

**Universidad Tecnológica Nacional**

Proyecto Final

---

Monitoreo y presentación digital de  
calidad de aire mediante detección de  
CO, HCHO, CO<sub>2</sub> y Partículas

---

*Autores:*

- *Meza, Lautaro Marcelo – lautaromeza@alu.frp.utn.edu.ar*
- *Ré, Boris Nahuel – borisre@alu.frp.utn.edu.ar*

*Director:*

- *Mg. Ing. José María Triano – jmtriano@bioingenieria.edu.ar*

*Proyecto final presentado para cumplimentar los requisitos académicos  
para acceder al título de Ingeniero en Electrónica.*

*en la*

**Facultad Regional Paraná**

Junio de 2025



## **Declaración de autoría:**

Nosotros declaramos que el Proyecto Final “Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas” y el trabajo realizado son propios. Declaramos:

- Este trabajo fue realizado en su totalidad, o principalmente, para acceder al título de grado de Ingeniero en Electrónica en la Universidad Tecnológica Nacional, Regional Paraná.
- Se establece claramente que el desarrollo realizado y el informe que lo acompaña no han sido previamente utilizados para acceder a otro título de grado o pregrado.
- Siempre que se ha utilizado trabajo de otros autores, el mismo ha sido correctamente citado. El resto del trabajo es de autoría propia.
- Se ha indicado y agradecido correctamente a todos aquellos que han colaborado con el presente trabajo.
- Cuando el trabajo forma parte de un trabajo de mayores dimensiones donde han participado otras personas, se ha indicado claramente el alcance del trabajo realizado.

Firmas:

- 
- 
- 

Fecha:

## Agradecimientos:

Al Magister Ingeniero José María Triano por habernos brindado su tiempo y conocimiento además de su predisposición en todo momento. A nuestra UTN Facultad Regional Paraná y su personal docente por brindarnos el espacio y formación profesional. A Alejandro Escobar por brindarnos los elementos 3D de nuestro proyecto.

Estoy profundamente agradecido a mis padres que me brindaron su ayuda y apoyo a lo largo de toda mi carrera. A mi novia por acompañarme y apoyarme siempre. A mis compañeros de facultad y entendidos del tema que brindaron su valioso tiempo para ayudarme, aportar con sus ideas y conocimiento.

Meza, Lautaro Marcelo

Agradezco a mi familia, especialmente a mis padres y hermanos, que me apoyaron y confiaron en mí. También agradezco a mi compañero de proyecto, a mis tíos, a mis abuelos y a mis amigos, quienes me ayudaron a lo largo de esta carrera. Por último, agradezco mis compañeros y profesores que compartieron sus ideas.

Ré, Boris Nahuel

Universidad Tecnológica Nacional

## *Abstract*

Facultad Regional Paraná

Ingeniero en Electrónica

# **Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO<sub>2</sub> y partículas.**

Meza, Lautaro Marcelo

Ré, Boris Nahuel

### **Abstract:**

*This project presents the design and development of a monitoring and digital display device for air quality, detecting CO, HCHO, CO<sub>2</sub>, and suspended particles. The main objective is to raise awareness about the air quality in the user's environment. For this purpose, Winsen sensors were employed, along with an ESP32 microcontroller for sensor management and IoT support, a touchscreen for local visualization, and a mobile application for remote monitoring of the data.*

*The microcontroller code has been developed in C and C++, while the mobile application was built using the Flutter framework, with code written in Dart, Java, and XML. The device can display real-time variables such as humidity, temperature, carbon monoxide, carbon dioxide, formal-dehyde, and PM<sub>10</sub> and PM<sub>2.5</sub> particles. Additionally, it can record these values over time and notify users through audible alarms and mobile app notifications.*

### **Keywords:**

*Air Quality, Monitoring, sensor, ESP32, Flutter, Firebase, Arduino UNO, JSON, particulate matter, gases.*

## **Resumen:**

*Este proyecto presenta el diseño y desarrollo de un dispositivo de monitoreo y visualización digital de la calidad del aire, mediante la detección de CO, HCHO, CO2 y partículas en sus-pensión. El objetivo principal es concientizar a las personas sobre la calidad del aire en su entorno. Para ello, se emplearon sensores Winsen, un microcontrolador ESP32 para la gestión de los sensores y soporte IoT, una pantalla táctil para la visualización local, y una aplicación móvil para el monitoreo remoto de los datos.*

*El desarrollo del código del microcontrolador se realizó en los lenguajes C y C++, mientras que la aplicación móvil se construyó utilizando el framework Flutter, con código en Dart, Java y XML. El dispositivo tiene la capacidad de mostrar en tiempo real variables como la humedad, temperatura, monóxido de carbono, dióxido de carbono, formaldehído, y partículas PM10 y PM2.5. Además, puede registrar estos datos a lo largo del tiempo y notificar a los usuarios mediante alarmas sonoras y notificaciones en la aplicación móvil.*

## **Palabras Clave:**

*Calidad del aire, Monitoreo, Sensores, ESP32, Flutter, Firebase, Arduino UNO, JSON, gases, material particulado.*

# Índice:

<b>CAPITULO 1: INTRODUCCIÓN.....</b>	<b>1</b>
1.1. FUNDAMENTACIÓN .....	1
1.2. DESCRIPCIÓN GENERAL.....	1
1.3. PRUEBA DE CONCEPTO.....	2
1.4. ESTADO DEL MERCADO.....	3
<b>CAPITULO 2: MARCO TEÓRICO.....</b>	<b>4</b>
2.1. CONTAMINACIÓN DEL AIRE .....	4
2.1.1. <i>Contaminantes</i> .....	4
2.1.2. <i>Valores máximos permisibles de contaminantes y sus efectos en la salud</i>	6
2.2. SENSORES .....	9
2.2.1. <i>Sensores electroquímicos</i> .....	9
2.2.2. <i>Sensores infrarrojos no dispersivos</i> .....	10
2.2.3. <i>Sensores de láser dispersivo</i> .....	11
<b>CAPITULO 3: DESARROLLO .....</b>	<b>13</b>
3.1. ARQUITECTURA .....	14
3.2. DISPOSITIVOS.....	15
3.2.1. <i>Sensor de Monóxido de carbono</i> .....	15
3.2.2. <i>Sensor de dióxido de carbono</i> .....	20
3.2.3. <i>Sensor de partículas</i> .....	25
3.2.4. <i>Sensor de Formaldehido</i> .....	30
3.2.5. <i>Sensor de humedad y temperatura</i> .....	33
3.2.6. <i>ESP32</i> .....	35
3.2.7. <i>Arduino UNO</i> .....	39
3.2.8. <i>Pantalla touch</i> .....	41
3.3. ADQUISICIÓN DE DATOS .....	42
3.3.1. <i>Lectura por ADC</i> .....	43
3.3.2. <i>Lectura por puerto serie UART</i> .....	44
3.4. VISUALIZACIÓN DE DATOS .....	48
3.4.1. <i>Transmisión de datos</i> .....	49
3.4.2. <i>Diseño de interfaz gráfica</i> .....	51
3.5. GESTIÓN DE DATOS.....	54
3.5.1. <i>Configuración del Timer</i> .....	57
3.6. COMUNICACIÓN CON BASE DE DATOS.....	57
3.6.1. <i>Establecer conexión</i> .....	59
3.6.2. <i>Lectura y escritura de registros</i> .....	61
3.7. APLICACIÓN MÓVIL .....	61
3.7.1. <i>Desarrollo de la aplicación</i> .....	62
3.8. DISEÑO DE PLACA .....	69
3.9. DISEÑO 3D.....	74
3.9.1. <i>Habitáculo</i> .....	74

3.9.2.	<i>Bases de los dispositivos interiores</i>	77
3.9.3.	<i>Diseño 3D completo</i>	80
3.10.	INTEGRACIÓN DE LAS PARTES	80
<b>CAPITULO 4: RESULTADOS</b>		<b>86</b>
4.1.	ANÁLISIS DE CONSUMO ELÉCTRICO	86
4.2.	PRUEBAS DE CAMPO	87
4.2.1.	<i>Descripción de las pruebas</i>	88
4.2.2.	<i>Descripción de los elementos</i>	88
4.2.3.	<i>CO<sub>2</sub>-Reacción vinagre bicarbonato</i>	89
4.2.4.	<i>CO – Combustión incompleta de papel y humo de cigarrillo</i>	101
4.2.5.	<i>Resultados</i>	102
4.2.6.	<i>Conclusiones de las pruebas</i>	109
<b>CAPITULO 5: ANÁLISIS DE COSTOS</b>		<b>111</b>
5.1.	ANÁLISIS	111
<b>CAPITULO 6: DISCUSIÓN Y CONCLUSIÓN</b>		<b>114</b>
6.1.	MEJORAS	115
<b>CAPITULO 7: BIBLIOGRAFÍA</b>		<b>116</b>
<b>CAPITULO 8: ANEXO</b>		<b>120</b>
8.1.	TABLA DE DATOS DE CURVA MQ7	120
8.2.	CODIGO APLICACIÓN MÓVIL	120
8.3.	CODIGO MICROCONTROLADOR ESP32	121
8.4.	ESP32 – CARACTERÍSTICAS UTILIZADAS	121
8.5.	FLUTTER	129
8.5.1.	<i>Anatomía de una aplicación</i>	130
8.5.2.	<i>Desarrollo de la aplicación</i>	131
8.6.	PANTALLA TOUCH	136
8.7.	ESTRUCTURA DE UN ARCHIVO JSON	138

## Figuras:

FIG. N°1 - EL IMPACTO EN EL SISTEMA RESPIRATORIO SEGÚN SU DIÁMETRO [2].	5
FIG. N°2 - ESTRUCTURA INTERNA DE UN SENSOR ELECTROQUÍMICO [9]	10
FIG. N°3 - PRINCIPIO DE FUNCIONAMIENTO SENSORES NDIR [12]	11
FIG. N°4 - PRINCIPIO DE LASER DISPERSIVO [9]	12
FIG. N°5 - DIAGRAMA EN BLOQUES GENERALIZADO – ELABORACIÓN PROPIA	13
FIG. N°6 - DIAGRAMA EN BLOQUES AMPLIADO- ELABORACIÓN PROPIA	14
FIG. N°7 - MQ7-GAS SENSOR [18]	15
FIG. N°8 - MQ-CIRCUITO INTERNO [17]	16
FIG. N°9 - CIRCUITO DEL SENSOR SIMPLIFICADO [18]	16
FIG. N°10 - CURVA CARACTERÍSTICA – SENSOR MQ7 [17]	17
FIG. N°11 - ANÁLISIS DE GRÁFICA – WEB PLOT DIGITIZER [20]	18
FIG. N°12 - TABLA DE DATOS DE LA CURVA MQ7 – VER ANEXO 8.1	18
FIG. N°13 - SENSOR DE DIÓXIDO DE CARBONO MHZ-19	21
FIG. N°14 - MHZ-19 HEADERS PINOUT	21
FIG. N°15 - MHZ-19 CONEXIÓN POR CABLES PINOUT	21
FIG. N°16 - SEÑALES DE EJEMPLO DE LECTURA POR PWM	22
FIG. N°17 - MHZ-19 COMANDO LEER CONCENTRACIÓN [21]	24
FIG. N°18 - MHZ-19 COMANDO AUTO CALIBRACIÓN ON/OFF [21]	24
FIG. N°19 - MHZ-19 COMANDO CONFIGURAR RANGO DE DETECCIÓN [21]	25
FIG. N°20 - ZH06 – PRINCIPIO DE FUNCIONAMIENTO [22]	25
FIG. N°21 - ZH06 - PINOUT [22]	26
FIG. N°22 - ZH06- EJEMPLO DE LECTURA POR PWM [22]	27
FIG. N°23 - ZH06- TRAMA EN MODO ACTIVO [22]	28
FIG. N°24 - ZH06- TRAMA DE PETICIÓN DE DATOS [22]	29
FIG. N°25 - ZH06- TRAMA DE RESPUESTA DE DATOS [22]	29
FIG. N°26 - ZH06 – TRAMA DE CAMBIO A MODO QA [22]	30
FIG. N°27 - ZH06 – TRAMA DE CAMBIO A MODO ACTIVO [22]	30
FIG. N°28 - MÓDULO SENSOR ZP16	30
FIG. N°29 - DEFINICIÓN DE PINES ZP16 [23]	30
FIG. N°30 - TRAMA CAMBIO DE ACTUALIZACIÓN ACTIVA A Q&A – ZP16 [23]	31
FIG. N°31 - TRAMA CAMBIO DE Q&A A ACTUALIZACIÓN ACTIVA – ZP16 [23]	31
FIG. N°32 - TRAMA PETICIÓN DE DATOS ZP16 [23]	32
FIG. N°33 - TRAMA RESPUESTA DE DATOS ZP16 [23]	32
FIG. N°34 - DHT11 PINOUT [24]	33
FIG. N°35 - DIAGRAMA DE CIRCUITO- DHT11 [24]	34
FIG. N°36 - SEÑAL DIGITAL DHT11 [24]	34
FIG. N°37 - ESP32-DEVKIT [26]	35
FIG. N°38 - ESP32- DIAGRAMA EN BLOQUES FUNCIONAL [25]	36
FIG. N°39 - ESP32-DEVKIT PINOUT [14]	36
FIG. N°40 - ARDUINO UNO – PINOUT [15]	40
FIG. N°41 - ARDUINO UNO Y ESP32 CON TXB0108[FUENTE PROPIA]	40
FIG. N°42 - PANTALLA ILI9486 [28]	41
FIG. N°43 - REVERSO DE PANTALLA TFT [29]	42
FIG. N°44 - PANTALLA TOUCH – PINOUT [29]	42

FIG. N°45 - ADQUISICIÓN DE DATOS ADC- DIAGRAMA DE FLUJO .....	43
FIG. N°46 - ADQUISICIÓN DE DATOS UART- DIAGRAMA DE FLUJO.....	44
FIG. N°47 - SENSOR MHZ-19 [21] .....	45
FIG. N°48 - CONVERTOR DE NIVELES LÓGICOS TXS0108 [31] .....	47
FIG. N°49 - MULTIPLEXOR CD4051 [9].....	48
FIG. N°50 - VISUALIZACIÓN DE DATOS – DIAGRAMA DE FLUJO[FUENTE PROPIA].....	50
FIG. N°51 - PREDISEÑO DE INTERFAZ[FUENTE PROPIA] .....	51
FIG. N°52 - BIENVENIDA – INTERFAZ GRÁFICA[FUENTE PROPIA] .....	52
FIG. N°53 - PANTALLA CARGADA EN ESPERA DE DATOS.....	53
FIG. N°54 - PANTALLA MOSTRANDO DATOS RECIBIDOS[FUENTE PROPIA] .....	54
FIG. N°55 - LOOP PRINCIPAL- DIAGRAMA DE FLUJO[FUENTE PROPIA] .....	56
FIG. N°56 - CONFIGURACIÓN DEL TIMER- DIAGRAMA DE FLUJO [ELABORACIÓN PROPIA]...	57
FIG. N°57 - DIAGRAMA ESP32-APLICACIÓN MÓVIL[FUENTE PROPIA] .....	58
FIG. N°58 - CAPTURA DE RTBD – DATOS EN TIEMPO REAL[FUENTE PROPIA] .....	58
FIG. N°59 - CAPTURA DE RTBD – REGISTROS HISTÓRICOS[FUENTE PROPIA].....	59
FIG. N°60 - DIAGRAMA DE FLUJO – CONEXIÓN A BASE DE DATOS [FUENTE PROPIA].....	60
FIG. N°61 - APLICACIÓN MÓVIL – ORGANIZACIÓN DE PÁGINAS[FUENTE PROPIA].....	63
FIG. N°62 - APLICACIÓN MÓVIL- PANTALLA DE INICIO [FUENTE PROPIA] .....	64
FIG. N°63 - APLICACIÓN MÓVIL – INICIO DE SESIÓN[FUENTE PROPIA] .....	65
FIG. N°64 - APLICACIÓN MÓVIL – CAPTURAS DE PÁGINA HOME[FUENTE PROPIA] .....	66
FIG. N°65 - APLICACIÓN MÓVIL- PÁGINA REGISTRO [FUENTE PROPIA].....	67
FIG. N°66 - APLICACIÓN MÓVIL – PÁGINA INFORMACIÓN [FUENTE PROPIA] .....	68
FIG. N°67 - APLICACIÓN MÓVIL – PÁGINA ACERCA DE [FUENTE PROPIA] .....	69
FIG. N°68 - DISEÑO DE PCB – DIAGRAMA CIRCUITAL [FUENTE PROPIA] .....	69
FIG. N°69 - DISEÑO PCB- CONECTORES A PERIFÉRICOS [FUENTE PROPIA].....	70
FIG. N°70 - DISEÑO PCB – CONEXIONADO CIRCUITOS INTEGRADOS [FUENTE PROPIA]....	71
FIG. N°71 - DISEÑO PCB- LAYOUT COMPLETO[FUENTE PROPIA] .....	72
FIG. N°72 - DISEÑO PCB – PLACA IMPRESA [FUENTE PROPIA] .....	73
FIG. N°73 - DISEÑO PCB- PLACA FINALIZADA TOP [FUENTE PROPIA].....	73
FIG. N°74 - DISEÑO PCB – PLACA FINALIZADA BOTTOM [FUENTE PROPIA] .....	74
FIG. N°75 - DISEÑO 3D – HABITÁCULO INFERIOR[AUTORÍA PROPIA].....	75
FIG. N°76 - DISEÑO 3D – ESQUINERO + PLACA CON LOGO UTN[AUTORÍA PROPIA].....	75
FIG. N°77 - DISEÑO 3D – HABITÁCULO SUP., CARCASA EXTERIOR. [AUTORÍA PROPIA]....	76
FIG. N°78 - DISEÑO 3D – HABITÁCULO SUPERIOR, BASE SENSORES [AUTORÍA PROPIA]...	77
FIG. N°79 - DISEÑO 3D – HABITÁCULO SUPERIOR – TAPA [AUTORÍA PROPIA].....	77
FIG. N°80 - DISEÑO 3D- CARCASA DE SENSORES 1 [AUTORÍA PROPIA] .....	78
FIG. N°81 - DISEÑO 3D – CARCASA DE SENSORES 2 [AUTORÍA PROPIA] .....	78
FIG. N°82 - DISEÑO 3D- BASE PARA ARDUINO UNO [AUTORÍA PROPIA].....	79
FIG. N°83 - DISEÑO 3D- CALCES PARA BASE ARDUINO UNO [AUTORÍA PROPIA].....	79
FIG. N°84 - DISEÑO 3D – VIGA SOPORTE DE BASE DE SENSORES. [AUTORÍA PROPIA] ....	79
FIG. N°85 - DISEÑO 3D – PARTES PRINCIPALES [AUTORÍA PROPIA] .....	80
FIG. N°86 - DEBUG – INICIO DE SISTEMAS [FUENTE PROPIA] .....	81
FIG. N°87 - DEBUG – LECTURA DE DATOS [FUENTE PROPIA] .....	82
FIG. N°88 - DEBUG – COMUNICACIÓN ARDUINO – ESP EXITOSA [FUENTE PROPIA] .....	82
FIG. N°89 - ARMADO DE SISTEMA – BASE [AUTORÍA PROPIA] .....	83
FIG. N°90 - ARMADO DE SISTEMA – SENSORES [AUTORÍA PROPIA] .....	84
FIG. N°91 - ARMADO DE SISTEMA – TAPA [FUENTE PROPIA] .....	84

