTA MARIA

DEPARTAMENTO DE ELECTRÓNICA

VALPARAÍSO - CHILE

“DISEÑO E IMPLEMENTACIÓN DE UN SENSOR DE CO₂ INALÁMBRICO Y ENERGÉTICAMENTE AUTÓNOMO”

MEMORIA DE TITULACIÓN PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL ELECTRÓNICO

JULIO ERIK CONTRERAS FUICA

Profesor guía
Dr. MARCELO PÉREZ LEIVA

Profesor correferente
Dr. SAMIR KOURO RENAER

Marzo de 2023

*Nada es demasiado maravilloso para ser cierto si obedece a las leyes de la
naturaleza.*
-Michael Faraday

*Para mi familia, en especial a mi madre. Dedicado también a todas las
personas que hicieron posible la gratuidad en la educación superior.*

AGRADECIMIENTOS

QUISIERA expresar mi gratitud por todo estos años en la academia a mis compañeros, amigos y familia que ha estado apoyándome siempre, también agradecer a mi profesor guía por permitirme ser parte de su proyecto.

Gracias a mi primo Sebastián, quien dedicó tiempo y esfuerzo para ayudarme a estudiar para la PSU semana a semana durante prácticamente todo un año. Lograr el puntaje necesario no hubiese podido ser posible sin su ayuda.

A mi familia que me apoyaron en mi decisión de estudiar en otra ciudad, con todas las dificultades que esto implica y quienes han sido mi pilar de apoyo incondicional.

A mis compañeros de la pensión, con quienes compartí mis primeros años universitarios, les agradezco por ser parte de mi vida y formar parte de mis recuerdos más queridos. También quiero agradecer a mi grupo de amigos Andrés, Ronald, Nacho y Miguel con los que nos coordinamos para tomar los ramos juntos, por no dejarme nunca.

Quiero agradecer a mis *roomies* Dani y Cris con los que he compartido este último año y medio de vuelta a la presencialidad, acompañándome en la aventura de vivir solos. Además, quiero reconocer a todas las personas que han pasado por mi vida, recuerdo con mucho cariño y llevo conmigo un poco de cada uno de ustedes, sin excepción.

Agradecer sin dudas a los ayudantes con los que me encontré a lo largo de mi estancia en la Universidad que hicieron posible, en muchas ocasiones, enseñarnos de manera mucho más clara y aterrizada que en las cátedras. Definitivamente todo hubiese sido mucho más difícil sin muchos de ellos.

Finalmente, agradecer a Cata por acompañarme de manera incondicional estos últimos meses.

Julio Erik Contreras Fuica

RESUMEN

EN el contexto de la Industria 4.0, la implementación de tecnologías IoT y sensores inteligentes se ha convertido en un requisito esencial para mejorar la eficiencia y productividad de los procesos industriales. En esta misma línea, las tendencias de hogar inteligente también aumentan la demanda de sensores inteligentes que proporcionen datos para, por ejemplo, tomar acciones correctivas.

En esta memoria de titulación se presenta el diseño, implementación y evaluación de un sensor de CO₂, temperatura y humedad, que utiliza tecnología de Energy Harvesting para su alimentación y transmite datos por Bluetooth Low Energy (BLE). Se presenta el desarrollo completo, comenzando con la idea inicial, pasando por el diseño y elección de componentes y terminando con el dispositivo ensamblado.

El principal objetivo de este dispositivo es monitorizar la calidad del aire en interiores, ya que es ahí donde las personas pasan la mayor cantidad de tiempo en su día a día. Otro motivo para esto es controlar la propagación de COVID-19, ya que la concentración de CO₂ es un buen indicador de la condición de ventilación de un lugar.

El proceso de diseño comienza con la descripción de los requerimientos para luego dividir el problema en partes, generando un diagrama de bloques. En base a este, se eligen los componentes y se diseña el diagrama esquemático. Teniendo en cuenta el tamaño de estos, se diseña una carcasa capaz de contenerlos. A partir de ese diseño se extraen las medidas máximas que puede tener la PCB y una vez diseñada se manda a fabricar. Para el ensamble se usaron distintas máquinas y herramientas, entre las que se incluyen stencil, pick and place manual, horno, caudín, pinzas y microscopio, debido al uso de pequeños componentes SMD.

El dispositivo desarrollado utiliza como microcontrolador el SoC nRF52840 compatible con BLE, incorpora tres sensores para tomar mediciones: SCD41, SGP40 y BME688, despliega los datos en una pantalla de tinta electrónica y utiliza el PMIC AEM10941 con un panel solar para cargar dos supercondensadores que proveen de energía el sistema.

Palabras Claves

Energy harvesting, diseño PCB, Internet de las cosas, calidad del aire, Bluetooth Low Energy

ABSTRACT

IN the context of Industry 4.0, the implementation of IoT technologies and smart sensors has become an essential requirement to improve the efficiency and productivity of industrial processes. In the same way, smart home trends also increase the demand for smart sensors that provide data to, for example, take corrective actions.

This graduation thesis presents the design, implementation, and evaluation of a CO₂, temperature, and humidity sensor that uses Energy Harvesting technology for its power supply and transmits data through Bluetooth Low Energy (BLE). The complete development is presented, starting with the initial idea, going through the design and component selection, ending with the assembled device.

The main objective of this device is to monitor indoor air quality, because people spend the most time in their daily lives indoors. Another reason for this is to control the spread of COVID-19, as CO₂ concentration is a good indicator of the ventilation of a place.

The design process contemplates everything from the idea to a minimum viable product. This starts with the description of the requirements and then breaking down the problem into parts, generating a block diagram. Based on this, the components are selected and the schematic diagram is designed. Taking into account the size of these components, a casing capable of containing them is designed. From this design, the maximum dimensions that the PCB can have are extracted, and once it's designed, it is sent for manufacturing. Different machines and tools were used for the assembly, including stencil, manual pick and place, oven, soldering iron, tweezers and microscope, due to the use of small SMD components.

The developed device uses the nRF52840 SoC compatible with BLE as a microcontroller, incorporates three sensors to take measurements: SCD41, SGP40, and BME688, displays the data on an e-ink screen, and uses the AEM10941 PMIC with a solar panel to charge two supercapacitors that provide energy to the system.

Keywords

Energy harvesting, PCB design, Internet of things, indoor air quality, Bluetooth Low Energy

ÍNDICE

AGRADECIMIENTOS	I
RESUMEN	II
ABSTRACT	III
ÍNDICE DE FIGURAS	VIII
ÍNDICE DE TABLAS	X
ABREVIACIONES	XI
1. INTRODUCCIÓN	1
1.1. Contexto	1
1.1.1. Calidad del aire y COVID-19	2
1.1.2. Trabajo a desarrollar	2
1.2. Estado del arte	2
1.2.1. Agentes contaminantes	2
1.2.2. Tecnologías para estimar la calidad del aire	3
1.2.2.a. Compuestos Volátiles Orgánicos (VOC)	3
1.2.2.b. Concentración de CO ₂	4
1.2.3. Dispositivos comerciales disponibles actualmente	4
1.3. Objetivos	4
1.4. Alcances y contribución	5
1.5. Estructura del documento	5
2. DISEÑO	8
2.1. Requerimientos de diseño	8
2.2. Esquema general	9
2.2.1. Subsistema de alimentación	9
2.2.2. Subsistema funcional	9
2.3. Criterios de elección de componentes	10
2.3.1. Microcontrolador/SoC	11
2.3.2. Tecnología de transmisión inalámbrica	11
2.3.3. Sensores	11
2.3.4. Sistema de recolección de energía	12

2.3.5.	Pantalla	12
2.4.	Elección de componentes principales	12
2.4.1.	Microcontrolador y tecnología de transmisión inalámbrica	12
2.4.1.a.	E73-2G4M08S1C	14
2.4.2.	Pantalla e-paper	14
2.4.2.a.	Estructura de las pantallas de tinta electrónica	14
2.4.2.b.	GDEH0154D67	15
2.4.2.c.	DESPI-C02	16
2.4.3.	AEM10941	16
2.4.4.	Sensores	17
2.4.4.a.	Sensor de CO2: SCD41	17
2.4.4.b.	Sensor de VOC: SGP40	17
2.4.4.c.	Sensor de gas: BME688	18
2.5.	Diagrama esquemático	18
2.5.1.	Elección de pines del microcontrolador	18
2.5.1.a.	Oscilador de cristal	18
2.5.1.b.	Botones y LEDs	18
2.5.2.	Componentes adicionales y configuración del AEM10941	19
2.5.2.a.	Configuración de batería y LDOs	20
2.5.2.b.	Capacitancia de supercondensadores	20
2.5.2.c.	Elección de supercondensadores	21
2.5.2.d.	Panel solar	22
2.5.2.e.	Configuración MPPT	22
2.5.2.f.	Configuración de arranque en frío	23
2.5.2.g.	Configuración de batería primaria	23
2.5.2.h.	Protección para entrada de 5V	24
2.5.2.i.	Componentes externos para el funcionamiento del AEM10941	24
2.5.3.	Divisor resistivo activable	25
2.5.4.	Resistencias pull-up I2C	25
2.5.5.	Reemplazo de DESPI-C02	25
2.5.6.	Condensadores de desacople	26
2.5.6.a.	Condensadores para SCD41	26
2.6.	Diseño de carcasa	26
2.6.1.	Justificación del diseño	27
2.6.2.	Diseño en Fusion 360	27
2.7.	Diseño de PCB	27
2.7.1.	Tamaño de los componentes	27
2.7.2.	Ubicación de los componentes	28
2.7.3.	Conexiones de prueba	29
2.7.4.	Lista de materiales	30
3.	DESARROLLO DE FIRMWARE	32
3.1.	Entorno de desarrollo	32
3.1.1.	Crear proyecto de PlatformIO	33
3.2.	Zephyr y sus archivos de configuración	34
3.3.	Bluetooth Low Energy (BLE)	34
3.4.	Lectura de sensor SCD41	36
3.5.	Configuración y manipulación de pines	36

3.6. Pantalla e-paper	39
3.7. Conversor análogo-digital	40
3.8. Interrupción del botón central	42
3.9. Sensor SGP40	42
3.10. Sensor BME688	43
4. PRUEBAS DE INTEGRACIÓN	44
4.1. Programación del microcontrolador	45
4.2. Pruebas de integración	45
4.3. Consumo de energía	46
4.4. Validación de paquetes Bluetooth	47
5. RESULTADOS EXPERIMENTALES	49
5.1. Primeras impresiones	49
5.2. Ensamble de la PCB	50
5.2.1. Ensamble y primer error	51
5.2.2. Segundo ensamble	51
5.2.3. Tercera y última placa	53
5.3. Pruebas del subsistema de alimentación	53
5.3.1. Configuración de las pruebas	53
5.3.1.a. Condiciones solares	55
5.3.2. Pruebas del divisor resistivo activable	55
5.3.3. Pruebas de carga desde cero con energía solar	56
5.3.4. Pruebas de autonomía	56
5.3.5. Carga en vacío	59
5.3.6. Carga con la salida conectada	59
5.3.7. Histéresis para evitar sobrevoltaje	61
5.3.8. Luz indirecta durante un día	61
5.3.9. Autodescarga	62
5.3.10. Balanceo de supercondensadores	63
5.4. Subsistema funcional y prueba de sensores	63
5.4.1. Medición de CO2 en un dormitorio	64
5.4.2. Comparación SCD41 versus SGP40	64
5.5. Ensamble final con la carcasa	65
6. CONCLUSIONES Y TRABAJO FUTURO	67
6.1. Resultados obtenidos	67
6.2. Conclusiones	68
6.3. Trabajo Futuro	69
A. HERRAMIENTAS	71
A.1. JouleScope	71
A.2. Programador	71
A.2.1. Black Magic Probe	72
A.2.2. J-Link	73
A.3. Generación de elementos gráficos	73
B. CAPTURA Y GENERACIÓN DE GRÁFICOS A PARTIR DEL PUERTO SERIAL	76

B.1. Conversor UART a USB	76
B.2. Captura de datos	77
B.3. Generación de gráficos	79
C. ANEXO: PCB Y DIAGRAMA ESQUEMÁTICO	80
BIBLIOGRAFÍA	86

Índice de figuras

1.1. Principio de funcionamiento de los sensores con tecnología MOx.	3
1.2. Principio de funcionamiento de los sensores con tecnología NDIR.	4
1.3. Sensores disponibles en el mercado.	6
2.1. Subsistema de alimentación para el resto del circuito.	9
2.2. Subsistema funcional, encargado de lectura, procesamiento, y disposición de los datos.	10
2.3. Detalle del módulo E73-2G4M08S1C.	14
2.4. Estructura de componentes para el funcionamiento de una pantalla e-paper.	15
2.5. Pantalla de tinta electrónica GDEH0154D67 comparada con una moneda de 100 pesos.	16
2.6. Placa adaptadora DESPI-C02 para conectar la pantalla GDEH0154D67 al microcontrolador.	16
2.7. Diagrama funcional simplificado del AEM10941.	17
2.8. LED y su resistencia limitadora.	19
2.9. Comparación entre EDLC (izquierda) y LIC (derecha).	22
2.10. Protección 5V.	24
2.11. Divisor resistivo activable.	26
2.12. Modelo 3D de la carcasa en Fusion 360.	28
2.13. Render 3D de la PCB.	29
3.1. Creación de proyecto en PlatformIO.	33
3.2. Jerarquía de los Perfiles, Servicios y Características de las transacciones GATT de BLE.	35
3.3. Elementos gráficos que forman la interfaz para mostrar la información.	40
4.1. Protoboard para las pruebas preliminares de integración de componentes.	45
4.2. Perfil de corriente de una medición del sensor SCD41.	46
4.3. Captura de pantalla de la aplicación nRF Connect.	48
5.1. Placa de circuito impreso real, ambas caras.	49
5.2. Proceso de soldado con stencil, máquina Pick and Place y horno.	50
5.3. Placa de circuito impreso con los componentes soldados.	51
5.4. Curva de carga del arreglo de supercondensadores.	54
5.5. Sol directo e indirecto.	55

5.6. Curva de carga del arreglo de supercondensadores.	57
5.7. Mediciones del consumo de potencia usando JouleScope.	57
5.8. Descarga de supercondensadores alimentando el sistema a través del PMIC.	58
5.9. Carga de los supercondensadores sin conectar la salida.	60
5.10. Carga de los supercondensadores con la salida conectada.	60
5.11. Histéresis cuando los supercondensadores están cargados y se sigue recibiendo luz solar.	61
5.12. Carga de los supercondensadores durante un día completo.	62
5.13. Detalle de carga de supercondensadores antes de recibir sol directo.	62
5.14. Autodescarga de supercondensadores durante 8 horas.	63
5.15. Concentración de CO ₂ durante la noche en un dormitorio cerrado.	64
5.16. Funcionamiento del SGP40 y su VOC index.	65
5.17. Comparación de los sensores SCD41 y SGP40.	65
5.18. Carcasa impresa en 3D ensamblada.	66
A.1. JouleScope y su diagrama interno.	72
A.2. Asignación de pines del J-Link en compatibilidad con SWD.	73
A.3. Ajuste de brillo y contraste para generar el número 3.	74
A.4. Captura del software Image2Lcd para convertir una flecha de la interfaz gráfica a un arreglo de bytes.	74
B.1. Módulo UART-USB CP2102.	77
B.2. Pestaña de configuración de puerto en RealTerm.	78
B.3. Pestaña de captura en Realterm.	78
B.4. Captura de Excel con los datos preparados para graficar.	79
C.1. Capas Top y Top Overlay de la PCB diseñada.	80
C.2. Capas Bottom y Bottom Overlay de la PCB diseñada.	81
C.3. Diagrama esquemático principal, con microcontrolador y circuitos asociados.	82
C.4. Diagrama esquemático relativo principalmente al subsistema de potencia.	83
C.5. Diagrama esquemático relativo al circuito adaptador para la pantalla de tinta electrónica.	84
C.6. Diagrama esquemático relativo a los sensores utilizados y las resistencias para el bus I2C.	85

Índice de tablas

1.1. Comparativa de sensores de CO2 disponibles en el mercado nacional.	5
2.1. Pines del microcontrolador elegidos.	19
2.2. Configuración elegida para los pines CFG.	20
2.3. Características eléctricas del panel solar SM141K07TF.	23
2.4. Pines para configurar el ratio de búsqueda del MPP.	23
2.5. Componentes externos para el AEM10941	25
2.6. Lista de materiales (BOM)	31
3.1. Rangos para calificar cualitativamente la concentración de CO2.	40
5.1. Verificación del balanceo en los supercondensadores.	63

ABREVIACIONES

Mayúsculas

PCB	: Printed Circuit Board
CO2	: Dióxido de Carbono
IAQ	: Indoor Air Quality
IAP	: Indoor Air Pollution
NDIR	: Non Dispersive Infrared
IoT	: Internet of Things
RTOS	: Real Time Operating System
BLE	: Bluetooth Low Energy
GATT	: Generic Attribute Profile
GAP	: Generic Access Profile
UUID	: Universally Unique Identifier
MPPT	: Maximum Power Point Tracking
SCAP	: Super Capacitor
EDLC	: Electrochemical Double Layer Capacitor
LIC	: Lithium-ion Capacitor
IC	: Integrated Circuit
PMIC	: Power Management Integrated Circuit
LDO	: Low-Dropout regulator
GND	: Tierra
SWD	: Serial Wire Debug
I2C	: Inter-Integrated Circuit
SDA	: Serial Data
SCL	: Serial Clock
SPI	: Serial Peripheral Interface
UART	: Universal Asynchronous Receiver-Transmitter
TX	: Transmisor
RX	: Receptor
USB	: Universal Serial Bus
ADC	: Analog to Digital Converter
EPD	: Electrophoretic Display
TCon	: Timing Controller
HAT	: Hardware Attached on Top
FPC	: Flat Panel Cable
SMD	: Surface-Mount Device

LED	: Light-Emitting Diode
PWM	: Pulse-Width Modulation
IDE	: Integrated Development Environment
PIO	: PlatformIO
SDK	: Software Development Kit
DK	: Development Kit
DT	: Device Tree
GPIO	: General Purpose Input/Output
MCU	: Microcontroller Unit
BMP	: Black Magic Probe

Minúsculas

dc	: direct current
ppm	: partes por millón
px	: pixeles

INTRODUCCIÓN

La calidad del aire en interiores (IAQ) es un factor importante para la salud y bienestar de las personas. El dióxido de carbono (CO₂) es uno de los principales indicadores de la IAQ y su concentración puede utilizarse como indicador del estado de ventilación. En interiores donde no haya combustión, el CO₂ es producido principalmente por la respiración de humanos y animales, por lo que la concentración de este está directamente relacionado con la ocupación y actividad de las personas presentes [1].

1.1. Contexto

A nivel mundial, la calidad del aire en interiores está relacionada a muertes prematuras de 3.2 millones de personas anualmente, según cifras de la OMS [2].

En general siempre se habla mucho respecto del smog y la mala calidad del aire en exteriores, así como sus diversas fuentes de contaminación tales como industrias, vehículos, calefacción a leña, etc. Sin embargo, poco se sabe de lo que sucede en el interior de las casas u oficinas, es muy raro escuchar acerca de las fuentes de contaminación y los problemas en la salud que puede provocar la mala calidad del aire en interiores.

Se estima que en la sociedad actual las personas pasan en torno al 90% del tiempo en ambientes cerrados, ya sea en el hogar, transporte, lugar de trabajo/estudio, etc [3]. Por este motivo, resulta importante tener un control y/o monitoreo de la calidad del aire que las personas respiran en su día a día, ya que no solo afecta en la salud física, sino también en aspectos como la concentración y la capacidad de aprendizaje [4].

Gracias a la popularización del internet de las cosas (IoT) y a las tecnologías de casas inteligentes cada vez es más común la utilización de sensores y actuadores [5]. Gracias a esto, un sensor de calidad del aire podría detectar una mala condición de ventilación y emitir una alerta al usuario o directamente activar el sistema de ventilación.

Además, en el contexto de la Industria 4.0, la implementación de tecnologías IoT y

sensores inteligentes se ha convertido en un requisito esencial para mejorar la eficiencia y productividad de los procesos industriales. En particular, el monitoreo y control de la calidad del aire en ambientes interiores es crítico en diversas aplicaciones industriales, tales como la fabricación de alimentos y productos farmacéuticos, la automatización de edificios, y la monitorización ambiental en instalaciones industriales. En este contexto, un sensor de CO₂ puede proporcionar una solución eficiente y de bajo costo para el monitoreo y control de la calidad del aire en aplicaciones industriales, de oficina y hogareñas.

1.1.1. Calidad del aire y COVID-19

Debido a la pandemia del COVID-19 se ha evidenciado la importancia de mantener los espacios cerrados correctamente ventilados. El virus que causa COVID-19 se propaga principalmente a través de gotas respiratorias, que pueden transmitirse por el aire cuando una persona infectada habla, tose o estornuda [6]. Una mala calidad del aire en interiores puede aumentar la concentración de estas gotas respiratorias en el aire, lo que puede aumentar el riesgo de transmisión [7].

Acciones como mantener la distancia social, lavado de manos o el uso de mascarillas son buenas medidas para reducir el contagio de coronavirus. Mantener una buena ventilación para reducir la exposición a aerosoles en espacios confinados también es una buena solución [8].

1.1.2. Trabajo a desarrollar

El trabajo principal de esta memoria de titulación es el desarrollo de un dispositivo capaz de medir la temperatura, humedad y, lo más importante, la concentración de CO₂ del entorno. Dichas mediciones deben ser mostradas a través de una pantalla incluida en el aparato, junto con enviarse de manera inalámbrica a través de Bluetooth. Para garantizar su autonomía en ausencia de fuentes de alimentación externas, el dispositivo debe generar su propia energía mediante un sistema de Energy Harvesting que aprovecha la energía lumínica del ambiente para su funcionamiento. En particular, se utiliza un panel solar incorporado y un supercondensador para almacenar la energía recolectada.

1.2. Estado del arte

El estudio del estado del arte se enfocó principalmente en tres temas: los agentes contaminantes del aire en interiores, las tecnologías existentes para medir la calidad del aire y la búsqueda y comparación de dispositivos disponibles en el mercado similares al que se va a desarrollar.

1.2.1. Agentes contaminantes

Hay muchas fuentes de contaminación del aire en interiores. Los compuestos que contaminan el aire pueden provenir de la actividad humana, como el uso de productos químicos para la limpieza, el humo del tabaco, el uso de materiales de construcción y de mobiliario, y la actividad de cocinar. También pueden provenir de fuentes externas, como el humo, el polvo, los ácaros del polvo, y los productos químicos que se filtran desde el exterior. Es importante tener en cuenta que incluso si una fuente de contaminación del aire está fuera

de un edificio, puede afectar la calidad del aire en su interior [3].

Los VOC (compuestos orgánicos volátiles, por sus siglas en inglés) son una clase de contaminantes del aire que se encuentran en muchos productos químicos y materiales de construcción. Los VOC pueden ser emitidos por productos como pinturas, disolventes, pegamentos, cosméticos, materiales de limpieza y muchos otros. Estos compuestos pueden tener efectos negativos en la salud humana, ya que algunos de ellos son considerados tóxicos o cancerígenos [9].

Debido a su impacto en la calidad del aire interior y la salud humana, medir los niveles de VOC es una práctica común en el monitoreo de la calidad del aire en interiores, especialmente en edificios de oficinas, hogares y otros espacios cerrados.

1.2.2. Tecnologías para estimar la calidad del aire

Existen distintas tecnologías para estimar la calidad del aire, dependiendo del tipo de contaminante se quiere medir, pero el estudio se centrará en VOCs y concentración de CO₂.

1.2.2.a. Compuestos Volátiles Orgánicos (VOC)

Generalmente, para medir gases, y más específicamente, VOCs se suelen emplear sensores que utilizan la tecnología MOx (Metal Oxide) [10].

Los sensores MOx consisten en un material semiconductor de óxido metálico, que cambia su resistencia eléctrica en presencia de ciertos gases. Este cambio en la conductividad eléctrica se puede medir y se correlaciona con la concentración de gas en el entorno.

Sin embargo, los sensores MOx también tienen algunas limitaciones, como su sensibilidad a la temperatura y la humedad, y su tendencia a desviar los resultados de las mediciones a lo largo del tiempo. Estas limitaciones pueden requerir ajustes y calibraciones periódicas para garantizar la precisión de las mediciones. El principio de funcionamiento se muestra en la Figura 1.1.

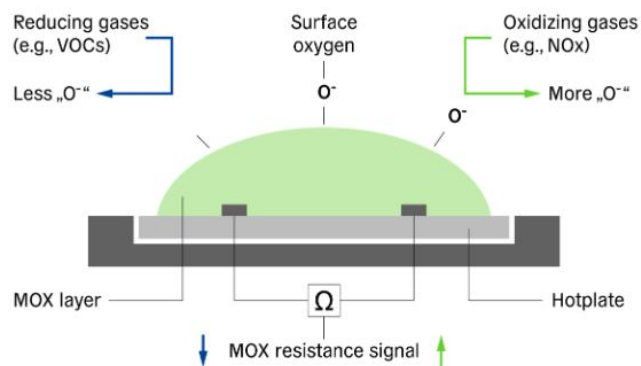


Figura 1.1: Principio de funcionamiento de los sensores con tecnología MOx.

1.2.2.b. Concentración de CO₂

Existen sensores que estiman la concentración de CO₂ a partir de la correlación que existe entre y otros gases, utilizando la tecnología MOx.

Sin embargo, para medir directamente el CO₂, la tecnología NDIR (Nondispersive infrared sensor) es la más utilizada en este tipo de sensores [11]. Se basan principalmente en la gran capacidad que tienen las moléculas de dióxido de carbono para absorber la luz infrarroja. Gracias a esta cualidad, es posible iluminar una muestra de gas con esta luz y luego calcular la cantidad que se absorbió para luego estimar la concentración de CO₂. Este principio se muestra en la Figura 1.2.

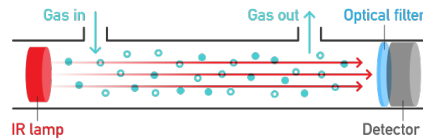


Figura 1.2: Principio de funcionamiento de los sensores con tecnología NDIR.

La tecnología NDIR tiene dos variantes [11]:

- Transmissive NDIR: esta variante detecta la luz absorbida de manera óptica, por ejemplo, mediante un fotodiodo.
- Photoacoustic NDIR: como indica su nombre, esta variante utiliza un micrófono para medir las vibraciones que producen las moléculas de CO₂ al absorber la luz infrarroja.

1.2.3. Dispositivos comerciales disponibles actualmente

El objetivo final del trabajo es desarrollar un producto comercial, por lo que corresponde analizar el estado del arte de dispositivos similares que ya están disponibles en el mercado.

Se analizan 4 dispositivos de distintos fabricantes/vendedores: Netatmo [12], Medidco [13], MedidoresChile [14] y uno genérico [15], que se encontró en distintas tiendas con distintos nombres. Sus principales características se resumen en la Tabla 1.1.

En [16] se comparan los sensores de Netatmo e IC-Meter [17], sin embargo, no fue posible hallar suficiente información de este último. En dicho estudio se concluye que el dispositivo de Netatmo indica valores distintos frente a cambios de temperatura.

Todos los modelos analizados indican usar la tecnología NDIR para medir el CO₂, pero no especifican la variante específica.

1.3. Objetivos

El objetivo principal es diseñar un dispositivo de monitoreo de CO₂ que supere la calidad y limitaciones de los dispositivos existentes en el mercado. En particular, se busca desarrollar un dispositivo que pueda ser utilizado en diferentes aplicaciones industriales y comerciales, así como también en hogares y oficinas.

Fabricante	Netatmo	Medidco	MedidoresChile	OEM
Modelo	Healthy Home Coach	AM34	M2	-
Dimensiones	45 x 45 x 155 mm	94 x 94 x 30 mm	15 x 8 x 10 cm	9 x 7 x 3,5 cm
Instalación	Sobremesa	Pared/sobremesa	Sobremesa	Pared
Sensores	CO2, humedad, temperatura y ruido	CO2, humedad y temperatura	CO2, humedad y temperatura	CO2, humedad y temperatura
Rango CO2	0 - 5 000 ppm	0 - 6 000 ppm	400 - 5 000 ppm	400 - 5 000ppm
Precisión	± 50 ppm (de 0 a 1 000 ppm)	Sin información	Sin información	Sin información
Visualización	No	Pantalla LCD 2.75 ”	Pantalla LCD	Pantalla LCD
Conectividad	Apple HomeKit (WiFi)	No	No	No
Alimentación	USB	USB y batería 1500mAh (32 h)	USB y batería 3000mAh	USB-C y batería de litio 1200mAh
Precio	159.990 CLP	75.990 CLP	149.990 CLP	39.990 CLP

Tabla 1.1: Comparativa de sensores de CO2 disponibles en el mercado nacional.

El dispositivo será diseñado desde cero y será la primera iteración de este, por lo que se tendrá que idear, diseñar, buscar e integrar cada uno de los componentes necesarios para su construcción, incluyendo sensores, el sistemas de transmisión de datos inalámbricos, sistemas de alimentación mediante Energy Harvesting, y otros componentes necesarios para lograr un dispositivo de calidad. Se espera que este dispositivo pueda ayudar a tomar acciones correctivas para mejorar la calidad del aire interior y garantizar la salud y bienestar de las personas.

1.4. Alcances y contribución

Se espera desarrollar un producto mínimo viable. Solo medirá temperatura, humedad relativa, concentración de CO2 y compuestos volátiles orgánicos, no considerando otros contaminantes del aire como el material particulado, por ejemplo.