FIG. N°92 - SISTEMA ARMADO COMPLETO CON CABLES PARA DEBUG [FUENTE PROPIA] ...	85
FIG. N°93 - SISTEMA ARMADO COMPLETO FINAL [FUENTE PROPIA].....	85
FIG. N°94 - PRUEBA DE CONSUMO ELÉCTRICO. [FUENTE PROPIA] .....	87
FIG. N°95 - PRUEBAS DE CAMPO – HABITÁCULO CONTENEDOR [FUENTE PROPIA] .....	88
FIG. N°96 - PRUEBAS DE CAMPO – HABITÁCULO DE REACCIÓN. [FUENTE PROPIA] .....	89
FIG. N°97 - PRUEBA DE SATURACIÓN – GRAFICA DE DATOS.....	102
FIG. N°98 - PRUEBA UNO DE ESTIMACIÓN – GRAFICA PRIMER ENSAYO .....	103
FIG. N°99 - PRUEBA UNO DE ESTIMACIÓN – GRAFICA SEGUNDO ENSAYO .....	104
FIG. N°100 - PRUEBA DOS DE ESTIMACIÓN – GRAFICA DE PRIMER ENSAYO .....	105
FIG. N°101 - PRUEBA DOS DE ESTIMACIÓN – GRAFICA DE SEGUNDO ENSAYO.....	106
FIG. N°102 - PRUEBA DE PAPEL SERVILLETA – GRAFICA DE CONCENTRACIÓN PPM .....	107
FIG. N°103 - PRUEBA DE PAPEL SERVILLETA – GRAFICA DE VOLTAJE LEÍDO .....	107
FIG. N°104 - PRUEBA DE CIGARRILLO – GRAFICA DE CONCENTRACIÓN PPM .....	108
FIG. N°105 - PRUEBA DE CIGARRILLO – GRAFICA DE VOLTAJE LEÍDO .....	109
FIG. N°106 - ESP32-DEVKIT [26] .....	121
FIG. N°107 - ESP32- FUNCIONAL BLOCK DIAGRAM [25] .....	122
FIG. N°108 - ESP32-DEVKIT PINOUT [14] .....	122
FIG. N°109 - ESP32- SAR ADC CONTROLLERS [14] .....	123
FIG. N°110 - ESP32- CARACTERÍSTICAS ADC [14] .....	124
FIG. N°111 - ESP32- ERROR DEL ADC [14].....	124
FIG. N°112 - ESP32-UART TRAMA DE DATOS [14] .....	125
FIG. N°113 - ESP32- UART CONTROLLER [14].....	125
FIG. N°114 - ARQUITECTURA DE FLUTTER [39].....	130
FIG. N°115 - ANATOMÍA DE UNA APLICACIÓN - FLUTTER [39] .....	131
FIG. N°116 - DESARROLLO DE APLICACIÓN – CONEXIÓN A FIREBASE[FUENTE PROPIA]....	132
FIG. N°117 - APLICACIÓN MÓVIL - VALIDACIÓN CON FIREBASE[FUENTE PROPIA].....	133
FIG. N°118 - APLICACIÓN MÓVIL- DECLARACIÓN DE PÁGINA HOME[FUENTE PROPIA] .....	134
FIG. N°119 - APLICACIÓN MÓVIL – RUTAS A LAS PÁGINAS[FUENTE PROPIA] .....	135
FIG. N°120 - CODIGO ARDUINO – PARÁMETROS DE CALIBRACIÓN [FUENTE PROPIA] .....	137
FIG. N°121 - CÓDIGO ARDUINO – ESTRUCTURA CLAVE/VALOR [FUENTE PROPIA].....	138

## Ecuaciones:

ECUACIÓN N°.1 - FÓRMULA PARA PASAR DE MG/M <sup>3</sup> A PPM. [7] .....	7
ECUACIÓN N°.2 - CÁLCULO DE RS – PLANTEO DE PRIMERA ECUACIÓN [12].....	17
ECUACIÓN N°.3 - CÁLCULO DE RS- DESPEJE FINAL [12].....	17
ECUACIÓN N°.4 - PRIMERA APROXIMACIÓN LOGARÍTMICA DE MQ7.....	18
ECUACIÓN N°.5 - ECUACIÓN LOGARÍTMICA DE MQ7 .....	19
ECUACIÓN N°.6 - EXPRESIÓN DE RS .....	19
ECUACIÓN N°.7 - OBTENCIÓN DE VRL.....	19
ECUACIÓN N°.8 - CO- CÁLCULO DE CONCENTRACIÓN EN PPM.....	20
ECUACIÓN N°.9 - CÁLCULO DE R0 – MQ7 .....	20
ECUACIÓN N°.10 -CONCENTRACIÓN DE CO2 POR PWM.....	22
ECUACIÓN N°.11 -ZH06- CÁLCULO DE CONCENTRACIÓN PM2.5 POR PWM.....	27
ECUACIÓN N°.12 -LECTURA DE HCHO.....	32
ECUACIÓN N°.13 -ZP16 - PASO DE MG/M3 A PPM.....	33
ECUACIÓN N°.14 -ESP32-TIMER, TIEMPO DE INTERRUPTIÓN .....	38
ECUACIÓN N°.15 -REACCIÓN DE VINAGRE CON BICARBONATO – ECUACIÓN ESTEQUIOMÉTRICA. [5].....	89
ECUACIÓN N°.16 -EJEMPLO – CALCULO DE GRAMOS DE BICARBONATO .....	90
ECUACIÓN N°.17 -EJEMPLO – CALCULO DE GRAMOS DE VINAGRE .....	90
ECUACIÓN N°.18 -EJEMPLO – AJUSTE DE CANTIDAD DE VINAGRE.....	90
ECUACIÓN N°.19 -PRUEBA DE SATURACIÓN – CALCULO DE CONCENTRACIÓN gm <sup>3</sup> .....	91
ECUACIÓN N°.20 -PRUEBA DE SATURACIÓN – CALCULO DE MASA CO <sub>2</sub> .....	91
ECUACIÓN N°.21 -PRUEBA DE SATURACIÓN – MOLES DE BICARBONATO .....	91
ECUACIÓN N°.22 -PRUEBA DE SATURACIÓN – MOLES DE ÁCIDO ACÉTICO .....	91
ECUACIÓN N°.23 -PRUEBA DE SATURACIÓN – GRAMOS DE CO <sub>2</sub> .....	92
ECUACIÓN N°.24 -PRUEBA DE SATURACIÓN – CONCENTRACIÓN DE CO <sub>2</sub> EN gm <sup>3</sup> .....	92
ECUACIÓN N°.25 -PRUEBA DE SATURACIÓN – CONCENTRACIÓN DE CO <sub>2</sub> EN [PPM] .....	92
ECUACIÓN N°.26 -PRUEBA DE ESTIMACIÓN 2000PPM – CÁLCULO DE CONCENTRACIÓN CO <sub>2</sub> gm <sup>3</sup> 93	
ECUACIÓN N°.27 - PRUEBA DE ESTIMACIÓN 2000 PPM – CALCULO DE CONCENTRACIÓN CO <sub>2</sub> gm <sup>3</sup> 93	
ECUACIÓN N°.28 -PRUEBA DE ESTIMACIÓN 2000PPM – CALCULO DE MASA CO <sub>2</sub> .....	93
ECUACIÓN N°.29 -PRUEBA DE ESTIMACIÓN 2000PPM – CALCULO DE MOLES CO <sub>2</sub> .....	93
ECUACIÓN N°.30 -PRUEBA DE ESTIMACIÓN 2000PPM – CALCULO DE MASA DE ÁCIDO ACÉTICO 94	
ECUACIÓN N°.31 -PRUEBA DE ESTIMACIÓN 2000PPM – CÁLCULO DE MASA DE VINAGRE....	94
ECUACIÓN N°.32 -PRUEBA DE ESTIMACIÓN 2000PPM – CÁLCULO DE MASA DE BICARBONATO DE SODIO 94	
ECUACIÓN N°.33 -PRIMER ENSAYO 2000PPM – MOLES BICARBONATO.....	94
ECUACIÓN N°.34 -PRIMER ENSAYO 2000PPM – MOLES VINAGRE .....	94
ECUACIÓN N°.35 -PRIMER ENSAYO 2000PPM – MASA CO <sub>2</sub> GENERADA.....	94
ECUACIÓN N°.36 -PRIMER ENSAYO 2000PPM – CONCENTRACIÓN mgm <sup>3</sup> CO <sub>2</sub> GENERADA	95
ECUACIÓN N°.37 -PRIMER ENSAYO 2000PPM – CONCENTRACIÓN PPM GENERADOS .....	95
ECUACIÓN N°.38 -SEGUNDO ENSAYO 2000PPM – MOLES DE BICARBONATO .....	95
ECUACIÓN N°.39 -SEGUNDO ENSAYO 2000PPM – MOLES DE ÁCIDO ACÉTICO.....	95

ECUACIÓN N°.40 -SEGUNDO ENSAYO 2000PPM – MASA DE CO <sub>2</sub> .....	95
ECUACIÓN N°.41 -SEGUNDO ENSAYO 2000PPM - CONCENTRACIÓN CO <sub>2</sub> GENERADA.....	96
ECUACIÓN N°.42 -SEGUNDO ENSAYO 2000PPM – CONCENTRACIÓN DE CO <sub>2</sub> GENERADA .	96
ECUACIÓN N°.43 -ECUACIÓN DE LOS GASES IDEALES [5].....	96
ECUACIÓN N°.44 -FORMULA - CALCULO DE VOLUMEN MOLAR .....	96
ECUACIÓN N°.45 -PRIMER ENSAYO 2000 PPM - AJUSTE DE VOLUMEN MOLAR SEGÚN CONDICIONES .....	97
ECUACIÓN N°.46 -PRIMER ENSAYO 2000 PPM – AJUSTE DE CONCENTRACIÓN <i>mgm</i> <sup>3</sup> CO <sub>2</sub> GENERADA	97
ECUACIÓN N°.47 -PRIMER ENSAYO 2000 PPM – AJUSTE DE CONCENTRACIÓN PPM CO <sub>2</sub> GENERADA	97
ECUACIÓN N°.48 -PRUEBA DE ESTIMACIÓN 4000 PPM- CONCENTRACIÓN <i>mgm</i> <sup>3</sup> CO <sub>2</sub> GENERADA	97
ECUACIÓN N°.49 -PRUEBA DE ESTIMACIÓN 4000 PPM- MASA DE CO <sub>2</sub> GENERADA .....	98
ECUACIÓN N°.50 -PRUEBA DE ESTIMACIÓN 4000 PPM – MOLES DE ÁCIDO ACÉTICO.....	98
ECUACIÓN N°.51 -PRUEBA DE ESTIMACIÓN 4000 PPM – MASA DE ÁCIDO ACÉTICO .....	98
ECUACIÓN N°.52 -PRUEBA DE ESTIMACIÓN 4000 PPM – MASA DE VINAGRE .....	98
ECUACIÓN N°.53 -PRUEBA DE ESTIMACIÓN 4000 PPM – MASA DE BICARBONATO .....	98
ECUACIÓN N°.54 -PRIMER ENSAYO 4000 PPM- MOLES DE BICARBONATO.....	99
ECUACIÓN N°.55 -PRIMER ENSAYO 4000 PPM- MOLES DE VINAGRE .....	99
ECUACIÓN N°.56 -PRIMER ENSAYO 4000 PPM- MASA DE CO <sub>2</sub> CALCULADA .....	99
ECUACIÓN N°.57 -PRIMER ENSAYO 4000PPM- CONCENTRACIÓN CO <sub>2</sub> CALCULADA.....	99
ECUACIÓN N°.58 -PRIMER ENSAYO 4000PPM- CONCENTRACIÓN CO <sub>2</sub> [PPM] CALCULADA..	99
ECUACIÓN N°.59 -SEGUNDO ENSAYO 4000PPM- MOLES BICARBONATO .....	99
ECUACIÓN N°.60 -SEGUNDO ENSAYO 4000PPM – MOLES DE VINAGRE.....	100
ECUACIÓN N°.61 -SEGUNDO ENSAYO 4000PPM – MASA DE CO <sub>2</sub> GENERADA .....	100
ECUACIÓN N°.62 -SEGUNDO ENSAYO 4000PPM – CONCENTRACIÓN DE CO <sub>2</sub> GENERADA	100
ECUACIÓN N°.63 -PRIMER ENSAYO 4000PPM – AJUSTE DE VOLUMEN MOLAR .....	100
ECUACIÓN N°.64 -PRIMER ENSAYO 4000PPM – AJUSTE DE CONCENTRACIÓN CO <sub>2</sub> .....	101
ECUACIÓN N°.65 -PRIMER ENSAYO 4000PPM – AJUSTE DE CONCENTRACIÓN PPM CO <sub>2</sub> .	101

## Tablas:

TABLA N°1 - VALORES MÁXIMOS [5] .....	8
TABLA N°2 - VALORES MÁXIMOS - PARTÍCULAS EN SUSPENSIÓN. [5] .....	9
TABLA N°3 - MHZ-19 CONFIGURACIÓN DE PUERTO SERIE [21] .....	23
TABLA N°4 - ZH06 - CONFIGURACIÓN DE PUERTO SERIE [22] .....	27
TABLA N°5 - ZP16- CONFIGURACIÓN DE PUERTO SERIE [23].....	31
TABLA N°6 - ESP32- ASIGNACIÓN DE UARTs [FUENTE PROPIA].....	37
TABLA N°7 - CD4051 SELECCIÓN DE CANAL [9].....	48
TABLA N°8 - CD4051- ASIGNACIÓN DE CANALES.....	48
TABLA N°9 - CONSUMO EN AMPER DE DISPOSITIVOS .....	86
TABLA N°10 -COSTOS FIJOS.....	111
TABLA N°11 -COSTO HORA HOMBRE .....	112
TABLA N°12 -DETALLE HORAS TRABAJADAS .....	112
TABLA N°13 -COSTO NETO MANO DE OBRA.....	112
TABLA N°14 -PRESUPUESTO TOTAL .....	113

## Lista de Abreviaciones y Símbolos

- CO<sub>2</sub>** Dióxido de carbono 1-3, 6, 11, 21, 23, 45, 51, 63, 90-107, 111, 116.
- CO** Monóxido de carbono 1-2, 6, 20, 44-45, 51, 63, 90, 103, 109-111, 116
- HCHO** Formaldehído 1, 34, 45, 48, 51, 63.
- PM 1.0** Material particulado menor o igual a 1  $\mu\text{m}$  1, 4, 9, 26, 30.
- PM 2.5** Material particulado menor o igual a 2.5  $\mu\text{m}$  1, 4, 9, 26-28, 30, 45, 47, 51, 63.
- PM 10** Material particulado menor o igual a 10  $\mu\text{m}$  1, 4, 9, 26, 30, 45, 47, 51, 63.
- MP** Material particulado 4.
- PST** Partículas suspendidas totales 4.
- IoT** Internet of Things (Internet de las Cosas) 2, 58, 117.
- JSON** Notación de objetos de JavaScript 51, 53, 55, 62.
- ADC** Convertidor analógico digital 20, 23, 38, 44-45.
- UART** Transmisor-Receptor asíncrono universal 14, 21, 24, 28, 32, 38-39, 42, 45-46, 48-49, 51, 71.
- PWM** Modulación por ancho de pulso 21, 23-24, 27-28, 40.
- DAC** Convertidor Digital-Analógico 21, 23, 37.
- CAPC** Concentración Admisible para Periodos Cortos 7, 9.
- CAPL** Concentración Admisible para Períodos Largos 7, 9.
- PPM** Partes por millón 6-7, 16, 18-21, 23-26, 34, 47-48, 90, 92-112, 116.
- $\mu$  Millonésima parte de una magnitud 4, 26-27.
- CPU** Unidad Central de Procesamiento 37.
- GPIOs** Entrada/Salidas de propósito general 37.
- SAR-ADC** Registro de aproximaciones sucesivas ADC 38.
- RTC** Reloj de tiempo real 37, 40.
- PC** Computadora personal 38.
- E/S** Entrada/Salida 40.
- RS232** Estándar de comunicación serie 24, 28.
- PCB** Placa de Circuito Impreso 40, 70-78, 113-114, 117.
- TFT** tecnología de pantalla 42-43.

**RTBD** Base de datos en tiempo real 59-60.

**VSCo** Visual Studio Code(aplicación de desarrollo de la aplicación) 63.


**ARS** Moneda Argentina 113-115.

**USD** Moneda Norteamericana 113-115.

**NDIR** Infrarrojo no dispersivo 10-11, 21.

## **Dedicado a:**

Nuestras familias, amigos y profesores por su apoyo incondicional a lo largo de todos estos años.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	1. Introducción

## Capítulo 1: Introducción

El aire es la mezcla homogénea de gases que componen la atmósfera, principalmente compuesto por un 20,94% de oxígeno, 78,08% de nitrógeno, 0,93% de argón, 0,035% de dióxido de carbono (CO<sub>2</sub>) y 0,40% de vapor de agua [1]. Es uno de los principales elementos de la naturaleza y tal vez, el menos tenido en cuenta por el individuo común. Esta composición del **aire** se replica en cualquier ambiente que cuente con la ventilación adecuada, en general existen otros gases y partículas presentes en los espacios comunes de una ciudad o determinados ambientes, cuando estos gases y partículas “*extras*” componen la mezcla o bien, cuando un gas que compone la mezcla homogénea supera cierto nivel nos encontramos frente a una contaminación del aire. El agregado de partículas y gases a la composición original ocurre debido a que el aire, así como el agua y la tierra es receptor de las sustancias emitidas durante procesos industriales o por la misma **naturaleza** (CO<sub>2</sub> en animales y humanos, metano por descomposición orgánica) [2].


### 1.1. Fundamentación

El acontecimiento global surgido en 2020, denominado “*Pandemia por COVID-19*” nos hizo repensar como país en la importancia de cuidar los espacios en donde el individuo convive y se desarrolla, esto nos enseñó que existen más factores o elementos en el aire que pueden afectar nuestra salud, surgió entonces la necesidad de conocer si el aire en el ambiente en cual nos encontramos era adecuado para **respirar**, de esta manera es que nos propusimos realizar un dispositivo que sea capaz de medir los niveles de los principales gases y material particulado que se encuentran en un ambiente doméstico, escolar o de oficina y determinar si es nocivo para la salud del individuo o no.

### 1.2. Descripción general

El dispositivo desarrollado es un instrumento con la capacidad de medir concentraciones de monóxido de carbono (CO) proveniente de la combustión, dióxido de carbono (CO<sub>2</sub>) proveniente de los humanos y animales, formaldehído (HCHO) y material particulado (PM<sub>10</sub>, PM<sub>2.5</sub> y PM<sub>1.0</sub>) provenientes de fuentes naturales como partículas del suelo, o actividades humanas como el transporte, la calefacción o las industriales [2]. Se pretende que el **dispositivo** sea capaz de:

- Lograr una medición eficiente de los valores a identificar.
- Enviar notificaciones al consumidor en casos de alarma
- La visualización de los valores pueda verse en tiempo real, pudiendo existir un pequeño retardo.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	1. Introducción

- Permitir al usuario apagar la alarma de notificación desde cualquier vía por donde interactúe.
- Medir las concentraciones de gases o partículas de un nivel nocivo en un periodo de tiempo.
- Que sea capaz de medir otras variables físicas, como ser temperatura, humedad, etc.

Este proyecto fue desarrollado bajo el área de la **Electrónica IoT** contemplando el desarrollo de hardware y software IoT siendo posible el monitoreo desde una aplicación con notificaciones y uso de pantalla para mostrar valores con la posibilidad de establecer niveles nocivos para la salud y activar una alarma en caso de **emergencia**. Su utilización está limitada a lugares cerrados con concurrencia habitual de personas como oficinas de empresas, aulas, espacios del hogar o cualquier sitio donde permanezca mucha gente por un tiempo determinado (sin considerar ambientes industriales) y que puedan actuar en consecuencia con la información mostrada. No se asegura un correcto funcionamiento en un ambiente industrial debido a la naturaleza de contaminación del aire existente en dicho ámbito [3].


### 1.3. Prueba de concepto

La calidad del aire es un tema de gran importancia para la salud humana y la exposición prolongada a la **contaminación** del aire puede tener graves consecuencias para la salud tanto como promotor o agravante de distintas afecciones. Es muy importante tener en consideración el ambiente en el cual habitamos diariamente, creemos que una vez presentado, el dispositivo tendrá éxito debido a que, en base a lo investigado, existen pocos dispositivos con la capacidad de ser monitoreado en tiempo real y notificar la **calidad del aire** en la habitación.

El sistema que se plantea permite la conexión remota, cuenta con seguridad de acceso al sistema y un registro de los niveles de concentración que se puede consultar en todo momento, en adición, es de bajo costo, de uso muy sencillo y de fácil interpretación de datos.

Existen situaciones donde el personal de una oficina siente fatiga, dolores de cabeza o de repente tiene alergias respiratorias y desconocen la fuente, al hacer uso de este **dispositivo** les indicará el estado del aire, es decir, si el *“ambiente está contaminado”* y a qué *“nivel”*. De esta manera, con la información brindada, se podrán dar cuenta del **estado** y tomar acciones en consecuencia (ventilación, filtrado del aire, etc.) evitando futuras enfermedades respiratorias o contagios virales en los ambientes laborales.

El caso mencionado anteriormente se puede replicar en cualquier ámbito cerrado donde exista concurrencia de muchas personas, como posibles usuarios tenemos a las oficinas de empresas, aulas escolares o incluso el hogar de una familia, no se ha considerado su uso en ámbitos industriales.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	1. Introducción


#### 1.4. Estado del mercado

Existen diversos instrumentos en el mercado capaces de medir **parámetros** como la temperatura, la humedad y el nivel de monóxido de carbono (CO) en el ambiente. También hay dispositivos diseñados para la medición de partículas en suspensión. Sin embargo, muchos de estos equipos presentan un costo elevado y resultan de difícil acceso en el país.

Además, según lo que pudimos investigar, existe una limitada variedad de **instrumentos** que ofrezcan la posibilidad de notificar al usuario, mediante una aplicación (App), el “*estado de las variables medidas*”. En general, estos dispositivos incorporan únicamente una pantalla para su interacción, y son pocos los que disponen de sistemas de advertencia ante valores nocivos. Asimismo, son escasos los que integran la medición de una amplia variedad de gases y partículas.

Nuestro proyecto consiste en desarrollar un instrumento capaz de medir **gases** y **partículas**, con interacción tanto local como remota, sistema de advertencias y monitoreo en tiempo real, todo a un costo accesible para el usuario.

Este dispositivo busca brindar información relevante sobre la calidad del aire, permitiendo a los usuarios tomar medidas para reducir la contaminación ambiental y proteger su salud.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO <sub>2</sub> y Partículas
Ingeniería en Electrónica	2. Marco teórico

## Capítulo 2: Marco Teórico

### 2.1. Contaminación del aire

La contaminación del aire es la presencia en la atmósfera de **sustancias** que implican una alteración de sus características naturales. Además, puede generar molestias o riesgos para la salud de las personas o de los seres vivos [4]. Esto ha sido un problema en los ambientes donde el ser humano se ha desarrollado, dañando su **salud** en su hábitat afectando seriamente su rendimiento en las actividades que llevaba a cabo. Por ejemplo, en las escuelas que tienen diversos contaminantes en el aire, hace que sus alumnos corran riesgo de dañar gravemente su salud, ya que inhalan una mayor cantidad de aire respecto a su peso corporal debido a que su sistema respiratorio aún se encuentra en desarrollo y la dosificación no es perfecta. Por lo tanto, es necesario mantener los sectores escolares limpios y ventilados, ya que la mala calidad del aire afecta el rendimiento del ser humano y puede desencadenar o agravar síntomas de las enfermedades respiratorias [2].

#### 2.1.1. Contaminantes

Hay una gran cantidad de contaminantes presentes en el planeta tierra. A continuación, vamos a explicar algunos de los contaminantes más importantes:

##### 2.1.1.1. Material particulado (MP)

El material particulado presente en aire corresponde a partículas sólidas y líquidas que están en suspensión en la atmósfera, estas se clasifican según su diámetro que oscilan entre  $2 \cdot 10^{-4}$  y  $5 \cdot 10^2$   $\mu\text{m}$ . Debido a que existen diferentes tamaños y forma, se agrupan en diferentes diámetros aerodinámicos, como por ejemplos partículas, PM 1 menores o iguales a 1  $\mu\text{m}$  de diámetro, PM 2.5 menores o iguales a 2.5  $\mu\text{m}$  y PM 10 menores o iguales a 10  $\mu\text{m}$ . Además, dentro de MP están las partículas suspendidas totales (PST) que son todas las partículas suspendidas en un determinado volumen de aire no perturbado [2].

Estas sustancias se encuentran ampliamente en la atmósfera debido a que se generan por una mezcla de diversas **sustancias**, como por ejemplo silicatos, carbonatos, sulfatos, cloruros, nitratos, óxidos, metales, carbón, alquitrán, resinas, polen, hongos, bacterias, etc. Estas pueden ser generadas por procesos naturales o por la acción humana, y se dispersan por el viento, llegando a afectar el sistema respiratorio humano dependiendo de su tamaño, como se indica en la siguiente imagen [2].

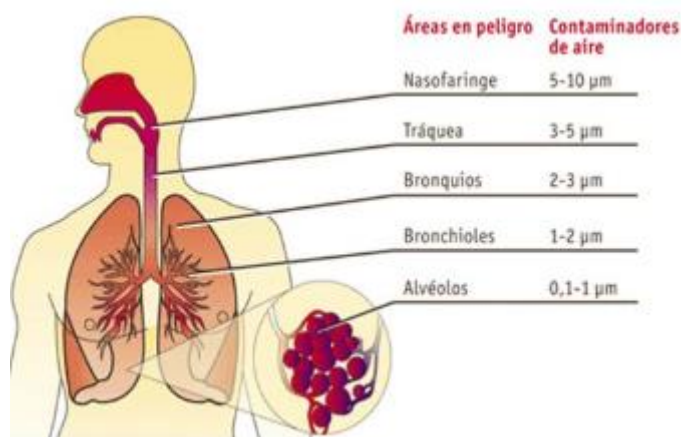


Fig. N°1 - El impacto en el sistema respiratorio según su diámetro [2].

#### 2.1.1.2. Compuesto de azufre


Los compuestos de azufre están presentes en la atmosfera debido a la acción humana. Los principales compuestos presente en la atmosfera son los **óxidos de azufre** como el dióxido de azufre (SO<sub>2</sub>) y el trióxido de azufre (SO<sub>3</sub>) esto es debido principalmente a la quema de combustibles fósiles ricos en azufre esto produce un tercio del SO<sub>2</sub> atmosférico. El SO<sub>2</sub> es un **gas incoloro**, no inflamable y no explosivo que se puede oxidar parcialmente y dar lugar al SO<sub>3</sub> que si tiene en el ambiente presencia de “*humedades altas*” puede dar lugar a ácido sulfúrico y sus sales [2].

#### 2.1.1.3. Óxido nitrógeno

El óxido nitrógeno que se encuentra en la atmosfera es producido en su gran mayoría por la combustión a altas temperaturas. Estos procesos dan lugar al N<sub>2</sub>, este se oxida dando lugar al óxido nítrico (NO) y por medio de reacciones fotoquímicas al dióxido de nitrógeno (NO<sub>2</sub>). Además, gran parte del NO<sub>2</sub> presente en la atmosfera se convierte en ácido nítrico y sales [2].

#### 2.1.1.4. Hidrocarburos

Los hidrocarburos aromáticos policíclicos (HAPs) son compuestos orgánicos que se componen de anillos aromáticos simples y derivan de la combustión de materiales orgánicos o de formas naturales en volcanes. Estos compuestos con dos a tres anillos aromáticos se denominan HAPs de bajo peso molecular, también conocidos como “*compuestos orgánicos volátiles*” (COVs), mientras que aquellos con 4 a 7 anillos se denominan HAPs de mayor peso molecular, también conocidos como contaminantes orgánicos persistentes (COPs). Los COVs, además de ser generados por la quema de compuestos orgánicos, también se generan por disolventes, pinturas y otros productos presentes en los hogares. Estos compuestos pueden generar compuestos aldehídicos, como el formaldehído y el acetaldehído [2].

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO <sub>2</sub> y Partículas
Ingeniería en Electrónica	2. Marco teórico

#### 2.1.1.5. Formaldehido

El formaldehido es un **gas incoloro**, contaminante del aire natural proveniente de muchas fuentes. En el hogar, es producido por cigarrillos, estufas de gas y chimeneas abiertas al aire, también se encuentra en antisépticos, medicamentos, cosméticos, suavizadores de telas, artículos para el cuidado de zapatos, pegamentos y adhesivos. En el aire se encuentra en muy pocas cantidades siendo casi benévolo, cuando las concentraciones exceden los 3 ppm comienzan síntomas de irritación en ojos, nariz y garganta. Concentraciones cercanas a los 20ppm sugieren un riesgo inmediato a la salud [2].

#### 2.1.1.6. Óxidos de carbono

Estos óxidos están constituidos por el monóxido de carbono (CO) y el dióxido de carbono (CO<sub>2</sub>). El CO<sub>2</sub> no se considera un contaminante atmosférico, pero últimamente hay preocupación por la gran cantidad que emite el ser humano. El CO es un contaminante que proviene de la combustión incompleta de cualquier tipo de material combustible [2].

#### 2.1.1.7. Metales pesados

Los Metales pesados se encuentran como material particulado o en estado gaseoso si tienen alta presión de vapor en la atmosfera. En este grupo de contaminantes están el, mercurio, tetraetilo de plomo, cadmio, cromo, cobre, zinc y arsénico. El problema de estos contaminantes es que son, en general, bioacumulable y persistentes en el ser humano siendo un problema para la salud [2].


### 2.1.2. Valores máximos permisibles de contaminantes y sus efectos en la salud

Los valores máximos permisibles se han establecidos en una gran variedad de reglamentaciones para poder proteger a los trabajadores en el ambiente laboral. Por eso, para poder entender los valores que vamos a desarrollar es necesario explicar los siguientes conceptos:

- a) Concentración máxima permisible ponderada en el tiempo (CMP).

Es la concentración máxima permisible a la que un trabajador puede estar expuesto repetidamente día tras día, sin sufrir efectos adversos, durante una jornada laboral de 8 horas al día y 40 horas a la semana [5].

- b) Concentración máxima permisible para cortos períodos de tiempo (CMP - CPT).

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	2. Marco teórico

La CMP-CPT es la exposición media ponderada en el tiempo, que no debe ser sobrepasado en ningún momento, en un periodo de 15 minutos. Estos valores no deben tener un tiempo de exposición mayor a 15 minutos ni se deben repetir 4 veces en el mismo día teniendo un espacio entre exposición y exposición de 60 minutos. Este límite existe para garantizarle al trabajador de que no va a tener daños a la salud como: Irritación. Daños en los tejidos (crónicos o irreversibles). Narcosis en grado suficiente que le impidan desarrollarse laboralmente de manera adecuada [5].

#### c) Concentración Máxima Permisible - Valor Techo (CMP-C)

Es la concentración que nunca debe ser sobrepasada. Si no la podemos medir de manera instantánea la podemos determinar cuándo las muestras son cortas, mediante muestreo durante un tiempo que no debe exceder los 15 minutos, con la excepción de aquellas sustancias que puedan causar irritación de forma inmediata [5].

#### d) Concentración Admisible para Períodos Cortos (CAPC).

Para periodos continuos de 20 minutos es la concentración que no debe ser sobrepasada [6].

#### e) Concentración Admisible para Períodos Largos (CAPL).

Para periodos continuos de 24 horas es la concentración que no debe ser sobrepasada en ese periodo [6].

#### 2.1.2.1. Unidades de medida

La unidad de medida para medir las concentraciones de los contaminantes en la atmosfera es miligramo u otro submúltiplo de gramo (unidad de masa) dividido por metro cubico (unidad de aire). Sin embargo, también podemos expresar las concentraciones en partes por millón (ppm) que se pueden pasar de una unidad a otra con la ecuación N° 1 [7].

$$concentracion(ppm) = concentracion\left(\frac{mg}{m^3}\right) \times \frac{24,5}{peso\ molecular}$$

Ecuación N°.1 - Fórmula para pasar de mg/m<sup>3</sup> a ppm. [7]

En la ecuación N°1 se asume que la presión es de 1 atmosfera y la temperatura es de 25 [°C].



Una vez aclarados estos conceptos, se presentarán en las siguientes tablas los contaminantes con sus valores límites de concentraciones, la mayoría han sido obtenidos principalmente de la reglamentación del **ENRE** salvo en dos, donde se ha utilizado como fuente de consulta, la “*Ley de Entre Ríos N° 6260*” para sistema particulado y para el Tetraetilo de Plomo se ha utilizado la “*OSHA y NOAA*” ambas organizaciones de Estados Unidos.

SUSTANCIAS	MMP		CMP-CPT CMP-C		PM	EFECTOS CRÍTICOS	
	Valor	Unidad	Valor	unidad			
Fluoruro de hidrogeno	-	-	C3	ppm	66,01	Irritación, pulmón, flucrosis	enre
Tetraetilo de Plomo	0,075	mg/m <sup>3</sup>	-	-	323,45	Extremadamente venenoso y daño en el sistema nervioso	<a href="#">osha</a> y <a href="#">noaa</a>
Cadmio y compuestos como (Cd)	0,01	mg/m <sup>3</sup>	-	-	112,4 Variable	Riñón	enre
Compuestos de Cromo insolubles (VI)	0,01	mg/m <sup>3</sup>	-	-	Variable	Cáncer e irritación	enre
Cobre (Cu) Humos	0,2	mg/m <sup>3</sup>	-	-	63,56	Irritación, gi, fiebre del metal	enre
Cobre (Cu) polvo y niebla	1	mg/m <sup>3</sup>	-	-	63,56	Irritación, gi, fiebre del metal	enre
Cloruro de Zinc (Zn)Humos	1	mg/m <sup>3</sup>	2	mg/m <sup>3</sup>	136,29	irritación edema pulmonar	enre
Arsénico y compuestos inorgánicos como As	0,01	mg/m <sup>3</sup>	-	-	74,92 Variable	cáncer (pulmón y piel), pulmon	enre
Monóxido de Carbono (CO)	25	ppm	-	-	28,01	Anoxia, SCV, SNC, reproduccion	enre
Dióxido de Carbono (CO <sub>2</sub> )	5000	ppm	30000	ppm	44,01	Asfixia	enre
Formaldehido (CH <sub>2</sub> O)	-	-	C 0,3	ppm	30,03	irritación y cáncer	enre
Nitrógeno (N)	-	-	-	-	14,01	Asfixiante simple	enre
Dióxido de Nitrógeno (NO <sub>2</sub> )	3	ppm	5	ppm	46,01	irritación, edema pulmonar	enre
Dióxido de Azufre (SO <sub>2</sub> )	2	ppm	5	ppm	64,07	irritación	enre

Tabla N°1 - Valores máximos [5]



Contaminantes del aire	CAPC		CAPL		
	Valor	Unidad	Valor	Unidad	
PM 1	0,5	mg/m3	0,15	mg/m3	Ley_6260 - Entrerios.gov
PM 2.5					
PM 10					

Tabla N°2 - Valores máximos - Partículas en suspensión. [5]

## 2.2. Sensores

Los sensores son **dispositivos** que detectan y responden a una magnitud física y/o variable de nuestro entorno. Se utilizan para interpretar el entorno que nos rodea. Para medir distintas **magnitudes físicas** y/o variables, existen una gran variedad de sensores, como los sensores de velocidad de flujo, corriente eléctrica, presión, temperatura, ópticos, químicos y ambientales [8].

Las magnitudes de interés para este proyecto que se requiere medir son los **gases** monóxido de carbono, dióxido de carbono, formaldehído y las **partículas** PM1, PM2.5, PM10. La medición de estas magnitudes se realizará mediante tres tipos de sensores diferentes.

### 2.2.1. Sensores electroquímicos

Los sensores electroquímicos son aquellos que producen una corriente linealmente proporcional a la fracción de volumen que miden. Están compuestos por tres elementos claves [9]:

- Electrodo: Componentes primarios donde ocurren las reacciones químicas
- Electrolito: Es el medio en el cual los iones pueden moverse entre electrodos
- Membrana: Separa la muestra del electrodo y permite el paso selectivo de sustancias específicas.

El principio de funcionamiento del sensor electroquímico implica tres pasos principales, el primero es la difusión del gas a través de la membrana llegando a la superficie del electrodo, que consiste en un metal catalítico que optimiza la reacción del gas a medir, dicho gas reacciona con el electrodo produciendo una corriente que circula hacia el contraelectrodo que actúa como una semicelda y completa el circuito. Un circuito externo mide el flujo de corriente y determina la concentración del gas. En algunos diseños, se incluye un tercer electrodo de "referencia" para mejorar la estabilidad, la relación señal-ruido y acelerar el tiempo de respuesta del sensor amperométrico básico [9] [10].

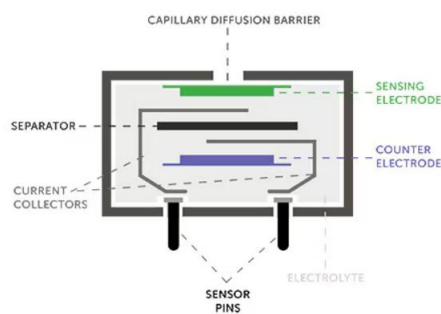


Fig. N°2 - Estructura interna de un sensor electroquímico [9]

### 2.2.2. Sensores infrarrojos no dispersivos

Un sensor de gas de tipo infrarrojo no dispersivo (NDIR) calcula la concentración de gas utilizando la propiedad de que las moléculas de gas absorben los rayos infrarrojos, la cantidad de luz absorbida es proporcional al número de moléculas del gas a detectar en la muestra [11].

Entre los componentes principales de los sensores NDIR podemos encontrar.

- Fuente de luz IR
- Célula de muestra de gas: Es una cámara a través de la cual pasa el gas a analizar. Permite que la luz infrarroja interactúe con las moléculas del gas, la construcción de la célula de muestra es tal que no hay absorción de luz IR en sus paredes y permita la máxima reflexión [12].
- Filtros ópticos: Son utilizados para seleccionar longitudes de onda específicas de luz IR absorbidas por las moléculas del gas. Ayuda a mejorar la precisión del sensor [12].
- Detector: Mide la cantidad de luz IR que pasa a través de la muestra de gas después de interactuar con sus moléculas. La señal del detector proporciona información sobre la concentración del gas en la muestra [12].
- Canal de referencia: Algunos sensores NDIR cuentan con un canal de referencia, el cual mide la cantidad de luz IR que pasa por la cámara de muestra sin ser absorbida por las moléculas del gas. Sirve como línea base para compensar las variaciones en la intensidad de la fuente de luz y las condiciones ambientales [12].
- Procesamiento de señales: La señal de salida del detector y la señal del canal de referencia son procesadas por circuitos integrados alrededor del sensor. Estos componentes electrónicos incluyen un convertidor de analógico a digital, calibración de fábrica y tablas de compensación almace-



nadas en la memoria, y algoritmos de compensación que toman información de sensores externos de temperatura, humedad y presión que pueden influir en la medición final [12].

Los sensores NDIR funcionan mediante el uso de una lámpara infrarroja (IR) para dirigir las ondas de luz a través de un tubo lleno de una muestra de aire. Este aire se mueve hacia un filtro óptico frente a un detector de luz infrarroja. El detector de luz IR mide la cantidad de luz IR que pasa a través del filtro óptico.

En un sensor de CO<sub>2</sub> NDIR, la banda de radiación IR producida por la lámpara está cerca de la banda de absorción de CO<sub>2</sub> de 4,26 micras. Debido a que el espectro IR de CO<sub>2</sub> es único, la coincidencia con la longitud de onda de la fuente de luz sirve como una firma o "huella digital" para identificar la molécula de CO<sub>2</sub>.

A medida que la luz infrarroja pasa a través del tubo de muestra de aire, las moléculas de gas CO<sub>2</sub> absorben la banda específica de luz infrarroja mientras dejan pasar otras longitudes de onda de luz. En el extremo del detector, la luz restante llega a un filtro óptico que absorbe todas las longitudes de onda de la luz, excepto la longitud de onda de 4,26 micrones absorbida por las moléculas de CO<sub>2</sub> en el tubo de muestra.

Finalmente, un detector de infrarrojos lee la cantidad restante de luz que no fue absorbida por las moléculas de CO<sub>2</sub> o el filtro óptico. La disminución de la luz infrarroja transmitida es proporcional a la concentración de gas CO<sub>2</sub> [12].