Por otro lado, es necesario destacar que este proyecto no incluirá el desarrollo de una aplicación para teléfonos móviles u otro dispositivo que reciba y procese los datos del sensor. Sin embargo, se espera que el sensor pueda comunicarse de forma inalámbrica a través de Bluetooth y que se puedan analizar los datos que envía utilizando una aplicación genérica para verificar su funcionamiento.

En cuanto a la contribución del proyecto, se espera que el prototipo desarrollado sirva como base para futuros dispositivos de monitoreo de calidad del aire en diferentes aplicaciones, ya sea en hogares, oficinas o industrias. Se espera que el uso de Energy Harvesting en la alimentación del dispositivo sea un aporte significativo para proyectos e investigaciones futuras.

1.5. Estructura del documento

Este documento está compuesto por seis capítulos principales y tres apéndices.

- **Capítulo 1: Introducción.** Introduce el tema a trabajar, presenta la relevancia



(a) Netatmo.



(b) Medidco.



(c) Medidores Chile.



(d) OEM.

Figura 1.3: Sensores disponibles en el mercado.

de este en base al contexto actual y se estudia el estado del arte en dos enfoques: dispositivos similares disponibles en el mercado y tecnologías para medir la calidad del aire.

- **Capítulo 2: Diseño.** Se presentan los requisitos de diseño y se propone un diagrama de bloques para enfrentar el problema. Se eligen componentes en base a ciertos criterios y se continúa el diseño en esa línea, para finalizar el capítulo con la placa de circuito impreso (PCB) ya diseñada.
- **Capítulo 3: Desarrollo de firmware.** Explica las decisiones tomadas respecto al entorno de desarrollo y la implementación del código relativo a de cada uno de los componentes.
- **Capítulo 4: Pruebas de integración.** A partir de los componentes de hardware

principales, se realizaron pruebas para asegurar que estos fueran compatibles entre sí.

- **Capítulo 5: Resultados experimentales.** Se analiza y ensambla la PCB para poder realizar pruebas experimentales principalmente de autonomía energética y de calidad de los sensores.
- **Capítulo 6: Conclusiones y trabajo futuro.** Se resumen y comentan los resultados obtenidos. Se cierran ideas y se proponen caminos a seguir para mejorar el sistema.
- **Apéndice A: Herramientas.** Se presentan las herramientas para programar y depurar el sistema. También se muestran las herramientas de software para generar elementos gráficos para la pantalla.
- **Apéndice B: Captura y generación de gráficos a partir del puerto serial.** Se explica el procedimiento llevado a cabo para generar gráficos.
- **Anexo C: PCB y diagrama esquemático.** Se adjuntan las capas del diseño de PCB así como su diagrama esquemático.

DISEÑO

En este capítulo se describe el proceso de diseño del dispositivo así como la justificación para las decisiones tomadas.

2.1. Requerimientos de diseño

En esta sección se enumeran los requerimientos de diseño para resumir de mejor manera las capacidades que este dispositivo debe cumplir.

1. Debe ser energéticamente autónomo, es decir, debe ser capaz de generar la energía necesaria para su funcionamiento y almacenar el excedente para usarlo cuando la fuente de energía no esté disponible.
2. Para propósitos del punto anterior, debe contar con un panel solar incorporado con la potencia necesaria para darle autonomía las 24 horas del día, a partir luz solar o artificial.
3. Sin desmedro de lo anterior, el dispositivo debe incluir un puerto USB con el cual se pueda alimentar desde una fuente de 5V DC en caso que este sea instalado en un lugar que no cuente con luz suficiente.
4. Para almacenar la energía se deben usar supercondensadores, quedando descartada la opción de usar baterías, ya sean recargables o no.
5. La medición de los sensores debe ser desplegada en una pantalla incluida en el dispositivo.
6. A lo anterior se incluye el envío periódico de los datos mediante algún protocolo de transmisión inalámbrica.
7. Resulta imperativo que el bajo consumo de energía esté considerado en el desarrollo de cada una de las etapas de este dispositivo. Esto, debido a las limitaciones de energía producto de la baja densidad de potencia de los supercondensadores (SCAP), en comparación a las baterías de litio, y que, idealmente, toda la energía provendrá de la luz ambiental (que está disponible solo unas horas del día).

2.2. Esquema general

En base a los requerimientos de diseño, se propone un diagrama de bloques funcionales. Para mayor claridad, se separa en dos partes; el Subsistema de alimentación (Fig. 2.1) y el Subsistema funcional (Fig. 2.2).

2.2.1. Subsistema de alimentación

Este Subsistema está encargado de proporcionar la energía para todo el resto del sistema, principalmente a partir de *energy harvesting*. Todas las flechas del Subsistema de alimentación representan transmisión de energía eléctrica.

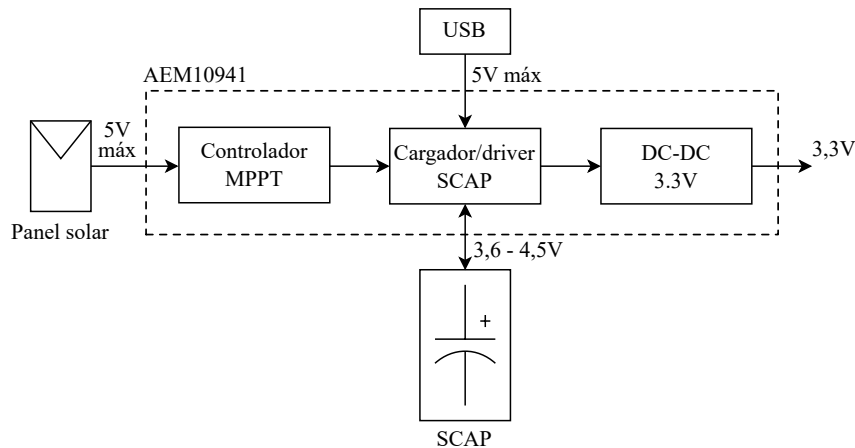


Figura 2.1: Subsistema de alimentación para el resto del circuito.

- **Panel solar:** Se encarga de convertir la energía lumínica en energía eléctrica.
- **Controlador MPPT:** Ejecuta un algoritmo MPPT para seguir el punto de máxima potencia (MPP) y operar el panel solar ahí, con el fin de aprovechar al máximo posible la energía.
- **Cargador/driver SCAP:** Este bloque se encarga de cargar y descargar el SCAP de manera controlada.
- **DC-DC 3.3V:** El voltaje del supercondensador bajará a medida que se descargue. Este bloque se encarga de entregar una salida estable de 3.3V DC.
- **USB:** Conector USB para alimentar el sistema en caso que no sea posible generar energía desde fuentes lumínicas.
- **SCAP:** Debe almacenar la energía recolectada por el panel solar para luego proporcionarla al resto del circuito según se requiera.

2.2.2. Subsistema funcional

Todos los componentes de este subsistema requieren una alimentación de 3.3V que está provista por el Subsistema de alimentación. Para mayor claridad solo se dibujan las

interacciones que representan intercambio de información. Notar que los buses de datos SPI e I2C en realidad son periféricos internos, propios del microcontrolador, pero están dibujados de esta manera para enfatizar que tienen contacto directo con el exterior y para especificar a qué está conectado cada elemento.

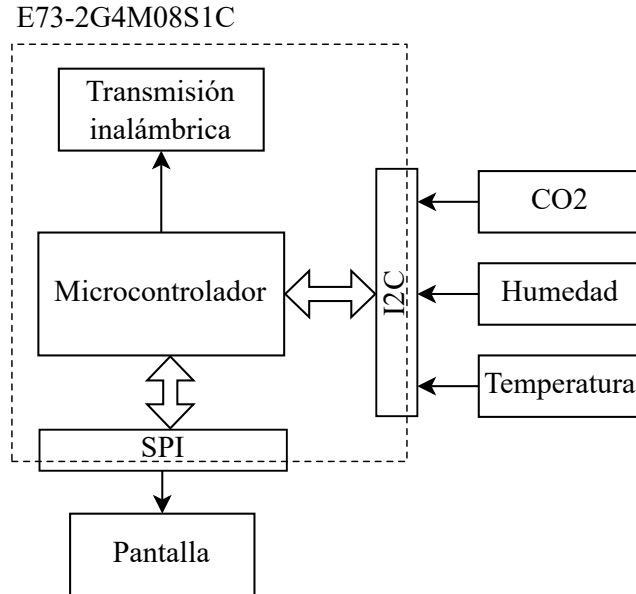


Figura 2.2: Subsistema funcional, encargado de lectura, procesamiento, y disposición de los datos.

- **Transmisión inalámbrica:** Los datos recolectados por los sensores se enviarán por un protocolo de transmisión inalámbrica.
- **Microcontrolador:** El microcontrolador estará encargado de solicitar y procesar la lectura de los datos de los sensores. Adicionalmente, debe decidir el momento en que los datos se envían por el protocolo de transmisión inalámbrica.
- **Pantalla:** Los datos de los sensores deben ser desplegados en un display o pantalla.
- **Sensores:** Los sensores son los que finalmente se encargan de medir las condiciones del ambiente y se comunican con el microcontrolador por medio del protocolo I2C.

Luego de analizar opciones se encontraron componentes que hacen el trabajo de más de un bloque funcional, por lo cual, en los diagramas de bloques anteriores, existen módulos en línea punteada que están compuestos por más de un bloque. De esta manera se enfatiza que se utilizan componentes que realizan más de una función.

2.3. Criterios de elección de componentes

En esta sección se describen los criterios por los cuales se escogieron los principales componentes del sistema, relacionando dichos criterios con el dispositivo a desarrollar y lo que se quiere lograr. También se enumeran alternativas de solución a modo de referencia, pero la elección final y su justificación se detallan en la sección siguiente.

2.3.1. Microcontrolador/SoC

- Consumo de energía: La energía es la mayor limitante en este proyecto, por esto la ponderación que se le asigna a este punto es alta.
- Tecnología de transmisión inalámbrica: Se valora que el microcontrolador/SoC (System On Chip) incorpore de serie compatibilidad con protocolos de transmisión inalámbrica, así como la variedad e importancia de estos.
- Calidad de recursos: La calidad de recursos o de ayuda disponible en Internet es muy relevante ya que agiliza el flujo de trabajo y permite la rápida resolución de problemas.

Alternativas: familias de microcontroladores nRF52, ESP32, STM32.

2.3.2. Tecnología de transmisión inalámbrica

- Velocidad de transmisión: Tasa de transmisión. Se considera muy poco relevante en este caso ya que la cantidad de datos a enviar es muy pequeña y de bajo riesgo.
- Popularidad/compatibilidad: Potencial compatibilidad con otros dispositivos. Se considera importante contar dispositivos potencialmente compatibles y que estos sean accesibles o sean de fácil acceso (como un teléfono móvil).
- Eficiencia energética: La eficiencia es muy relevante dadas las limitaciones de energía presentes.
- Rango de alcance: El rango o distancia para transmitir datos es poco relevante ya que el dispositivo a desarrollar está orientado a interiores y no a grandes distancias.
- Incluido con el microcontrolador: Se valora que el protocolo se incluya de serie en algún microcontrolador/SoC.
- Facilidad de implementación: La calidad de recursos y facilidad/dificultad de implementación es importante para el éxito del sistema.

Alternativas: Bluetooth, LoRa, Wi-Fi, Zigbee, Matter.

2.3.3. Sensores

- Tecnología y gases que mide: La calidad de la tecnología usada para detectar y cuáles gases detecta.
- Rango de medición y precisión: Esta característica es importante para asegurar la calidad del producto final.
- Sensor digital: Se prefiere contar con un sensor digital en lugar de uno análogo. Son más modernos y se pueden conectar múltiples sensores en el mismo bus I2C.
- Consumo de energía: Dadas las limitaciones de energía, se considera importante que sea eficiente.
- Calidad de documentación: Calidad de los recursos proporcionados por el fabricante, como ejemplos, hojas de datos y notas de aplicación.

- Incluye sensor de temperatura y humedad: Se valora la inclusión de sensor de temperatura y humedad, a pesar de que se pueda incluir un sensor para este motivo específico, es útil contar con redundancia para asegurar una buena medición.

Alternativas: SCD40/41, SGP30/40, BME680/688, MiCS-VZ-89TE, CCS811, SCD30.

2.3.4. Sistema de recolección de energía

- MPPT: Inclusión de algoritmo de búsqueda del máximo punto de potencia.
- Fuente auxiliar: Capacidad para alimentar el sistema desde una fuente auxiliar.
- Salida regulada: Calidad y eficiencia de la salida de voltaje.

Alternativas: AEM10941, SVP1040/1050, bq25570, bq25505, MAX38889.

2.3.5. Pantalla

- Consumo de energía: Dadas las limitaciones de energía de la aplicación, se considera muy importante.
- Tasa de refresco: Cada cuánto tiempo se puede actualizar el contenido en pantalla. Es muy poco relevante para la aplicación.
- Colores: Cantidad de colores que puede representar. Se valora pero no es de gran relevancia para la aplicación.
- Contraste en ambientes luminosos: Es importante ya que se usarán paneles solares, es decir, el dispositivo estará expuesto a la luz solar o a interiores bien iluminados.
- Resolución: Cantidad de pixeles o capacidad de representar figuras. Se valora pero no es indispensable para la aplicación.

Alternativas: tecnologías de pantalla e-paper, OLED, LCD, TFT.

2.4. Elección de componentes principales

A continuación se presentan los componentes seleccionados equivalentes a los bloques de las figuras 2.1 y 2.2. Se explican los motivos de la decisión en base a los criterios de elección, pero sin considerar el diseño de componentes pasivos u otros circuitos adicionales.

2.4.1. Microcontrolador y tecnología de transmisión inalámbrica

La elección del microcontrolador y el protocolo de transmisión inalámbrica se presentan juntos debido a que están íntimamente relacionados. También es posible usar un microcontrolador común y un módulo inalámbrico dedicado, pero las opciones integradas disponibles actualmente satisfacen perfectamente los requerimientos.

LoRa se descarta porque su principal ventaja es la transmisión a distancias muy largas e idealmente en línea de vista. Situaciones muy distintas a las que se tiene en este caso, que está ideado para interiores.

Wi-Fi se descartó tempranamente porque es un protocolo muy complejo, está desarrollado para Internet, tiene una alta tasa de transmisión que en este caso es irrelevante y lo más importante, Wi-Fi consume mucha energía, lo que puede ser un problema en un dispositivo que se alimenta con *energy harvesting*. Dado que la energía disponible es limitada, no es suficiente para mantener el Wi-Fi activo.

Usar Wi-Fi es la principal razón para elegir ESP32, que es una alternativa típica para dispositivos de IoT. Sin embargo, como se descartó usar Wi-Fi, esta familia de microcontroladores pierde su principal ventaja. Además, no está diseñada para aplicaciones de bajo consumo.

Finalmente, se optó por usar un chip de la familia nRF52 de Nordic Semiconductor. Esta decisión proviene inicialmente de tomar como inspiración una pulsera de actividad (Xiaomi Mi Band 5). Ese tipo de dispositivos tiene una batería muy pequeña (debido a su restricción de espacio), una pantalla, una serie de sensores y conexión Bluetooth que envía datos constantemente al celular. Todo esto con una duración media de batería de una semana corrida.

Lo mencionado anteriormente describe bastante bien lo que se busca en el dispositivo a desarrollar; que lea un sensor, muestre el dato en pantalla, envíe estos de manera inalámbrica y que tenga el menor consumo posible de energía.

El modelo de pulsera referido usa un SoC (System On Chip) de la empresa Dialog Semiconductor, pero al seguir investigando se descubrió a PineTime, un reloj Open Source desarrollado con un chip nRF52832 [18]. Adicional a esto, se descubrieron dispositivos similares Open Hardware que usan la misma familia de chips [19].

Recientemente, otro reloj inteligente Open Source ha sido noticia, el ZSWatch que también utiliza un SoC de la familia nRF52, específicamente el nRF52833 [20] [21]. Además, está basado en Zephyr, algo que será relevante más adelante.

De los distintos SoC que componen la familia nRF52 de Nordic Semiconductor, se optó en particular por el nRF52840 [22], ya que en el momento de la elección era el que tenía mejores prestaciones y mejor compatibilidad con distintos protocolos de transmisión. Cuenta con una CPU ARM Cortex-M4 de 32 bits, 1 MB de Flash, 256 KB de RAM, frecuencia de hasta 64 MHz e incorpora un sistema para administrar el consumo de potencia con el cual logra un bajo consumo de potencia, lo que lo hace ideal para esta aplicación en la que se dispone de poca energía. Su voltaje de alimentación es 3,3 V.

Entre las tecnologías de transmisión inalámbrica que dispone, se encuentran Bluetooth Low Energy (BLE), Bluetooth Mesh, Thread, Zigbee y Matter. Se eligió trabajar con la primera opción, BLE, debido principalmente a que Bluetooth está masificado y la transmisión de datos se puede comprobar fácilmente con un teléfono celular.

Este tipo de chips necesitan de otros componentes para funcionar, entre ellos, la antena. Para evitar diseñar esta última y las dificultades que conlleva, se eligió utilizar un módulo, específicamente el E73-2G4M08S1C.

Por lo general, este tipo de pantallas siguen una estructura como la que se muestra en la Figura 2.4, la cual se extrajo desde [24].

Estas pantallas se componen principalmente por el panel EPD (Electrophoretic Display), que corresponde a la parte donde se visualizan las imágenes. El Driver IC o TCon (Timing controller), que genera las señales específicas para desplegar la imagen a partir de los comandos que recibe del MCU (Microcontroller Unit), que en este caso específico corresponde al nRF52840. Y el EPD Driving Circuit, que genera los voltajes necesarios para manejar el panel con una serie de transistores, diodos, inductores y condensadores.

Usualmente, además de vender el panel, los fabricantes también comercializan el EPD Driving Circuit, pero por lo general lo llaman HAT (Hardware Attached on Top). Esta se vende como una placa que, en un lado se conecta al microcontrolador y en la otra lleva un conector FPC para la pantalla. En estas placas, las señales SPI y la alimentación pasan directo, tal como lo muestra la Figura mencionada. Se referirá a esto como placa o circuito adaptador.

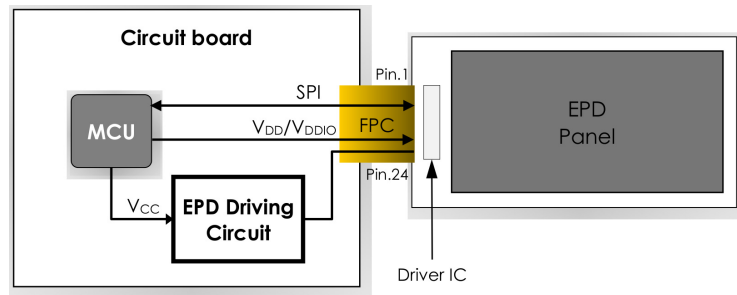


Figura 2.4: Estructura de componentes para el funcionamiento de una pantalla e-paper.

2.4.2.b. GDEH0154D67

Entre los diversos modelos de pantalla e-paper disponibles, se eligió el GDEH0154D67. Esta es una pantalla de tinta electrónica de solo 1.54 pulgadas, fabricada por la compañía Good Display [25]. Tiene una resolución de 200x200 píxeles y solo puede desplegar imágenes monocromáticas (blanco y negro). Se comunica con el MCU mediante el protocolo SPI y otros tres pines: BUSY, DAT/COM y RES. Permite tanto actualización parcial como completa y se alimenta con 3,3 V. En la Figura 2.5 se muestra su aspecto físico comparado con una moneda chilena de 100 pesos, para tener referencia de su tamaño.

En su interior, esta pantalla incluye el driver SSD1681 [26], el cual, como se mencionó, es el que realmente recibe las instrucciones del microcontrolador. Aparte de esto, se necesita el circuito adaptador que genera los voltajes necesarios para manejar el panel.

En una primera instancia, para verificar el funcionamiento de la pantalla, se utiliza el adaptador desarrollado y recomendado por la misma empresa fabricante, denominado DESPI-C02 [27].

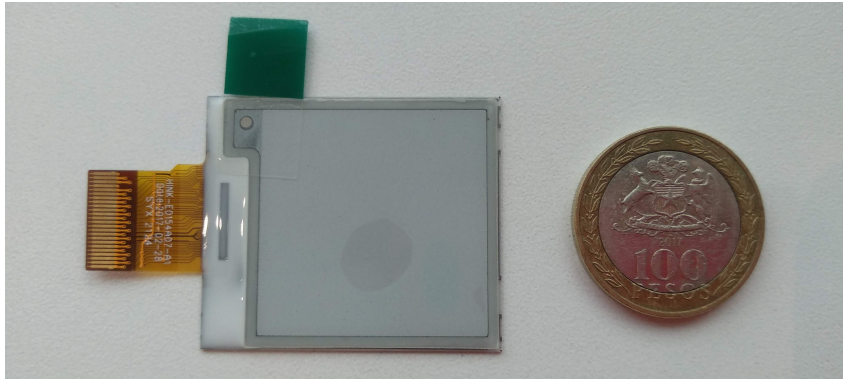


Figura 2.5: Pantalla de tinta electrónica GDEH0154D67 comparada con una moneda de 100 pesos.

2.4.2.c. DESPI-C02

En la Figura 2.6 se muestra la placa del DESPI-C02. Notar que cuenta con un interruptor que permite seleccionar el valor de una de sus resistencias, puesto que está pensada para ser compatible con múltiples modelos de pantalla que dispone el fabricante.

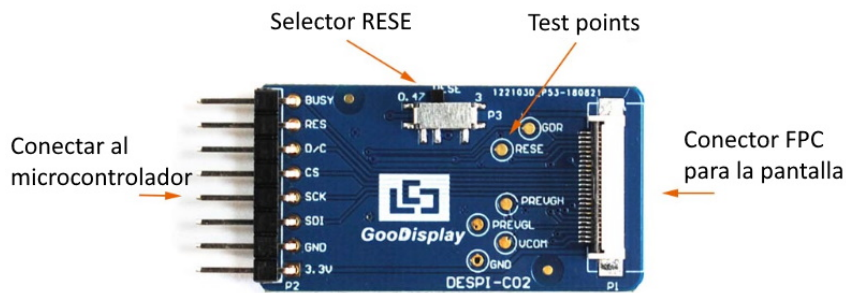


Figura 2.6: Placa adaptadora DESPI-C02 para conectar la pantalla GDEH0154D67 al microcontrolador.

2.4.3. AEM10941

Se tomó la decisión de usar un circuito integrado que sea capaz de resolver todas las principales tareas relacionadas al Subsistema de alimentación. Estas son: carga de supercondensadores por a partir de la luz solar, búsqueda del MPP, salida regulada de 3,3 V y posibilidad de alimentar por USB en caso de ser necesario.

El AEM10941 [28] es un PMIC (Power Management Integrated Circuit) fabricado por e-peas, especialmente diseñado para cosechar energía del sol. Puede extraer potencia del panel fotovoltaico con voltajes desde 50mV hasta 5V y con esto alimentar su salida, a la vez que se recarga el elemento almacenador de energía. Gracias a los pines de configuración con

los que cuenta se pueden seleccionar distintos voltajes de corte superior e inferior, lo que permite utilizar supercondensadores o baterías de distintas tecnologías.

En la figura 2.7 se muestra un diagrama simplificado de este PMIC, donde se observan principalmente las entradas y salidas, así como los principales componentes pasivos necesarios para su funcionamiento. En la hoja de datos [29] se encuentra disponible una versión más detallada de este diagrama.

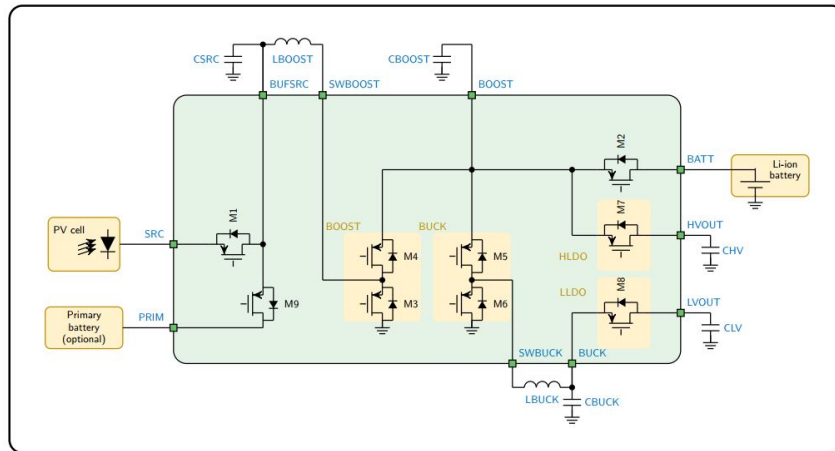


Figura 2.7: Diagrama funcional simplificado del AEM10941.

2.4.4. Sensores

Se eligió el SCD41 para ser el sensor principal del dispositivo a desarrollar. Además, se consideraron otros dos sensores para probarlos y comparar sus mediciones con el sensor principal. Gracias a que los tres sensores se comunican con el MCU mediante el bus I2C, solo se necesitan dos pines para la lectura de datos.

2.4.4.a. Sensor de CO₂: SCD41

El fabricante Sensirion cuenta con la familia de sensores de CO₂ SCD4X, compuesta por el SCD40 y SCD41 [30]. Este último, destaca sobre el primero porque tiene la capacidad de generar mediciones bajo demanda (single-shot), con lo que se logra un menor consumo de energía. En cambio, el SCD40 solo permite realizar mediciones periódicas a un intervalo fijo de 5 o 30 segundos dependiendo el modo, pero sin posibilidad de ajuste manual. Además de la concentración de CO₂, este sensor también mide la temperatura y la humedad relativa. Este sensor mide CO₂ utilizando la tecnología Photoacoustic NDIR [11].

2.4.4.b. Sensor de VOC: SGP40

Otro de los productos de Sensirion es el sensor de compuestos volátiles orgánicos (VOCs) SGP40 [31]. Este sensor entrega un índice de VOC basado las mediciones de las últimas 24 horas. Para tomar mediciones de VOC necesita que se le entreguen valores actuales de

temperatura y humedad relativa, puesto que no incorpora hardware para hacer esto por sí mismo [32]. Este sensor utiliza la tecnología MOx [10].

2.4.4.c. Sensor de gas: BME688

El sensor BME688 está fabricado por Bosch Sensortec [33]. Además de medir VOCs, es capaz de detectar otros gases como compuestos volátiles de azufre, monóxido de carbono e hidrógeno. Adicional a esto, también mide temperatura, humedad relativa y presión atmosférica [34]. Todos estos datos se pueden postprocesar para obtener salidas digitales entre las que se incluye CO₂ equivalente, por lo que es un buen candidato para reemplazar el SCD41 en caso de ser necesario. Este sensor también utiliza la tecnología MOx para medir los gases.

2.5. Diagrama esquemático

En esta sección se presenta el diseño del diagrama esquemático así como la elección de los valores para ciertos componentes.

2.5.1. Elección de pines del microcontrolador

El módulo E73-2G4M08S1C no expone todos los pines que dispone el nRF52840 que lleva dentro. Revisar hoja de datos para mayor información [23]. Además, la mayoría de los pines tienen distintas capacidades o funcionalidades asignadas y/o reservadas. Para evitar complicaciones, la asignación de pines se basó principalmente en la del kit de desarrollo de este SoC, el nRF52840-DK [35]. Los pines adicionales se eligieron de manera arbitraria, pero teniendo en cuenta las capacidades de cada uno.

En la Tabla 2.1 se resumen todos los pines utilizados. Aquellos que no aparecen aquí, se dejaron sin conexión. La configuración para VDD, VDDH y VBUS se basó en el *Circuit configuration no. 6*, sección 7.3.6 del Product Specification del nRF52840 [36].

En la Figura C.3 disponible en el Anexo C se muestra el diagrama esquemático asociado.

2.5.1.a. Oscilador de cristal

El nRF52840 puede funcionar con dos osciladores de cristal: uno *rápido* de 32 MHz y otro *lento* de 32.768 kHz. El microcontrolador usará uno u otro dependiendo de la tarea que esté realizando. El módulo incluye un oscilador de 32 MHz pero no así el de 32.768 kHz. A pesar que se puede prescindir de este último porque se puede generar dentro del chip con un oscilador RC, se prefirió incluir uno dedicado para lograr un mejor desempeño y fiabilidad. Adicionalmente, el oscilador necesita dos condensadores de 12 pF, como lo recomienda la hoja de datos del nRF52840 [36].

2.5.1.b. Botones y LEDs

Se agregan botones para reiniciar el microcontrolador y otro para realizar una función a definir.

Se agregan dos LEDs indicadores conectados por medio de una resistencia de 1k Ω a pines GPIO del MCU. Estos están destinados principalmente para debuggear y conocer el estado de ciertas variables o similar.

Pin E73-	Pin nRF52840	Designación	Descripción
1	P1.11	SCL	Señal SCL del bus I2C
2	P1.10	BUSY	Señal BUSY de la pantalla
3	P0.03/AIN1	BATT_OUT	Salida del div. resistivo, entrada análoga
5	GND	GND	Tierra
6	P1.13	LED2	LED 2
9	P0.31	SCK	Señal SCK del bus SPI, para la pantalla
10	P0.30	MOSI	Señal MOSI del bus SPI, para la pantalla
11	P0.00/XL1	LFCLK	Oscilador de baja frecuencia
13	P0.01/XL2	LFCLK	Oscilador de baja frecuencia
14	P0.06	UART TX	Señal de transmisión del puerto serial
16	P0.08	UART RX	Señal de recepción del puerto serial
17	P1.09	SDA	Señal SDA del bus I2C
19	VDD	VDD	Conectar a 3,3 V
20	P0.12	PB2	Botón central
21	GND	GND	Tierra
23	VDDH	VDDH	Conectar a 3,3 V
24	GND	GND	Tierra
26	P0.18/RESET	RESET	Botón de reinicio
27	VBUS	VBUS	Conectar a tierra
30	P0.17	RES	Señal RES de la pantalla
33	P0.13	LED1	LED 1
34	P0.22	DAT/COM	Señal D/C de la pantalla
35	P0.24	SHUTDOWN_SOON	Señal STATUS 1 del AEM10941
37	SWDIO	SWDIO	Señal de datos para programar por SWD
39	SWDCLK	SWDCLK	Señal de reloj para programar por SWD
40	P1.04	CS	Señal CS del bus SPI, para la pantalla
42	P1.06	BATT_CNTRL	Señal para activar el div. resistivo activable

Tabla 2.1: Pines del microcontrolador elegidos.