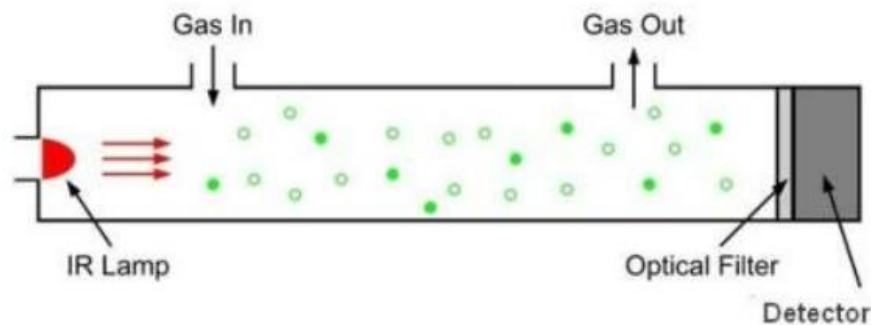


Fig. N°3 - Principio de funcionamiento sensores NDIR [12]

### 2.2.3. Sensores de láser dispersivo

Los contadores ópticos de partículas (OPC) son una tecnología específica de PM que utiliza para su detección un método en donde hace pasar el aire a medir por una célula de medición que contiene un láser y un fotodetector. Las partículas del aire dispersan la luz del láser y el detector mide la luz dispersada para calcular la cantidad de partículas en el ambiente [9].



Las mediciones por difracción láser captan información relacionada con la distribución del tamaño de las partículas midiendo la intensidad de dispersión como una función del ángulo de dispersión, la longitud de onda y la polarización de la luz según los modelos de dispersión aplicables [13]. Este es un método absoluto que no requiere calibración. La difracción láser ofrece una serie de ventajas, como facilidad de uso, funcionamiento rápido, alta reproducibilidad y un amplio intervalo de tamaños dinámicos, que abarca casi cinco órdenes de magnitud (de nanómetros a milímetros) [13].

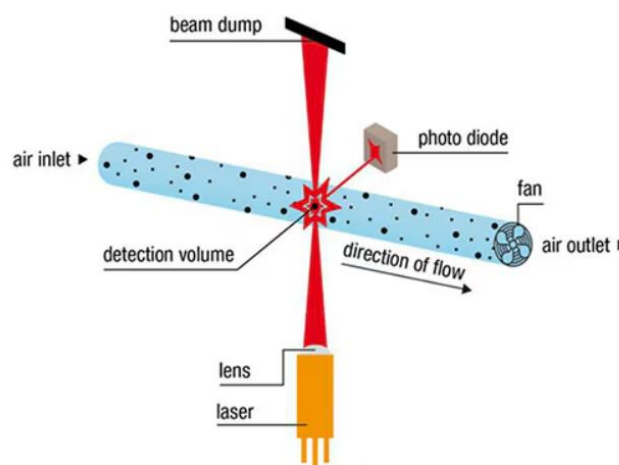



Fig. N°4 - Principio de laser dispersivo [9]

El proceso inicia con una fuente de luz que genera un haz monocromático. Luego de pasar por varios componentes ópticos, el haz sin procesar crea un rayo colimado expandido que ilumina las partículas en el volumen de dispersión. Las partículas dispersan luz, generando patrones de dispersión angular únicos. Estos patrones de dispersión se transforman en un patrón de intensidad espacial que se detecta mediante un arreglo de fotodetectores multi elemento. Posteriormente se procesa y digitaliza la fotocorriente, para crear un patrón de flujo de intensidad que se convierte en una distribución del tamaño de las partículas [13].

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

### Capítulo 3: Desarrollo

En este capítulo se abordará el desarrollo completo del proyecto, incluyendo sus recursos físicos necesarios. Para un correcto acompañamiento al lector se irá relatando el desarrollo en etapas, donde en primer lugar se aborda la arquitectura general del proyecto brindando un panorama descriptivo del mismo para un entendimiento inicial, para luego adentrarse en cada una de las etapas especificando recursos utilizados, técnicas de desarrollo, entre otras cosas.

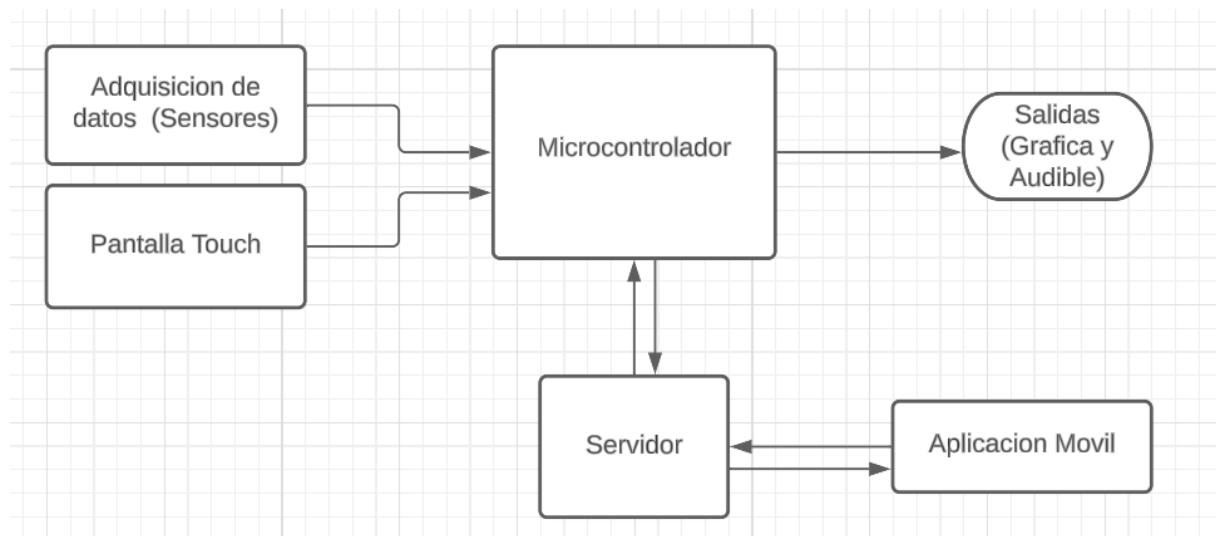



Fig. N°5 - Diagrama en bloques generalizado – Elaboración propia

El diagrama Figura N°2 ilustra la arquitectura general del proyecto. A continuación, se detalla brevemente cada bloque:

- **Adquisición de datos:** Este bloque es la comunicación con el mundo exterior, el ambiente a monitorear y representa a los sensores, que se encargan de realizar la lectura de las magnitudes físicas a monitorear para luego ser transferidas al microcontrolador.
- **Salidas:** Representa la comunicación al usuario, o sea la pantalla y el notificador acústico (buzzer) activo. La pantalla muestra toda la información al usuario mientras que el buzzer es el actuador de una alarma sonora que se activará ante un evento que lo amerite.
- **Microcontrolador:** Es el núcleo del proyecto, interpreta las lecturas de los sensores, procesa la información para comunicarla a la pantalla. También realiza la comunicación con el servidor para el control por la aplicación móvil(app).

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO <sub>2</sub> y Partículas
Ingeniería en Electrónica	3. Desarrollo

- **Servidor:** Es el nexo entre el microcontrolador y la aplicación móvil, recibe la información desde el microcontrolador, permitiendo realizar consultas desde la aplicación. Toda comunicación entre microcontrolador y aplicación se realiza a través del servidor.
- **Aplicación Móvil:** Lleva a cabo el monitoreo a distancia del instrumento, gestión de alarmas, datos históricos, entre otras cosas.

### 3.1. Arquitectura

Desglosando un poco más el diagrama general, presentamos la arquitectura ampliada del proyecto.

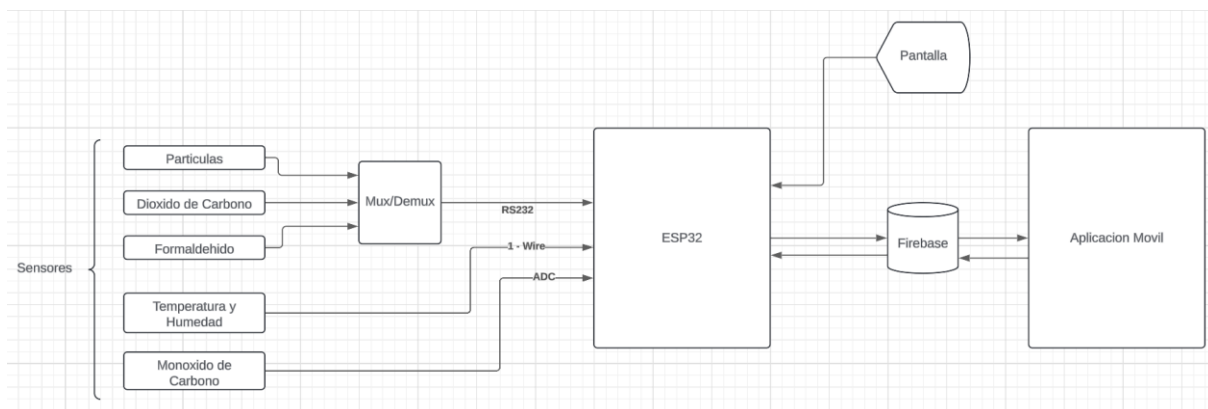


Fig. N°6 - Diagrama en bloques ampliado- Elaboración propia

El bloque de adquisición de datos es ampliado a sensores y multiplexores/demultiplexores aclarando las magnitudes físicas a medir, tres de estos sensores se comunican por protocolo serie **UART**, desarrollaremos este tema más adelante. Luego se incorpora un sensor por comunicación analógica y el último por protocolo serie denominado 1-wire, que será desarrollado en otra sección más adelante.

El microcontrolador por utilizar será el **ESP32** [14] que cumple con los requisitos necesarios para el proyecto como una placa de red integrada, 240 [MHz] de frecuencia de procesamiento y memorias RAM y flash suficientes. Recibirá la información proporcionada por los sensores mediante sus puertos, la procesará, mostrará por pantalla y se sincronizará con la base de datos para la comunicación con la aplicación móvil.

La pantalla será del tipo táctil comandada por un módulo Arduino UNO [15] que será el encargado de comunicarse con el ESP32 solicitándole información para luego cargarla a la pantalla y presentar los datos.



**Firestore** [16] es una plataforma de Google que provee herramientas para el desarrollo de aplicaciones móviles o páginas web, en este caso se utiliza para crear una base de datos en tiempo real que registrará los datos y comandos necesarios para la aplicación móvil.

La aplicación móvil toma los registros enviados por la placa, los visualiza en tiempo real y permite ver registros anteriores en el tiempo de forma listada y en forma gráfica. Adicionalmente se incorporan notificaciones y control de sesiones de usuarios.

## 3.2. Dispositivos

### 3.2.1. Sensor de Monóxido de carbono

El dispositivo por utilizar para las mediciones de monóxido de carbono es el MQ7 [17] de la familia de los sensores MQ.



Fig. N°7 - MQ7-Gas Sensor [18]

Estos sensores son del tipo electroquímicos [18] y cuentan con un material sensible que reacciona ante la presencia de monóxido de carbono, dicho material es SnO<sub>2</sub>, es un material semiconductor transparente con gran estabilidad química y mecánica, tiene una fase estable que presenta una estructura cristalina tetragonal conocida como casiterita. Tiene una conductividad baja en aire limpio, que aumenta a medida que aumenta la concentración del gas [17]. El sensor funciona como una resistencia que varía su valor cuando se exponen a diferentes gases.

El principio de funcionamiento se basa en un calentador que poseen internamente encargado de aumentar la temperatura interna para que sea posible una reacción química en el sensor y de esta manera varía su resistencia ante la presencia de ciertos gases, en este caso monóxido de carbono [17].

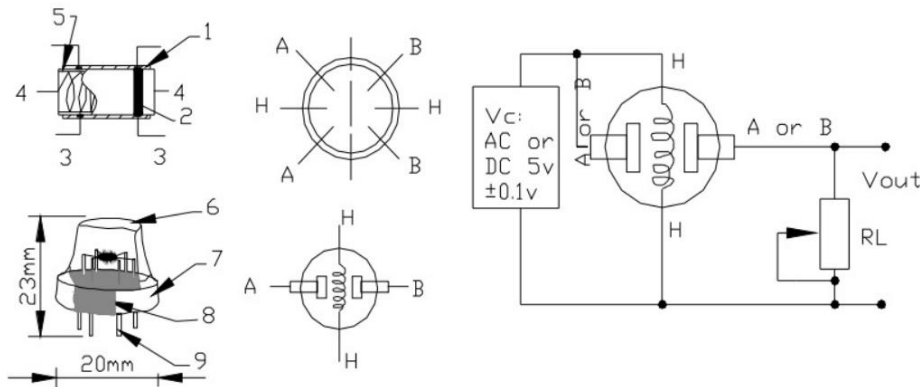


Fig. N°8 - MQ-circuito interno [17]

Como se observa en la figura, el sensor se comporta como una resistencia y para poder leerlo es necesario colocar una resistencia de carga  $R_L$  para cerrar el circuito.

3.2.1.1. Cálculo de resistencia del sensor ( $R_S$ )

Para conocer el valor en ppm del gas, primero debemos saber qué valor resistivo tiene el sensor y para ello debemos plantearlo como circuito.

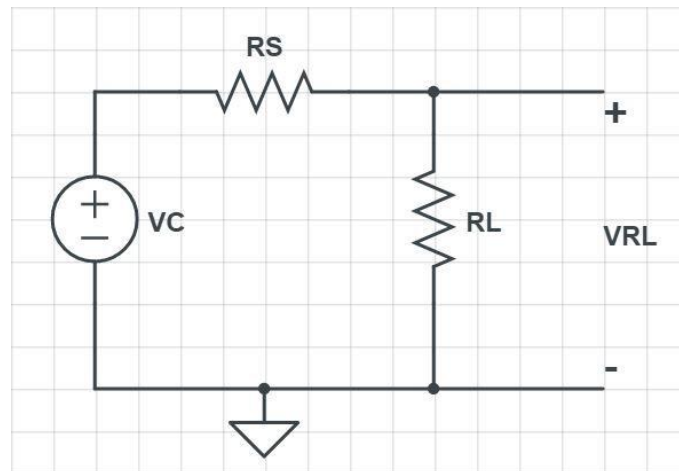


Fig. N°9 - Circuito del sensor simplificado [18]

Donde " $R_S$ " representa al sensor, y " $R_L$ " es la resistencia de carga. " $V_C$ " es la tensión de alimentación que le proporcionamos al sensor. En nuestro caso  $R_L = 1[K\Omega]$  y  $V_C = 5[V]$ .

Resolvemos el circuito para obtener  $R_S$ .



$$V_{RL} = \frac{V_C}{R_S + R_L} * R_L$$

Ecuación N°.2 - Cálculo de RS – Planteo de primera ecuación [12]

$$V_{RL} * R_S = (V_C * R_L) - (V_{RL} * R_L)$$

$$R_S = \frac{(V_C * R_L) - (V_{RL} * R_L)}{V_{RL}}$$

Ecuación N°.3 - Cálculo de RS- Despeje final [12]

### 3.2.1.2. Interpretación de gráfica y obtención de ecuación del sensor

Una vez obtenida la ecuación para el cálculo de la resistencia del sensor (RS), nos vamos a la hoja de datos o datasheet del sensor, vamos a su gráfica característica.

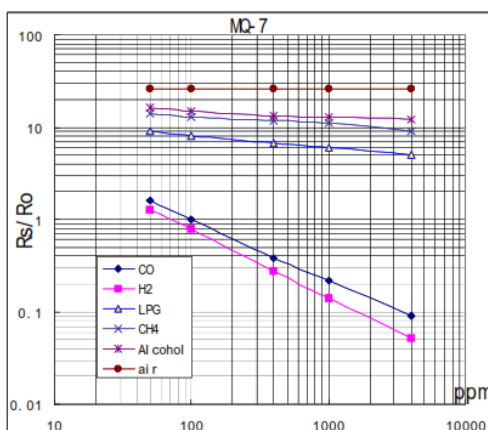


Fig. N°10 -Curva característica – Sensor MQ7 [17]

De la gráfica observamos varias rectas, cada una representa la respuesta a los gases que es capaz de medir el sensor, en el eje de abscisas con escala logarítmica veremos los valores ppm y en el eje de ordenada veremos la relación o “ratio” entre “Rs” y “Ro” con escala logarítmica también. “Ro” es la resistencia que tiene el sensor al estar expuesto al aire limpio. Nótese en la gráfica, la recta de aire limpio es constante en todos los valores ppm, esto es porque “Rs = Ro”.

Habiendo explicado la gráfica se procede a extraer los datos para obtener una aproximación lineal o logarítmica y poder determinar los valores ppm según el valor “Rs”. Para analizar el gráfico se utiliza una herramienta de software online llamada “WebplotDigitizer” la cual analizará el gráfico y proporcionará una tabla de valores a ser procesada mediante un libro de Excel (Ver Anexo) obteniéndose una ecuación de aproximación lineal o logarítmica según se requiera. Debido a que los cálculos son realizados por el microcontrolador y, a fines de ser más exactos, se trabaja con la ecuación **logarítmica**.



En “WebplotDigitizer” se marcaron los puntos conocidos de la gráfica con el objetivo de minimizar el error y para luego generar la tabla de puntos [19].

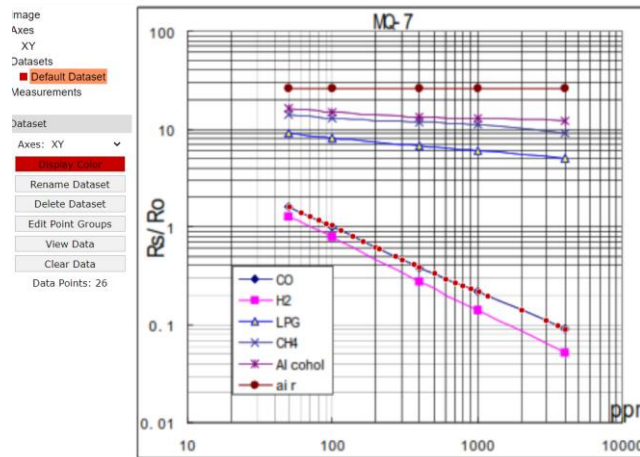


Fig. N°11 -Análisis de gráfica – Web Plot Digitizer [20]

Ya marcados los puntos, generamos la tabla de valores y los exportamos en formato Excel. Con los datos exportados creamos un gráfico, luego mediante herramientas graficas de Excel marcamos una línea de tendencia “potencial” y ponemos que nos marque ecuación (ecuación N°4).

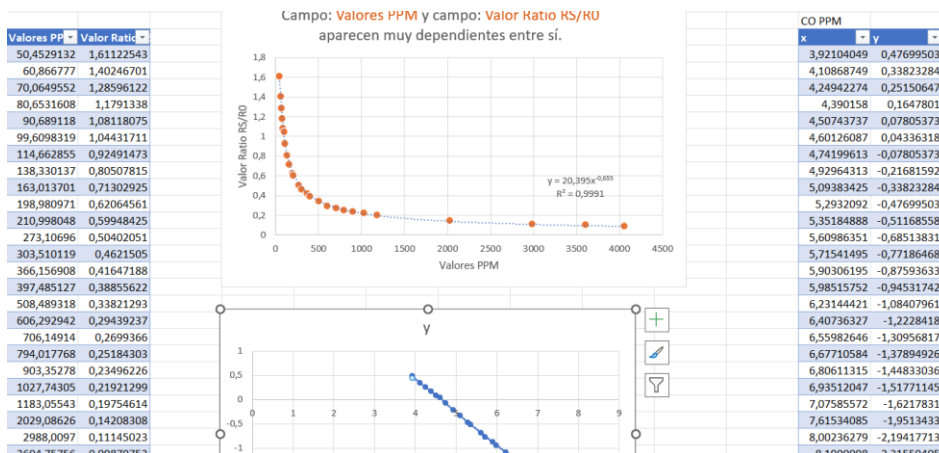



Fig. N°12 - Tabla de datos de la curva MQ7 – Ver Anexo 8.1

De esta manera, se obtiene la linealización de la curva de calibración del sensor MQ7. Una vez obtenida la curva pasamos a escribir nuestra ecuación que nos resultó

$$y = 98,045 * X^{-1,533}$$

Ecuación N°.4 - Primera aproximación logarítmica de MQ7

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

Pasando a nuestros valores

$$(CO_{ppm}) = 98,045 * \left(\frac{R_s}{R_0}\right)^{-1,533}$$

Ecuación N°.5 - Ecuación logarítmica de MQ7

Siendo  $R_s \rightarrow$  La resistencia que debo calcular con el valor analógico leído desde el sensor

$R_0 \rightarrow$  La resistencia del sensor a 100 ppm de CO en aire limpio.