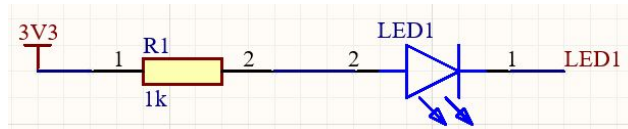


Figura 2.8: LED y su resistencia limitadora.

El diagrama esquemático completo asociado al microcontrolador se adjunta en el Anexo C, en la Figura C.3.

2.5.2. Componentes adicionales y configuración del AEM10941

El AEM10941 necesita de ciertos componentes pasivos y configurar algunas opciones para operar.

En su hoja de datos [29] se definen ciertos voltajes importantes para el funcionamiento

del PMIC. Se referirá como BATT al elemento almacenador de energía, pues se conecta en el pin con ese nombre.

- Vovch: (*Overcharge*) Voltaje máximo de BATT antes de apagar el convertidor Boost.
- Vchrdy: (*Charge ready*) Voltaje mínimo en BATT para encender los LDOs después de un arranque frío.
- Vovdis: (*Overdischarge*) Voltaje en el que se considera que BATT está descargada y no se seguirá extrayendo energía.
- Vhv: (*High voltage*) Voltaje del LDO de *alto* voltaje.
- Vlv: (*Low voltage*) Voltaje del LDO de bajo voltaje.

2.5.2.a. Configuración de batería y LDOs

El AEM10941 cuenta con dos salidas de voltaje que provienen de dos LDOs independientes. El de bajo voltaje es capaz de proporcionar un voltaje de entre 1,2 a 1,8 V con una corriente máxima de 20 mA. El de *alto* voltaje puede entregar tensiones desde 1,8 hasta $V_{batt} - 0,3$ V a un máximo de 80 mA. En esta aplicación, solo es útil el de *alto* voltaje, por lo que solo este se mantendrá activo, mientras que el de bajo voltaje se desactivará. Para esto, se deben configurar los pines ENHV con un 1 lógico y el ENLV con un 0 lógico.

En la Tabla 7 del *datasheet* de este PMIC se muestran las opciones para configurar los voltajes listados anteriormente. Esto se realiza mediante tres pines, CFG[2:0] en base a configuraciones predefinidas, aunque también se incluye un modo personalizado. Las configuraciones disponibles, relativas a supercondensadores se muestran en la Tabla 2.2. Se omitió la columna correspondiente a la salida del LDO de bajo voltaje (Vlv), ya que no se usará. La opción elegida es la última fila de la Tabla. Notar que esta es la mejor opción para el requerimiento de tener una salida de 3,3 V, ya que, debido al funcionamiento del LDO de *alto* voltaje, este necesita que su voltaje de entrada esté 0,3 V sobre su salida. Con el modo personalizado no se hubiese podido conseguir un Vovdis menor. En el límite superior ocurre lo mismo, el voltaje máximo de este IC en el pin BATT es 4,5 V, lo mismo que se consigue con la configuración elegida.

CFG[2]	CFG[1]	CFG[0]	Vovch	Vchrdy	Vovdis	Vhv	Uso típico
1	0	0	2.7 V	2.3 V	2.2 V	1.8 V	Single-cell SCAP
0	1	1	4.5 V	3.67 V	2.80 V	2.5 V	Dual-cell SCAP
0	1	0	4.50 V	3.92 V	3.60 V	3.3 V	Dual-cell SCAP

Tabla 2.2: Configuración elegida para los pines CFG.

Definir el pin con un 1 lógico significa conectarlo al pin BUCK, mientras que el 0 lógico es tierra (GND).

2.5.2.b. Capacitancia de supercondensadores

Dadas las limitaciones interpuestas por el PMIC mencionadas con anterioridad, es necesario utilizar dos supercondensadores en serie, por lo que, cualquiera que se elija, verá

reducida su capacitancia a la mitad, obteniendo a cambio un voltaje máximo duplicado.

Se considerará un promedio de 6 horas de sol, por lo que los supercondensadores deben abastecer al sistema por las otras 18 horas restantes. Para estimar la capacitancia, se considera un consumo promedio de corriente de 2 mA a 3,3 V. Con lo cual se puede calcular la potencia:

$$P = V \cdot I = 3,3 \cdot 2 \cdot 10^{-3} = 6,6 \text{ [mW]} \quad (2.1)$$

A partir de la potencia, se calcula la energía necesaria E_n , considerando las 18 horas mencionadas:

$$E_n = 6,6 \cdot 10^{-3} \cdot 18 = 0,1188 \text{ [W} \cdot \text{h]} \rightarrow E_n = 0,1188 \cdot 3600 = 427,68 \text{ [J]} \quad (2.2)$$

La energía se convirtió de Watt-hora a Joules para comparar más sencillamente con la ecuación de energía de un condensador. Teniendo en cuenta las limitaciones en el voltaje mínimo y máximo que impone el AEM10941; 3,6 y 4,5 respectivamente, se tiene que la energía de los supercondensadores E_{SCAP} debe al menos igualar la energía necesaria E_n .

$$E_{SCAP} = \frac{1}{2}C(4,5)^2 - \frac{1}{2}C(3,6)^2 = 427,68 \text{ [J]} \quad (2.3)$$

En la ecuación anterior se consideró C como la capacitancia en serie, por lo que la capacitancia individual C_{ind} de cada SCAP está dada por $C_{ind} = 2 \cdot C$.

Despejando la capacitancia, se obtiene $C = 117,42 \text{ F}$, por lo tanto, cada supercondensador deber ser de:

$$C_{ind} = 234,84 \text{ [F]}. \quad (2.4)$$

2.5.2.c. Elección de supercondensadores

De las opciones de compra disponibles y para intentar cumplir el requerimiento de diseño, se eligió el supercondensador modelo TPLC-3R8/200MR16X25 del fabricante Tecate Group [37].

Este supercondensador es de 200 F y es de tecnología LIC, acrónimo de Lithium-ion Capacitor, lo que significa que es un híbrido entre una batería de iones de litio y un condensador, intentando reunir las ventajas de ambos. Esto le permite lograr una densidad de energía mayor y otras ventajas [38], pero a cambio, genera otros inconvenientes.

Problema con tecnología LIC

Durante las pruebas realizadas en Resultados Experimentales 5 se descubrió un error en la elección del supercondensador, debido a las limitaciones de su tecnología. Esto se explica en la sección referenciada.

Como alternativa al SCAP LIC elegido primeramente, se optó por utilizar unos de tecnología EDLC (Electric Double Layer Capacitor). Esta tecnología no tiene las limitaciones

de la primera, ya que su comportamiento es más similar a un condensador tradicional.

La desventaja es que los supercondensadores EDLC tienen menor densidad energética, por lo que, para lograr la misma capacitancia de 200 F se usaría un espacio considerablemente mayor.

Para lograr un tamaño físico similar al que tenía el primer modelo elegido, fue necesario reducir la capacitancia a 100 F. A pesar esta reducción, el nuevo SCAP, modelo TPL-100/22X45F [39], del mismo fabricante, es bastante más voluminoso que el anterior. La Figura 2.9 muestra la comparación directa entre ambos modelos adquiridos.

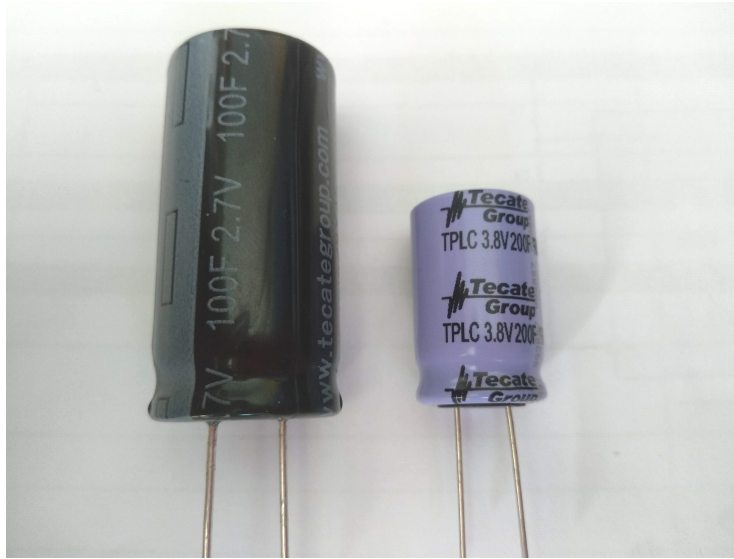


Figura 2.9: Comparación entre EDLC (izquierda) y LIC (derecha).

2.5.2.d. Panel solar

De acuerdo con la hoja de datos del AEM10941, soporta paneles solares de hasta 5 V, lo que equivale a un máximo de 7 celdas. Al analizar las opciones disponibles en tiendas como Mouser o Digikey, teniendo en cuenta la restricción mencionada, el tamaño y la energía que puede proveer, se eligió el modelo SM141K07TF de ANYSOLAR. La Tabla 2.3 muestra sus principales características eléctricas. El tamaño es de 50x23 mm.

2.5.2.e. Configuración MPPT

Con el panel solar elegido, se puede definir la configuración para el ratio de búsqueda del MPP. Para esto se dispone de dos pines, SELMPP[1:0] de manera análoga a lo realizado en la configuración de BATT. En la Tabla 2.4 se muestran las configuraciones disponibles. La elección se basa en el cociente entre V_{mpp} y V_{OC} (2.5) extraídos de la hoja de datos del panel solar, mismos que se presentaron en la Tabla 2.3.

Símbolo	Parámetro	Valor típico	Unidad
V_{OC}	Voltaje de circuito abierto	4,84	V
I_{SC}	Corriente de cortocircuito	58,6	mA
V_{mpp}	Voltaje en MPP	3,91	V
I_{mpp}	Corriente en MPP	55,1	mA
P_{mpp}	Peak máximo de potencia	215	mW
η	Eficiencia de la celda solar	25	%

Tabla 2.3: Características eléctricas del panel solar SM141K07TF.

$$\frac{V_{mpp}}{V_{oc}} = \frac{3,91}{4,84} = 0,8079 \quad (2.5)$$

SELMPP[1]	SELMPP[0]	V_{mpp}/V_{oc}
0	0	70 %
0	1	75 %
1	0	85 %
1	1	90 %

Tabla 2.4: Pines para configurar el ratio de búsqueda del MPP.

Como el resultado de la ecuación 2.5 es de 80,79 %, se opta por usar la tercera opción, es decir, SELMPP[1] = 1 y SELMPP[0] = 0, con lo que se tiene un ratio de búsqueda MPP del 85 %.

Al igual que en la configuración de BATT; definir el pin con un 1 lógico significa conectarlo al BUCK, mientras que el 0 lógico es GND.

2.5.2.f. Configuración de arranque en frío

El arranque en frío corresponde a la situación en la que BATT está descargada y no hay energía para ser *cosechada*. Para iniciar el arranque, se necesita un voltaje mínimo de 380 mV y una potencia de 3 μ W en la fuente (panel solar en este caso), correspondiente al pin SRC. Dicho voltaje mínimo se puede configurar para que sea más alto, por medio de un divisor resistivo. Sin embargo, se requiere justamente que el arranque del PMIC comience lo más pronto posible. Para esto, simplemente hay que conectar el pin FB_COLD directamente a SRC.

2.5.2.g. Configuración de batería primaria

El pin de batería primaria (PRIM) está ideado para conectar una batería convencional y usarla de respaldo en caso que los SCAP se descarguen y no haya energía para *cosechar* del sol.

En este caso, se pretende usar este pin para usarlo como alimentación alternativa a través de un conector USB. Para poder usar PRIM, es necesario definir el voltaje al cual el PMIC asumirá como que la batería primaria está descargada. Esto se realiza mediante un divisor

resistivo. El datasheet define la resistencia total R_P como $R_P = R_7 + R_8$. Para esta se recomiendan valores entre 100 k y 500 k Ω . Optando por este último se tiene:

$$R_P = 500 [k\Omega] \quad (2.6)$$

Aplicando la ecuación del datasheet, considerando $V_{prim_min} = 2,2$ V, se tiene

$$R_7 = \frac{\frac{1}{4}V_{prim_min}R_P}{2,2} \approx 120k [k\Omega]. \quad (2.7)$$

Con lo cual,

$$R_8 = R_P - R_7 = 380 [k\Omega]. \quad (2.8)$$

2.5.2.h. Protección para entrada de 5V

El pin PRIM del AEM soporta un voltaje máximo de 5 V. Puede darse el caso que el voltaje USB supere ligeramente su voltaje nominal de 5 V, por ejemplo, por defectos en la fuente de voltaje. Para evitar posibles problemas que esto pueda provocar, se propone un circuito simple compuesto principalmente por un diodo Zener.

El diodo elegido tiene un voltaje de ruptura en polarización inversa de 4,7 V y se incluye una resistencia pequeña para absorber la caída de voltaje. También se agrega un condensador para evitar que el pin quede flotando cuando no se esté usando, de acuerdo con las instrucciones de la hoja de datos.

Se espera evitar un sobrevoltaje máximo de 0,5 V.

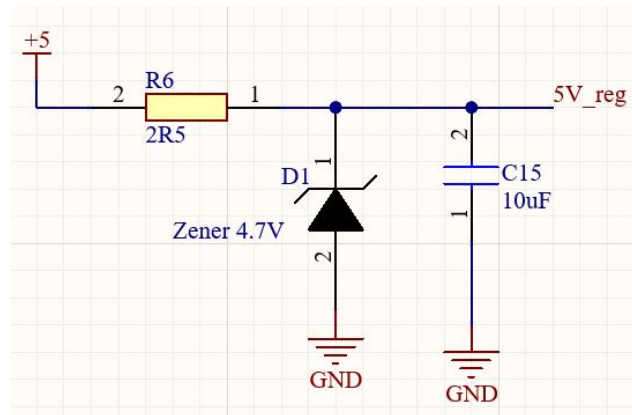


Figura 2.10: Protección 5V.

2.5.2.i. Componentes externos para el funcionamiento del AEM10941

Además de las configuraciones ya establecidas, este PMIC requiere de ciertos inductores y condensadores para funcionar. Los valores recomendados se detallan explícitamente en su hoja de datos y se resumen en la Tabla 2.5, diferenciando la denominación del fabricante y la propia.

El diagrama esquemático relacionado al AEM10941 se encuentra en la Figura C.4.

Designación		Valor	Unidad
AEM10941	Propia		
LBOOST	L1	10	μH
LBUCK	L2	10	μH
CSRC	C13	10	μF
CBUCK	C16	10	μF
CBOOST	C14	22	μF
CHV	C17	10	μF
CLV	-	-	-

Tabla 2.5: Componentes externos para el AEM10941

2.5.3. Divisor resistivo activable

El voltaje máximo al que el PMIC carga el arreglo de SCAP es 4.5V, lo cual es mayor al voltaje máximo que admite el conversor análogo digital (ADC) del microcontrolador, que a su vez está relacionado con el voltaje de alimentación (VDD), con la relación $VDD + 0,3 = 3,6$ V.

Por este motivo, es necesario añadir un divisor resistivo que pueda convertir los 4,5 V a 3,6 V o similar. Sin embargo, usar este arreglo directamente provocaría un constante flujo de corriente que drenaría la carga del SCAP. Para evitar este efecto, se diseña un divisor resistivo activable por medio de una de las salidas digitales del microcontrolador.

De esta manera, es posible activar el divisor de resistivo por un corto periodo de tiempo, suficiente para medir el voltaje del SCAP pero desperdiciando muy poca energía.

El diagrama esquemático se presenta en 2.11.

2.5.4. Resistencias pull-up I2C

Los tres sensores elegidos: SCD41, SGP40 y BME688, utilizan el bus I2C para comunicarse con el MCU. De acuerdo con el protocolo I2C [40], los pines correspondientes a SDA (Serial Data) y SCL (Serial Clock) requieren una resistencia de pull-up debido a su configuración colector abierto (open-collector).

Se escogieron resistencia de 10 k Ω para este propósito.

2.5.5. Reemplazo de DESPI-C02

El circuito adaptador para la pantalla de tinta electrónica se incluye directamente en la PCB, basándose en el diagrama esquemático del DESPI-C02 [41] y usando los valores para los componentes recomendados en la hoja de datos del driver de la pantalla, el SSD1681 [26]. Estas dos referencias tienen ciertas discrepancias entre sí, por lo que no fue tan sencillo llegar a un circuito final. Además, los modelos específicos de los componentes ahí mencionados, ya no se encuentran disponibles para comprar, por lo que fue necesario elegir componentes con valores cercanos y características similares.

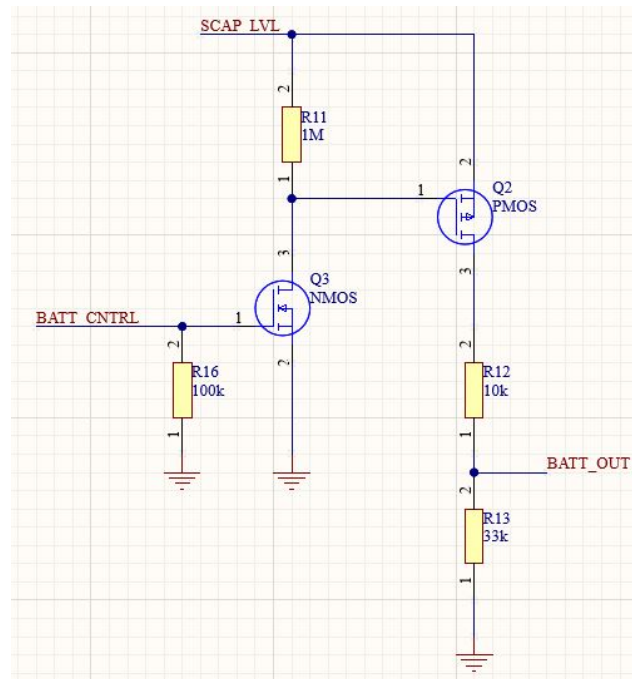


Figura 2.11: Divisor resistivo activable.

El diagrama esquemático para el reemplazo del DESPI-C02 y el conector FPC para la pantalla, se muestran en la Figura C.5.

2.5.6. Condensadores de desacople

Por recomendaciones de diseño electrónico, se agregaron condensadores de desacople (*decoupling capacitor*) en cada uno de los pines de alimentación de cada circuito integrado. El valor elegido para este propósito es de $0.1 \mu\text{F}$.

2.5.6.a. Condensadores para SCD41

Luego de realizar las Pruebas de integración, sección 4.3, se decidió agregar condensadores especiales para el sensor SCD41, para intentar reducir los efectos que producen los peaks de corriente en la fuente de alimentación.

Esto y el resto del diagrama esquemático asociado a los sensores se encuentra en la Figura C.6 del Anexo C.

2.6. Diseño de carcasa

El diseño de la carcasa se realiza posterior a la elección de los componentes principales y más grandes del circuito, para poder ubicarlos y reservar espacio en la PCB, motivo por el cual la carcasa se diseña primero.

2.6.1. Justificación del diseño

Este dispositivo está pensado para ser instalado en una pared o sobremesa, de ahí la necesidad de contar con una carcasa que sea capaz de adaptarse a ambos escenarios.

La razón principal del tamaño de la carcasa del dispositivo es su pantalla de tinta electrónica, ya que debe ser el elemento que más resalte. Por este motivo debe existir cierta armonía entre el tamaño de ambas para lograr un diseño agradable a la vista.

Dado que el dispositivo debe contar con un panel solar desde el cual obtiene su energía, es imperativo ubicar esta de manera estratégica para intentar conseguir el mejor ángulo de incidencia en la mayoría de las situaciones a las que pueda estar expuesto. Para esto, una buena opción es instalar el panel solar en un ángulo de 45°, de esta manera se logra una buena cobertura a fuentes lumínicas tanto desde la parte superior como de la parte frontal.

Para proteger el sensor, este será instalado dentro de la carcasa. Sin embargo, obviamente es necesario contar con ventilación para poder medir la temperatura, humedad y calidad del aire. Por este motivo se incorporan agujeros de ventilación en ambos laterales de la carcasa.

Este dispositivo intenta ahorrar energía en todo su diseño; una de las maneras de implementar esto es tomando mediciones cada cierta cantidad de minutos. Es posible que esta tasa de actualización de datos no sea suficiente para las necesidades del usuario y se puede dar la situación en que requiera una medición instantánea. Para cubrir este requerimiento, se incluye un botón en la parte frontal de la carcasa, justamente para generar una medición bajo demanda, en base a la solicitud del usuario.

2.6.2. Diseño en Fusion 360

Para diseñar el modelo 3D de la carcasa se utilizó el software Fusion 360 [42]. El modelo resultante se muestra en la figura 2.12.

2.7. Diseño de PCB

Principalmente, el diseño de la PCB está restringido por el diseño de la carcasa, la que a su vez, como se explica en la sección 2.6.1, está condicionada por el tamaño de la pantalla, del panel solar y por la manera en la que se pretende usar el dispositivo. A continuación se explican las consideraciones tomadas para poder cumplir con las distintas restricciones de espacio y ubicación.

2.7.1. Tamaño de los componentes

Para ahorrar espacio y poder ubicar los componentes importantes o más grandes con mayor facilidad es que se elije trabajar con componentes pequeños pero que aún sean sencillos de manipular. Por esto es que se decide que todas las resistencias y condensadores (a excepción de uno electrolítico) serán de montaje superficial (SMD) de tamaño 0603 (pulg.), equivalente a 1608 (mm), es decir, 1.6x0.8 mm.

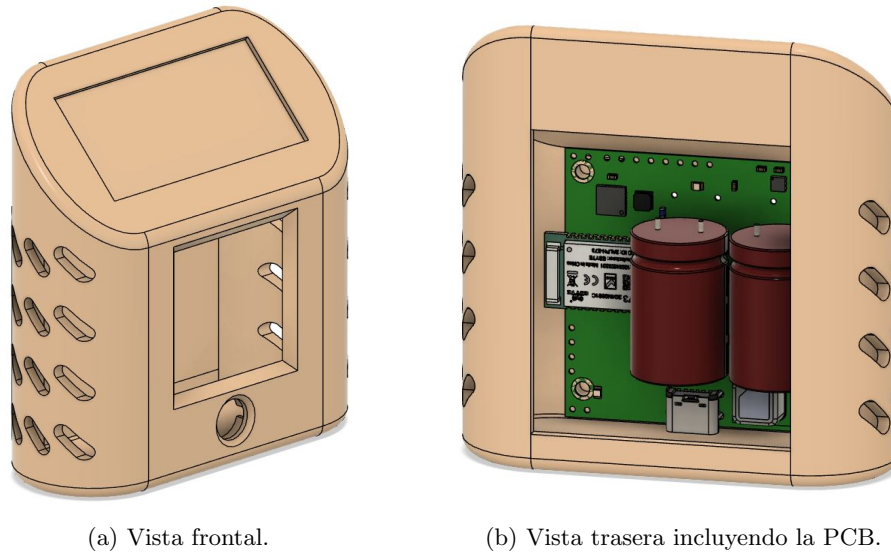


Figura 2.12: Modelo 3D de la carcasa en Fusion 360.

2.7.2. Ubicación de los componentes

A partir del modelo 3D de la carcasa, se generó un archivo de dibujo técnico (bidimensional) que indica la posición de los elementos inamovibles tales como el botón central, tornillos de montaje, pantalla y el tamaño máximo que puede tener la PCB para que pueda ser instalada sin problemas. Este archivo de dibujo se importó a Altium [43] y se usó como base para el diseño de la placa de circuito impreso.

- En el diseño de la carcasa se consideró que posicionar el conector USB en la parte inferior era más cómodo y agradable visualmente, por lo que su ubicación en la PCB está predeterminado.
- La posición de la pantalla y la corta longitud de su cable plano determinan directamente la ubicación del conector FPC en la PCB, a su costado derecho.
- En este tipo de módulos que incorporan una antena, se recomienda que esta quede alejada de metales, pistas u otras zonas con cobre de la PCB. La mejor manera de posicionar el módulo es dejando su antena fuera del resto de la placa. Tal como lo muestra el datasheet del MK02 [44], un módulo similar al que se está usando y tal como fue instalado por un desarrollador [45] que compartió su proceso de diseño en un blog.

A fin de cumplir esta recomendación, el módulo microcontrolador se ubicó en uno de los bordes libres de la placa, específicamente el izquierdo, evitando las esquinas para facilitar el ruteo de las pistas. Los bordes superior e inferior se descartan por la cercanía con el panel solar y por el conector USB, respectivamente.

- Siguiendo las reglas de diseño del sensor SCD41 [46], este se posicionó lo más lejos posible de posibles fuentes de calor como el microcontrolador o el PMIC

- Los SCAP fueron de los componentes más complicados de ubicar debido su gran tamaño en comparación a los demás. Una de las opciones consideradas fue instalarlos fuera de la PCB mediante cables, sin embargo, fue difícil implementar esta solución debido al poco espacio disponible en la carcasa y su dificultad para fijarlos a esta, además habría que considerar una PCB auxiliar que pudiese conectar ambos en serie. Finalmente, la solución adoptada fue aprovechar sus terminales para doblar los SCAP sobre otros componentes, de esta manera prácticamente no usan espacio en la PCB, como lo muestra la Figura 2.13.
- Para montar la placa en la carcasa, se consideraron 3 tornillos M2.5, los cuales se posicionaron cerca de las esquinas, en las posiciones que se definieron durante el diseño 3D de esta.

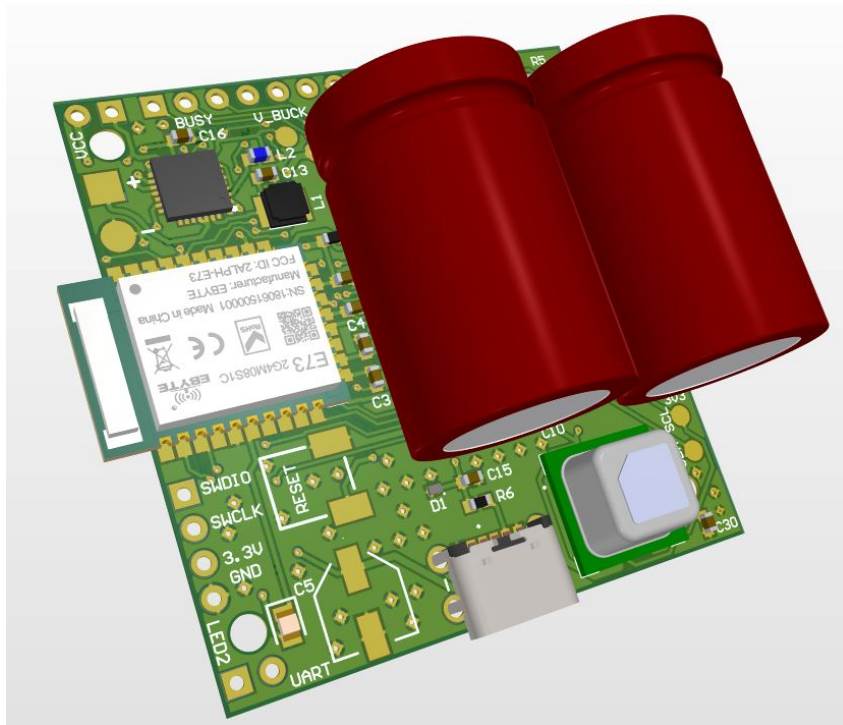


Figura 2.13: Render 3D de la PCB.

2.7.3. Conexiones de prueba

Dado que esta es la primera iteración de este dispositivo, se incluyeron pads en la PCB que permiten medir voltajes de interés (test-point) o realizar conexiones provisionarias, como se listan a continuación.

- Test-points para medir voltajes de interés.
- Pines para acceder al puerto UART.

- Conexión alternativa para DESPI-C02.
- Jumper para separar el Subsistema de alimentación del Subsistema funcional.
- Pines para programación mediante SWD.

2.7.4. Lista de materiales

La Tabla 2.6 lista todos los componentes utilizados para armar la PCB. Prestar atención a las Notas, pues explican modificaciones o aclaraciones importantes respecto a los componentes.

Notas

- (1) La PCB incluye un conector para DESPI-C02, que se usaría solo en caso que la implementación de este circuito en la placa no funcionara. Afortunadamente no fue necesario utilizarlo.
- (2) Como se explicó, finalmente se usaron los supercondensadores TPL-100/22X45F, de 100 F.
- (3) Fue necesario usar condensadores con voltaje máximo de 50 V para el circuito adaptador para la pantalla, se recomienda, por ejemplo, el Part Number UMK107AB7105KA-T.
- (4) Se usaron pin header macho genéricos para esto.
- (5) El footprint para estos botones se ideó en base a Kailih Silent, pero por temas de espacio, sobretodo para el boton central (PB2), se recomienda el Part Number TL3315NF100Q.
- (6) Para el conector SWD, simplemente se usó un pin header hembra genérico.