$ratio = \frac{R_s}{R_0} \rightarrow$  Según los datos de la tabla que brinda la hoja de datos (datasheet) este valor es **26,3**.

Cálculo  $R_s$  según  $V_{RL}$  que es la tensión de la salida analógica del modulo MQ7

$$R_s = \frac{V_c - V_{RL}}{V_{RL}} * R_L$$

Ecuación N°.6 - Expresión de Rs

Siendo  $R_L = 1 K\Omega$  y  $V_c = 5 V$

El valor de " $R_0$ " se calcula en laboratorio, " $R_0$ " es el valor de " $R_s$ " cuando se usa una muestra de aire con 100 [ppm]. Y hay que hacer esto para cada sensor puesto que " $R_0$ " es diferente en cada uno.


Se debe tomar muchas muestras hasta obtener un valor estable de " $R_s$ " en aire limpio y convertir ese valor " $R_s$ " en " $R_0$ ". Luego de eso puedo realizar mediciones de " $R_s$ " y calcular la " $ratio$ " [17].

### 3.2.1.3. Lectura de valores

La lectura de valores se realiza mediante el microcontrolador que cuenta con una etapa ADC el cual nos proporcionara una lectura analógica que depende de la tensión de entrada y la resolución de este. Repasando la "*ecuación N°6*" vemos que es necesario obtener el valor de tensión proporcionado por el sensor, dicho valor es  $V_{RL}$  en la ecuación. En el microcontrolador se deben realizar los siguientes cálculos

$$Voltios leidos(V_{RL}) = \frac{(Medicion\ analogica)}{Resolucion\ Bit\ maxima(4096)} * Voltaje\ de\ resolucion$$

Ecuación N°.7 - Obtención de  $V_{RL}$

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

Ya con este valor de  $V_{RL}$  es posible obtener  $R_S$ . Luego solo necesitamos la ratio (variación), que surge de la relación de  $R_S$  con  $R_0$ .

Con todos los datos obtenidos, ya es posible calcular la concentración en ppm de nuestro gas. Traemos a colación la ecuación N°5

$$(CO_{ppm}) = 98,045 * \left(\frac{R_S}{R_0}\right)^{-1,533}$$

Ecuación N°.8 - CO- Cálculo de concentración en ppm

Se debe tener en cuenta que sólo  $R_S$  varía, lo demás permanece constante.  $R_0$  se debe calcular una sola vez y ese valor queda fijo. El cálculo de dicha resistencia se llama calibración [17].

#### 3.2.1.4. Calibración del sensor

La calibración consiste en obtener el valor resistivo  $R_0$  que tomará el sensor al estar expuesto al aire limpio libre de contaminantes. Este parámetro, en conjunto con la ratio proporcionada por su hoja de datos (datasheet), establece el criterio mediante el cual se realizarán las demás mediciones para poder calcular la concentración en partículas por millón (ppm) de dióxido de carbono (CO2).

Los pasos para el cálculo de  $R_0$  son los mismos que para obtener  $R_S$  con la diferencia que debe realizarse en un ambiente libre de contaminantes. Para una mejor exactitud se realizarán múltiples lecturas para promediarlas.

Tomando las medidas en aire limpio, primero se obtiene el valor de tensión con la ecuación N°7, luego de dicha lectura se calcula  $R_S$  con la ecuación N°6. Por último, es posible calcular  $R_0$  según la ratio dado por su hoja de datos (datasheet) [17].

$$R_0 = \frac{R_S}{ratio}$$

Ecuación N°.9 - Cálculo de  $R_0$  – MQ7

#### 3.2.2. Sensor de dióxido de carbono

Para la lectura de dióxido de carbono (CO2) se hará uso del sensor infrarrojo **MH-Z19B** de Winsen Sensors. Este módulo sensor se basa en el principio **infrarrojo no dispersivo** (NDIR) para detectar la presencia de CO2 en el aire con una gran selectividad y de larga duración. Está desarrollado mediante una ajustada integración con tecnología de detección de gas de absorción infrarroja por medios ópticos



[21]. Este módulo debe precalentarse por 3 minutos, integra compensación de temperatura y tiene 3 salidas, PWM, DAC y serie UART.



Fig. N°13 -Sensor de dióxido de Carbono MHZ-19

Dispone de dos formas de conexión, por medio de pines headers macho o por medio de un terminal con cable, se hará uso de la segunda forma ya que el sensor no irá ubicado sobre la placa.

Pin	Pin Definition
Vin	Positive pole of power (Vin)
GND	Negative pole of power (GND)
Vo	Analog output(0.4~2V)
PWM	PWM
Hd	HD(zero point calibration, low level lasting for over 7s is effective)
Rx	UART(RXD)TTL Level data input
Tx	UART(TXD)TTL Level data output

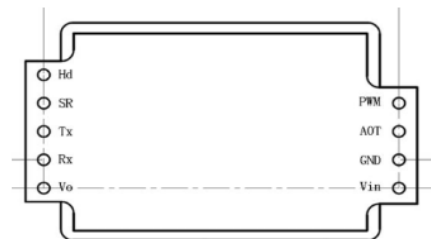


Fig. N°14 -MHZ-19 Headers pinout

Pin	Terminal pin Definition
Pin 1	Analog Output Vo (0.4~2V)
Pin 2	None
Pin 3	Negative Pole(GND)
Pin 4	Positive Pole(Vin)
Pin 5	UART(RXD)TTL Level data input
Pin 6	UART(TXD)TTL Level data output
Pin 7	None

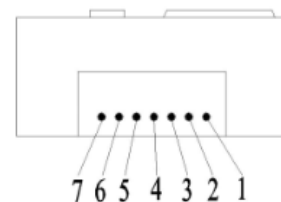



Fig. N°15 -MHZ-19 Conexión por cables pinout

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

### 3.2.2.1. Salida de Datos

#### a) Analógica

El dispositivo en uno de sus pines cuenta con salida analógica en un rango de 0.4[V] a 2 [V]. La detección infrarroja toma el valor de concentración del gas y mediante un DAC integrado del sensor se realiza la conversión a tensión para mostrarlo en la salida de su Pin 1. El rango de detección para esta salida solo es 0-2000 [ppm]. Para realizar la lectura desde nuestro microcontrolador será necesario configurar nuestro ADC de manera que realice lecturas en el rango del sensor. Esta **salida de datos** es la menos exacta de las tres posibles que ofrece el sensor, por lo que no se utilizará en este proyecto.

#### b) PWM

El dispositivo también cuenta con salida PWM, pero solo lo hace a través de las salidas por pines, no cuenta con salida PWM por cable ya que por este medio queda susceptible al ruido y existe riesgo de **corrupción de datos**. Por salida PWM, el rango de detección es 0-2000 [ppm] y trabaja de la siguiente manera:

Cada ciclo tiene una duración de 1004 [ms], el inicio y fin de ciclo se identifica con un pulso de 2 [ms] para realizar la lectura en ppm de concentración del gas.

Para el cálculo de la concentración ppm del gas se deben quitar estos pulsos de inicio y fin de ciclo. Cuanta más concentración de gas exista, mayor es el **ancho del pulso**. La fórmula para su cálculo es como sigue:

$$CO2_{ppm} = 2000 * \frac{TH - 2ms}{TH + TL - 4ms}$$

Ecuación N°.10 - Concentración de CO2 por PWM

Siendo *TH* el tiempo en estado lógico “alto” y *TL* el tiempo en estado lógico “bajo”.

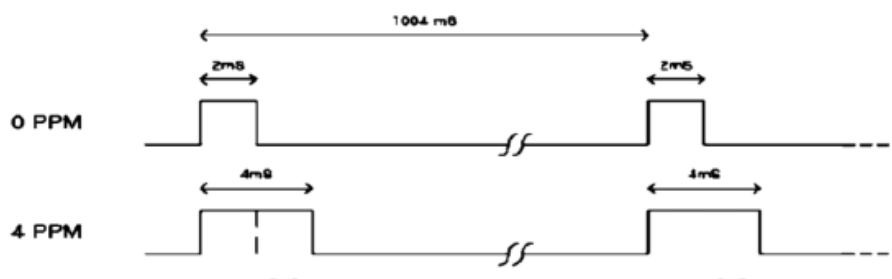



Fig. N°16 -Señales de ejemplo de lectura por PWM

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

Realizar lecturas PWM por medio del microcontrolador requiere el uso de interrupciones en sus periféricos. El ciclo de lectura inicia con un cambio de estado, esto lanza una interrupción y se comienza a contabilizar el tiempo. Para evitar falsas lecturas se realizan dos mediciones seguidas y estas deben coincidir, el método de medición es contabilizando el **tiempo por ventanas**. La secuencia de tratamiento de interrupciones debería ser como sigue:

Cada cambio de estado lanza una interrupción, esta metodología se conoce como “*interrupción change*”, por cada interrupción se registra el tiempo y número de interrupción, en el bucle principal, mediante una función, se realiza el tratamiento de tiempos y el cálculo de valores de concentración del gas, para cada lectura deben ocurrir un total de 4 interrupciones y según el número de interrupción se realiza una tarea diferente, luego de la primer interrupción se registra el valor de tiempo, luego de la segunda se registra la diferencia de tiempos entre la primer y segunda interrupción, luego de la tercer interrupción se repite el paso uno, luego de la cuarta interrupción se repite el paso dos y se procede a calcular el valor de concentración del gas según los tiempos registrados. Al finalizar, se reinician las variables permitiendo que el **ciclo de lectura** se repita y se registre otra lectura [21].

### c) UART

La comunicación por puerto serie UART es la más fiable y permite establecer rangos de medición de 0-2000 [ppm], 0-5000 [ppm] y 0-10000 [ppm]. La concentración de gas se lee directamente, no es necesario realizar una conversión.

La comunicación se realiza por medio de RS232 full dúplex, el módulo sensor espera un comando y responde en consecuencia. En nuestro microcontrolador debemos configurar el puerto serie de la siguiente manera:

Baud Rate	9600
Data byte	8 byte
Stop Byte	1 byte
Parity	none

Tabla N°3 - MHZ-19 Configuración de puerto serie [21]

Cada comando enviado y cada respuesta tiene un tamaño de 8 bytes, con un bit de start fijo **0xFF**, en su trama contiene también el número de sensor, por defecto es **0x01** y el ultimo byte corresponde al checksum. Es importante calcular el checksum en cada caso ya que nos indica si el dato enviado o recibido es correcto y si la comunicación fue exitosa [21].



c.1) Lectura de datos

La lectura de concentración en valores ppm se realiza por medio del siguiente comando y se espera otro comando respuesta [21].

0x86- Read CO2 concentration								
Sending command								
Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
Start Byte	Reserved	Command	-	-	-	-	-	Checksum
0xFF	0x01	0x86	0x00	0x00	0x00	0x00	0x00	0x79
Return value								
Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
Start Byte	Command	Concentration (High 8 Byte)	Concentration (Low 8 Byte)	-	-	-	-	Checksum
0xFF	0x86	HIGH	LOW	-	-	-	-	Checksum
<b>CO2 concentration = HIGH * 256 + LOW</b> <b>For example:</b> <b>Send command FF 01 86 00 00 00 00 00 79, Return value FF 86 02 20 00 00 00 00 58</b> <b>How to calculate concentration: convert hexadecimal 02 into decimal 2, hexadecimal 20 into decimal 32, then 2*256+32=544ppm</b>								

Fig. N°17 -MHZ-19 Comando Leer concentración [21]


c.2) Auto Calibración

Este módulo trae la opción de auto calibración, esto significa que el sensor pasadas 24 horas de uso seguidas, se calibrará automáticamente. Esta opción puede activarse o desactivarse mediante el siguiente comando [21].

0x79- On/Off Self-calibration for Zero Point								
Send command-No return value								
Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
Start Byte	Reserved	Command	-	-	-	-	-	Checksum
0xFF	0x01	0x79	0xA0/0x00	0x00	0x00	0x00	0x00	Checksum
<b>For example:</b> <b>ON this function, send command: FF 01 79 A0 00 00 00 00 E6</b> <b>OFF this function, send command: FF 01 79 00 00 00 00 00 86</b> <b>NOTE: This function is on when Byte3 is 0xA0 while this function is off when Byte3 is 0x00.</b> <b>Default status is "this function is on".</b>								

Fig. N°18 -MHZ-19 Comando Auto calibración on/off [21]

Por defecto, la auto calibración está activada. Debe entenderse que esta opción es útil cuando el sensor no se encuentra en un ambiente altamente contaminado, en aquellos casos debe calibrarse de forma manual en un ambiente limpio. El fabricante llama a un ambiente limpio a una cantidad de partículas menor a 400 ppm [21].

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

### c.3) Rango de detección

Es posible modificar el rango de detección como se había mencionado, las posibilidades son de 0 a 2000ppm, de 0 a 5000 ppm o de 0 a 10000 ppm. En los bytes 4, 5, 6 y 7 se debe colocar el valor en hexadecimal del rango a detectar o cambiar.

0x99- Detection range setting								
Send command-No return value								
Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
Start Byte	Reserved	Command	Reserved	Detection range 24~32 bit	Detection range 16~23 bit	Detection range 8~15 bit	Detection range 0~7 bit	Check sum
0xFF	0x01	0x99	0x00	Data 1	Data 2	Data 3	Data 4	Check sum

Fig. N°19 -MHZ-19 Comando configurar rango de detección [21]

### 3.2.3. Sensor de partículas

Para el sensado de partículas se hará uso del sensor **ZH06** de Winsen Sensors [22]. Este es un sensor de partículas que se basa en el principio de **dispersión laser** para la detección de materia particulado presente en el aire, es capaz de detectar partículas del tamaño de 1 [µm] (PM1.0), 2.5 [µm] (PM2.5) y 10 [µm] (PM10).

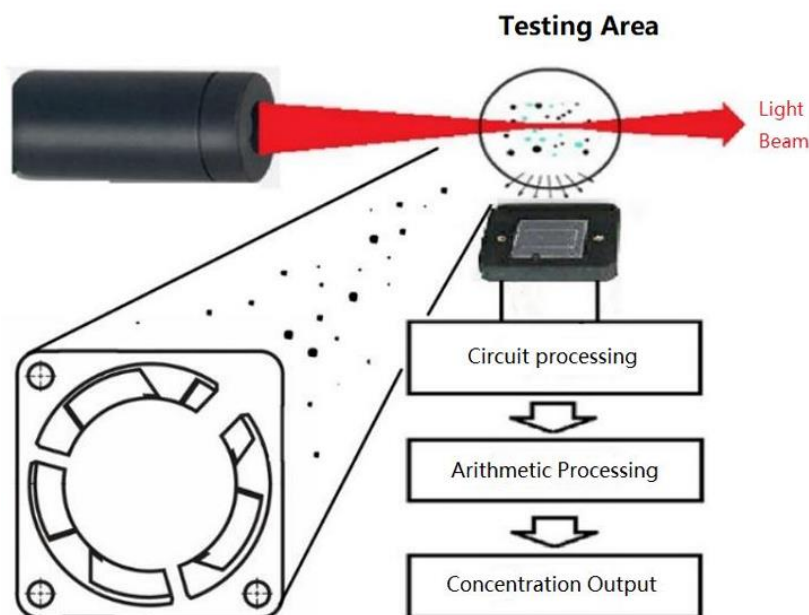


Fig. N°20 -ZH06 – principio de funcionamiento [22]



En su interior contiene una rendija de sensado, en dicho espacio se bombardea con una luz laser y por medios ópticos se detecta la concentración de material particulado presente en el aire. Al ser óptico, este sensor ofrece una gran estabilidad y precisión de medición, y su lectura se realiza mediante una señal PWM o por protocolo serie UART.

El conexionado y distribución de pines es como sigue

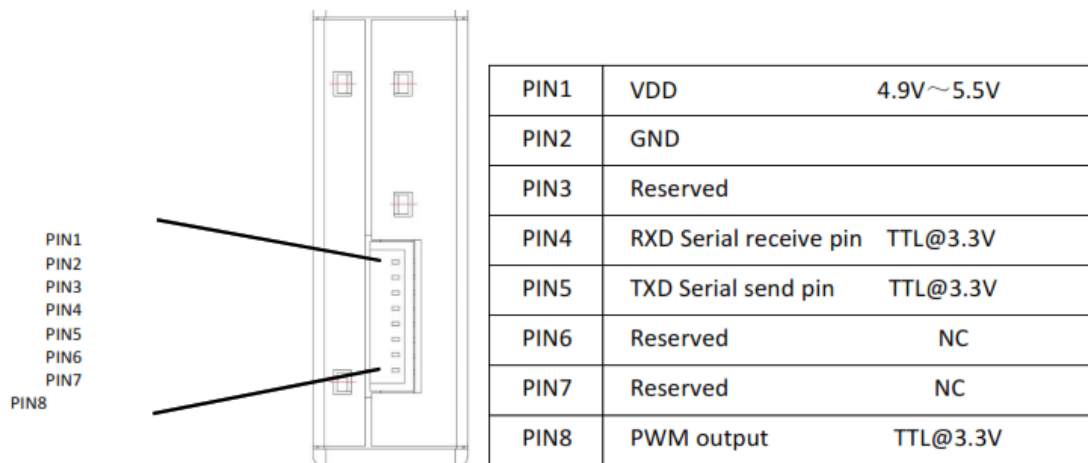


Fig. N°21 -ZH06 - Pinout [22]

Debe ser instalado de manera que las ventilas de entrada y salida de aire del sensor estén en contacto con el aire exterior. Dicho de otra manera, debe circular el aire por este [22].

### 3.2.3.1. Salida de datos

#### a) PWM

El dispositivo cuenta con salida PWM a través del pin 8, es una **señal TTL** de 3.3 [V] con un rango de detección de 0-1000 [ $\frac{\mu\text{g}}{\text{m}^3}$ ] con la desventaja que solo es capaz de comunicar la concentración de partículas PM2.5. Cada periodo dura 1000 [ms], iniciando con un pulso start de 200 [ $\mu\text{s}$ ] y finalizando con un pulso de 200 [ $\mu\text{s}$ ]. La presencia de PM2.5 es un nivel alto, es decir, para medir la concentración debo contar el tiempo en alto de la señal.