Cant.	Designación	Descripción	Valor	Part number
1	IC1	Módulo E73-2G4M08S1C	nRF52840	E73-2G4M08S1C
1	IC3	Sensor de gas BME688	No aplica	BME688
1	IC4	Sensor de CO2 SCD41	400 - 5000 ppm	SCD41-D-R2
1	IC6	Sensor de VOC SGP40	0 - 500 index	SGP40-D-R4
1	IC7	PMIC AEM10941	No aplica	AEM10941-QFN
1 ⁽¹⁾	IC8	HAT DESPI-C02	No aplica	DESPI-C02
1	No aplica	Pantalla tinta electrónica	200x200 px	GDEH0154D67
1	Y1	Oscilador de cristal SMD	32,768 kHz	ECS-.327-CDX-2216
2 ⁽²⁾	SCAP1, SCAP2	Supercondensadores LIC	200 F	TPLC-3R8/200MR16X25
1	D1	Diodo Zenner 0402	4,7 V	NZ8F4V7MX2WT5G
3	D2, D3, D4	Diodo Schottky SOD-123	30 V 0.5 A	SS0530_R1_00001
2	LED1, LED2	LED SMD 0805	Verde	APT2012LZGCK
1	PV1	Panel solar 7 celdas	215mW 4.84V	SM141K07TF
1	Q2	MOSFET canal P SOT-23	20 V	PMV75UP,215
2	Q1, Q3	MOSFET canal N SOT-23	30 V	PJA3432-AU_R1.000A1
1	L1	Inductor SMD WE-LQS	10 μ H	74404032100
1	L2	Inductor SMD 0603	10 μ H	MLZ1608E100MT
1	L3	Inductor SMD 1515	47 μ H	LQH44PH470MPRL
6	C1, C2, C6, C7, C10, C30	Capacitor cerámico 0603	0,1 uF 50V	C0603C104K5RAC3121
2	C3, C4	Capacitor cerámico 0603	12 pF 6,3V	C0603C120K9RAC7411
1	C5	Capacitor electrolítico	470 μ F 6,3V	EEE-FNJ471XUP
5	C8, C13, C15, C16, C17	Capacitor cerámico 0603	10 uF 10V	GRM188Z71A106KA73D
15 ⁽³⁾	C9, C11, C12, C18-C29	Capacitor cerámico 0603	1 uF 16V	EMK107B7105KA8T
1	C14	Capacitor cerámico 0603	22 μ F 10V	GRM187R61A226ME15D
2	R1, R2	Resistencia SMD 0603	1 k Ω	CR0603AFX-1001ELF
6	R3, R4, R9, R12, R14, R15	Resistencia SMD 0603	10 k Ω	SFR03EZPF1002
1	R8	Resistencia SMD 0603	380 k Ω	RT0603DRE07379KL
1	R7	Resistencia SMD 0603	120 k Ω	SFR03EZPF1203
3	R5, R6, R10	Resistencia SMD 0603	2,2 Ω	SDR03EZPF2R20
1	R11	Resistencia SMD 0603	1 M Ω	CRCW06031M00FKEAC
1	R13	Resistencia SMD 0603	33 k Ω	SFR03EZPF3302
1	R16	Resistencia SMD 0603	100 k Ω	SDR03EZPF1003
2 ⁽⁴⁾	UART1, VCC	Pin header genérico	2 pines	No aplica
2 ⁽⁵⁾	PB1, PB2	Botón push genérico	No aplica	No aplica
1	J1	Conector FPC	24 pin 0.5 pitch	FH34SRJ-24S-0.5SH(99)
1 ⁽⁶⁾	CONN1	Conector para SWD	4 pines	M20-7820446
1	CONN2	Conector USB tipo C	6 contactos	217175-0001

Tabla 2.6: Lista de materiales (BOM)

DESARROLLO DE FIRMWARE

Tras diseñar el hardware del dispositivo, resta programar su funcionalidad para que los distintos elementos se comuniquen entre sí, para lo cual se desarrolla el código o firmware que se montará en el microcontrolador.

3.1. Entorno de desarrollo

El desarrollo de firmware de este proyecto se construyó bajo el sistema operativo de tiempo real (RTOS) Zephyr [47]. Esto significó un reto importante por la inexperiencia y por tratarse de una tecnología relativamente nueva y menos conocida como Arduino, por ejemplo. Este RTOS está diseñado para ser usado en sistemas con recursos limitados y con énfasis en IoT [48].

Entre las razones para usar Zephyr se encuentra principalmente que el kit de desarrollo de software oficial de Nordic Semiconductor, nRF Connect SDK, integra Zephyr [49], de hecho, la mayoría de los ejemplos que ambos incluyen son los mismos. En ambos casos, utilizar herramientas recomendadas por el fabricante resulta ideal. Otra de las ventajas de Zephyr es que es de código abierto y multiplataforma, lo que disminuye la probabilidad de errores en el código fuente por ser de acceso público, a su vez esto ayuda a su masificación y que poder encontrar ayuda sea menos complicado en comparación a SDKs privativos. Que sea multiplataforma permite portar de manera más sencilla el código a otra arquitectura o fabricante.

Para usar Zephyr, este provee un entorno de desarrollo basado principalmente en línea de comandos, lo cual es poco amigable para el desarrollador. Nordic Semiconductor, por su parte, cuenta con nRF Connect for VS Code [50], que, como su nombre indica, corre sobre Visual Studio Code. Esta opción es bastante más amigable ya que incluye menús para configurar parámetros importantes. La desventaja es que está limitado exclusivamente a los productos de Nordic y solo se puede cargar el código con el programador J-Link. Si bien estas desventajas no eran problemáticas en este caso, se prefirió optar por PlatformIO, principalmente por estar más familiarizado con él.

PlatformIO [51] también corre sobre VS Code. Este IDE permite desarrollar para una

gran variedad de placas de desarrollo de distintas arquitecturas, con distintos frameworks y programadores. Por ejemplo, para este caso que se está usando el microcontrolador nRF52840, se puede desarrollar usando tres frameworks: Arduino, Mbed y Zephyr RTOS, siendo este último el de interés. Y se puede programar con seis protocolos, entre los que encuentran ST-Link, Black Magic Probe y J-Link [52].

3.1.1. Crear proyecto de PlatformIO

Como se explicó en la etapa de Diseño (sección 2.5.1), se usará como base la configuración de la placa de desarrollo oficial de Nordic Semiconductor, llamada nRF52840-DK. En la Figura 3.1 se muestra el menú para crear un nuevo proyecto en PlatformIO. Aquí es importante elegir la placa mencionada y el framework, Zephyr RTOS. También es posible definir la carpeta donde se almacenarán los archivos generados.

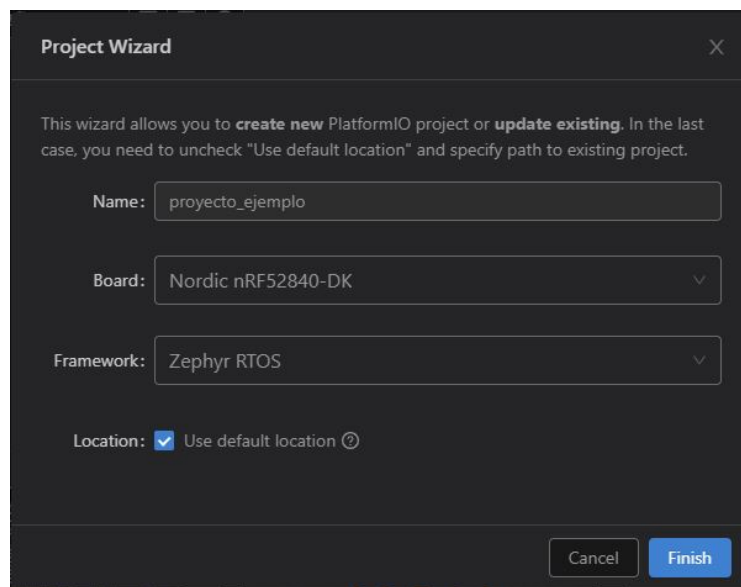


Figura 3.1: Creación de proyecto en PlatformIO.

Entre las carpetas importantes a tener en cuenta y que se generan automáticamente se encuentran:

- src: En ella deben estar los archivos de código fuente principales (.c).
- include: Aquí se agregan los archivos de encabezado (.h).
- lib: En ella se dejan las librerías que se usarán en el proyecto.
- zephyr: En esta se alojan los archivos importantes relacionados exclusivamente con Zephyr.
- .pio: Esta carpeta se genera al compilar el proyecto. Aquí se encuentran los archivos temporales o resultados de compilación.

Por último, el archivo `platformio.ini` es importante para definir ciertas configuraciones del entorno de desarrollo, como por ejemplo, el programador o la herramienta de depuración [53].

3.2. Zephyr y sus archivos de configuración

Zephyr está desarrollado principalmente por la Linux Foundation, por lo cual hereda algunos aspectos del kernel de Linux. Debido a esto, puede ser complicado de aprender en un comienzo. En primer lugar, utiliza archivos de tipo Device Tree para describir hardware y de esta manera independizar el código fuente de la arquitectura que se quiere controlar [54]. DT fue creado y es utilizado por Linux para conocer y controlar la máquina sin necesidad de describir detalles específicos de esta en el código fuente del sistema operativo [55].

En conjunto a los archivos Devicetree, Zephyr también utiliza archivos Kconfig [56], el cual también es utilizado en el kernel de Linux. Se utiliza para configurar el kernel y los subsistemas, así como activar o desactivar módulos de la aplicación específica. El kernel es la base de los sistemas operativos y se encarga principalmente de comunicar el hardware con las aplicaciones, por lo que tiene que estar cargado en memoria mientras el sistema está en funcionamiento [57].

A modo de ejemplo, en [58] se muestra el proceso para controlar el brillo un LED con PWM en una placa de desarrollo ESP32 usando Zephyr. Se puede ver lo engorroso que puede llegar a ser el trabajar con los archivos Devicetree y Kconfig, sobretodo en un comienzo. Luego de lograr los primeros pasos, estos se pueden “extrapolar” para conseguir controlar otros periféricos. Además, como se mencionó, estos archivos son principalmente para eso: controlar el hardware desde el firmware; una vez que se tiene acceso a estos desde el código, casi no es necesario modificar los archivos de configuración.

3.3. Bluetooth Low Energy (BLE)

Como se explicó en la sección 2.4.1, para la comunicación inalámbrica de este dispositivo se usará Bluetooth Low Energy. Para comunicarse con otros dispositivos, BLE utiliza Servicios y Características, los que están contenidos dentro del Generic ATtribute Profile (GATT) [59] [60]. Es importante tener en cuenta que un dispositivo periférico (como este caso) se puede conectar con un solo dispositivo central (como un celular). También existen otros métodos de comunicación sin conexión, como el modo Broadcaster.

El dispositivo periférico es el encargado de generar el servidor GATT e indicarle a la central qué Servicios y Características tiene disponible para leer, escribir y/o notificar.

Los Perfiles, Servicios y Características forman parte de las transacciones GATT y están jerarquizadas como lo muestra la Figura 3.2.

Los Perfiles están predefinidos en la especificación de Bluetooth [61], así como los Servicios y Características que pueden/deben incluir. Estos parámetros predefinidos usan identificadores UUID de 16 bits, mientras que los definidos por el desarrollador deben ser de 128 bits [62].

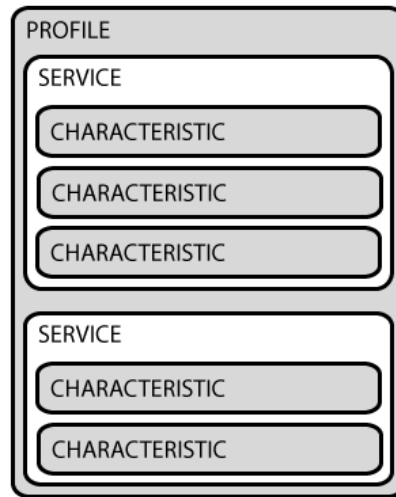


Figura 3.2: Jerarquía de los Perfiles, Servicios y Características de las transacciones GATT de BLE.

Para el desarrollo de firmware de este dispositivo se decidió usar principalmente el Perfil Environment Service Profile (ESP), puesto que existe un ejemplo en el repositorio de Zephyr que lo utiliza [63]. Dicho ejemplo simula sensores de temperatura y humedad y las expone a través de dos Características que también son parte del estándar Bluetooth. Para completar el comportamiento deseado, se añade una tercera Característica para alojar en ella la medición del sensor de CO₂. Como la especificación de Bluetooth no contempla Características predefinidas para esto, fue necesario definir una propia con UUID de 128 bits, la cual se obtuvo con un generador [64]. El comportamiento de este ejemplo se comprobó usando la aplicación nRF Connect [65]. El funcionamiento de la aplicación se explica en el capítulo siguiente.

Las modificaciones en el archivo Kconfig para que esta parte del código funcione, se basaron principalmente en el archivo `prj.conf` que se proporciona en el ejemplo referido. La totalidad de estas se enumeran a continuación:

```

1 CONFIG_BT=y
2 CONFIG_BT_PERIPHERAL=y
3 CONFIG_BT_SIGNING=y
4 CONFIG_BT_DIS=y
5 CONFIG_BT_DIS_PNP=n
6 CONFIG_BT_BAS=y
7 CONFIG_BT_DEBUG_LOG=y
8 CONFIG_BT_SMP=y
9 CONFIG_BT_DEVICE_NAME="Sensor CO2"
10 CONFIG_BT_DEVICE_NAME_DYNAMIC=y
11 CONFIG_BT_DEVICE_NAME_MAX=12

```

En general, las opciones de Kconfig están mal documentadas y es difícil hallar qué opción específica hay que activar para lograr el funcionamiento de algo. La mayoría de las veces se activaron opciones a raíz de investigar en foros de Internet. En otras, errores de compilación daban pistas de las opciones a activar, puesto que finalmente este archivo simplemente modifica macros, que típicamente activan o desactivan partes del código fuente con sentencias del preprocesador de C del tipo `#ifdef ... #endif`.

Una vez que se logró el comportamiento deseado con respecto al Bluetooth, queda implementar la lectura de los sensores. De esta manera, en lugar de enviar datos simulados por BLE, se enviarán datos reales.

Al ejemplo se le agregó también uno de los Servicios más comunes en los dispositivos BLE: el Battery Service (BAS), el cual expone una Característica para llevar la cuenta del porcentaje restante de batería, en rangos de 0 a 100.

3.4. Lectura de sensor SCD41

Como se ha mencionado, el principal sensor de este dispositivo es el SCD41. Afortunadamente, el fabricante proporciona código en C para comunicarse al sensor mediante I2C con funciones de bajo nivel genéricas, las cuales se tienen que modificar para compatibilizarlas con la plataforma específica [66]. Convenientemente, también proporciona ejemplos de implementación para estas funciones de bajo nivel para distintas plataformas, entre las que se incluye Zephyr.

Los ejemplos proporcionados por Sensirion fueron de gran utilidad para entender cómo se vincula el hardware descrito en el archivo Devicetree con el código en C.

3.5. Configuración y manipulación de pines

La elección de los pines se explicó en el capítulo de Diseño 2.5.1. Dado que esto se basa principalmente en la asignación de pines original del kit de desarrollo (DK) del microcontrolador, el nRF52840-DK, pero con ciertas diferencias, se creará un archivo overlay.

Un archivo overlay es parte de los archivos que usa Devicetree. Su trabajo básicamente es sobrescribir o añadir definiciones a los archivos DT base (.dts). La idea es modificar solo lo necesario y mantener las definiciones originales si es que son útiles.

Para añadir un archivo overlay al proyecto, primero se debe crear dentro de la carpeta `zephyr`, en este caso se llamará `nrf52840.overlay`. Luego, para que tenga efecto, es necesario enunciarlo en el archivo `CmakeLists.txt` que se creó automáticamente al iniciar el proyecto, en la misma carpeta mencionada anteriormente. Esto se realiza con la siguiente sentencia y es importante que esté ubicada en la segunda línea.

```
2 set(DTC_OVERLAY_FILE "nrf52840.overlay")
```

Siguiendo la definición de puertos propuesta en la etapa de diseño, el archivo overlay debe contener lo siguiente:

```

1 &i2c0 {
2     sda-pin = <41>; //P1.09 -> 1x32 + 9 = 41
3     scl-pin = <43>; //P1.11 -> 1x32 + 11 = 43
4 };
5
6 / {
7     aliases {
8         respin = "/epd_out/res";
9         cspin = "/epd_out/cs";
10        dcpin = "/epd_out/dc";
11        busypin = "/epd_in/busy";
12        btncentral = &button1;
13        battcntrl = &led3;
14    };
15    epd_out {
16        compatible = "gpio-leds";
17        res {
18            gpios = <&gpio0 17 GPIO_ACTIVE_HIGH>;
19            label = "EDP_RES";
20        };
21        cs {
22            gpios = <&gpio1 4 GPIO_ACTIVE_HIGH>;
23            label = "EDP_CS";
24        };
25        dc {
26            gpios = <&gpio0 22 GPIO_ACTIVE_HIGH>;
27            label = "EDP_DC";
28        };
29    };
30
31    epd_in {
32        compatible = "gpio-keys";
33        busy {
34            gpios = <&gpio1 10 GPIO_ACTIVE_HIGH>;
35            label = "EDP_BUSY";
36        };
37    };
38
39    buttons {
40        compatible = "gpio-keys";
41        button1: button_1 {
42            gpios = < &gpio0 12 (GPIO_ACTIVE_HIGH |
43                ↪ GPIO_PULL_DOWN) >;
44            label = "boton_central";
45        };
46    };

```

```

47     leds {
48         compatible = "gpio-leds";
49         led2: led_2 {
50             gpios = < &gpio1 13 GPIO_ACTIVE_LOW >;
51             label = "Green LED 2";
52         };
53         led3: led_3 {
54             gpios = < &gpio1 6 GPIO_ACTIVE_HIGH >;
55             label = "BATT_CNTRL";
56         };
57     };
58 };

```

Estas definiciones modifican los pines asignados al i2c0, que corresponde a uno de los dos periféricos I2C que dispone el nRF52840. En el caso del periférico SPI, se mantuvieron los pines original del spi1, ya que coincidía con los pines disponibles en el E73-2G4M08S1C.

Los pines para la pantalla e-paper se agruparon en epd_in y epd_out, para los pines de entrada y salida. Notar que para estas se usó la etiqueta `compatible` con `gpio-keys` y `gpio-leds`. Estas etiquetas están ideadas para controlar botones y LEDs, pero dado que en la práctica no hay diferencia, se usaron de manera genérica para definir entradas y salidas digitales, respectivamente.

Para que los cambios realizados tengan efecto y no hayan errores de compilación, también es necesario agregar las siguientes líneas al archivo Kconfig.

```

1 CONFIG_GPIO=y
2 CONFIG_I2C=y
3 CONFIG_SPI=y

```

Cada vez que se modifica el archivo overlay o Kconfig (`prj.conf`), es necesario hacer una compilación limpia, es decir, primero usar el botón Clean y luego Build/Upload en PlatformIO. El archivo Devicetree que se genera a partir de los archivos base y el overlay, se encuentra en `.pio/build/nrf52840_dk/zephyr/zephyr.dts`. Revisarlo es útil para verificar que los cambios se realizaron exitosamente y para revisar cómo está configurado el resto de hardware.

Ahora solo queda vincular esta descripción de hardware con el código en C. Por ejemplo para el LED2, esto se haría con la siguientes líneas de código:

```

1 #define LED2_NODE DT_ALIAS(led2) //P1.13
2 static const struct gpio_dt_spec led2 = GPIO_DT_SPEC_GET(
   ↪ LED2_NODE, gpios);

```

Luego, para inicializar los puertos:

```
gpio_pin_configure_dt(&led2, GPIO_OUTPUT_ACTIVE);
```

Finalmente, para encender el LED:

```
gpio_pin_set_dt(&led2, 1); //0 para apagar
```

3.6. Pantalla e-paper

En un comienzo se intentó usar o portar librerías especializadas en desplegar gráficos en pantalla, tales como U8glib [67], GxEPD2 [68], lvgl [69] o la desarrollada por HelTec Automation [70]. Sin embargo, portar las librerías, que, a excepción de lvgl, están todas desarrolladas para Arduino, es demasiado laborioso. Además de lo anterior, en caso de poder portarlas a Zephyr, el resultado habría sido una librería muy pesada con muchas funcionalidades inutilizadas, debido a la naturaleza propia de estas, sus dependencias y la definición de los caracteres de distintas fuentes y tamaños. Por otro lado, a pesar de que proveen una gran cantidad de funcionalidades para posicionar los elementos gráficos en pantalla, a menudo resulta engorroso lograr exactamente lo que se desea. En resumen, debido al gran esfuerzo necesario y al resultado poco satisfactorio, se descartó esta opción.

Por otro lado, el fabricante de la pantalla, Good Display, proporciona ejemplos para probar el funcionamiento de esta. Uno de los ejemplos se puede encontrar en [71], el cual está desarrollado para la plataforma STM32.

El ejemplo mencionado se utilizó como base para portarlo a Zephyr. Al tratarse de un código mucho más simple y autocontenido, se obtuvo un resultado rápido y exitoso. Se deja a libre disposición el resultado de esta conversión en un repositorio de GitHub [72].

Lo anterior fue sumamente útil para entender el funcionamiento de la pantalla y la manera en que se despliegan las imágenes. Básicamente, el ejemplo se basa en una serie de arreglos de 8 bits, donde cada bit representa el color de un pixel. Cada uno de los arreglos corresponde a una imagen, que en su mayoría son del tamaño máximo, 200x200 px, y se intercambian mediante actualización completa. También existen imágenes más pequeñas que corresponden a números, los cuales se usan para ejemplificar la visualización de un contador mediante actualización parcial.

A partir de lo anterior, se optó por seguir modificando el ejemplo para evaluar la utilización del mismo método para lo que se desea conseguir finalmente.

Finalmente, se llegó a la conclusión que lo mejor sería contar una imagen principal o base, que se cargue al inicio mediante actualización completa y luego, con actualización parcial añadir los números o datos adicionales en forma de imágenes más pequeñas, como se muestra en la Figura 3.3. El proceso para generar los elementos gráficos y convertirlos a arreglos se explica detalladamente en el Apéndice A.3.

Notar que bajo la medición de CO₂ de la Figura 3.3b aparece el mensaje “malo”. Esto corresponde a un mensaje que califica cualitativamente la calidad del aire en base al valor actual de la concentración de CO₂. Los rangos para estos valores se obtuvieron principalmente de [73], aunque que existen normativas más estrictas, como la del Ministerio de Salud en el contexto de la pandemia de COVID-19 [74]. En la Tabla 3.1 se muestran los rangos utilizados en este caso. Lo mínimo que puede medir el sensor SCD41 es 400 ppm, que es la concentración real típica que se tiene al aire libre. El límite superior de este sensor es 5000 ppm.

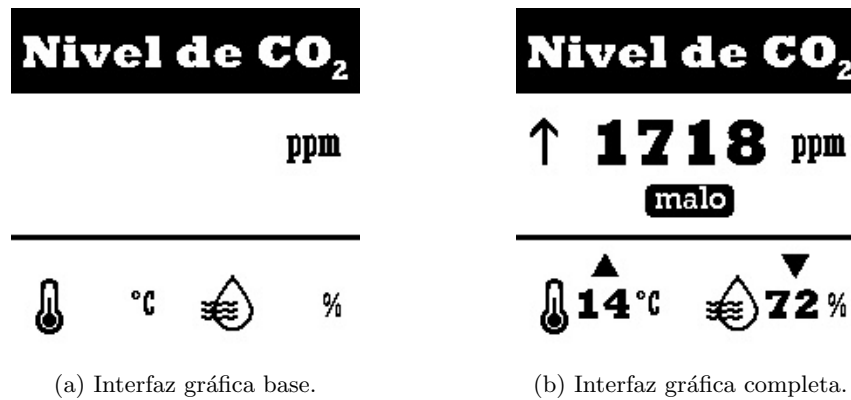


Figura 3.3: Elementos gráficos que forman la interfaz para mostrar la información.

Rango [ppm]	Calificación	Elemento gráfico
400 a 699	Excelente	excelente
700 a 999	Bueno	bueno
1000 a 1499	Mediocre	mediocre
1500 a 1999	Malo	malo
2000 a 2499	Peligroso	peligroso
mayor a 2500	Muy peligroso	muy peligroso

Tabla 3.1: Rangos para calificar cualitativamente la concentración de CO₂.

3.7. Conversor análogo-digital

Dado que este dispositivo, idealmente, debe funcionar con energía de los supercondensadores, es importante conocer el porcentaje de carga de estos. Para lograrlo, se usará el divisor resistivo activable descrito en la etapa de Diseño.

El primer paso es inicializar el pin activador (`batt_cntrl`) y el periférico ADC, de manera análoga a lo explicado en la sección PINES. Por ejemplo, de la siguiente manera:

```

1 static const struct gpio_dt_spec batt_cntrl = GPIO_DT_SPEC_GET(
  ↪ CNTRL_NODE, gpios);
2 static const struct device *adc_dev = DEVICE_DT_GET(ADC_NODE);

```

Luego, se definen valores constantes y ciertas estructuras para configurar el ADC. Nota: aquí, para las resistencias del divisor resistivo se usaron dos resistencias iguales, de 33 kΩ, por lo cual el pin de medición recibe la mitad del voltaje de los SCAP.

```

1 #define ADC_GAIN ADC_GAIN_1_6
2 #define ADC_REFERENCE ADC_REF_INTERNAL
3 #define ADC_ACQUISITION_TIME ADC_ACQ_TIME(
  ↪ ADC_ACQ_TIME_MICROSECONDS, 40)

```

```

4 #define ADC_RESOLUTION 10
5 #define MAX_VOLTAGE_RANGE 3.6 //ADC_REFERENCE / ADC_GAIN
6 #define DIV_RESISTIVO 2 //((33k/(33k+33k))^-1
7
8 static const struct adc_channel_cfg m_1st_channel_cfg = {
9     .gain = ADC_GAIN,
10    .reference = ADC_REFERENCE,
11    .acquisition_time = ADC_ACQUISITION_TIME,
12    .channel_id = 1,
13    .input_positive = 2, //NRF_SAADC_INPUT_AIN1 = 2
14 };
15
16 static int16_t m_sample_buffer[6];
17
18 const struct adc_sequence sequence = {
19     .channels = BIT(1),
20     .buffer = m_sample_buffer,
21     .buffer_size = sizeof(m_sample_buffer),
22     .resolution = ADC_RESOLUTION,
23 };

```

La función que realiza la medición se define de la siguiente manera. Primero activa el divisor resistivo, lee el ADC, convierte el valor bruto a milivolt, calcula el porcentaje, actualiza la característica BLE con el porcentaje, imprime este por UART y finalmente desactiva el divisor.

```

1 void medir_Vbatt()
2 {
3     gpio_pin_set_dt(&batt_cntrl, 1);
4     adc_read(adc_dev, &sequence);
5     int16_t adc_raw = m_sample_buffer[0];
6     int16_t batt_volt = adc_raw * (MAX_VOLTAGE_RANGE / 1023) *
7         ↪ DIV_RESISTIVO * 1000;
8     uint8_t porcentaje = (batt_volt - 3600) / 9;
9     //4.5 - 3.6 = 0.9 = 900 mV      (100/900 = 1/9)
10    if(porcentaje < 0){
11        porcentaje = 0;
12    }else if(porcentaje >100){
13        porcentaje = 100;
14    }
15    bt_bas_set_battery_level(porcentaje);
16    printk("%d\n", batt_volt);
17    gpio_pin_set_dt(&batt_cntrl, 0);
18 }

```

Esta función se ejecuta periódicamente con una interrupción por temporizador, configurada de la siguiente manera.

```

1 static struct k_timer timer_adc;
2

```

```

3 void timer_ADC_callback(struct k_timer *timer_id)
4 {
5     medir_Vbatt();
6 }

```

El temporizador se debe iniciar llamando a las siguientes funciones:

```

1 k_timer_init(&timer_adc, timer_ADC_callback, NULL);
2 k_timer_start(&timer_adc, K_SECONDS(10), K_SECONDS(10));

```

Para que todo esto funcione correctamente, es necesario añadir las siguientes configuraciones al archivo Kconfig (prj.conf).

```

1 CONFIG_ADC=y
2 CONFIG_ADC_ASYNC=y

```

3.8. Interrupción del botón central

En la etapa de Diseño se consideró incorporar un botón central (PB2). Para conocer su estado se prefirió usar interrupciones de hardware en lugar de *polling*, para liberar al MCU de estar consultando periódicamente, lo cual también sería un gasto innecesario de energía.

Para definir la interrupción por hardware se usaron las siguientes líneas de código, además de las definiciones del archivo Devicetree mencionadas en la sección 3.5.

```

1 static const struct gpio_dt_spec btn_central = GPIO_DT_SPEC_GET
   ↪ (BTN_CENTRAL_NODE, gpios);
2
3 static struct gpio_callback btn_central_cb;
4
5 static void button_pressed(const struct device *dev, struct
   ↪ gpio_callback *cb, uint32_t pins)
6 {
7     printk("El boton central se ha presionado\n");
8 }

```

3.9. Sensor SGP40

Al igual que con el sensor SCD41, para el SGP40 el fabricante Sensirion también proporciona el código fuente para controlarlo [75]. De hecho, las funciones de bajo nivel son las mismas, por lo que estas se pueden reutilizar.

La salida directa del sensor es un valor que por sí mismo no tiene sentido. Para tener datos útiles, es necesario procesar los datos directos con un algoritmo que proporciona el fabricante: Sensirion VOC index [76]. Esto se explica con más detalle en el capítulo de Pruebas Experimentales.

El SGP40 no incluye sensor de temperatura y humedad, sin embargo, necesita que se le entreguen mediciones de estas magnitudes para utilizarlas como referencia. Desde luego, se

le pueden definir valores fijos para este propósito, pero lo ideal es entregárselo para mejorar la calidad de la medición. En este caso, se le proporcionan mediciones de temperatura y humedad que se obtienen desde el SCD41.