El cálculo de concentración es como sigue:

$$p\left(\frac{\mu\text{g}}{\text{m}^3}\right) = \frac{1000 * TH}{TH + TL}$$



Ecuación N°.11 - ZH06- Cálculo de concentración PM2.5 por PWM

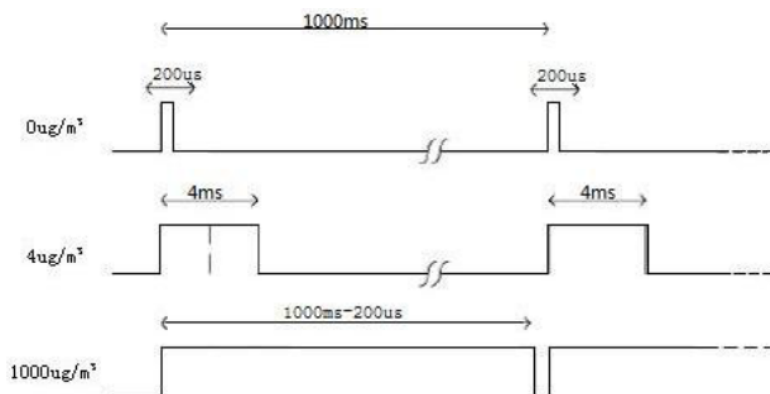


Fig. N°22 -ZH06- Ejemplo de lectura por PWM [22]

La lectura de concentración mediante el microcontrolador debe ser mediante el uso de interrupciones utilizando la misma metodología que en la sección 3.2.2.1 parte PWM de este documento.

b) UART

La lectura mediante puerto serie es la más fiable y es la utilizada en este proyecto. Este dispositivo trabaja con protocolo UART TTL 3.3[V] mediante topología RS232. Para iniciar la comunicación primero configuramos nuestro puerto serie de la siguiente manera:

Baud Rate	9600
Data byte	8 byte
Stop Byte	1 byte
Parity	none

Tabla N°4 - ZH06 - Configuración de puerto serie [22]


Luego de configurado, se inicia la comunicación. Cuenta con dos modos de trabajo, **modo activo** y **modo pregunta y respuesta (QA)**. En modo activo, el sensor envía una trama de 32 bits cada un segundo de forma continua. La trama es como sigue:



Byte 1	Start byte 1		0x42
Byte 2	Start byte 2		0x4D
Byte 3	Frame length	high level 8	0x00
Byte 4		low level 8	0x1C
Byte 5	Data 1	High Level 8	Reserved
Byte 6		Low Level 8	
Byte 7	Data 2	High Level 8	Reserved
Byte 8		Low Level 8	
Byte 9	Data 3	High Level 8	Reserved
Byte 10		Low Level 8	
Byte 11	Data 4	High Level 8	PM1.0 concentration (ug/m <sup>3</sup> )
Byte 12		Low Level 8	
Byte 13	Data 5	High Level 8	PM2.5 concentration (ug/m <sup>3</sup> )
Byte 14		Low Level 8	
Byte 15	Data 6	High Level 8	PM10 concentration (ug/m <sup>3</sup> )
Byte 16		Low Level 8	
Byte 17	Data 7	High Level 8	reserved
Byte 18		Low Level 8	
Byte 19	Data 8	High Level 8	reserved
Byte 20		Low Level 8	
Byte 21	Data 9	High Level 8	reserved
Byte 22		Low Level 8	
Byte 23	Data 10	High Level 8	reserved
Byte 24		Low Level 8	
Byte 25	Data 11	High Level 8	reserved
Byte 26		Low Level 8	
Byte 27	Data 12	High Level 8	reserved
Byte 28		Low Level 8	
Byte 29	Data 13	High Level 8	reserved
Byte 30		Low Level 8	
Byte 31	Check	High Level 8	Initiative upload check= = byte1+.....+byte 22
Byte 32		Low Level 8	

Fig. N°23 -ZH06- Trama en modo activo [22]

Los valores de concentración para las diferentes partículas se encuentran desde el byte 11 hasta el byte 16. Este modo de trabajo resulta ineficiente para nuestra aplicación ya que se debe sincronizar el microcontrolador con el sensor para tomar las lecturas lo cual no es conveniente. En modo QA, la comunicación con el dispositivo es **asíncrona**, es decir, debo enviar una petición o comando y esperar una res-

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

puesta. De esta manera se puede controlar los tiempos de comunicación con el sensor y tomar lecturas cuando sea necesario [22]. La petición de información se realiza enviando la siguiente trama.

0	1	2	3	4	5	6	7	8
Starting	Reserve	command	reserve	reserve	reserve	reserve	reserve	Check value
0xFF	0x01	0x86	0x00	0x00	0x00	0x00	0x00	0x79

Fig. N°24 -ZH06- Trama de petición de datos [22]

Por parte del sensor recibimos la siguiente trama:

0	1	2	3	4	5	6	7	8
Starting Command		PM2.5(ug/m3)		PM10(ug/m3)		PM1.0(ug/m3)		Check value
		High 8 Level	Low 8 Level	High 8 Level	Low 8 Level	High 8 Level	Low 8 Level	
0xFF	0x86	0x00	0x85	0x00	0x96	0x00	0x65	0xFA

Fig. N°25 -ZH06- Trama de respuesta de datos [22]

Los datos de la trama de respuesta se encuentran en Hexadecimal, desde el bit 2 al bit 7 se tiene la concentración del material particulado separados en parte alta y parte baja de cada uno. La interpretación para cada una de las partículas debe entenderse de la siguiente manera:

- $PM2.5 = (\text{bit } 3 + 256) + \text{bit } 2$
- $PM10 = (\text{bit } 4 + 256) + \text{bit } 5$
- $PM1.0 = (\text{bit } 6 + 256) + \text{bit } 7$

Debe entenderse que sumar 256 significa un corrimiento de bits para luego sumarse a la parte baja, es importante entender que sumar bits no es lo mismo que sumar números.

Debido a que el sensor inicia en modo activo, es necesario realizar el cambio a modo QA mediante el envío de una petición al sensor [22].

Para pasar a modo QA se envía la siguiente trama:



0	1	2	3	4	5	6	7	8
Starting	Reserve	command	Q&A	Reserve	Reserve	Reserve	Reserve	Check value
0xFF	0x01	0x78	0x41	0x00	0x00	0x00	0x00	0x46

Fig. N°26 -ZH06 – Trama de cambio a modo QA [22]

Para volver al modo activo se debe enviar:

0	1	2	3	4	5	6	7	8
Starting	Reserve	Command	Upload	Reserve	Reserve	Reserve	Reserve	Check value
0xFF	0x01	0x78	0x40	0x00	0x00	0x00	0x00	0x47

Fig. N°27 -ZH06 – Trama de cambio a modo activo [22]

### 3.2.4. Sensor de Formaldehido

Para mediciones de gases volátiles como **Formaldehido** utilizamos el módulo de calidad de aire **ZP16** de Winsen Sensors, el cual hace uso de un chip avanzado de película gruesa sensible a gases volátiles. Es un módulo que ha sido probado, ajustado y **calibrado**, lo cual lo hace consistente a largo plazo y con una gran sensibilidad.

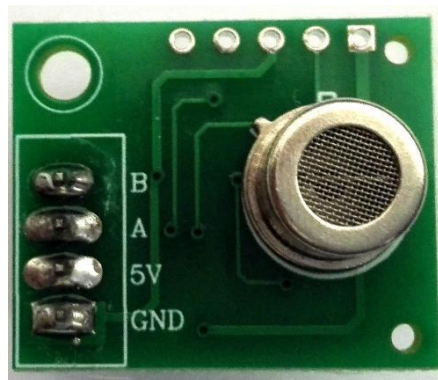



Fig. N°28 -Módulo Sensor Zp16

Name	Function
GND	Input power -
5V	Input power +
A	UART(TX) 0-5V Output
B	UART(RX) 0-5V Input

Fig. N°29 -Definición de pines Zp16 [23]

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

No requiere calibración, se alimenta con 5[V], cuenta con una resolución de 0.05 mg por metro cúbico, y mide hasta 10 mg por metro cúbico de concentración máxima. Requiere un precalentamiento de alrededor de 3 minutos antes de la **primera lectura** [23].

### 3.2.4.1. Protocolo de comunicación

Este módulo se comunica por medio de protocolo serie UART RS-232 con la siguiente configuración.

Baud Rate	9600
Data byte	8 byte
Stop Byte	1 byte
Parity	none

Tabla N°5 - Zp16- Configuración de puerto serie [23]

### 3.2.4.2. Comandos de comunicación

Las comunicaciones con el módulo pueden ser de dos formas, actualización activa y modo pregunta y respuesta (QA). El modo por defecto es actualización activa donde el módulo envía la concentración de gas cada un segundo. En modo QA, se debe enviar un comando de petición de dato y se espera una respuesta con el dato de concentración del gas.

Para pasar de modo actualización activa a modo QA se debe enviar el siguiente comando.


Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
Start Byte	Reserved	Switch Command	Q&A	Reserved	Reserved	Reserved	Reserved	Checksum
0xFF	0x00	0x78	0x41	0x00	0x00	0x00	0x00	0x47

Fig. N°30 -Trama cambio de actualización activa a Q&A – Zp16 [23]

Para pasar de modo QA a actualización activa se debe enviar el siguiente comando.

Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
Start Byte	Reserved	Switch Command	active upload	Reserved	Reserved	Reserved	Reserved	Checksum
0xFF	0x00	0x78	0x40	0x00	0x00	0x00	0x00	0x48

Fig. N°31 -Trama cambio de Q&A a actualización activa – Zp16 [23]

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

Ambos comandos son solo escritura, no vamos a recibir una respuesta por parte del módulo ZP16 [23].

### 3.2.4.3. Lectura de datos

La trama de datos recibida es la misma independientemente del modo en el que nos encontremos. Estando en modo QA para solicitar el dato debemos enviar

Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
Start Byte	Reserved	Command	Q&A	Reserved	Reserved	Reserved	Reserved	Checksum
0xFF	0x00	0x86	0x00	0x00	0x00	0x00	0x00	0x7A

Fig. N°32 -Trama petición de datos Zp16 [23]

Vamos a recibir una trama según la siguiente estructura

Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7	Byte8
Start Byte	Command	High byte of gas concentration (MG/M3)	Low byte of gas concentration (MG/M3)	Decimal point	Reserved	High byte of full scale	Low byte of full scale	Checksum
0xFF	0x86	0x01	0xC2	0x02	0x00	0x03	0xE8	0xCA

Fig. N°33 -Trama respuesta de datos Zp16 [23]

La lectura de concentración en  $\frac{mg}{m^3}$

Concentración de gas = byte2\*256 + byte3

Ejemplo → FF 86 01 C2 02 00 03 E8 CA


byte2 = 1; byte3 = C2 = 194;

256\*1+194 = 450.

Ahora miramos el **byte4** de punto decimal.... *Byte4=02*; se divide por 10 a la potencia del byte.

$$\frac{450}{10^2} = 4.5 \left[ \frac{mg}{m^3} \right]$$

Ecuación N°.12 - Lectura de HCHO

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

Luego lo paso a ppm.

$$ppm = \frac{24.45 * \frac{mg}{m^3}}{\text{peso molecular formalehido}}$$

Ecuación N°.13 - ZP16 - Paso de mg/m3 a ppm

Peso molecular del formaldehido = 30.026 [ $\frac{g}{mol}$ ] [7].

### 3.2.5. Sensor de humedad y temperatura

Como segunda prioridad se propone dar a conocer los valores de humedad y temperatura, el sensor elegido para la tarea es el **DHT11**. Este sensor incluye un componente de medición de humedad de tipo resistivo y un componente de medición de temperatura NTC y se conecta a un microcontrolador de 8 bits que ofrece una salida digital calibrada.

El sensor ya viene calibrado en laboratorio, los **coeficientes de calibración** son almacenados como programas en la memoria OTP y son utilizados internamente para el proceso de detección [24].

#### 3.2.5.1. Pinout

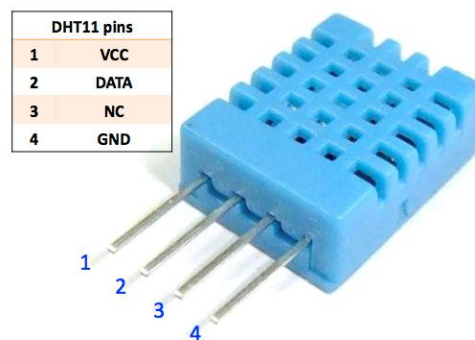


Fig. N°34 -DHT11 pinout [24]

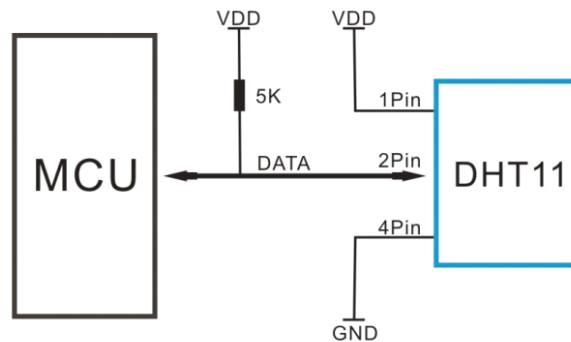


Fig. N°35 -Diagrama de circuito- DHT11 [24]

La conexión es sencilla, solo cuenta con 3 terminales de uso, 2 para la alimentación y uno para la salida de datos, como se puede ver se recomienda el uso de una resistencia de 5k trabajando como pull-up para asegurar una lectura correcta del sensor.

### 3.2.5.2. Proceso de comunicación

El formato de dato single-bus es utilizado para la comunicación y sincronización entre el microcontrolador y el sensor. Cada proceso de comunicación dura 4ms, el dato es de 40bit.

Cuando el MCU envía la señal de start, el sensor cambia de modo "bajo consumo" a "modo de trabajo" esperando del MCU la señal de start completa. Una vez es completada, el sensor envía una señal de respuesta de 40 bit que incluye la información de humedad relativa y temperatura al MCU. La comunicación es **asíncrona**, es decir se debe solicitar el dato, si el MCU no envía la petición o señal de start, el sensor cambiará a modo de bajo consumo nuevamente [24]. El proceso es descrito mediante la siguiente imagen.

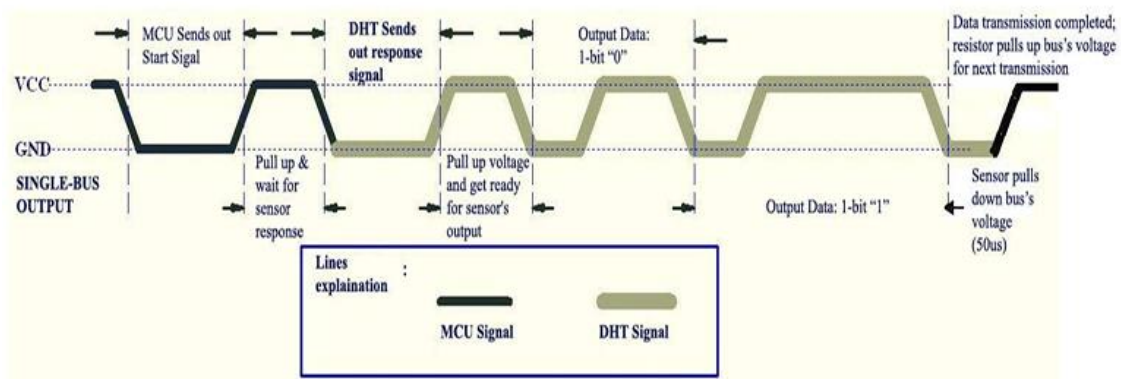


Fig. N°36 -Señal digital DHT11 [24]



### 3.2.6. ESP32

La lectura de datos, la conexión a base de datos, la conexión a internet Wifi, el generador de registros y la gestión de sensores, se realizan a través del microcontrolador. Un dispositivo que sea capaz de realizar todas estas tareas de manera eficiente es el ESP32, de Spresif Systems [25]. Se hará uso de este a través de la placa de desarrollo ofrecida por el fabricante, ESP32-DevKit [26].



Fig. N°37 -ESP32-DevKit [26]

Esta placa de desarrollo nos permite acceder a la totalidad de salidas utilizables que tiene el microcontrolador. Dentro de sus características más importantes destacamos [25]:

1. Placa Wifi, 802.11n (2.4 GHz), hasta 150 Mbps
2. Bluetooth v4.2
3. CPU 240 MHz
4. 3 interfaces UARTs
5. 4 Timers. Interno a cristal de 8 [MHz], Interno RC ajustable, interno RTC, y RTC Watchdog
6. 34 GPIOs programables
7. 2 DAC de 8 bits

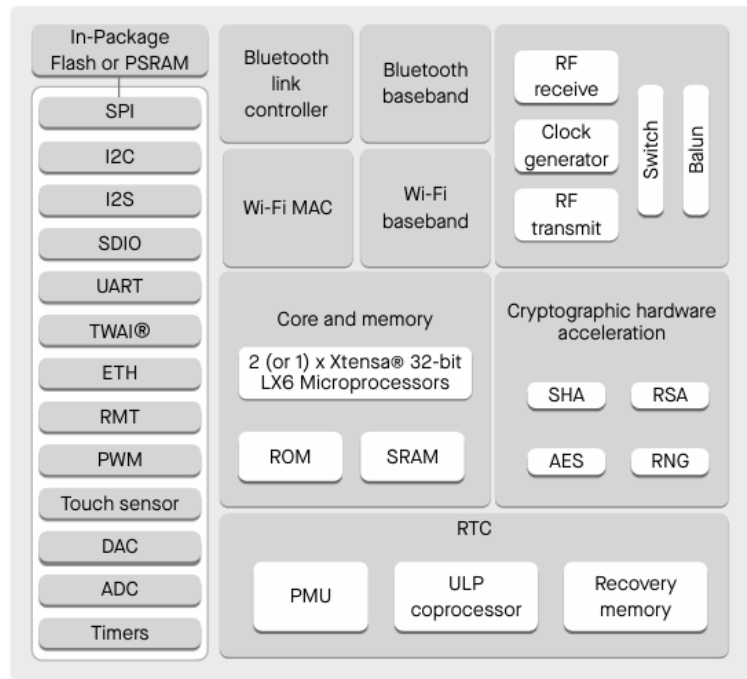


Fig. N°38 -ESP32- Diagrama en bloques funcional [25]

En la siguiente imagen se puede ver con más detalles los GPIOs utilizables que tiene la placa.

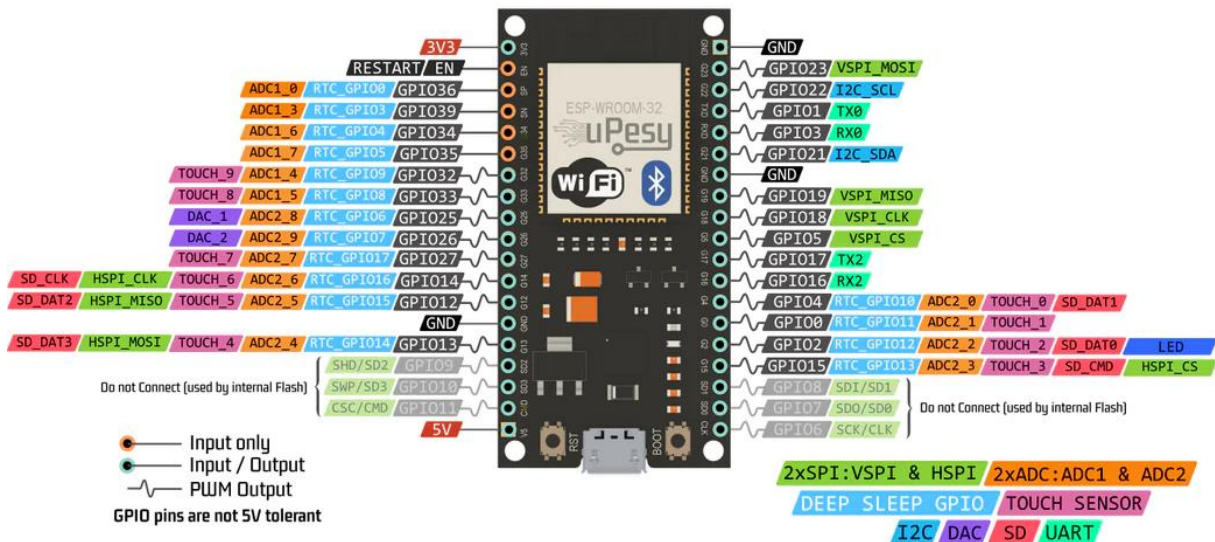



Fig. N°39 -ESP32-DevKit Pinout [14]

A continuación, se realizará una breve descripción de las características del ESP32 de las que haremos uso.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

Dentro de todas las posibilidades que ofrece esta placa de desarrollo, en este proyecto se hará uso de las siguientes características.

### 3.2.6.1. ADC


ESP32 integra dos SAR-ADC de 12 bits y soporta mediciones en 18 canales, 8 canales en para el primer ADC y 10 canales para el segundo ADC, la palabra “SAR” refiere al término **aproximaciones sucesivas** el cual es el método utilizado por el microcontrolador para la conversión analógica-digital, por más detalle sobre el funcionamiento del ADC de ESP32 se puede consultar el **anexo sección 8.4** de este documento. En este proyecto se hará uso de solamente un canal ADC para la lectura de valores de solamente uno de los sensores, el asignado para esta tarea es el GPIO35 correspondiente al canal 7 del primer ADC.