3.10. Sensor BME688

Este sensor es similar al anterior en el sentido en que también hay que postprocesar los datos que entrega. Bosch proporciona una API [77] para controlar el BME688 desde un MCU. Mismamente, incluye funciones de bajo nivel que deben ser portadas a la plataforma específica.

Lamentablemente, no se tuvieron resultados exitosos con respecto a dicha API. A pesar de poder comunicarse con el sensor, no se lograron obtener datos válidos. Al probar el ejemplo de self-test se obtuvo siempre el error `No new data found`.

En caso de haber tenido éxito en la etapa anterior, el flujo de trabajo habría continuado con la implementación del algoritmo que postprocesa los datos, que en este caso se llama BSEC2 [78]. Este cuanta con múltiples “salidas virtuales”, entre las que se encuentran temperatura, humedad relativa, CO2 equivalente, VOCs y presión atmosférica.

PRUEBAS DE INTEGRACIÓN

Se realizaron pruebas preliminares para estudiar la integración y compatibilidad entre los componentes, con el fin de verificar que las elecciones fueron correctas.

Cronológicamente, estas pruebas se realizaron antes del diseño de la PCB y la carcasa, pero de manera simultánea a gran parte del Desarrollo de firmware. Se presenta en esta ubicación para seguir un orden lógico acorde.

En esta etapa se probaron el módulo E73-2G4M08S1C que tiene en su interior el SoC nRF52840, la pantalla de tinta electrónica GDEH0154D67 con su circuito adaptador DESPI-C02 y el sensor de CO₂ SCD41. Desafortunadamente, no fue posible probar el PMIC AEM10941, los supercondensadores ni el panel solar porque estos se adquirieron en etapas posteriores.

El objetivo de esta etapa es poder realizar mediciones de la concentración de CO₂ del ambiente, mostrar los datos por la pantalla y enviarlos por Bluetooth. Estas pruebas preliminares no incluyen el Subsistema de alimentación, pero sí contempla mediciones del consumo de energía para verificar o modificar parte del sistema en caso de ser necesario.

Los principales puntos a evaluar son:

- Desbloqueo y programación de ejemplos en el microcontrolador.
- Manejo de pines, periféricos, etc.
- Prueba del sensor de CO₂ con la librería provista por el fabricante.
- Pruebas básicas de la pantalla de tinta electrónica.
- Personalización de los elementos gráficos de la pantalla para mostrar los datos.
- Estudio del funcionamiento de Bluetooth y envío de datos del sensor.
- Pruebas de consumo de energía.

4.1. Programación del microcontrolador

Para comenzar las pruebas de integración, lo primero a realizar es programar el microcontrolador, de esta manera se podrá probar en conjunto con el sensor y la pantalla. Hay que tener en cuenta que el módulo que se adquirió (E73-2G4M08S1C) viene con protección contra escritura, es decir, no se puede modificar el firmware de fábrica directamente [79]. Para conocer más sobre los métodos de desbloqueo y programadores, revisar el Anexo A.2.

Una vez desbloqueado el módulo, se puede programar sin problemas. El entorno de desarrollo y parte del firmware se presentaron en el capítulo anterior.

4.2. Pruebas de integración

En la Figura 4.1 se muestra la configuración de los componentes en las primeras pruebas de integración realizadas. Se adquirió un módulo del sensor SCD41 ideal para pruebas de este tipo ya que cuenta con pin header para conectar directamente en una protoboard.

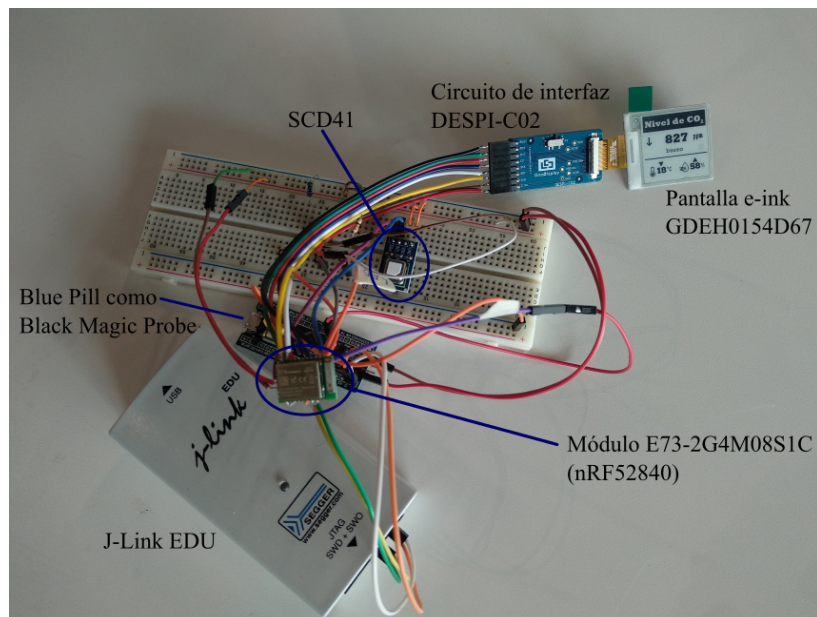


Figura 4.1: Protoboard para las pruebas preliminares de integración de componentes.

El módulo con el SoC nRF52840 está en formato SMD (Surface-mount Device), lo que significa que no se puede instalar en una protoboard, por lo cual fue necesario soldar cables directamente en sus pines para lograr conectarlo con el resto de componentes.

En la Figura, la placa Blue Pill se está utilizando como fuente de 3.3V a partir de los 5V de su puerto USB y como convertidor UART a USB, a raíz de lo explicado en el Anexo A.2. Con esto, es posible recibir mensajes desde el MCU al computador para debuggear fácilmente.

Para programar el MCU se utiliza el J-Link EDU. Este no tiene salida de 3.3V y necesita de una señal V_{Tref} de referencia para indicarle el voltaje al cual trabaja el microcontrolador que se quiere programar. Esto es parte de lo que se explica en el Anexo A.2.

Para que el módulo E73-2G4M08S1C funcione sin el reloj externo de 32.768 kHz, es necesario indicarle al microcontrolador que utilice el oscilador RC interno, mediante las siguientes líneas en el archivo KConfig (prj.conf).

```
1 CONFIG_CLOCK_CONTROL_NRF_K32SRC_RC=y
2 CONFIG_CLOCK_CONTROL_NRF_K32SRC_XTAL=n
```

4.3. Consumo de energía

Gracias a la herramienta JouleScope, que se presenta en el Anexo A.1, fue posible medir voltajes, corriente y potencia consumida por el circuito.

Al medir la corriente, se observó que una medición (simultánea) de temperatura, humedad y CO_2 del sensor SCD41 genera con un consumo como el de la Figura 4.2. Este cuenta con una duración de 1,22 segundos, con varios peaks de 12.5 ms con corrientes de hasta 136 mA, espaciados por un intervalo de la misma duración.

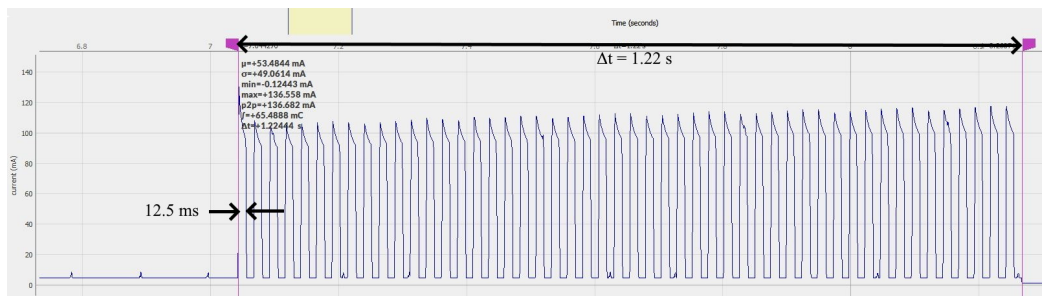


Figura 4.2: Perfil de corriente de una medición del sensor SCD41.

De las mediciones realizadas, se desprenden principalmente dos observaciones.

1. La corriente peak exigida por el sensor SCD41 durante una medición (136 mA) es mayor a la que el PMIC puede entregar (80 mA). Una de las propuestas para mitigar este problema es añadir un condensador de gran capacidad para que ayude a proporcionar la corriente que el sensor necesita. En caso de que esta solución no sea suficiente se podría intentar utilizando un sensor de menor consumo o cambiar el subsistema de alimentación, utilizando otro(s) PMIC. Finalmente, se optó por usar la primera opción. Esta es una de las razones por las que la PCB diseñada cuenta con dos sensores adicionales, el SGP40 y el BME688.
2. Durante la prueba de consumo de potencia del sistema con los componentes disponibles se realizaron mediciones cada 2,5 minutos. Al medir el consumo de potencia durante 5 minutos (dos mediciones), JouleScope detecta un consumo de 3 J. Estas pruebas se realizaron con los supercondensadores LIC de 200 F cada uno. Considerando que

la capacitancia se reduce a la mitad al usarlos en serie, se tiene una capacitancia equivalente de 100 F. Por otro lado, teniendo en cuenta que el rango de operación entre los que el PMIC puede trabajar con el SCAP es entre 4.5 y 3.6 V, se tiene que la energía que se puede aprovechar está dada por la siguiente ecuación.

$$E = \frac{1}{2}CV^2 \quad (4.1)$$

$$= \frac{1}{2}CV_{max}^2 - \frac{1}{2}CV_{min}^2 \quad (4.2)$$

$$= \frac{1}{2} \cdot 100 \cdot 4,5^2 - \frac{1}{2} \cdot 100 \cdot 3,6^2 = 364,5 \text{ J} \quad (4.3)$$

Calculando la autonomía que se podría lograr, se obtiene un aproximado de

$$364,5 \text{ J} \cdot \frac{5 \text{ min}}{3 \text{ J}} \cdot \frac{1 \text{ hora}}{60 \text{ min}} \approx 10 \text{ horas.} \quad (4.4)$$

Desde luego, 10 horas no es un tiempo suficiente para lograr una autonomía ininterrumpida, puesto que en la etapa de Diseño se estableció que esta debería ser de al menos 18 horas. A pesar de esto, se decide seguir adelante ya que por restricciones de espacio no se pueden usar supercondensadores de mayor capacitancia. Además, aún queda espacio de mejora para disminuir el consumo de energía por medio del firmware, por ejemplo, desactivando periféricos que no se utilicen.

4.4. Validación de paquetes Bluetooth

Para validar que el dispositivo está enviando los datos por Bluetooth correctamente, se utiliza la aplicación para Android nRF Connect [65], con la cual se pueden analizar las conexiones y los paquetes que comparten los dispositivos que se vinculan al celular que ejecuta la aplicación.

En la figura 4.3 se muestra una captura de pantalla de la aplicación nRF Connect corriendo en un celular con Android. En particular se observan los datos enviados por el sensor y recibidos por el celular. Estos datos están agrupados en Características: Temperature, Humidity y Unknown Characteristic. Esta última tiene ese nombre debido a que, a diferencia de la temperatura y humedad, la especificación de Bluetooth no incluye una Característica predefinida para el nivel de CO₂, por lo que fue necesario crear una personalizada con UUID de 128 bits.

Notar que las primeras dos Características tienen sus valores (Value) legibles, salvo que están escaladas por 10, es decir, los valores correctos son 17,954°C y 64,761%. En cambio, el valor de CO₂ se muestra codificado en hexadecimal, puesto que la aplicación desconoce cómo presentar su valor, por lo explicado anteriormente. Además, los bytes están invertidos, por lo que el valor 0x5603 en realidad corresponde a 0x0356, que en sistema decimal equivale a 854 ppm.

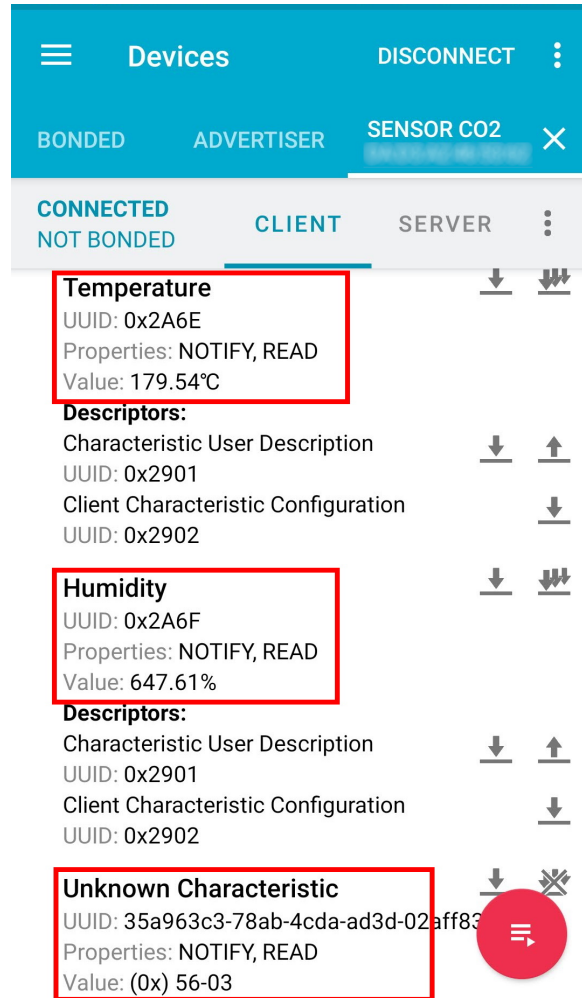


Figura 4.3: Captura de pantalla de la aplicación nRF Connect.

RESULTADOS EXPERIMENTALES

En este capítulo se presentan todas las pruebas y resultados obtenidos desde la recepción de los componentes y la PCB, por lo que se menciona desde las primeras impresiones de la placa hasta las pruebas del sistema completo, pasando por el proceso de soldado y ensamble.

5.1. Primeras impresiones

Al ver la PCB, lo primero que llama la atención es la alta densidad de pads, pistas y vias. Un problema derivado de esto es que algunas vias quedaron muy cerca de pads correspondientes a señales distintas. Además, las vias están fabricadas sin máscara de soldar, lo que podría provocar que la soldadura de alguno de los pads haga cortocircuito con una vía, provocando una conexión indeseada.

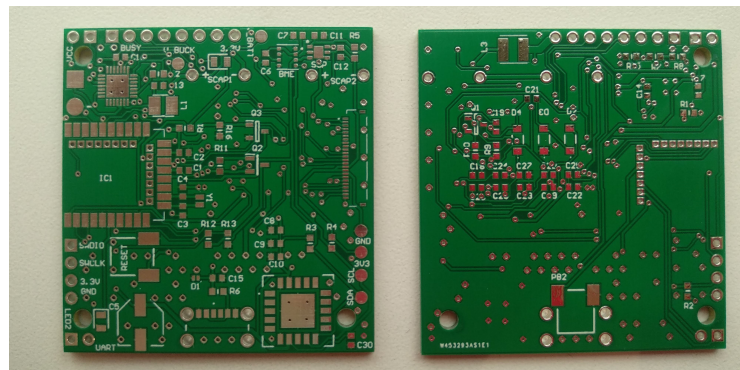
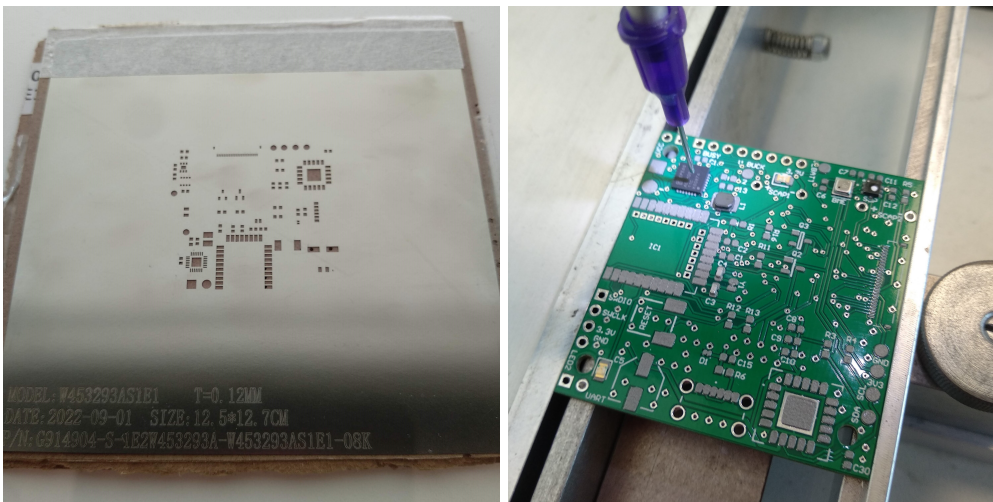


Figura 5.1: Placa de circuito impreso real, ambas caras.

5.2. Ensamble de la PCB

El ensamble de la placa de circuito impreso se realizó en dos etapas, una para cada cara. Para la cara superior o Top se cuenta con un stencil (Fig. 5.2a), con el cual se esparce pasta de soldar, depositando esta solo en los pads requeridos. Luego, se pobla con los componentes con la ayuda de una máquina Pick and Place (Fig 5.2b), la cual genera vacío para sujetar y posicionar cada componente. El uso de esta máquina fue de gran utilidad, ya que es mucho más precisa que usar pinzas, por ejemplo. Tras esto, la PCB se introduce en el horno para soldar, donde finalmente los componentes quedan fijos.

La cara inferior o Bottom se soldó a mano usando cautiñ, microscopio y pinzas. No se fabricó stencil para esta cara ya que lleva menos componentes y ninguno tiene pads tan pequeños como para dificultar demasiado el proceso.



(a) Stencil.

(b) Posicionamiento de componentes.

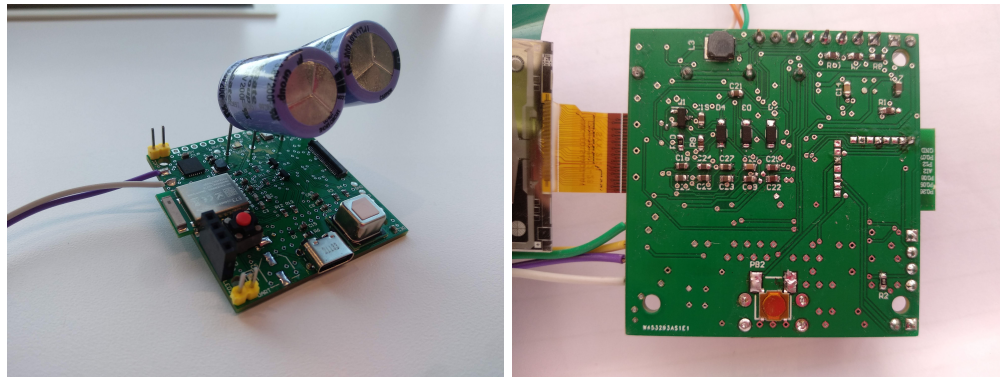
Figura 5.2: Proceso de soldado con stencil, máquina Pick and Place y horno.

La Figura 5.3 muestra el resultado del proceso de ensamble de la PCB. Se alcanza a notar que la soldadura de la cara inferior es menos prolija que la superior al tratarse de un proceso manual.

No se tuvieron grandes inconvenientes durante este proceso, salvo el conector FPC, que en su hoja de datos no indica la temperatura máxima que soporta. Por lo que, para evitar posibles problemas, se prefirió soldarlo a mano a mano en lugar de introducirlo al horno, con la dificultad de que sus pads son bastante pequeños.

Otra dificultad fue lograr un buen contacto de los pines inferiores del E73-2G4M08S1C con la PCB. Esto se produjo porque en la PCB, estos pads se fabricaron con vías, teniendo que soldarlos con cautiñ desde la cara inferior.

En total se armaron tres placas, la primera se armó completa pero se tuvieron problemas



(a) Cara superior.

(b) Cara inferior.

Figura 5.3: Placa de circuito impreso con los componentes soldados.

con el Subsistema de alimentación, la segunda se ensambló solo con los componentes correspondientes al subsistema dañado en la anterior y la tercera se construyó completamente, teniendo en cuenta y evitando los errores cometidos en las anteriores. Esto se explica con más detalle a continuación.

5.2.1. Ensamble y primer error

Durante el proceso de ensamble de la placa, se instalaron los dos supercondensadores LIC de 200 F que se tenían considerados inicialmente, pero sin haber medido previamente el voltaje de estos. El problema es que, este tipo de supercondensadores vienen cargados de fábrica a un voltaje aproximado de 3,5 V, lo cual no está informado como debería en su hoja de datos. Debido a que estos componentes se conectan en serie en la placa, el voltaje total, es decir, el que recibe el pin BATT del AEM10941 superó el voltaje máximo de 4,5 V, llegando a un valor cercano a 7 V.

Este error tuvo dos efectos importantes 1) el PMIC quedó totalmente inoperante, no pudiendo obtener ninguno de los resultados esperados y 2) sospechar que este tipo de supercondensadores es incompatible con esta aplicación junto al AEM10941, o al menos, dista mucho de ser la mejor opción. Esto acabaría confirmándose al construir la segunda placa.

5.2.2. Segundo ensamble

Luego del primer intento fallido, se armó una segunda placa, pero solo soldando los componentes relacionados al Subsistema de alimentación, el que se dañó en la primera placa.

Para evitar el error del ensamble anterior, primero se descargaron los supercondensadores conectándoles una resistencia de potencia en paralelo, previo a incorporarlos a la PCB. Hasta ese entonces se desconocían las limitaciones en el voltaje mínimo de los supercondensadores híbridos de litio (LIC), por lo que se descargaron indiscriminadamente. Con esto se descubrió un comportamiento inesperado, ya que, después de descargarlos a cerca de 0 V,

aumentaban su voltaje progresivamente al dejarlos reposar unos minutos. Este comportamiento fue la base para investigar y descubrir sus limitaciones.

Al notar que dichas limitaciones no eran compatibles con esta aplicación, se decidió dejar de utilizar los supercondensadores LIC, principalmente por los motivos que se listan a continuación:

- Voltaje máximo en serie muy alto: El voltaje máximo de los SCAP LIC es 3,8 V. Al conectar dos en serie se obtiene un máximo de 7,6 V, lo cual es un despropósito para el máximo que puede controlar el PMIC (4,5 V).
- Voltaje máximo individual: En caso de usarlos individualmente, el voltaje máximo es demasiado bajo para lograr 3,3 V a la salida, considerando que la salida de voltaje del PMIC trabaja hasta un voltaje mínimo a la entrada de 3,6 V. Además, desde el arranque en frío, el AEM10941 requiere al menos 3,9 V en su pin BATT, es decir, en esta situación nunca se activaría su salida. Esto solo tendría sentido en una situación que requiera 1,8 V a la salida en lugar de 3,3 V, pero ese no es el caso. Peor aún, esto no es válido por el siguiente punto.
- Voltaje mínimo: Finalmente, la mayor limitación que tiene la tecnología LIC es su voltaje mínimo. Tal como las baterías de litio, estos SCAP ven afectada su vida útil en caso que se descarguen demasiado. En particular, no se recomienda descargarlos a menos de 2,2 V. Considerando la configuración en serie, esto se traduce a 4,4 V, lo que dejaría un rango de solo 0,1 V hasta alcanzar los 4,5 V máximos del AEM10941.

Por las razones mencionadas, se optó por utilizar supercondensadores electroquímicos de doble capa (EDLC), los cuales tienen un comportamiento más similar a los condensadores tradicionales, por lo que es posible descargarlos hasta cero sin problemas. Específicamente, se eligió el modelo TPL-100/22X45F.

En síntesis, finalmente se armó esta segunda placa solo con los componentes del subsistema de alimentación, pero ahora usando los supercondensadores EDLC, evitando así el error cometido anteriormente. De esta manera fue posible obtener los resultados esperados, es decir, lograr que estos se carguen a partir de la energía del sol y obtener un voltaje de salida del AEM10941 de 3,3 V para alimentar el Subsistema funcional.

Con el funcionamiento básico ya comprobado, se procedió a intentar alimentar la placa anterior con esta. Recordar que los subsistemas funcional y de alimentación se pueden conectar y desconectar a voluntad por medio de un jumper, de esta manera, es posible alimentar la parte funcional de manera externa, con una fuente de alimentación o, como en este caso, usando una placa con el Subsistema de alimentación operativo.

Al conectar ambas placas, se observa que el sistema se enciende, pero inmediatamente presenta intermitencias, presumiblemente debido a lo previsto en la sección 4.3, donde se descubrieron los altos peaks de corriente que requiere el sensor SCD41. Esta hipótesis se comprobó rápidamente cargando un código que no hacía uso de este sensor. El nuevo código que se cargó hace uso de Bluetooth Low Energy (BLE) y la pantalla de tinta electrónica. En este caso el AEM10941 sí fue capaz de proporcionar el voltaje y la corriente necesaria para operar, por lo que se corrobora que el sensor produjo el problema de intermitencia.

Con esa parte parte comprobada, se procedió a probar el funcionamiento del conector USB. Se espera que el sistema continúe operando aún si los supercondensadores se descargan. Al intentar probar esto, inicialmente no se tuvieron resultados satisfactorios, por lo que se intentó cambiar el condensador CSRC (C13 en la designación propia) de $10\ \mu\text{F}$ a $150\ \mu\text{F}$, según recomienda el datasheet [29] en caso de utilizar batería auxiliar. Con este cambio tampoco se obtuvieron los resultados esperados.

Se probó también la protección para entrada de 5V, logrando que voltajes de entrada ligeramente superiores a 5 V se mantengan por debajo de este umbral, para proteger el pin PRIM del AEM10941. En particular, con pruebas en vacío, se logró reducir de 5,05 V a 4,89 V.

Lamentablemente, después de estas últimas pruebas, el PMIC dejó de funcionar por motivos desconocidos, pero muy probablemente debido a alguno de los siguientes motivos: 1) conexión momentánea accidental de nodos que no debían conectarse, durante el proceso de cambio del condensador CSRC o 2) dejar flotando el pin BATT. Hasta ese entonces se desconocía que, según el datasheet, dicho pin no puede quedar desconectado nunca, por lo que se recomienda dejar siempre un condensador de al menos $150\ \mu\text{F}$ en ese pin en caso de desconectar la batería. Esto último se tomó en consideración para construir la tercera PCB.

5.2.3. Tercera y última placa

Con todo lo aprendido de los errores anteriores, se procedió a construir una tercera y última placa. Con esta, al tener ambos subsistemas funcionales, fue más sencillo poder tomar mediciones más exhaustivas.

5.3. Pruebas del subsistema de alimentación

Se enfatizan las pruebas del Subsistema de alimentación ya que en el capítulo de Pruebas de integración no se contaba con ningún componente de este subsistema, por lo que no fue posible hacer pruebas en esa oportunidad.

5.3.1. Configuración de las pruebas

En la Figura 5.4 se muestra la PCB ensamblada con sus principales elementos enumerados. Estos se listan a continuación y se referenciarán más adelante como “puntos”.

1. Alimentación positiva del Subsistema funcional (3,3 V)
2. Salida positiva del LDO de alto voltaje del AEM10941 (3,3 V)
3. PMIC AEM10941
4. Alimentación positiva del subsistema funcional (3,3 V), parte del conector auxiliar del DESPI-C02
5. GND, parte del conector auxiliar del DESPI-C02
6. Cables para conectar los supercondensadores de manera externa

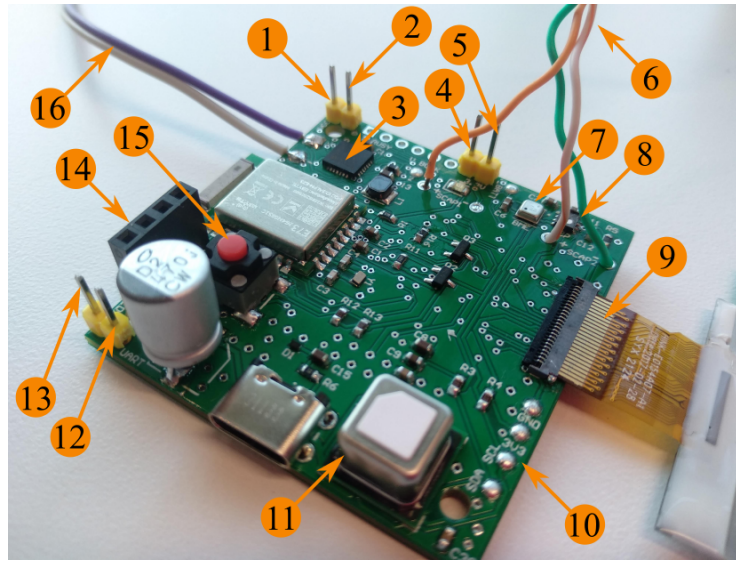


Figura 5.4: Curva de carga del arreglo de supercondensadores.

7. Sensor de gas BME688
8. Sensor de VOC SGP40
9. Conector FPC para la pantalla
10. Pads para debuggear bus I2C, con alimentación
11. Sensor de CO2 SCD41
12. UART RX
13. UART TX
14. Conector para programar mediante SWD
15. Botón de reinicio
16. Cable para conectar el panel solar de manera externa

Para las pruebas del subsistema de alimentación se trabajó en dos escenarios, dependiendo si (A) se quería extraer la energía desde los supercondensadores o (B) no, alimentando el sistema desde un computador. Se referirá a estos como “configuraciones”.

- (A) En este caso, se conecta un jumper entre los puntos 1 y 2, lo que provoca que ambos subsistemas queden unidos, es decir, el de alimentación proveyéndole energía al funcional, desde los supercondensadores y a través del LDO del AEM10941.

Para poder leer datos desde el puerto serial desde un computador sin que este alimente la PCB, solo se conecta GND, para poder tener una referencia de voltaje. Revisar Apéndice B.

las resistencias del divisor resistivo y con el pin del conversor análogo-digital (ADC).