### 3.2.6.2. UART

El transmisor-receptor asíncrono universal (UART) incluido en el chip ESP32 facilita la comunicación asíncrona de datos entre el chip y cualquier dispositivo compatible con UART. El microcontrolador cuenta con tres UARTs (UART0, UART1 y UART2) independientes entre sí, es decir, es posible utilizar todas en simultáneo, aunque no es recomendable utilizar la UART0 ya que es la encargada de establecer la **comunicación** con la PC para la programación del microcontrolador, sobre más detalles de funcionamiento y control de UART se puede consultar el Anexo sección 8.4 de este documento. Para el desarrollo de nuestro proyecto, haremos uso de las tres UARTs disponibles y se asignaran basados en la siguiente tabla.

UART	Asignación/Función
UART0	Carga de código de programación a la placa, debug de funciones internas.
UART1	Control y gestión de sensores. Al ser tres sensores por comunicación UART se realiza una multiplexación para realizar comunicación por turnos.
UART2	Control y gestión de visualización de datos. La tarea principal será el envío de datos al Arduino uno que luego gestionará la pantalla para la visualización de datos.

Tabla N°6 - ESP32- Asignación de UARTs [fuente propia]

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

### 3.2.6.3. Timers

La placa ESP32 cuenta con dos **módulos timers**, cada uno con dos temporizadores ó timers, en total se disponen de cuatro temporizadores/timers de propósito general. Todos son de 64-bits basados en preescalers de 16-bits, el contador de timer puede ser de incremento o de decremento, ante un desborde del contador se produce una **interrupción** que puede ser controlada o también mediante el valor de un registro se puede comparar valores y lanzar una interrupción. En ambos casos, el contador vuelve a recargarse, es automático.

En el desarrollo de este proyecto se hará uso de uno de estos timers para la realización de tareas de aproximadamente 1 segundo, a la primera alarma se realizará la lectura de datos de los sensores, en la segunda alarma se realizará la carga de datos hacia la pantalla y la carga de datos hacia la base de datos. El cálculo de tiempo de alarma es el siguiente.

$$T[s] = \frac{T\_C * T\_DIV}{APB\_CLK[Hz]}$$

Ecuación N°.14 - ESP32-Timer, tiempo de interrupción


### 3.2.6.4. Watchdog

El watchdog es un **timer** específico utilizado para recuperar al sistema ante una falla. El microcontrolador cuenta con tres Watchdog timers, uno en cada módulo timer sistema principal watchdog timer con sus registros asociados (WDT) y uno en el módulo RTC (RWDT). Existen casos en que el contador de programa sufre una alteración y se pierde la secuencia normal o queda en un bucle infinito, y es en estos casos donde entra en acción el watchdog reiniciando el programa.

Es vital que se realice correctamente la calibración inicial de los sensores ya que son los dispositivos principales del proyecto, si no se realiza correctamente debe volver a intentarse, aquí es donde entra en acción el watchdog timer [25].

### 3.2.6.5. Wifi

Una de las principales características por la cual se opta por usar este microcontrolador es la inclusión de un módulo wifi en placa. **ESP32** está equipado con el módulo radio Wifi el cual consiste en emisor-receptor de 2.4 [GHz], bias y reguladores, balun de RF, switchs emisor-receptor y un generador de reloj. La ventaja de

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

estar integrado en placa es que se evita integrarlo mediante PCB lo cual podría inducir en errores no deseados, para este proyecto se utiliza en la comunicación a la **base de datos** y al **monitoreo remoto**.

### 3.2.7. Arduino UNO

En este proyecto se utiliza el microcontrolador Arduino UNO **ATmega328P** para gestionar y controlar la pantalla, ya que está diseñada para calzar en la plataforma de Arduino UNO y MEGA sin necesidad de otro dispositivo. En la página del fabricante se informa que esta placa de desarrollo cuenta con las siguientes características [15]:

- Procesador ATmega328P 16 MHz.
- Procesador USB-Serial ATmega16U2 16 MHz.
- 13 pines LED incorporados.
- 14 pines de E/S digitales.
- 6 pines de entrada analógica.
- 6 pines PWM.
- Comunicación UART, 12C e inspección de la inducción.
- 5 [V] de voltaje de E/S.
- 20 [mA] de corriente continua por pin de E/S.
- Memoria de 2 KB SRAM.
- Memoria de 32 KB FLASH.
- Memoria de 1 KB EEPROM.

En la siguiente figura se pueden ver la distribución de pines del Arduino UNO:

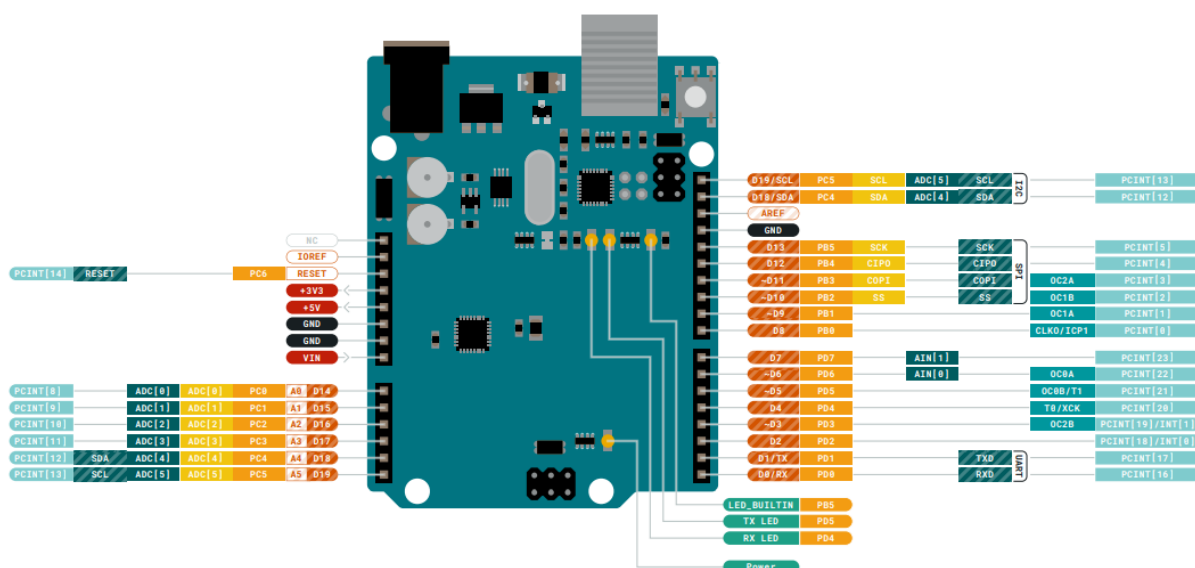


Fig. N°40 -Arduino UNO – Pinout [15]

Los datos que el Arduino UNO muestra en la pantalla se los comunica el ESP32 a través de puerto UART. La placa del **ESP32** está conectado por el puerto **UART2** con el Arduino UNO. Para poder conectar estas placas se tuvo que usar un “*conversor de niveles*” TXB0108 ya que el ESP entrega señales de entre 0-3.3V y el Arduino Uno señales entre 0-5 V.

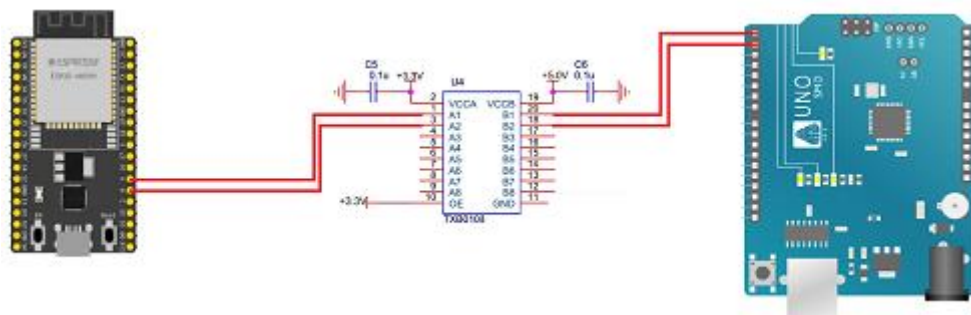



Fig. N°41 - Arduino UNO y ESP32 con TXB0108[Fuente propia]

En el Arduino UNO los pines encargados son el 0 y el 1. El 0 es el de lectura (Rx) y el 1 es el de escritura (Tx). Este puerto permite una comunicación serie asíncrona donde son configurables el formato de transmisión de datos y la velocidad. La clase serial habilita la comunicación a través del puerto UART con diversos métodos disponibles. El receptor, el transmisor y la velocidad en baudios son los componentes clave de UART. El Rx actúa como receptor de datos, luego los procesa y los convierte en datos paralelos para el sistema host. El transmisor recopila datos de

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

una fuente, los formatea en **bits seriales** y los envía a través de un pin TX (Transmisión). Por último, la velocidad de transmisión de datos la determina la velocidad en baudios [15].

### 3.2.8. Pantalla touch

Para visualizar los datos, en este proyecto, se optó por la pantalla **TFT ILI9486** de 3,5" con una resolución de 420x380 pixeles. Esta pantalla, tiene el controlador ILI9486 que nos brinda una gran variedad de comandos gráficos y procesos de visualización. Además, esta cuenta con colores de 65K en 16 bits, lo que permite que cada píxel se represente con 16 bits. Esto significa que cada píxel puede tener  $2^{16}$  (65,536) posibles combinaciones de colores [27].



Fig. N°42 -Pantalla ILI9486 [28]

Si bien esta pantalla goza de la comunicación SPI, esta no es usada en el proyecto, ya que esta se encarga de la memoria SD, la cual no se utiliza. Las pantallas TFT por lo general usan la librería adafruit, pero esta no es compatible en pantalla de 3,5 pulgadas y el controlador ILI9486. Por ende, para esta pantalla se usa la librería MCFRIEND\_kbv. Esta librería cuenta con aun más facilidades que la anteriormente mencionada, ya que, no requiere la especificación de los pines solo necesita la instanciación de un objeto TFT. A continuación, se detallan otras librerías necesarias para trabajar con esta pantalla [28].

- TouchScreen.h: Para utilizar la pantalla táctil.
- bitmap\_mono.h: Para mostrar una imagen de mapa de bits de la biblioteca.
- bitmap\_RGB.h: Para mostrar una imagen de mapa de bits de la biblioteca.
- SPI.h y SD.h: Para usar la tarjeta SD.

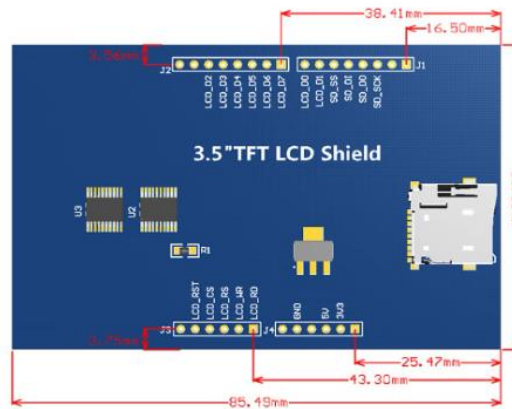


Fig. N°43 -Reverso de Pantalla TFT [29]




No	Pin	Description
1	LCD_RST	LCD bus reset signal, low level reset
2	LCD_CS	LCD bus chip select signal, low level enable
3	LCD_RS	LCD bus command / data selection signal, low level: command, high level: data
4	LCD_WR	LCD bus write signal
5	LCD_RD	LCD bus read signal
6	GND	Power ground
7	5V	5V power input
8	3V3	3.3V power input, this pin can be disconnected
9	LCD_D0	LCD 8-bit data Bit0
10	LCD_D1	LCD 8-bit data Bit1
11	LCD_D2	LCD 8-bit data Bit2
12	LCD_D3	LCD 8-bit data Bit3
13	LCD_D4	LCD 8-bit data Bit4
14	LCD_D5	LCD 8-bit data Bit5
15	LCD_D6	LCD 8-bit data Bit6
16	LCD_D7	LCD 8-bit data Bit7
17	SD_SS	SD card SPI bus chip select signal, low level enable
18	SD_DI	SD card SPI bus MOSI signal
19	SD_DO	SD card SPI bus MISO signal
20	SD_SCK	SD card SPI bus clock signal

Fig. N°44 -Pantalla Touch – Pinout [29]

En el anexo, se muestran las funciones con las que se pueden dibujar patrones o hacer determinadas acciones en la pantalla.

### 3.3. Adquisición de datos

En esta sección se abordará la metodología optada para la adquisición de datos desde los sensores, se distinguen dos métodos, el primero tiene que ver con lectura mediante uso del **ADC** del microcontrolador, el segundo es el uso de uno de los puertos **UART**. Se explicarán procesos y técnicas utilizadas sin inclusión de código de programación, si se desea ver el detalle se puede recurrir al **anexo**.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

La lectura de temperatura y humedad se realiza por medio del sensor DHT11 del cual se hace uso de una librería de ESP para la lectura de dato.

### 3.3.1. Lectura por ADC

El gas Monóxido de carbono (CO) será medido a través del sensor MQ7, el cual cuenta con una salida analógica que brinda la concentración de CO. Esta salida será tomada e interpretada por el microcontrolador a través de su ADC, en este caso se hace uso del canal 7 del ADC 1. El proceso de lectura de CO por ADC es el siguiente:

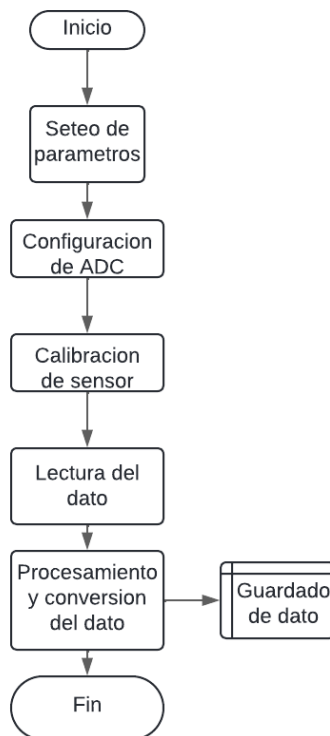



Fig. N°45 -Adquisición de datos ADC- Diagrama de flujo

Lo primero a realizar, es establecer los parámetros necesarios para el cálculo de la concentración del gas una vez leído el dato. Nos referimos a constantes del sensor calculadas en la **sección 3.2.1** de este documento y al método de cálculo de concentración del gas (lineal o logarítmica). Aquí mismo, se indica bajo que pin se realizaran las lecturas del gas y, sabiendo que en valores de voltaje nuestras lecturas no superan los 1.3[V], es conveniente realizarlas en este rango para lograr una mayor resolución.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

Se establecen valores de **atenuación de ADC** para que realice lecturas entre los rangos de 100 [mV] y 1.325 [V], se realizará el cálculo por método logarítmico.

Una vez seteados los parámetros y configurado el ADC se procede a calibrar los sensores que así lo requieran, como el de CO y CO2. Para calibrarlos será necesario ubicar el dispositivo en un ambiente libre de contaminantes, como un laboratorio o una sala limpia. Luego, para el caso del sensor de CO, realizar lecturas y establecer el parámetro “*ratio*” del sensor, para luego ser comparado con las nuevas lecturas en el ambiente a medir y calcular la concentración de CO en ese ambiente. Ya calibrado el sensor, es posible realizar lecturas continuas de concentración de CO.

### 3.3.2. Lectura por puerto serie UART

El gas formaldehído (HCHO), dióxido de carbono (CO2) y las partículas PM10 y PM2.5 serán medidos a través de los sensores **ZP16**, **MHZ-19** y **ZH06** respectivamente, los cuales soportan protocolo serie para la comunicación y lectura de datos con el microcontrolador. El proceso de lectura del dato realizado por el microcontrolador para cada uno de los sensores, salvo la trama, es el mismo.

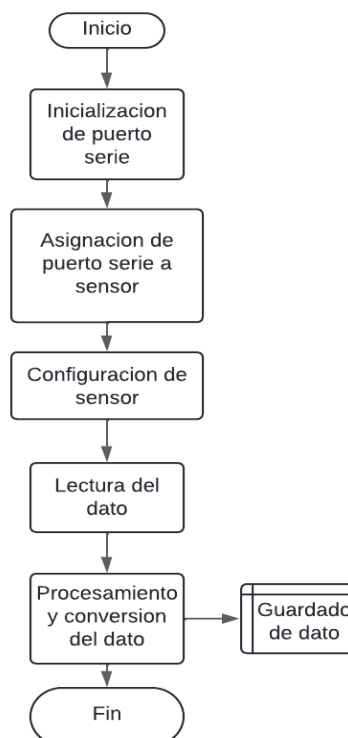



Fig. N°46 -Adquisición de datos UART- Diagrama de flujo

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO <sub>2</sub> y Partículas
Ingeniería en Electrónica	3. Desarrollo

Contamos con la ventaja de que todos los sensores requieren la misma configuración UART para su comunicación (ver tablas N°3, N°4 y N°5), lo primero a realizar es la elección del puerto UART y su inicialización. Como se había mencionado en la **sección 3.2.6.2** de este documento, contamos con tres controladoras UART, la UART1 es la asignada para la comunicación con los sensores.


Ya inicializada la UART1, se procede a configurar cada uno de los sensores, pero surge un problema. El protocolo UART puro es 1 a 1 o punto a punto, es decir, que un dispositivo transmisor (sensor) se conecta directamente a un dispositivo receptor (microcontrolador) y en nuestro caso, contamos con **tres transmisores** (sensores) para conectarse a un solo receptor, tampoco es posible realizar un conexasión paralelo directo (habría colisión de datos). La solución a este problema es la multiplexación mediante un **MUX/DEMUX** digital bidireccional CD4051, el cual desarrollaremos más adelante, de modo que mediante control digital estableceremos la conexión física entre microcontrolador y cada uno de los sensores por separado [30].

### 3.3.2.1. MHZ-19



Fig. N°47 -Sensor MHZ-19 [21]

Este sensor nos brindará la concentración del gas dióxido de carbono, tiene una sola forma de funcionamiento, se le debe enviar la petición y esperar una respuesta. Además, permite establecer rangos de medición según se requiera (2000/5000/10000 ppm) a fin de obtener mayor o menor resolución en la medición. Aunque ya viene calibrado de fábrica y trae consigo la opción de *auto calibración*, es posible realizar una calibración manual para la cual el sensor debe estar expuesto a un ambiente de baja concentración de CO<sub>2</sub> (~400ppm) al menos unos 20 minutos antes de mandar la trama de puesta a cero. Esta trama le indicará al sensor, que la próxima lectura que realice será un valor de baja concentración de CO<sub>2</sub> (~400ppm) [21].

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

### 3.3.2.2. ZH06

Este sensor, nos brindará la concentración de partículas en unidades de  $\frac{\mu g}{m^3}$ . A diferencia del sensor anterior, no contamos con una librería para su gestión, se deben armar las tramas y enviar las peticiones manualmente pero el proceso es el mismo al indicado más arriba y, para su gestión se creó una clase con métodos y atributos con el fin de ser más eficientes en escritura de código (ver anexo).

La lectura de la concentración de partículas conlleva un proceso, en primera instancia se debe enviar la petición del dato, luego se debe analizar la respuesta por parte del sensor mediante un “checksum” para determinar que la trama fue recibida correctamente. Una analizada la respuesta y verificada que es correcta, se procede a extraer el dato de PM10, PM2.5 y PM1. Una vez extraídos los datos, están disponibles para ser enviados a la base de datos y/o a la pantalla en unidades  $[\frac{\mu g}{m^3}]$  o en unidades ppm (se realiza la conversión manualmente) según se desee. Este sensor es calibrado de fábrica y no permite la calibración de forma manual [22].

### 3.3.2.3. ZP16

Zp16 es el modelo del sensor utilizado para realizar lecturas de formaldehído (HCHO), comunica concentraciones de HCHO en unidades de  $[\frac{mg}{m^3}]$ . En similitud al sensor ZH06 se deben armar las tramas para enviar las peticiones de datos o control del sensor. Tiene la particularidad de que la comunicación serie es por UART con niveles lógicos **TTL 5[V]**, a diferencia de los otros dos sensores que tienen niveles **TTL 3.3[V]**, ya que nuestra placa trabaja con niveles lógicos de 3.3 [V] se debe hacer una conversión de niveles. Para ello, utilizamos el conversor de niveles de 8 canales bidireccionales **TXS0108** [31], que nos permite comunicarnos al microcontrolador en ambos sentidos sin problemas. En cuanto a software también se creó una clase para su tratamiento, el proceso de lectura es el mismo y tiene comandos similares a los demás sensores.