Las pruebas consistieron principalmente en medir, con un voltímetro, el nodo central del divisor resistivo (BATT_OUT de la figura 2.11) y verificar que había voltaje solamente cuando se activaba el pin BATT_CNTRL, que corresponde a una salida del microcontrolador. El comportamiento observado fue el esperado, midiendo el voltaje correspondiente a la ecuación de un divisor resistivo típico.

Se observó que, cuando el voltaje de los supercondensadores era máximo, el voltaje de salida de este circuito era muy cercano al voltaje máximo admitido por el microcontrolador, por lo que se decidió cambiar una de las resistencias para así obtener un voltaje máximo menor. Finalmente, el valor para R12 y R13 fue 33 k Ω .

Verificar el correcto funcionamiento de este circuito fue sumamente útil para las mediciones posteriores ya que permitió medir el voltaje de los SCAP de manera automática con el microcontrolador y enviar estos datos por medio del puerto serial (UART) para almacenarlos y analizarlos con detenimiento, como se explica en el Apéndice B.

5.3.3. Pruebas de carga desde cero con energía solar

Para la primera prueba se usó un par de SCAP EDLC completamente nuevos, cuyos voltajes iniciales estaban en torno a 0.1 V. La prueba consistió en cargar este par de supercondensadores únicamente usando la energía del sol, gestionada por el PMIC. Estas mediciones se tomaron cada 15 minutos aproximadamente, directamente con un voltímetro (no con el ADC del microcontrolador). Se procuró que los rayos de sol llegaran directamente al panel fotovoltaico, moviendo el dispositivo para seguir la trayectoria del sol, pero sin cambiar el ángulo de inclinación, el cual se mantuvo siempre directamente sobre la superficie.

El gráfico obtenido se muestra en la Figura 5.6. Se aprecia que esta sigue una curva ligeramente exponencial, similar al de la carga de típica de condensador convencional con voltaje constante. Sin embargo, esta curva puede deberse al cambio de la inclinación del sol a lo largo del día. Se observa que las 4,5 horas que estuvo con luz solar directa no fueron suficientes para cargar los SCAP hasta el máximo permitido por el PMIC (4,5 V), llegando solo hasta 3,74 V. Notar además que esta prueba es bastante sintética y no representa una situación real, del total de horas que recibió sol directo esta ventana con orientación poniente, el dispositivo solo lo habría recibido por poco más de una hora si es que no se hubiese trasladado de manualmente siguiendo los rayos solares.

5.3.4. Pruebas de autonomía

Debido a que no fue posible utilizar el sensor principal (SCD41) en conjunto con el PMIC, se optó por usar otro de los sensores disponibles en la placa para llevar a cabo las pruebas de autonomía. Aquí lo importante es conocer de antemano el consumo promedio de corriente y medir cuánto tiempo el sistema es capaz de alimentarse por sí solo.

Como se explicó en el capítulo de Desarrollo de firmware, de los otros dos sensores disponibles, solo se logró obtener datos del SGP40. De todas formas, en esta prueba no es relevante el dato del sensor, sino el tiempo de autonomía.

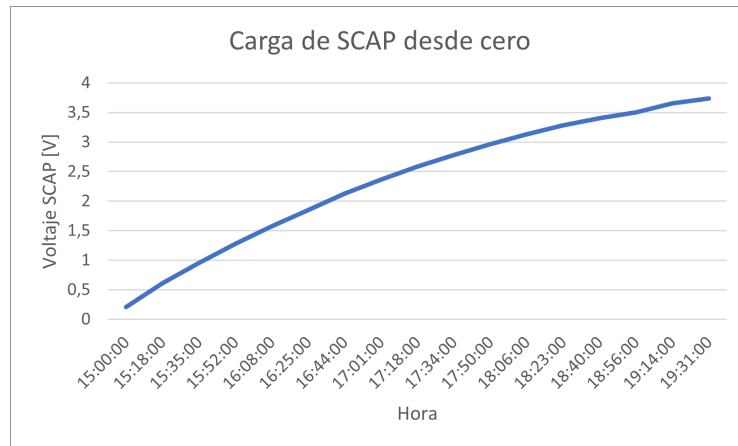


Figura 5.6: Curva de carga del arreglo de supercondensadores.

Se desarrolló un código que muestra la medición del sensor por la pantalla e-paper, a la vez que utiliza Bluetooth Low Energy. De este modo se logra un funcionamiento similar al que se esperaba originalmente.

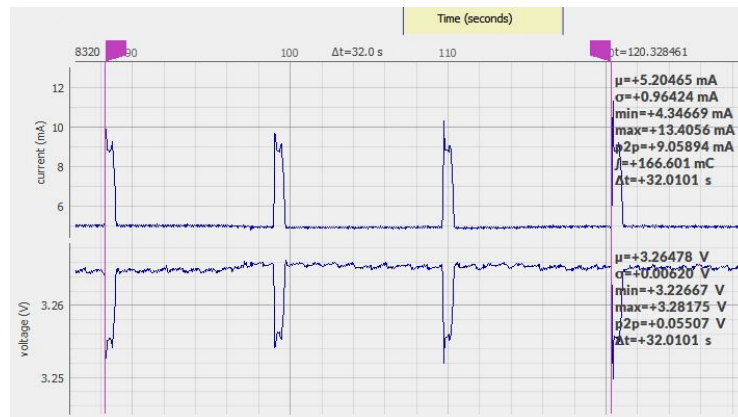


Figura 5.7: Mediciones del consumo de potencia usando JouleScope.

Se usó un JouleScope para medir el consumo de potencia y se obtuvieron los resultados de la Figura 5.7. A partir de las mediciones, se tiene que en estas condiciones el sistema consume un promedio de 5,2 mA. Además, otro punto interesante de notar es la estabilidad del voltaje, proveniente de la salida del PMIC AEM10941. Aquí se aprecia que con peaks de corriente de hasta 13,4 mA, el voltaje tiene una variación de 55 mV, es decir, un 1,68 %.

Lamentablemente, luego de tomar estas últimas mediciones, se dejó de tener acceso al dispositivo JouleScope, por lo que no se pudo contar con este para las mediciones posteriores.

Por este motivo y para tener un método alternativo para medir la potencia del sistema, posteriormente se agregó una rutina para enviar los datos por el puerto serial. Las pruebas consistieron en cargar los supercondensadores a un voltaje alto (4.2 V en este caso) y conectar el jumper para que el Subsistema de alimentación le provea energía al subsistema funcional. Las mediciones del ADC convertidas a milivolt se transmitieron por UART y se graficaron con el procedimiento explicado en el Apéndice B, con lo cual se obtuvo el gráfico de la Figura 5.8.



Figura 5.8: Descarga de supercondensadores alimentando el sistema a través del PMIC.

Del gráfico se observa que los supercondensadores se descargan de manera lineal, con una recta descrita por (5.1), con un coeficiente de determinación $R^2 = 0,9991$.

$$v(t) = -8,42 \cdot 10^{-5}t + 4,206 \quad (5.1)$$

Lo anterior se puede validar con la medición tomada con el JouleScope y con la ecuación (5.2), de un condensador tradicional.

$$i(t) = C \frac{dv}{dt} \quad (5.2)$$

Donde, si se considera una corriente de descarga constante $i(t) = I$ e integrando se obtiene

$$v(t) = \frac{1}{C} \int_0^t i(t)dt + v_0 = \frac{-I}{C}t + v_0, \quad (5.3)$$

que corresponde a una recta lineal con pendiente $m = I/C$. A partir de esto, se pueden comparar ambos métodos utilizados para validarlos mutuamente. La capacitancia se conoce de antemano, $C = 50$ F, debido a los dos supercondensadores de 100 F en serie.

$$m = -8,42 \cdot 10^{-5} = \frac{-I}{C} \quad v_0 = 4,2051 \quad (5.4)$$

De la ecuación anterior, se obtiene $|I| = 8,42 \cdot 10^{-5} \cdot 50 = 4,2$ mA. Esto significa una diferencia del 19,23 % en comparación a la corriente promedio de 5,2 mA que se obtuvo con

el JouleScope.

Extrapolando la recta de la ecuación 5.1, considerando la descarga desde el máximo, 4,5 V y comprobando esto mismo con mediciones empíricas realizadas posteriormente, se obtiene una autonomía promedio de 11000 segundos, es decir, 3 horas, 3 minutos y 20 segundos.

Con estas mediciones también es posible corroborar que los cálculos de la ecuación (2.3), correspondiente a la etapa de Diseño, fueron realizados de correctamente. Considerando el voltaje y la corriente obtenida del JouleScope, 3,26 V y 5,2 mA, la autonomía de 11000 segundos y ejecutando los mismos pasos, se tiene que la potencia promedio consumida es:

$$P = V \cdot I = 3,26 \cdot 5,2 \cdot 10^{-3} = 16,952 \text{ [mW]}. \quad (5.5)$$

Con lo cual, la energía que efectivamente entregan los supercondensadores a través del AEM10941 con estas condiciones es

$$E_{ef} = 16,952 \cdot 10^{-3} \cdot 11000 = 186,472 \text{ [J]}. \quad (5.6)$$

Utilizando este resultado para igualarlo a la ecuación de energía de los SCAP y despejando la capacitancia equivalente en serie desde

$$E_{SCAP} = \frac{1}{2}C(4,5)^2 - \frac{1}{2}C(3,6)^2 = 186,472 \text{ [J]}, \quad (5.7)$$

se obtiene $C = 51,1583$, lo que equivale a $C_{ind} = 102,317 \text{ F}$, muy similar a la capacitancia nominal de los supercondensadores EDLC que se están utilizando (100 F), por lo cual, se concluye que este método para calcular la autonomía y/o la capacitancia es válido.

Gracias a que el método está comprobado, se puede realizar el cálculo inverso para estimar que se necesita un consumo promedio menor a 0,68 mA para lograr una autonomía de 23 horas, dejando reservando solo una hora para cargar el sistema, lo que tendrá sentido más adelante.

5.3.5. Carga en vacío

Se sometió el sistema a pruebas de carga en vacío, es decir, sin conectar la salida, usando la configuración B.

El resultado de esta prueba se muestra en la Figura 5.9, donde se aprecia que, apenas se recibe sol directo, la pendiente de carga aumenta drásticamente. El tiempo que tardó en cargarse con sol directo desde $V_{ovdis} = 3,6 \text{ V}$ hasta $V_{ovch} = 4,5 \text{ V}$ fueron 2340 segundos, es decir, 39 minutos. Al repetir esta prueba en otra ocasión, se obtuvo un tiempo cercano, de 36 minutos.

5.3.6. Carga con la salida conectada

Esta prueba se realizó usando la configuración A.

Si bien el LDO del AEM10941 tiene salida con voltajes hasta un mínimo de 3,6 V, este no se activa mientras no se alcance el valor de $V_{chrdy} = 3,9 \text{ V}$. Es decir, cuando los supercondensadores están descargados a $V_{ovdis} = 3,6 \text{ V}$ o menor, y luego, el sistema comienza a



Figura 5.9: Carga de los supercondensadores sin conectar la salida.

recibir energía al panel solar, estos se cargan progresivamente. Pero mientras no se alcance V_{chrdy} , no hay voltaje de salida que alimente el Subsistema funcional y con ello, el MCU.

La Figura 5.10 muestra justamente esto. Esta prueba se realizó durante dos días seguidos, dejando el dispositivo en un lugar fijo. Como se está usando la configuración A, solo se registran datos cuando el MCU se alimenta, al activarse el LDO una vez superado $V_{chrdy} = 3,9$ V.



Figura 5.10: Carga de los supercondensadores con la salida conectada.

Se puede apreciar que el comportamiento es casi idéntico ambos días, con mediciones desde V_{chrdy} hasta 4,07 y 4,03 V, respectivamente. Notar que en ninguno de los dos días se alcanzó el máximo de $V_{ovch} = 4,5$ V ya que, a diferencia de la “Carga en vacío”, aquí se está cargando a la vez que funciona por sus propios medios. Además, no se “siguió” el rayo de sol, sino que se dejó en una posición fija.

5.3.7. Histéresis para evitar sobrevoltaje

Esta prueba se realizó usando la configuración A.

Cuando se alcanza el voltaje máximo en los supercondensadores, $V_{ovch} = 4,5 \text{ V}$, el AEM10941 entra en modo *overvoltage*, en el cual se deja de entregarle energía al elemento almacenador de energía para evitar que el voltaje siga subiendo. El proceso de carga se reanuda cuando el voltaje haya caído, según la hoja de datos, *algunos mV*, lo cual produce una histéresis que se puede ver en la Figura 5.11. De este modo, los SCAP se mantienen cargados pero evitando la sobrecarga.

De la Figura se observa que el voltaje debe caer hasta 4,46 V, es decir, un aproximado de 40 mV.

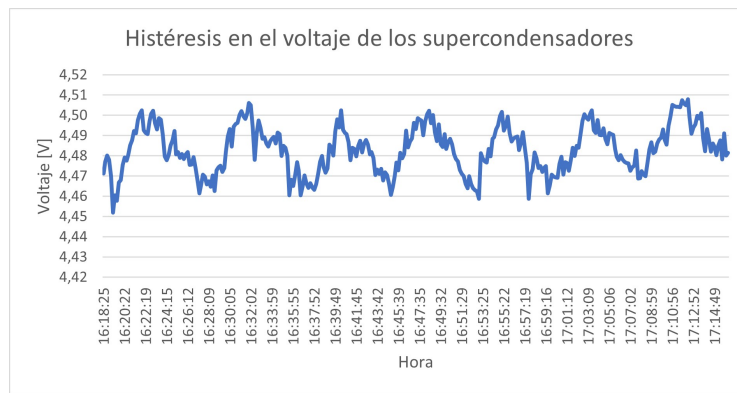


Figura 5.11: Histéresis cuando los supercondensadores están cargados y se sigue recibiendo luz solar.

5.3.8. Luz indirecta durante un día

Se dejó el dispositivo en una posición fija y usando la configuración B, durante toda la mañana y la tarde, aproximadamente de 6:50 a 17:30.

El resultado de esta medición se muestra en la Figura 5.12, donde se aprecia claramente la diferencia en la pendiente de carga antes y después de recibir sol directo.

En la Figura 5.13 se muestra solo la porción de carga con sol indirecto, con su línea de tendencia descrita por (5.8). Si se extrapola esta línea de tendencia, el sistema tardaría 995766 segundos (11 días, 12 horas y 36 minutos) en alcanzar el mismo voltaje (4,10 V) que se alcanzó con solo 1 hora y 8 minutos de minutos de sol directo.

$$y = 4,96 \cdot 10^{-7}x + 3,6061 \quad R^2 = 0,2937 \quad (5.8)$$

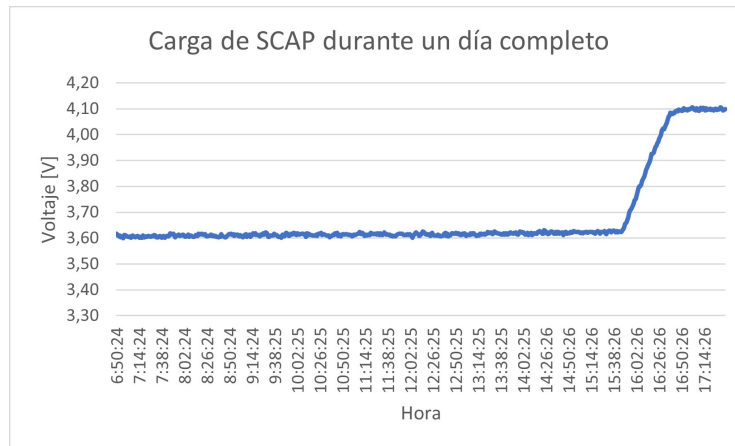


Figura 5.12: Carga de los supercondensadores durante un día completo.

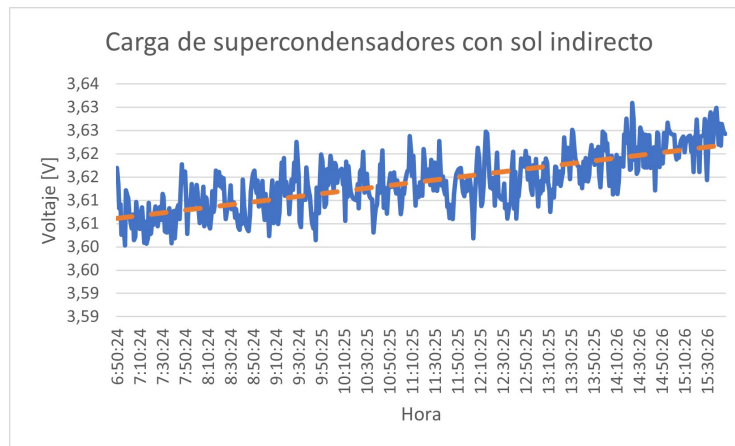


Figura 5.13: Detalle de carga de supercondensadores antes de recibir sol directo.

5.3.9. Autodescarga

Tanto las baterías como los condensadores sufren de autodescarga producto de corrientes de fuga. Por supuesto, los supercondensadores no están exentos de esto. En particular, los SCAP tipo EDLC se autodescargan mucho más rápido que los LIC [38].

Para medir la autodescarga de los supercondensadores usados se utilizó la configuración B. Notar que, además de la corriente de fuga propia de este, también se pierde una pequeña corriente al momento de realizar la medición con el divisor resistivo activable. La característica activable de este último, permite que esta corriente exista solo por un tiempo del orden de los microsegundos.

La Figura 5.14 muestra el resultado de esta prueba realizada durante 8 horas. Lamentablemente, el ADC del nRF52840 genera bastante ruido en la medición, por lo que

se traza una línea de tendencia descrita por (5.9), con un $R^2 = 0,3791$. Se puede estimar que durante el tiempo que se realizó la prueba, el voltaje bajó 10,66 mV.

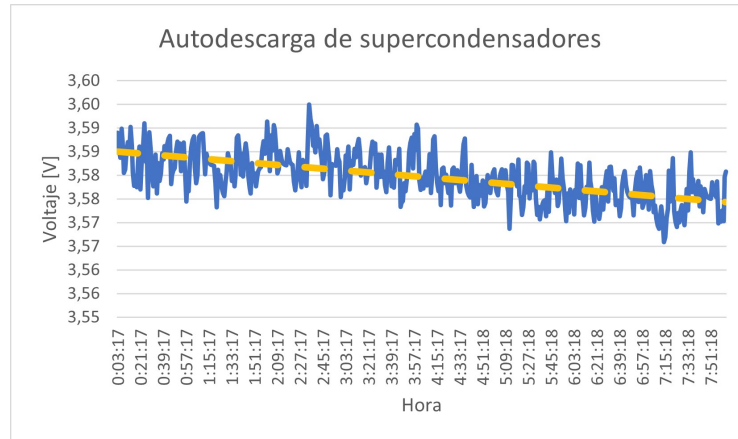


Figura 5.14: Autodescarga de supercondensadores durante 8 horas.

$$y = -3,7 \cdot 10^{-7}x + 3,5851 \quad (5.9)$$

5.3.10. Balanceo de supercondensadores

Una de las características del AEM10941 es que balancea ambas celdas de los supercondensadores. En esta prueba se estudió esta característica simplemente midiendo con un voltímetro el voltaje en cada una de ellas mientras se estaban cargando. El resultado se muestra en la Tabla 5.1, donde V_{batt} es el voltaje total en serie, V_{SCAP1} es el SCAP inferior, es decir, entre GND y el voltaje medio y V_{SCAP2} es el superior. La diferencia entre estos dos se muestra en mV y se puede apreciar que en todo momento se mantuvo una diferencia pequeña, de hasta 60 mV.

V_{batt} [V]	V_{SCAP1} [V]	V_{SCAP2} [V]	Diferencia [mV]
3,53	1,75	1,8	50
3,68	1,82	1,86	40
4,09	2,02	2,08	60
4,44	2,19	2,25	60

Tabla 5.1: Verificación del balanceo en los supercondensadores.

5.4. Subsistema funcional y prueba de sensores

La mayor parte del Subsistema funcional se verificó en el capítulo de Pruebas de integración. En ella se demostró que tanto la pantalla como la comunicación por Bluetooth funcionaron con éxito.

En esta sección se prueban los sensores SCD41 y SGP40. Desafortunadamente, como se mencionó en el capítulo de Desarrollo de Firmware, no fue posible obtener mediciones útiles del sensor BME688.

5.4.1. Medición de CO₂ en un dormitorio

Se realizó una prueba que consistió en medir la concentración de CO₂ de un dormitorio durante la noche, con la puerta y la ventana cerrada. El resultado de esta prueba se muestra en la Figura 5.15. Se observa que apenas se cierra la puerta, a eso de las 2:00 AM, la concentración de CO₂ sube progresivamente, comenzando en un nivel “Excelente” y terminando en uno “Muy peligroso” (de acuerdo con la Tabla 3.1). Al final de la mañana, momento en el cual se abren puertas y ventanas, se produce una drástica disminución en la concentración y por lo tanto, mejorando la calidad del aire de vuelta a los rangos iniciales.

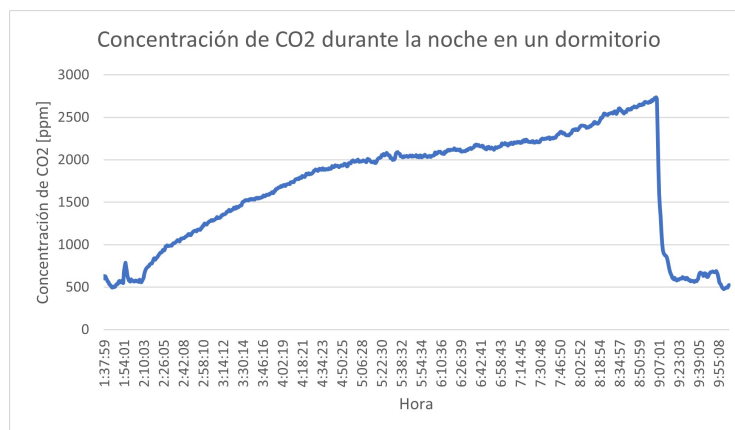


Figura 5.15: Concentración de CO₂ durante la noche en un dormitorio cerrado.

5.4.2. Comparación SCD41 versus SGP40

El sensor de VOC SGP40 genera una salida “cruda”, la cual se tiene que procesar en el MCU con el algoritmo [76], que produce un valor denominado VOC Index [80], el cual tiene rangos entre 0 y 500. Su funcionamiento se basa en un promedio móvil de las últimas 24 horas, por lo cual, el VOC index es un valor relativo, donde un número cercano a 100 indica que se tiene una medición cercana al promedio. Valores menores a 100 significa que la medición de VOCs es menor al promedio, por lo tanto, la calidad del aire es mejor. Viceversa para valores mayores a 100. La figura 5.16, obtenida desde [81], explica este comportamiento.

Se comparan los sensores SCD41 y SGP40, de CO₂ y VOC, respectivamente para probar si tienen comportamientos similares bajo las mismas condiciones. A partir de esto, se puede establecer correlación entre ambos datos y/o validar que los datos se relacionan con las condiciones del aire.

El procedimiento fue el mismo que el de la prueba anterior, solo que ahora se toman mediciones en ambos sensores simultáneamente y durante tres noches. El resultado de esta

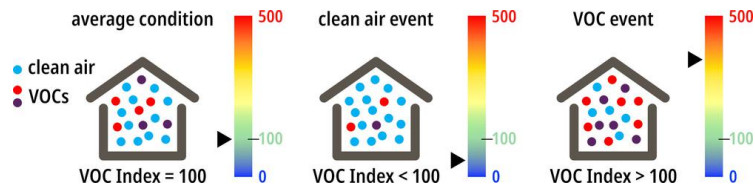


Figura 5.16: Funcionamiento del SGP40 y su VOC index.

prueba se muestra en la Figura 5.17.

Considerando que las durante las tres noches se mantuvieron prácticamente las mismas condiciones, se espera una respuesta similar de los sensores. Esto sí se cumplió para el sensor de CO₂, pero no así para el sensor de VOC, el cual tiene un comportamiento mucho más errático. Un claro ejemplo es lo observado en la segunda noche, pues presenta un comportamiento casi inverso al de CO₂ y los de sí mismo en comparación a las otras noches. Además, en general presenta peaks que se escapan de la tendencia y no se relacionan a ningún evento especial.

El coeficiente de correlación de la prueba completa es del 66,55 %. Mientras que considerando solo la última noche es de 78,30 %.

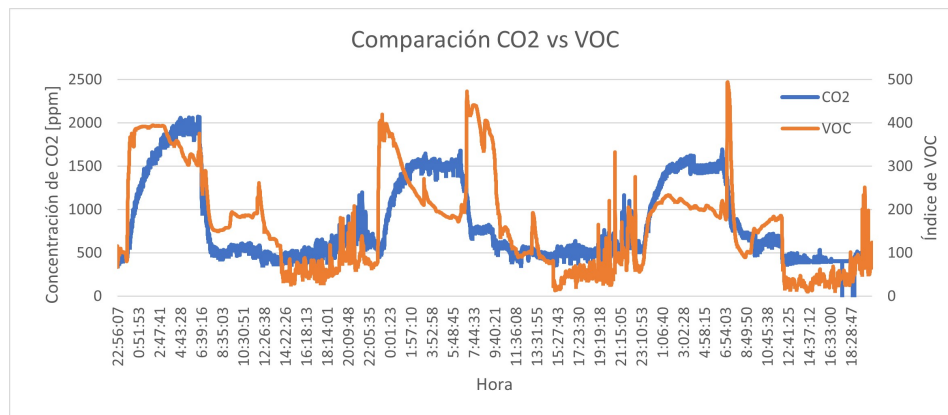


Figura 5.17: Comparación de los sensores SCD41 y SGP40.

5.5. Ensamble final con la carcasa

El diseño de la carcasa se continuó mejorando para poder incluirle tornillos con insertos metálicos (Fig. 5.18a) y una tapa que se pudiera abrir y cerrar fácilmente. En la Figura 5.18b se muestra la PCB ya instalada en la carcasa, usando 3 tornillos M2.5. Esto se hizo principalmente por motivos demostrativos, razón por la cual el panel solar ni los supercondensadores están conectados.

La Figura 5.18c muestra la carcasa ensamblada con el panel solar montado en la parte

superior y la pantalla indicando una excelente condición de ventilación. En la vista trasera se observa el agujero para montar el dispositivo en la pared (Fig. 5.18d).



(a) Detalle insertos metálicos.



(b) PCB montada.



(c) Vista frontal.



(d) Vista trasera.

Figura 5.18: Carcasa impresa en 3D ensamblada.

CONCLUSIONES Y TRABAJO FUTURO

6.1. Resultados obtenidos

ENTRE los resultados positivos, se puede destacar que el circuito y la PCB funcionaron de acuerdo a lo esperado. Por ejemplo, no fue necesario agregar cables para reparar pistas faltantes o agregar componentes además de los considerados inicialmente. Salvo quizá el condensador añadido entre el pin BATT y GND del AEM10941, pero este se agregó solo a modo de seguridad debido a la constante conexión y desconexión de los SCAP por motivo de las pruebas realizadas.

El Subsistema de alimentación, si bien no logró satisfacer todos los requerimientos, sí funcionó de acuerdo a su hoja de datos. El problema principalmente radicó en malas interpretaciones o en esperar un desempeño mejor del que es capaz el AEM10941.

Por otro lado, el subsistema funcional también funcionó bien. Haber probado este de manera preliminar en una protoboard fue de gran ayuda para traspasar a la PCB una versión sin errores, por lo que el mayor riesgo fue incorporar la DESPI-C02 en la misma. Ahí fue donde se tenía la mayor cantidad de posibles fuentes de error, debido a que se tuvieron que elegir componentes con valores cercanos a los que se recomiendan en las hojas de datos relacionadas a la pantalla y al DESPI-C02. Los modelos específicos de los componentes ahí presentes no se encontraban disponibles para comprar. Asimismo, también se podían haber cometido errores al momento de traspasar los diagramas eléctricos de los datasheet a Altium, principalmente porque dichos diagramas diferían levemente. Afortunadamente, el único problema que se tuvo con respecto a esto fue la elección del voltaje máximo de los condensadores relacionados a la pantalla. Estos debían ser de 50 V y se adquirieron de 10 V. Por suerte, esto se pudo solucionar al encontrar stock disponible en fuentes cercanas.

El principal problema se generó al juntar los dos subsistemas, algo que se había previsto en las Pruebas Preliminares y que, pese a que se intentó mitigar, no fue suficiente para

evitarlo. El sensor principal, el SCD41, al tomar mediciones requiere una corriente mayor de la que el PMIC es capaz de proveer. Esto provoca que el LDO del AEM10941 colapse y deje de funcionar, apagando todo el sistema.

Haber desconocido las limitaciones de los supercondensadores de tecnología LIC trajo bastantes dificultades y contratiempos. Provocó la pérdida de un AEM10941 y fue necesario conseguir una alternativa. Por otro lado, el espacio que se le había destinado en la PCB no fue suficiente para el nuevo SCAP EDLC.

6.2. Conclusiones

De las principales conclusiones que se pueden obtener de este proyecto, se tiene el desempeño del AEM10941. En un comienzo fue complicado ponerlo en operación, provocando incluso la pérdida de dos unidades de estos IC, en gran parte debido al problema en la elección de los supercondensadores. Este PMIC resultó tener un buen desempeño en el algoritmo MPPT y la carga de supercondensadores, siempre y cuando se contara con luz solar directa. Entre sus desventajas se encuentra la baja corriente de salida que es capaz de proveer, que fue insuficiente para alimentar el SCD41. La carga mediante USB no funcionó, sin embargo, ese no es el propósito del pin que se usó, por lo que no se considera como un problema del PMIC.

El sensor SCD41 funcionó muy bien y su implementación fue sencilla. Además del CO₂ también mide temperatura y humedad, lo cual es muy útil para esta aplicación. Es sumamente reactivo a los eventos del ambiente, como se pudo ver en las pruebas realizadas, al ventilar la habitación, por ejemplo. En dicha prueba se descubrió lo peligroso que resulta dormir en lugar sin ventilación debido a las grandes concentraciones de CO₂ que se alcanzan. El gran problema de este sensor es la cantidad de corriente que necesita para tomar mediciones.

El sensor SGP40 fue sencillo de utilizar y no se tuvieron mayores problemas. Sin embargo, las pruebas que se le realizaron reflejaron un comportamiento extraño con poca relación a los eventos del ambiente.

El sensor BME688 aparece como una alternativa prometedora, según su hoja de datos tiene un bajo consumo de energía y proporciona una gran cantidad de mediciones, por lo que podría proveer al dispositivo de todos los datos que necesita. Lamentablemente, fue imposible de hacer funcionar, además la librería BSEC2 que postprocesa los datos parece complicada de incluir y está precompilada, lo cual dificulta su uso y comprensión.

El módulo con el SoC nRF52840 permitió tener un dispositivo con conexión Bluetooth de manera rápida y sencilla, pese a las dificultades que se tuvieron en un inicio con respecto a la curva de aprendizaje de Zephyr y el bloqueo contra escritura que incluía el módulo. Asimismo, Bluetooth Low Energy fue un gran reto en un comienzo, pero que finalmente se logró comprender y aprovechar sus funcionalidades.

La pantalla de tinta electrónica no resultó tan difícil de utilizar después de todo, y las expectativas que se tenían se cumplieron plenamente. Se logró una buena visualización en todos los escenarios previstos y las tasas de refresco de las imágenes estuvieron acorde a la aplicación.

El concepto de *energy harvesting* con energía lumínica realmente tiene muchísimo potencial y campos de aplicación. Sin embargo, para aplicaciones como esta, en las cuales el dispositivo está pensado para trabajar en interiores, resulta complicado lograr un resultado fiable. De las pruebas realizadas se llegó a la conclusión de que el panel solar debe recibir luz solar directa por al menos una hora al día para lograr cargar los supercondensadores al máximo. Aún así, el consumo de energía del dispositivo debe ser ínfimo, menor a 0,68 mA según los cálculos realizados.

Realizar pruebas relacionadas a la energía solar es complicado y requiere de mucho tiempo. Estas solo se pueden realizar durante ciertas horas al día y son difíciles de replicar ya que está sujeto a las condiciones meteorológicas.

6.3. Trabajo Futuro

El reto más importante que queda por resolver es solucionar la incompatibilidad energética entre el sensor SCD41 y el PMIC AEM10941. La forma más sencilla es simplemente cambiar el sensor, por ejemplo, por uno de los que se probaron, cuyos puntos a considerar ya se explicaron anteriormente. En caso de seguir este camino y el LDO interno del AEM10941 es suficiente para alimentar el nuevo sensor, aún queda considerar 1) la carga por USB, que no funcionó en absoluto y 2) el bajo aprovechamiento de los supercondensadores que se logra con este PMIC. Ya que, de las mayores ventajas que presentan los supercondensadores EDLC en comparación a las baterías recargables es que estos pueden descargarse hasta cero. Es decir, es posible aprovechar la energía de los supercondensadores por debajo de los 3,6 V que es capaz el AEM10941.

De lo anterior se extrae que, una buena alternativa es seguir utilizando el AEM10941, pero solo en las etapas en las que demostró un buen desempeño, principalmente en la carga balanceada de los supercondensadores a partir de energía solar y en la implementación del algoritmo MPPT.

Mientras que, para proveer una salida de 3,3 V, en lugar de usar el LDO del AEM10941, una mejor opción es utilizar un convertidor Buck-Boost dedicado, capaz de proporcionar la salida requerida con voltajes tanto superiores como inferiores a este, con lo que se logra el objetivo de aprovechar de mejor manera las ventajas de los supercondensadores.

En esta misma línea, se podría implementar un cargador de SCAP mediante USB dedicado, con el fin de limitar la corriente de carga para proteger la fuente del puerto USB y limitar el voltaje máximo del SCAP.

Entre las mejoras que se le pueden hacer al dispositivo es utilizar una pantalla de tinta electrónica de mayor tamaño, lo que permitiría mostrar más información en pantalla, como la hora, fecha, porcentaje de batería y el estado de la conexión Bluetooth. Asimismo, proporcionaría más espacio para esconder los supercondensadores, permitiendo, por ejemplo, usar unos de mayor tamaño.

Respecto de la tecnología de transmisión inalámbrica, resulta interesante considerar la

implementación de Matter, el cual es una nueva especificación que está siendo impulsada por Apple, Google y Amazon, entre otros, para detener la fragmentación que existe actualmente en relación a las tecnologías para dispositivos IoT. Al momento de comenzar este proyecto se consideró su uso, pero se descartó por tratarse de una tecnología demasiado nueva y poco masificada. El SoC nRF52840 incluye soporte para Matter.

También se podrían evaluar otros tipos de conexión por Bluetooth, como Bluetooth Mesh, el rol Broadcast o funcionamiento tipo Beacon, con el cual se pueden transmitir datos sin necesidad de conexión.

A pesar de que los datos que maneja el dispositivo no son sensibles en cuanto a seguridad, en caso de requerirlo, se le podría añadir una capa de seguridad o encriptación de datos. En cualquier caso, una vez que se establece una conexión Bluetooth, no se puede conectar a otro dispositivo, quedando una comunicación exclusiva entre ambos.

Queda pendiente el desarrollo de una aplicación con la cual el dispositivo pueda comunicarse. Con esto los datos podrían, por ejemplo, procesarse, subirse a la nube y graficarse en un tablero virtual. También podría implementarse un sistema de buffer en el dispositivo, para que almacene sus datos por un tiempo determinado. Luego, al conectarse con la aplicación le comparte los datos en forma de ráfaga. De esta manera, no es necesario tener comunicación constante entre la aplicación y el dispositivo, evitando la pérdida de datos y ahorrando energía.

Finalmente, queda continuar estudiando los modos de bajo consumo del nRF52840 y la desactivación de periféricos para ahorrar energía. En esta misma línea, se recomienda utilizar resistencias de mayor valor para el divisor resistivo, de manera que la medición del voltaje requiera aún menos corriente.

HERRAMIENTAS

En este apéndice se describen y explican algunas herramientas utilizadas a lo largo del proyecto. Esto incluye el analizador de energía JouleScope, los programadores utilizados y las herramientas de software para generar los elementos gráficos de la pantalla de tinta electrónica.

A.1. JouleScope

JouleScope es un analizador de energía DC, capaz de medir corriente y voltaje de un circuito electrónico. Luego, a partir de estas variables calcula la potencia, energía y carga eléctrica. Posee un alto rango dinámico pasando por 9 órdenes de magnitud, desde amperes hasta nanoamperes. [82].

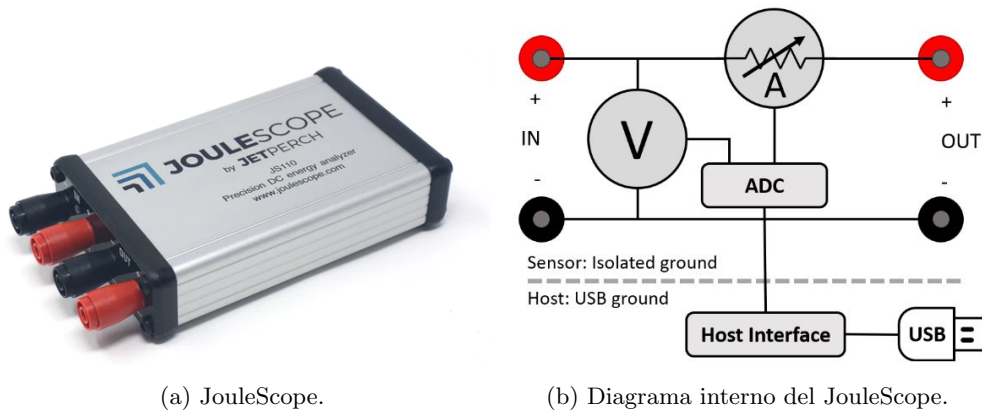
Para utilizarlo hay que conectar sus terminales de entrada (IN+ e IN-) a la fuente y los terminales de salida (OUT+ y OUT-) al circuito que se quiere medir. Luego se conecta al computador por medio de un cable USB y se visualizan los datos en tiempo real en el software homónimo. La figura A.1 muestra el dispositivo y su diagrama interno que ayuda a entender su funcionamiento.

A.2. Programador

En palabras simples, un programador es un dispositivo que se utiliza generalmente para grabar un código, es decir, una serie de instrucciones en la memoria no volátil de los microcontroladores. Por lo general también incluyen la función de *debugger*, lo que significa que el usuario puede, por ejemplo, analizar el estado interno del microcontrolador y/o detener la ejecución del programa en instantes específicos en tiempo real.

Para programar el microcontrolador se utilizó un programador J-Link, aunque en un principio se intentó con un Black Magic Probe (BMP). Las razones para esto se detallan a continuación.

De fábrica, el módulo E73-2G4M08S1C está protegido contra escritura de firmware, lo



(a) JouleScope.

(b) Diagrama interno del JouleScope.

Figura A.1: JouleScope y su diagrama interno.

que significa que es necesario un programador capaz de ejecutar instrucciones específicas para desbloquearlo. Esto es importante, porque si no estuviese protegido, podría programarse directamente con un ST-Link, que es sin duda la opción más accesible debido a su bajo costo. El problema es que un ST-Link no soporta las instrucciones para desbloquear el módulo.

La opción más sencilla, pero también más costosa, para desbloquear este módulo es utilizar un programador J-Link. Este, soporta programación y depuración mediante JTAG y SWD (Serial Wire Debug), donde, este último es que se necesita en este caso. De hecho, la placa de desarrollo nRF52840-DK incluye J-Link embebido en ella y utiliza SWD.

Debido a que conseguir un J-Link puede resultar costoso, primero se intentó utilizar una alternativa Open Source, llamada Black Magic Probe.

A.2.1. Black Magic Probe

La opción más económica, y que se usó en un inicio pero se descartó porque luego se pudo tener acceso a un J-Link, es usar un Black Magic Probe (BMP). Más específicamente, construir uno. En [79] y [83] se explica cómo hacerlo, pero básicamente el procedimiento consiste en usar una placa Blue Pill con microcontrolador STBM32F103C8T6 y programarle el firmware de BMP con un ST-LINK o con un convertidor UART a USB. Básicamente, programar un microcontrolador para convertirlo en un programador más potente. La ventaja, además del costo monetario, es que además del puerto para programar por SWD, genera un puerto COM para usarlo como conversor UART a USB. Adicionalmente, la placa Blue Pill ya incluye un regulador con salida de 3,3 V. La desventaja es que su uso como depurador (debugger) es menos fluido que el J-LINK.

La línea de comando que resultó para desbloquear el E73-2G4M08S1C con Black Magic Probe fue la siguiente:

```
1 arm-none-eabi-gdb --batch -ex "target extended-remote \\.COM7"
  ↪ -ex "mon swdp_scan" -ex "att 2" -ex "mon erase_mass"
```

A.2.2. J-Link

Como se mencionó, finalmente fue posible acceder a un J-Link, por lo que se decidió utilizar este, por tratarse de un dispositivo comercial y con mejor soporte.

El único inconveniente del J-Link es que no permite alimentar con 3,3 V el sistema a programar, por lo que es necesario usar una fuente externa. Para usar este programador es necesario instalar nRF Command Line Tools [84] y los drivers de J-Link [85]. Para ejecutar el desbloqueo se usa la siguiente línea de comando:

```
1 nrfjprog --recover --log
```

Los pines de entrada/salida para programar mediante SWD se muestran en la figura A.2. Notar que el orden de los pines en la figura corresponden a la salida directa del programador. Esto es importante ya que al conectar el cable plano, en el extremo libre de este, el orden de los pines queda en sentido inverso.

VTref	1 ● ● 2	NC
Not used	3 ● ● 4	GND
Not used	5 ● ● 6	GND
SWDIO	7 ● ● 8	GND
SWCLK	9 ● ● 10	GND
Not used	11 ● ● 12	GND
SWO	13 ● ● 14	GND*
RESET	15 ● ● 16	GND*
Not used	17 ● ● 18	GND*
5V-Supply	19 ● ● 20	GND*

Figura A.2: Asignación de pines del J-Link en compatibilidad con SWD.

Una vez desbloqueado, el módulo se puede programar sin problemas con cualquiera de los programadores mencionados, e incluso con un ST-Link.

A.3. Generación de elementos gráficos

Para generar los elementos gráficos y que estos se puedan usar con la pantalla de tinta electrónica se utilizaron principalmente dos software: paint.net [86] e Image2Lcd [87]. Con el primero se realizó el diseño gráfico, mientras que con el segundo se convierten los elementos gráficos a un arreglo de bits.

La Figura A.3 muestra parte del proceso para generar un número. En este caso particular se trata del número 3 de la fuente Rockwell Extra Bold tamaño 28. De izquierda a derecha, se observa que el primer número tiene pixeles grises, sin embargo, la pantalla no puede representar esos colores porque es monocromática. Para eliminar estos pixeles, se eleva el

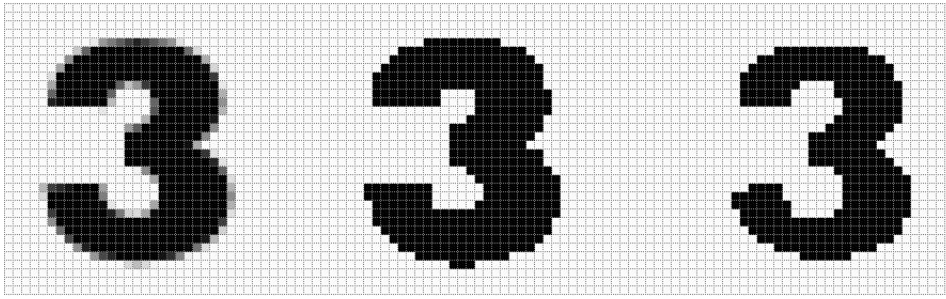


Figura A.3: Ajuste de brillo y contraste para generar el número 3.

contraste al máximo y se ajusta el brillo hasta encontrar un resultado cercano al deseado. Los siguientes dos “3” corresponden a ajustar el brillo al mínimo y al máximo, respectivamente. También es útil usar la herramienta Lápiz para pintar cada pixel individualmente y así ajustar el elemento gráfico.

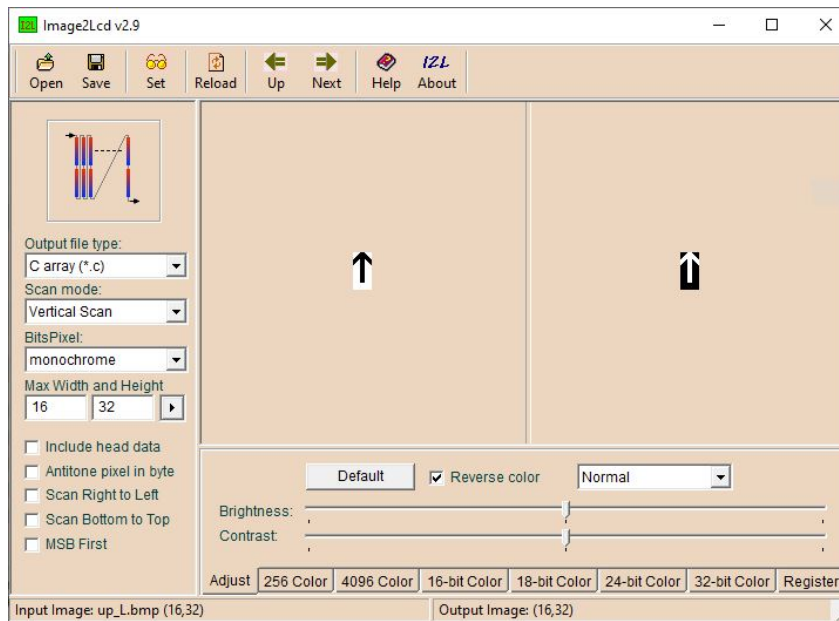


Figura A.4: Captura del software Image2Lcd para convertir una flecha de la interfaz gráfica a un arreglo de bytes.

En la captura del software Image2Lcd (Fig. A.4) se muestran las configuraciones necesarias para generar un arreglo de bytes compatible con la pantalla GDEH0154D67. La imagen de entrada mide 16x32 píxeles (ancho x alto), por lo que tiene un total de $16 \cdot 32 = 512$ píxeles. Al tratarse de una pantalla monocromática, cada pixel es equivalente a un solo bit. El driver de la pantalla tiene registros de 8 bit (1 byte), de esta manera, el arreglo resultante consta de $512/8 = 64$ bytes. Es importante mencionar que los registros están asignados verticalmente con respecto a la pantalla, por lo que la imagen de entrada

debe tener una altura múltiplo de 8. Horizontalmente no hay limitaciones de este sentido, salvo obviamente, la dimensión máxima que es de 200x200 pixeles.

CAPTURA Y GENERACIÓN DE GRÁFICOS A PARTIR DEL PUERTO SERIAL

En este apéndice se explica el procedimiento para generar gráficos en Excel a partir de los datos que envía el microcontrolador a través del puerto serial (UART). Asimismo, se explica brevemente que el conversor UART-USB también se utiliza como fuente de voltaje de 3,3 V.

B.1. Conversor UART a USB

La placa fue diseñada para brindar acceso directo a los pines UART RX y TX del microcontrolador (puntos 12 y 13), sin embargo, para poder enviar o recibir mensajes entre este y un computador, es necesario utilizar un conversor UART a USB.

Tal como se explicó en el capítulo 4, en un comienzo se utilizó una placa de desarrollo Blue Pill programada como Black Magic Probe, ya que se pretendía usar también como programador. Finalmente, no fue necesario usar dicha placa, puesto que se pudo contar con un J-Link para programar. Esto se explica detalladamente en el Apéndice A.

Finalmente, se usó un módulo CP2102 como el de la Figura B.1. Con este también es posible alimentar la placa con 3,3 V en caso de que se quiera prescindir del Subsistema de alimentación (Configuración A). Si se desea usar con este subsistema, solo se debe conectar GND (Configuración B). En ambos casos, dado que el microcontrolador no recibe datos, basta con conectar el pin RX del CP2102 al TX de la placa. Mencionar que para usar el módulo CP2102 con el computador, es necesario tener instalado su driver [88].

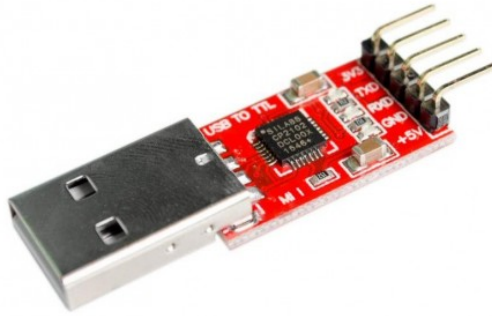


Figura B.1: Módulo UART-USB CP2102.

B.2. Captura de datos

A pesar que PlatformIO incluye un monitor serial, para capturar datos que se desean analizar posteriormente, se prefiere utilizar el programa RealTerm. Este contiene herramientas para almacenar en un archivo los datos recibidos, con la posibilidad de agregarle una marca temporal. De esta manera es posible recibir y guardar datos por incluso días sin perderlos y conociendo exactamente cuándo se capturó cada uno.

En la configuración de puertos, figura B.2, es importante definir la tasa de baudios y el número de puerto, que en caso de desconocerlo, se puede revisar en el Administrador de dispositivos de Windows. Los demás parámetros deben mantenerse por defecto. Luego presionar Open y/o Change según corresponda. Esto permitirá visualizar los datos recibidos por UART en la sección de color negro.

En la pestaña de captura, figura B.3, se debe definir el archivo en el cual se desean registrar los datos recibidos. Además, se puede elegir el tipo de marca de tiempo y el carácter separador. Se recomienda usar Unix y space, respectivamente. Las demás opciones pueden quedar por defecto.

Para comenzar la captura, presionar Start: Overwrite o Append, dependiendo si se quiere crear/sobreescribir el archivo o seguir rellenándolo. Para la segunda opción el archivo debe estar creado.

Al comenzar la captura, una parte de la ventana cambia a color rojo y lamentablemente, no se pueden ver los datos en tiempo real, sin embargo, sí se es posible abrir el archivo mientras RealTerm continúa escribiendo datos en él. Para terminar la captura, simplemente presionar Stop Capture.

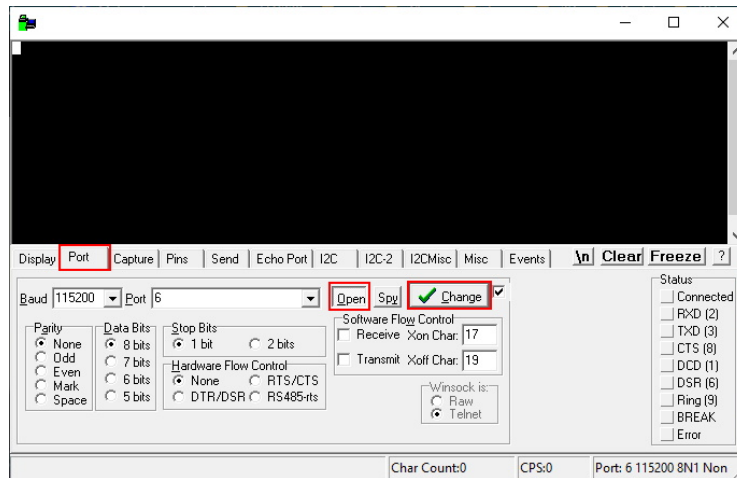


Figura B.2: Pestaña de configuración de puerto en RealTerm.

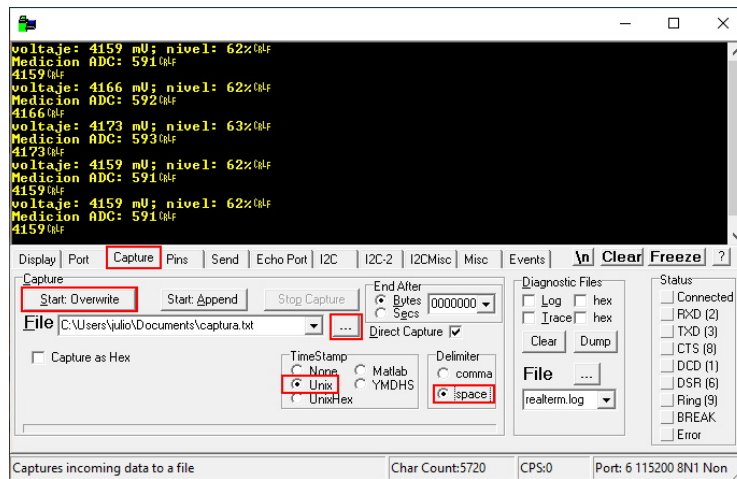


Figura B.3: Pestaña de captura en Realterm.

Para posteriormente poder generar gráficos, es importante que los datos estén organizados en columnas. Para esto es necesario que todos los datos que se transmitan por UART se envíen de una sola vez por iteración y sin otros mensajes que interfieran. Por ejemplo, si se quieren enviar dos datos: medición del ADC y temperatura, estos deben imprimirse con una línea de código como la siguiente:

```
printk("%d %d\n", ADC, temperatura);
//Esto se ejecuta cada x segundos
```

Con la recomendación anterior sobre la marca temporal, en el archivo de captura se obtiene algo como

```
1674683131 592 28103
```

1674683141	592	28281
1674683151	591	28235

En este caso se enviaron datos cada 10 segundos y los dos datos corresponden a enteros (la temperatura está en m°C).

B.3. Generación de gráficos

Para generar los gráficos a partir del archivo de captura creado anteriormente, se usó el programa Excel. Primeramente se deben importar los datos, una opción es a través del menú Datos > Obtener datos > De un archivo > De texto/CSV.

Sin embargo, esta opción no se recomienda, ya que es más difícil de sobrepasar errores que hayan podido ocurrir durante la captura. Por ejemplo, para hacer la prueba de descarga de los SCAP, se mantuvo la captura de datos hasta que el sistema se apagara. El problema es que en ese momento el monitor serial capturó caracteres inesperados que escapaban de la organización en columnas, lo que provocó que Excel no las detectara.

Por esto, se recomienda copiar y pegar el contenido del archivo de texto a un libro en blanco de Excel. Luego, en Opciones de pegado, seleccionar Usar el asistente para importar texto. Hacer clic en Siguiente para elegir el separador y luego Finalizar.

Para convertir la marca temporal desde Unix (segundos desde 01/01/1970) a una hora legible, se puede usar la siguiente función de Excel:

=A1 / (24*60*60) + FECHA (1970 ; 1 ; 1)

Luego, cambiar el formato de General a Hora desde el menú Inicio. Aplicar esto a toda la columna. Se recomienda copiar los datos a la derecha de la nueva columna de marca temporal y aprovechar de convertirlos en caso de ser necesario, como se muestra en la Figura B.4, donde se convierte un voltaje en mV a V con dos decimales.

	A	B	C	D	E	F
1	1673125623	4194			21:07:03	4,19
2	1673125634	4187			21:07:14	4,19
3	1673125644	4201			21:07:24	4,20
4	1673125655	4194			21:07:35	4,19
5	1673125666	4194			21:07:46	4,19
6	1673125676	4201			21:07:56	4,20
7	1673125687	4187			21:08:07	4,19
8	1673125697	4187			21:08:17	4,19
9	1673125708	4194			21:08:28	4,19
10	1673125719	4201			21:08:39	4,20
11	1673125729	4180			21:08:49	4,19

Figura B.4: Captura de Excel con los datos preparados para graficar.

Finalmente, para generar el gráfico, seleccionar las columnas que se desean y usar la opción Gráficos recomendados desde el menú Insertar.

ANEXO: PCB Y DIAGRAMA ESQUEMÁTICO

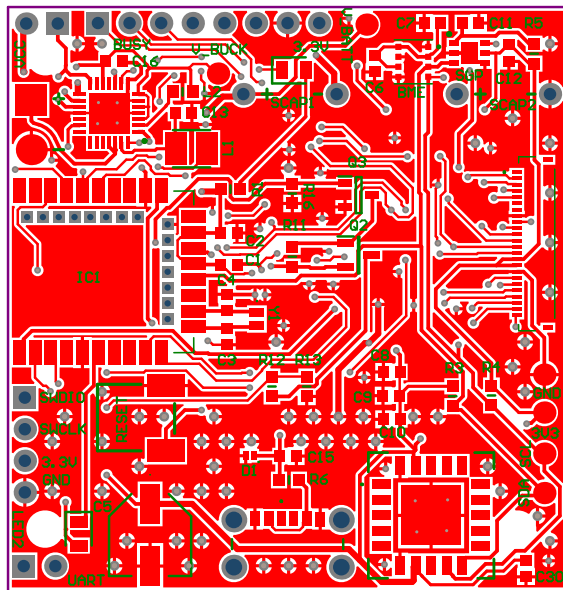


Figura C.1: Capas Top y Top Overlay de la PCB diseñada.

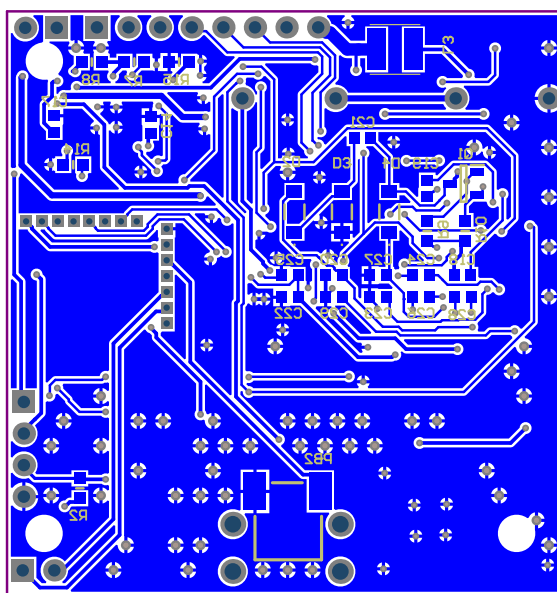


Figura C.2: Capas Bottom y Bottom Overlay de la PCB diseñada.

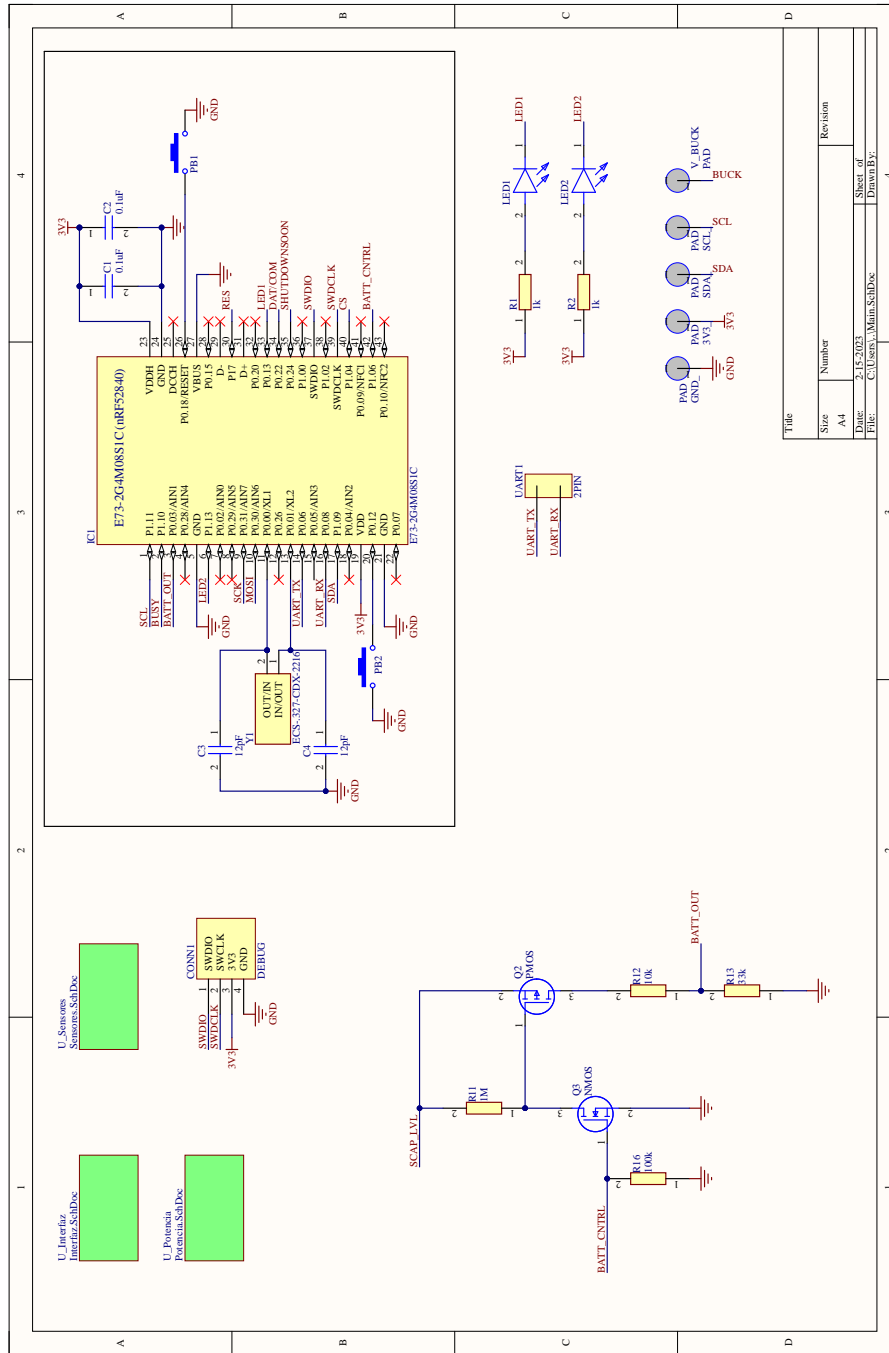


Figura C.3: Diagrama esquemático principal, con microcontrolador y circuitos asociados.

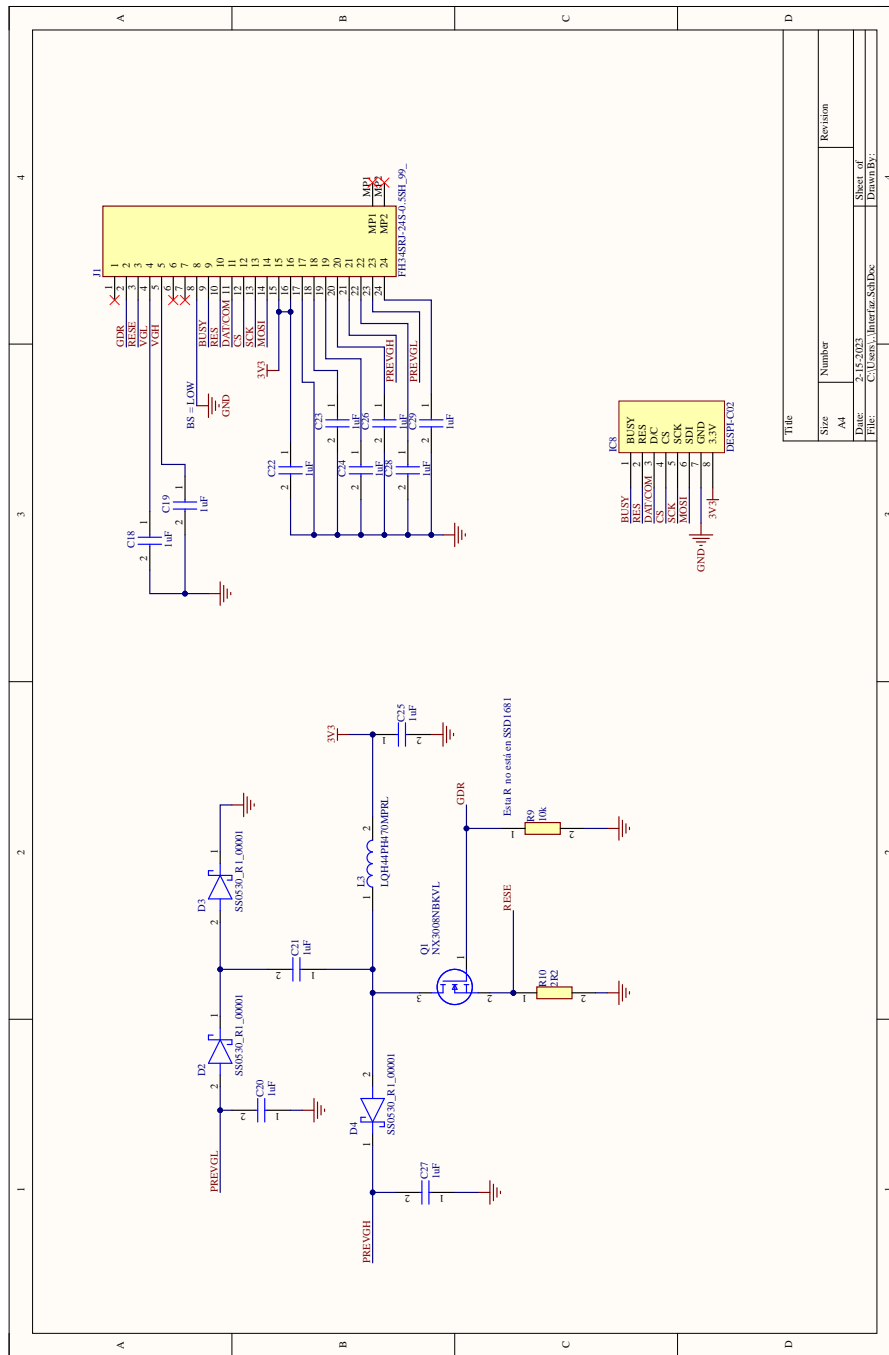


Figura C.5: Diagrama esquemático relativo al circuito adaptador para la pantalla de tinta electrónica.

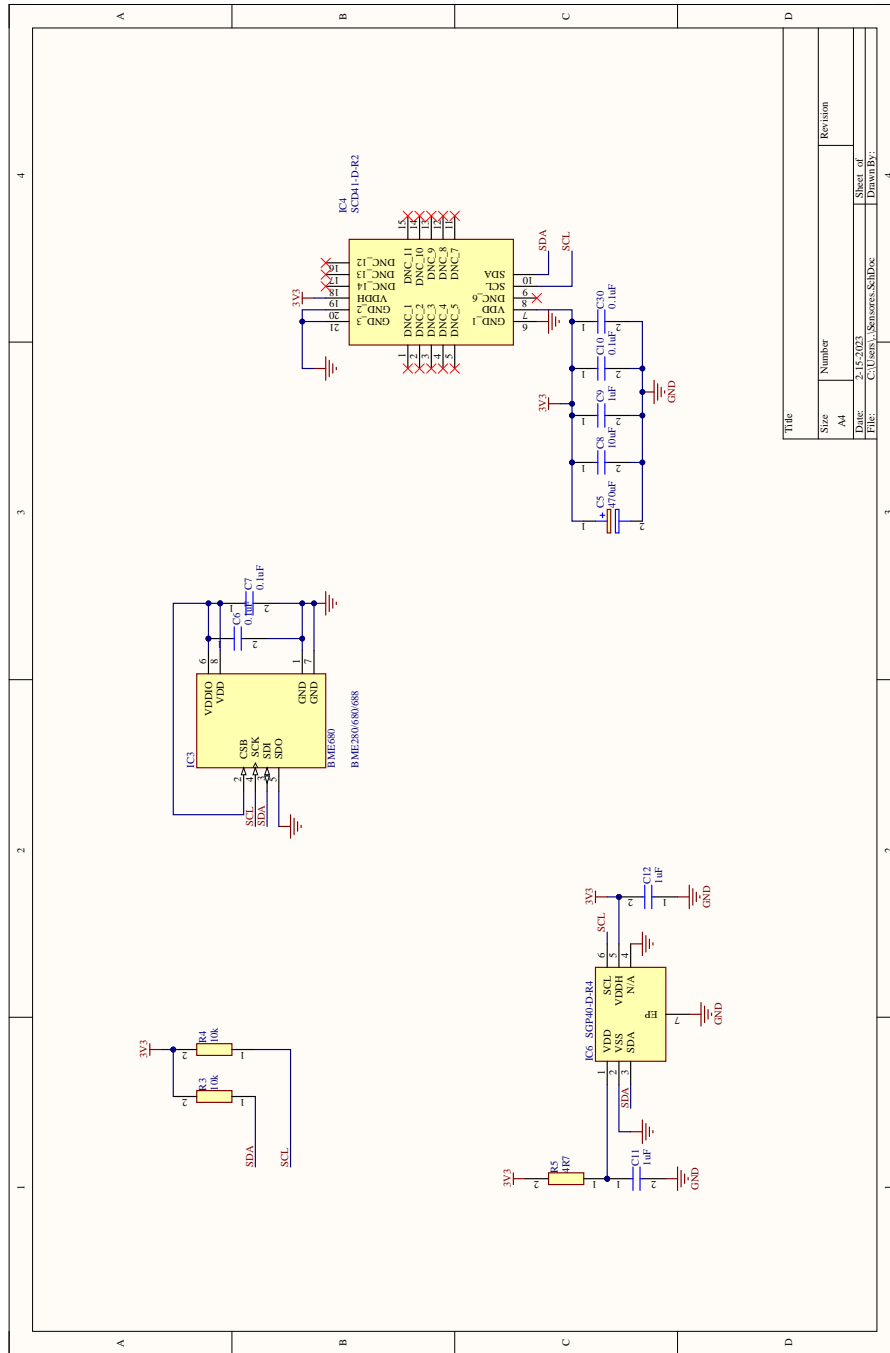


Figura C.6: Diagrama esquemático relativo a los sensores utilizados y las resistencias para el bus I2C.

BIBLIOGRAFÍA

- [1] N. Kappelt, H. S. Russell, S. Kwiatkowski, A. Afshari, and M. S. Johnson, “Correlation of respiratory aerosols and metabolic carbon dioxide,” *Sustainability*, vol. 13, no. 21, 2021. [Online]. Available: <https://doi.org/10.3390/su132112203>
- [2] World Health Organization, “Household air pollution.” [Online]. Available: <https://www.who.int/news-room/fact-sheets/detail/household-air-pollution-and-health> (Accessed 09/12/2022).
- [3] V. V. Tran, D. Park, and Y.-C. Lee, “Indoor air pollution, related human diseases, and recent trends in the control and improvement of indoor air quality,” *International Journal of Environmental Research and Public Health*, vol. 17, no. 8, p. 2927, Apr. 2020. [Online]. Available: <https://doi.org/10.3390/ijerph17082927>
- [4] Sensirion, “Indoor air quality brochure.” [Online]. Available: https://sensirion.com/media/documents/87EDEFB9/62601F86/Sensirion_Indoor-Air-Quality_Brochure.pdf (Accessed 09/12/2022).
- [5] A. Schieweck, E. Uhde, T. Salthammer, L. C. Salthammer, L. Morawska, M. Mazaheri, and P. Kumar, “Smart homes and the control of indoor air quality,” *Renewable and Sustainable Energy Reviews*, vol. 94, pp. 705–718, Oct. 2018. [Online]. Available: <https://doi.org/10.1016/j.rser.2018.05.057>
- [6] US EPA, “Indoor Air and Coronavirus (COVID-19).” [Online]. Available: <https://www.epa.gov/coronavirus/indoor-air-and-coronavirus-covid-19> (Accessed 09/12/2022).
- [7] Z. Peng and J. L. Jimenez, “Exhaled co2 as covid-19 infection risk proxy for different indoor environments and activities,” *medRxiv*, 2021. [Online]. Available: <https://doi.org/10.1101/2020.09.09.20191676>
- [8] N. Agarwal, C. S. Meena, B. P. Raj, L. Saini, A. Kumar, N. Gopalakrishnan, A. Kumar, N. B. Balam, T. Alam, N. R. Kapoor, and V. Aggarwal, “Indoor air quality improvement in COVID-19 pandemic: Review,” *Sustainable Cities and Society*, vol. 70, p. 102942, Jul. 2021. [Online]. Available: <https://doi.org/10.1016/j.scs.2021.102942>
- [9] P. Wolkoff, “Indoor air humidity, air quality, and health – an overview,” *International Journal of Hygiene and Environmental Health*, vol. 221, no. 3, pp. 376–390, Apr. 2018. [Online]. Available: <https://doi.org/10.1016/j.ijheh.2018.01.015>

- [10] Sensirion, “What is a metaloxide (MOX) sensor?” [Online]. Available: https://sensirion.com/media/documents/0083CDF4/6294DFEA/Info_Note_MOX_sensor.pdf (Accessed 09/12/2022).
- [11] Sensirion, “What types of NDIR sensors exist and how do they work?” [Online]. Available: https://sensirion.com/media/documents/9E7DA521/627C2C8D/CD_IN_SCDxx_Transmissive_and_photoacoustic_NDIR_sensing_D1.pdf (Accessed 05/03/2023).
- [12] Netatmo, “Medidor de Calidad del Aire interior inteligente.” [Online]. Available: <https://www.netatmo.com/es-es/aircare/homecoach> (Accessed 09/12/2022).
- [13] Medidco, “Medidor de Aire CO2 de Pared: Temperatura, Humedad Y CO2.” [Online]. Available: <https://medidco.cl/producto/medidor-de-aire-co2-de-pared-temperatura-humedad-y-co2/> (Accessed 09/12/2022).
- [14] Medidores Chile, “M2: Portátil Escritorio medidor detector de co2 ventilación.” [Online]. Available: <https://www.medidoreschile.cl/medidor-detector-co2-aire-mc02-sobremesa-escritorio> (Accessed 09/12/2022).
- [15] Detectores y Sensores, “Detector CO2 Dióxido de Carbono.” [Online]. Available: <https://www.detectores.cl/tienda/detector-co2-dioxido-de-carbono/> (Accessed 09/12/2022).
- [16] J. Petersen, J. Kristensen, H. Elarga, and R. Andersen, “Accuracy and air temperature dependency of commercial low-cost ndir co2 sensors: An experimental investigation,” 02 2018.
- [17] IC-Meter, “Indoor climate and energy.” [Online]. Available: https://www.ic-meter.com/wp-content/uploads/2014/05/IC-Meter_folder_energy_ENG.4.2014.pdf (Accessed 09/12/2022).
- [18] PINE64, “PineTime.” [Online]. Available: <https://www.pine64.org/pinetime/> (Accessed 11/02/2023).
- [19] Berk, “Air quality and weather forecast sensor.” [Online]. Available: <https://www.openhardware.io/view/8259/Air-quality-and-weather-forecast-sensor> (Accessed 11/02/2023).
- [20] jakkra, “ZSWatch.” [Online]. Available: <https://github.com/jakkra/ZSWatch> (Accessed 12/02/2023).
- [21] DesdeLinux, “ZSWatch, un smartwatch de diseño libre basado en Zephyr OS.” [Online]. Available: <https://blog.desdelinux.net/zswatch-un-smartwatch-de-diseno-libre-basado-en-zephyr-os/> (Accessed 11/02/2023).
- [22] Nordic Semiconductor, “nRF52840 - Bluetooth 5.2 SoC.” [Online]. Available: <https://www.nordicsemi.com/products/nrf52840> (Accessed 09/12/2022).
- [23] Ebyte, “E73-2G4M08S1C.” [Online]. Available: <https://www.ebyte.com/en/product-view-news.aspx?id=445> (Accessed 09/12/2022).
- [24] Pervasive Displays, “EPD Driving Circuit.” [Online]. Available: <https://www.pervaisivedisplays.com/wp-content/uploads/2020/12/EPD-driving-circuit-of-Pervasive-Displays-rev.01.pdf> (Accessed 09/12/2022).

- [25] Good Display, “1.54 inch e-paper display module partial refresh E-ink screen, GDEH0154D67.” [Online]. Available: <https://www.good-display.com/product/1.54-inch-e-paper-display-module-partial-refresh-E-ink-screen,-GDEH0154D67-208.html> (Accessed 09/12/2022).
- [26] Solomon Systech, “SSD1681 Product Preview.” [Online]. Available: <https://www.e-paper-display.com/SSD1681%20V0.13%20Spec903d.pdf> (Accessed 09/12/2022).
- [27] Good Display, “Connection adapter board HAT connect for e-Paper display DESPIC02.” [Online]. Available: https://www.e-paper-display.com/products_detail/productId=403.html (Accessed 09/12/2022).
- [28] e-peas, “AEM10941 Solar Harvesting — Photovoltaic Energy Harvesting.” [Online]. Available: <https://e-peas.com/product/aem10941/> (Accessed 09/12/2022).
- [29] e-peas, “AEM10941 Datasheet - Highly efficient, regulated dual-output, ambient energy manager for up to 7-cell solar panels with optional primary battery.” [Online]. Available: <https://e-peas.com/wp-content/uploads/2022/09/e-peas-AEM10941-datasheet-solar-energy-harvesting.pdf> (Accessed 09/12/2022).
- [30] Sensirion, “SCD41 - CO₂ accuracy of $\pm(40 \text{ ppm} + 5\% \text{ MV})$ @400-5000 ppm, low power.” [Online]. Available: <https://sensirion.com/products/catalog/SCD41/> (Accessed 09/12/2022).
- [31] Sensirion, “SGP40 - VOC sensor for HVAC and air quality applications.” [Online]. Available: <https://sensirion.com/products/catalog/SGP40/> (Accessed 09/02/2023).
- [32] Sensirion, “Datasheet SGP40.” [Online]. Available: https://sensirion.com/media/documents/296373BB/6203C5DF/Sensirion_Gas_Sensors_Datasheet_SGP40.pdf (Accessed 09/02/2023).
- [33] Bosch Sensortec, “Gas sensor BME688.” [Online]. Available: <https://www.bosch-sensortec.com/products/environmental-sensors/gas-sensors/bme688/> (Accessed 09/02/2023).
- [34] Bosch Sensortec, “BME688 - Datasheet.” [Online]. Available: <https://www.bosch-sensortec.com/media/boschsensortec/downloads/datasheets/bst-bme688-ds000.pdf> (Accessed 09/02/2023).
- [35] ARM Mbed, “Nordic nRF52840-DK.” [Online]. Available: <https://os.mbed.com/platforms/Nordic-nRF52840-DK/> (Accessed 14/02/2023).
- [36] Nordic Semiconductor, “nRF52840 Product Specification.” [Online]. Available: https://infocenter.nordicsemi.com/pdf/nRF52840_PS_v1.7.pdf (Accessed 19/02/2023).
- [37] Tecate Group, “TPLC-3R8/200MR16X25 Datasheet.” [Online]. Available: <https://www.tecategroup.com/products/data.sheet.php?i=TPLC-3R8/200MR16X25> (Accessed 19/02/2023).
- [38] TAIYO YUDEN, “Lithium Ion Capacitors: An Effective EDLC Replacement.” [Online]. Available: https://www.yuden.co.jp/include/english/solutions/lic/LIC_White_Paper_Final.pdf (Accessed 19/02/2023).

- [39] Tecate Group, “TPL-100/22X45F Datasheet.” [Online]. Available: https://www.tecategroup.com/products/data_sheet.php?i=TPL-100/22X45F (Accessed 19/02/2023).
- [40] NXP Semiconductors, “I2C-bus specification and user manual - UM10204.” [Online]. Available: <https://www.pololu.com/file/0J435/UM10204.pdf> (Accessed 09/12/2022).
- [41] Good Display, “DESPI-C02 Schematic.” [Online]. Available: https://www.e-paper-display.com/DESPI-C02_SCH%20V1.025e8.pdf (Accessed 09/12/2022).
- [42] Autodesk, “Fusion 360 — 3D CAD, CAM, CAE, & PCB Cloud-Based Software.” [Online]. Available: <https://www.autodesk.com/products/fusion-360/> (Accessed 12/12/2022).
- [43] Altium Limited, “Altium Designer - Software de diseño de PCB.” [Online]. Available: <https://www.altium.com/es/altium-designer/> (Accessed 09/12/2022).
- [44] Moko Blue, “MK02 Bluetooth Module Datasheet.” [Online]. Available: https://www.mokosmart.com/pdf/MK02_Bluetooth_Moduel_Datasheet-V1.1.20101501.pdf (Accessed 09/12/2022).
- [45] Uri Shaked, “The Smart Conference Badge We Almost Failed Shipping,” *Medium*, 2019. [Online]. Available: <https://urish.medium.com/the-smart-conference-badge-we-almost-failed-shipping-edb2b1ae85b6>
- [46] Sensirion, “Design-in Guide SCD4x CO₂ Sensor.” [Online]. Available: https://sensirion.com/media/documents/0D0C9129/623B1183/Sensirion_CO2_Sensors_SCD4x_design-in_guide.pdf (Accessed 09/12/2022).
- [47] Zephyr Project, “Zephyr Project.” [Online]. Available: <https://www.zephyrproject.org/> (Accessed 09/12/2022).
- [48] Zephyr Project, “Introduction – Zephyr Project Documentation.” [Online]. Available: <https://docs.zephyrproject.org/latest/introduction/index.html> (Accessed 28/01/2023).
- [49] Nordic Semiconductor, “nRF Connect SDK.” [Online]. Available: <https://www.nordicsemi.com/Products/Development-software/nRF-Connect-SDK> (Accessed 09/12/2022).
- [50] Nordic Semiconductor, “nRF Connect for VS Code.” [Online]. Available: <https://www.nordicsemi.com/Products/Development-tools/nrf-connect-for-vs-code> (Accessed 28/01/2023).
- [51] PlatformIO, “A professional collaborative platform for embedded development.” [Online]. Available: <https://platformio.org/> (Accessed 09/12/2022).
- [52] PlatformIO, “Nordic nRF52840-DK – PlatformIO v6.1 documentation.” [Online]. Available: https://docs.platformio.org/en/stable/boards/nordicnrf52/nrf52840_dk.html (Accessed 28/01/2023).
- [53] PlatformIO, ““platformio.ini” (Project Configuration File).” [Online]. Available: <https://docs.platformio.org/en/latest/projectconf/index.html> (Accessed 07/02/2023).
- [54] Zephyr Project, “Devicetree – Zephyr Project Documentation.” [Online]. Available: <https://docs.zephyrproject.org/latest/build/dts/index.html> (Accessed 30/01/2023).

- [55] The kernel development community, “Linux and the Devicetree – The Linux Kernel documentation.” [Online]. Available: <https://docs.kernel.org/devicetree/usage-model.html> (Accessed 28/01/2023).
- [56] Zephyr Project, “Configuration System (Kconfig) – Zephyr Project Documentation.” [Online]. Available: <https://docs.zephyrproject.org/latest/build/kconfig/index.html> (Accessed 30/01/2023).
- [57] The Linux Information Project, “Kernel Definition.” [Online]. Available: <http://www.linfo.org/kernel.html> (Accessed 30/01/2023).
- [58] The Pull-up Resistor Channel, “ESP32 on Zephyr OS: PWM, pinctrl, Timer [Part 4].” [Online]. Available: <https://www.youtube.com/watch?v=HTKgqJVZ-EI> (Accessed 30/01/2023).
- [59] Adafruit Learning System. GATT — Introduction to Bluetooth Low Energy. [Online]. Available: <https://learn.adafruit.com/introduction-to-bluetooth-low-energy/gatt> (Accessed 09/12/2022).
- [60] O’Reilly Media, “GATT (Services and Characteristics) - Getting Started with Bluetooth Low Energy.” [Online]. Available: <https://www.oreilly.com/library/view/getting-started-with/9781491900550/ch04.html> (Accessed 09/02/2023).
- [61] Bluetooth SIG, “Specifications – Bluetooth® Technology Website.” [Online]. Available: <https://www.bluetooth.com/specifications/specs/> (Accessed 31/01/2023).
- [62] Embedded Centric, “BLE profiles, services, characteristics, device roles and network topology.” [Online]. Available: <https://embeddedcentric.com/lesson-2-ble-profiles-services-characteristics-device-roles-and-network-topology/> (Accessed 31/01/2023).
- [63] Zephyr Project, “Bluetooth: Peripheral ESP – Zephyr Samples.” [Online]. Available: https://github.com/zephyrproject-rtos/zephyr/tree/main/samples/bluetooth/peripheral_esp (Accessed 31/01/2023).
- [64] TransparenTech LLC, “Online UUID Generator Tool.” [Online]. Available: <https://www.uuidgenerator.net/> (Accessed 02/02/2023).
- [65] Google Play, “nRF Connect for Mobile.” [Online]. Available: <https://play.google.com/store/apps/details?id=no.nordicsemi.android.mcp> (Accessed 09/12/2022).
- [66] Sensirion, “Sensirion Embedded I2C SCD4x Driver.” [Online]. Available: <https://github.com/Sensirion/embedded-i2c-scd4x> (Accessed 02/02/2023).
- [67] olikraus, “U8glib - Arduino Monochrom Graphics Library for LCDs and OLEDs.” [Online]. Available: <https://github.com/olikraus/u8glib> (Accessed 08/02/2023).
- [68] ZinggJM, “GxEPD2 - Arduino Display Library for SPI E-Paper Displays.” [Online]. Available: <https://github.com/ZinggJM/GxEPD2> (Accessed 08/02/2023).
- [69] LVGL LLC, “LVGL - Light and Versatile Embedded Graphics Library.” [Online]. Available: <https://lvgl.io/> (Accessed 08/02/2023).

- [70] HelTec Automation, “e-ink - Arduino library for E-Ink display.” [Online]. Available: <https://github.com/HelTecAutomation/e-ink> (Accessed 08/02/2023).
- [71] Good Display, “CODE-GDEH0154D67-20191012.” [Online]. Available: https://www.e-paper-display.com/download_detail/downloadsId%3d806.html (Accessed 07/02/2023).
- [72] jcontrerasf, “E-ink-display-Zephyr.” [Online]. Available: <https://github.com/jcontrerasf/E-ink-display-Zephyr> (Accessed 07/02/2023).
- [73] AEP - Asociación Española de Pediatría, “COVID-19: Un contaminante que flota en el aire.” [Online]. Available: <https://www.aeped.es/sites/default/files/covid19flotaenelaire212.pdf> (Accessed 04/03/2023).
- [74] ISP - Instituto de Salud Pública, “Ventilación en edificios modernos contexto COVID-19.” [Online]. Available: <https://www.ispch.cl/wp-content/uploads/2021/12/NT-VENTILACION-EN-EDIFICIOS-MODERNOS-CONTEXTO-COVID19.pdf> (Accessed 26/02/2023).
- [75] Sensirion, “embedded-i2c-sgp40.” [Online]. Available: <https://github.com/Sensirion/embedded-i2c-sgp40> (Accessed 09/02/2023).
- [76] Sensirion, “gas-index-algorithm.” [Online]. Available: <https://github.com/Sensirion/gas-index-algorithm> (Accessed 09/02/2023).
- [77] Bosch Sensortec, “BME68X Sensor API - Common Sensor API for the BME680 and BME688 sensors.” [Online]. Available: <https://github.com/boschsensortec/BME68x-Sensor-API> (Accessed 09/02/2023).
- [78] Bosch Sensortec, “BSEC - BME688 Software.” [Online]. Available: <https://www.bosch-sensortec.com/software-tools/software/bme688-software/> (Accessed 09/02/2023).
- [79] joric, “nrfmicro - Bootloader,” *GitHub Wiki*. [Online]. Available: <https://github.com/joric/nrfmicro/wiki/Bootloader> (Accessed 09/12/2022).
- [80] Sensirion, “What is Sensirion’s VOC Index?” [Online]. Available: https://sensirion.com/media/documents/02232963/6294E043/Info_Note_VOC_Index.pdf (Accessed 05/03/2023).
- [81] SOS electronic, “Forget about TVOC measuring. VOC index and SGP40 are coming to the scene.” [Online]. Available: <https://www.soselectronic.com/articles/sensirion/forget-about-tvoc-measuring-voc-index-and-sgp40-are-coming-to-the-scene-2581> (Accessed 22/02/2023).
- [82] Jetperch LLC, “Joulescope — DC Energy Analyzer Measures Current & Voltage.” [Online]. Available: <https://www.joulescope.com/> (Accessed 09/12/2022).
- [83] Jorge García, “Convertir una Black Pill en una Black Magic Probe.” [Online]. Available: https://jorgegarcia.dev.gitlab.io/post/mis_primeros_pasos_con_stm32_ii/ (Accessed 26/02/2023).
- [84] Nordic Semiconductor, “nRF Command Line Tools.” [Online]. Available: <https://www.nordicsemi.com/Products/Development-tools/nrf-command-line-tools/download> (Accessed 26/02/2023).

- [85] SEGGER, “J-Link Downloads.” [Online]. Available: <https://www.segger.com/downloads/jlink/> (Accessed 26/02/2023).
- [86] Paint.NET, “Paint.NET - Free Software for Digital Photo Editing.” [Online]. Available: <https://www.getpaint.net/> (Accessed 09/12/2022).
- [87] Good Display, “Image2Lcd v3.2.” [Online]. Available: https://www.e-paper-display.com/download_detail/downloadsId=625.html (Accessed 09/12/2022).
- [88] Silicon Labs, “CP210x USB to UART Bridge VCP Drivers.” [Online]. Available: <https://www.silabs.com/developers/usb-to-uart-bridge-vcp-drivers?tab=downloads> (Accessed 28/01/2023).