Como se había mencionado, las unidades de concentración que ofrece este sensor son diferentes a la mayoría, para una mayor facilidad de lectura por parte del usuario se realiza la conversión a valores ppm aplicando la ecuación N° 12 descrita en la sección 3.2.4.3 [23].

### 3.3.2.4. Conversor de niveles

El conversor de niveles TXS0108E es un conversor **bidireccional** de 8 bits y será utilizado para adaptar niveles lógicos de 5[v] a niveles lógicos de 3.3[v] y viceversa [31].

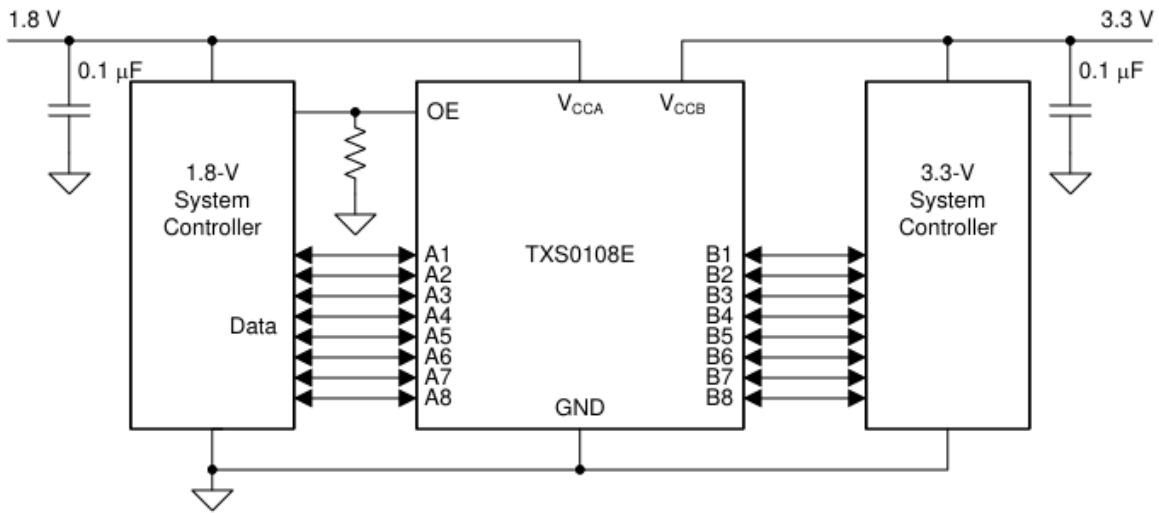
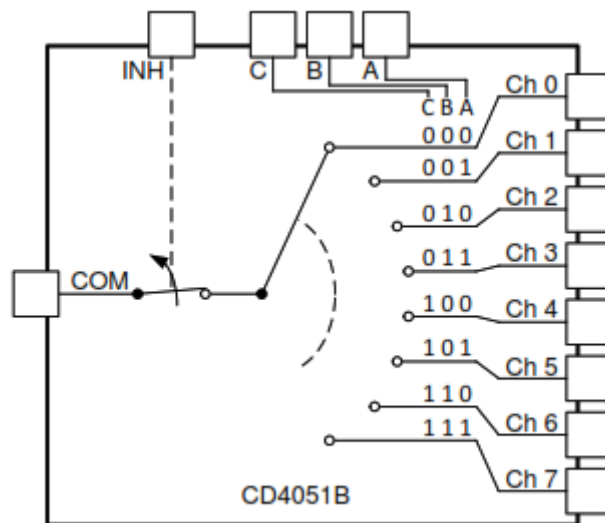


Fig. N°48 -Convertor de niveles lógicos TXS0108 [31]

La imagen extraída de su datasheet indica un ejemplo típico de aplicación, soporta dos alimentaciones diferentes, pero deben unirse sus masas. El lado A debe alimentarse con la tensión más baja, en nuestro caso 3.3 [v] soportando niveles de alimentación de 1.4 [v] a 3.6 [v] y el lado B con la tensión más alta que en nuestro caso es 5 [v] soportando niveles de alimentación de 1.65 [v] a 5.5 [v]. Mas adelante se podrá ver cómo es implementado en nuestro circuito [31].

### 3.3.2.5. Multiplexación de señales UART

Para esta tarea, hacemos uso del multiplexor/demultiplexor **CD4051** controlado digitalmente por 3bits. El funcionamiento es simple, dependiendo el valor digital de control se conmuta la salida con la entrada correspondiente.




 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

Fig. N°49 -Multiplexor CD4051 [9]

Es de utilidad para conmutar, cada lazo transmisor de cada sensor con el receptor del microcontrolador, y cada lazo receptor de cada sensor con el transmisor del microcontrolador, en síntesis, será necesario el uso de dos integrados. Ofrece la ventaja de ser de bajo ruido y ser de baja impedancia lo que resulta muy útil para la función requerida.

La selección del canal se realiza según la siguiente tabla

INPUT STATES				ON CHANNEL(S)
INHIBIT	C	B	A	
<b>CD4051B</b>				
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	X	X	X	None

Tabla N°7 - CD4051 selección de canal [9]


Como sólo debemos conmutar entre tres sensores, solo nos basta con el control de dos bits. El bit C lo pusimos a masa asignando un valor digital cero, las entradas sobrantes fueron conducidas a masa mediante una resistencia pull-down con la finalidad de reducir la entrada de ruido al circuito integrado. Basado en la figura aquí arriba presentamos la tabla con la asignación de canales para cada sensor [31].

Sensor	Canal	B	A
ZH06	0	0	0
MHZ-19	1	0	1
ZP16	2	1	0

Tabla N°8 - CD4051- Asignación de canales

### 3.4. Visualización de datos

En la introducción de este proyecto se abordaron los contaminantes del aire que afectan a la salud humana y medio ambiente. Por ello, la **visualización** y la fácil

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

interpretación de estos en el dispositivo es de vital importancia. Esto facilita la toma de decisiones de los usuarios para poner en marcha un plan de cuidados para ellos y sus seres queridos. En esta sección se abordará el desarrollo de la visualización de datos sin detallar código de programación (para ello ver Anexo), sino que, detallando la **metodología** utilizada.

#### 3.4.1. Transmisión de datos

Los contaminantes que se visualizan en nuestra pantalla son PM1, PM2.5, PM10, CO2, CO, HCHO, temperatura y humedad, como se mencionó anteriormente.

Los datos son recogidos por el ESP32 y transmitidos por la UART 2 al Arduino uno. Los archivos elegidos para transmitir los datos entre las placas es del tipo **JavaScript Object Notation** (JSON), ya que, permiten una forma fácil y simple de compartir datos estructurados. Si bien los archivos JSON fueron originariamente hechos para JavaScript, estos gozan de una simplicidad de uso, versatilidad y compatibilidad con la mayoría de los lenguajes de programación [32]. Para más información sobre la estructura del archivo JSON ver anexo.

##### 3.4.1.1. Diagrama de flujo de programa

Tal y como se mencionó, la placa de desarrollo Arduino UNO se encarga de recibir los datos y mostrarlos a través de la pantalla táctil. A continuación, se ilustra mediante diagrama de flujo, como es el programa que lleva a cabo la visualización de los datos.

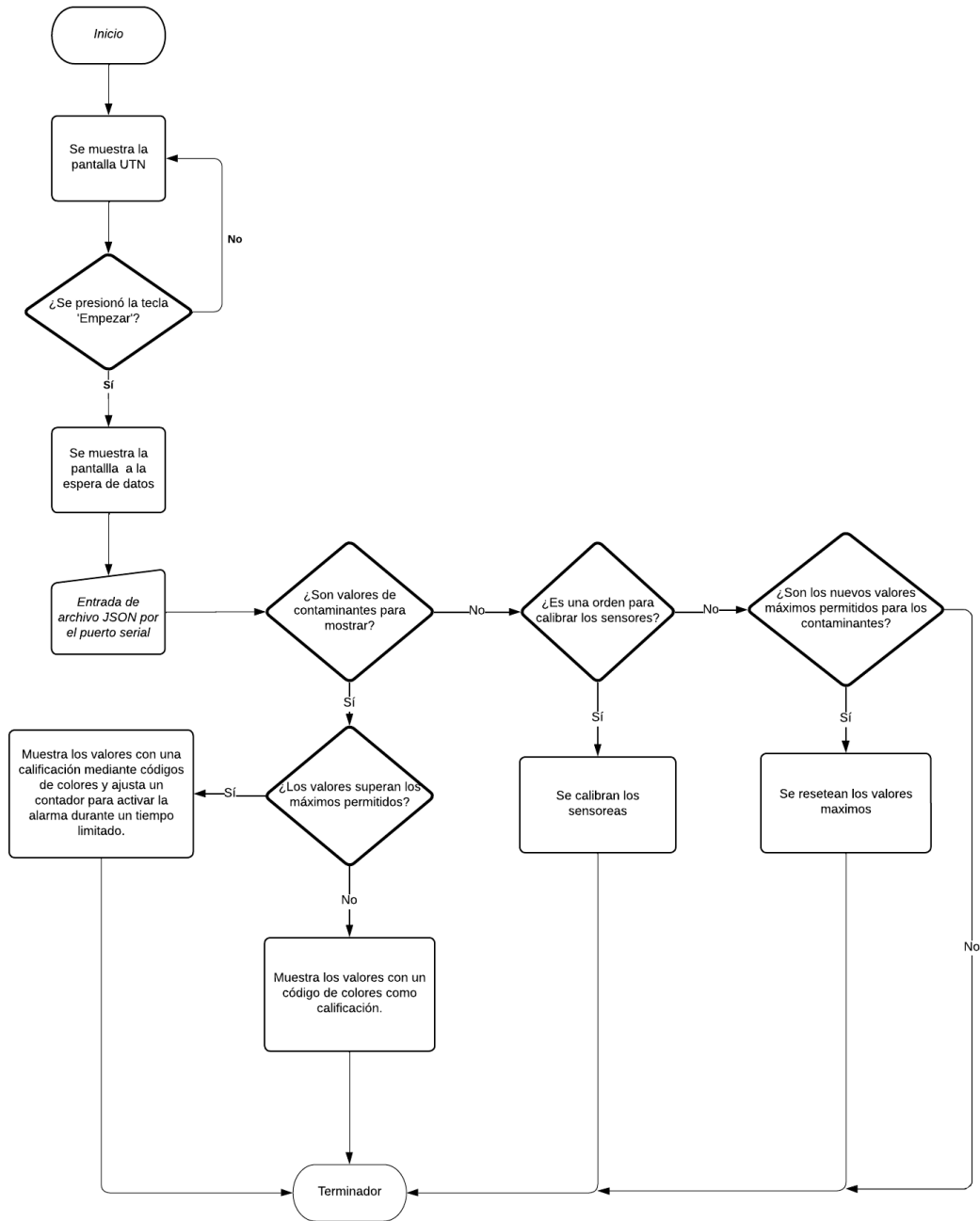



Fig. N°50 -Visualización de datos – Diagrama de flujo[Fuente propia]

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

Una vez que inicia el programa, queda a la espera de entrada de datos, al llegar el archivo JSON por su puerto serial, los datos son leídos y almacenados en un **buffer**. Este último, es analizado a fin de verificar la validez de los datos, si son correctos y no existe ningún error se interpreta y se actúa en consecuencia según lo detalla el diagrama aquí arriba.

### 3.4.2. Diseño de interfaz gráfica

La interfaz gráfica se prediseño bajo la herramienta “Canva”, una plataforma de diseño gráfico que permite crear plantillas personalizadas, con el fin de ordenar los datos para que el usuario pueda visualizar la información de manera rápida y clara. Este prediseño lo replicamos luego en código Arduino mediante funciones de las librerías de la pantalla.



Fig. N°51 -Prediseño de Interfaz[fuente propia]

La aplicación se inicia con una pantalla de bienvenida y debe pulsarse “empezar” para ver la información.



 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo



Fig. N°52 -Bienvenida – Interfaz gráfica[fuente propia]

La imagen es cargada en la interfaz mediante un llamado a la función “*utn()*” (ver anexo) en el *setup*. La siguiente imagen que aparece es el fondo de la pantalla sin el ingreso de datos. Este fondo se crea con el llamado de la función “*preparar()*” (ver anexo), después de interactuar con el botón *empezar*.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

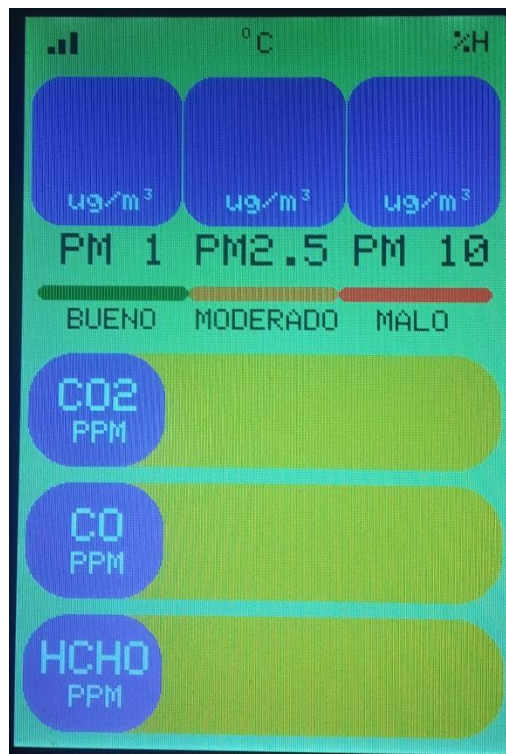


Fig. N°53 -Pantalla cargada en espera de datos

La siguiente pantalla va a describir la anterior con los datos y las alertas visuales. Esta pantalla se abre cuando el Arduino recibe un archivo JSON y abre la función "mostrar()"(ver anexo). Además, se grafican rectángulos del mismo color que el fondo para borrar el dato anterior [15].


 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo




Fig. N°54 -Pantalla mostrando datos recibidos[fuente propia]

El código verifica si alguna medición ha alcanzado un valor crítico para activar una alerta visual y sonora modificando el contador llamado “*alarma*”. La alerta visual se representa mediante un círculo verde o rojo en el indicador de conexión Wifi y un rectángulo de color verde, anaranjado o rojo en los indicadores de partículas y gases. En caso de alerta se indica con el color pertinente y se acciona un “*buzzer*” para notificarle al usuario de lo sucedido.

### 3.5. Gestión de datos

Resuelto el método para la adquisición de datos de los sensores, debemos realizar la gestión de estos. Sin importar la cantidad de parámetros leídos, el tratamiento debe ser el mismo, homogeneizar unidades en la medida que sea posible, agrupar los datos en un vector y publicarlos tanto en pantalla como en la base de datos. En resumen, la tarea del firmware **ESP32** será gestionar los sensores, adquirir y publicar datos.

Estas tareas no son realizadas al azar, por intervalo de tiempo definido leemos y publicamos los datos. El **control por tiempo** es realizado por **interrupciones** por timer del mismo ESP32, se realizó la configuración de un solo timer para que lance una interrupción cada aproximadamente 2 segundos, dentro de cada una se incrementa una variable y se habilita una “*bandera*” que indica la entrada a la interrupción, luego de la “*primer entrada*” se realiza la adquisición de datos, el método para la adquisición fue explicado anteriormente y su lógica consiste en adquirir el dato, acon-

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

dicionarlo de ser necesario y almacenarlo en un vector de datos. Luego de la “segunda entrada” se procede a publicarlos, se tomará los valores almacenados anteriormente en el vector, se muestra en monitor serie para “debug”, se carga en la **base de datos** y se envían por puerto serie a la pantalla para su visualización local, cada 30 minutos se publican datos en la base de datos para realizar un **registro histórico**. Debe entenderse que, al ser la segunda entrada de interrupción, los datos son publicados y actualizados cada 5 segundos para lograr el objetivo de **monitoreo en tiempo real**. A continuación, se ilustra el diagrama de flujo de nuestro “loop principal” para un entendimiento gráfico [25].

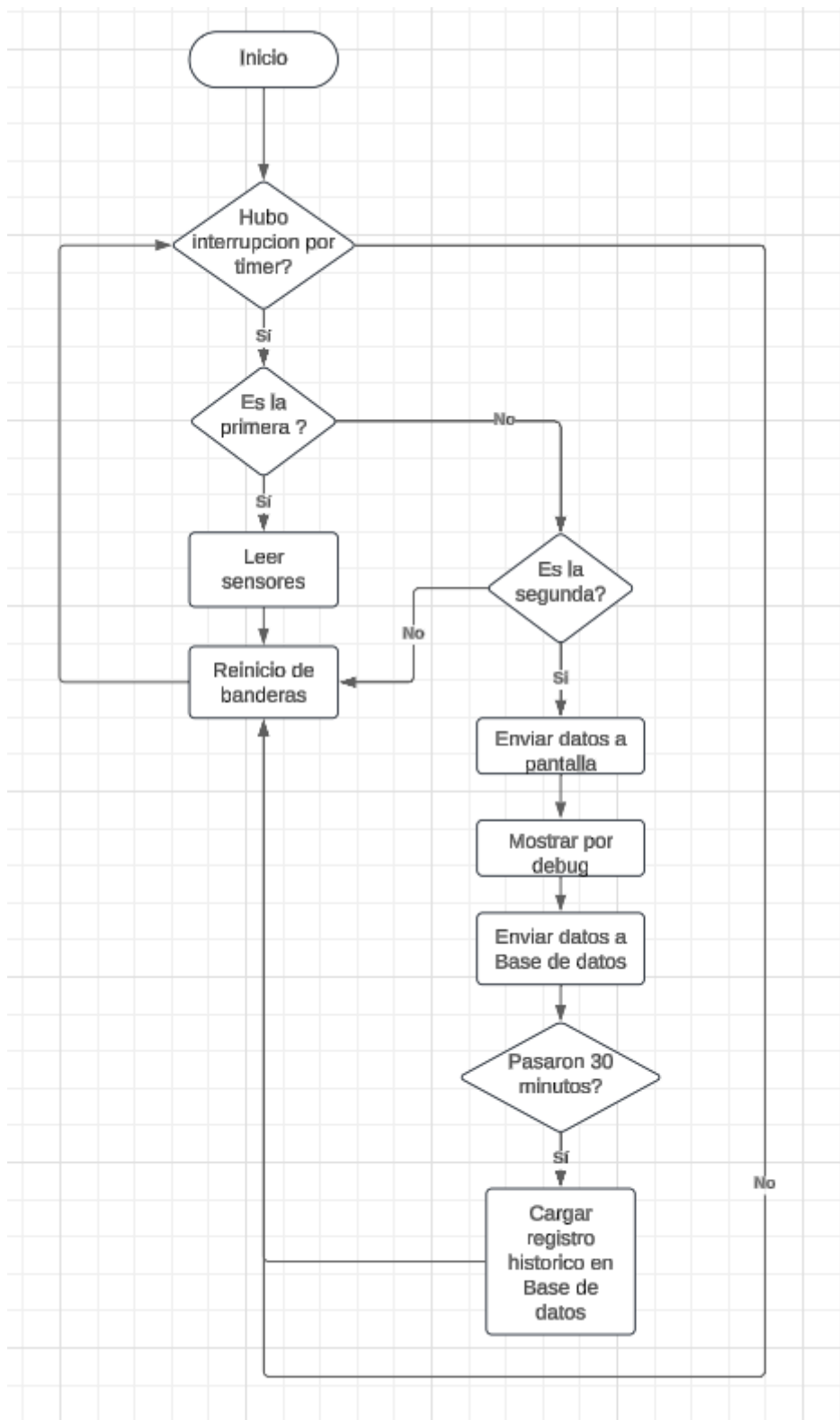


Fig. N°55 -Loop principal- Diagrama de flujo[fuente propia]



### 3.5.1. Configuración del Timer

Usamos un solo timer para controlar la gestión del programa. Ya que todos son iguales, utilizamos el “*Timer0 (Timer 0, channel 0)*” [25]. Para explicar el proceso de utilizar el timer se presenta el siguiente diagrama de flujo:

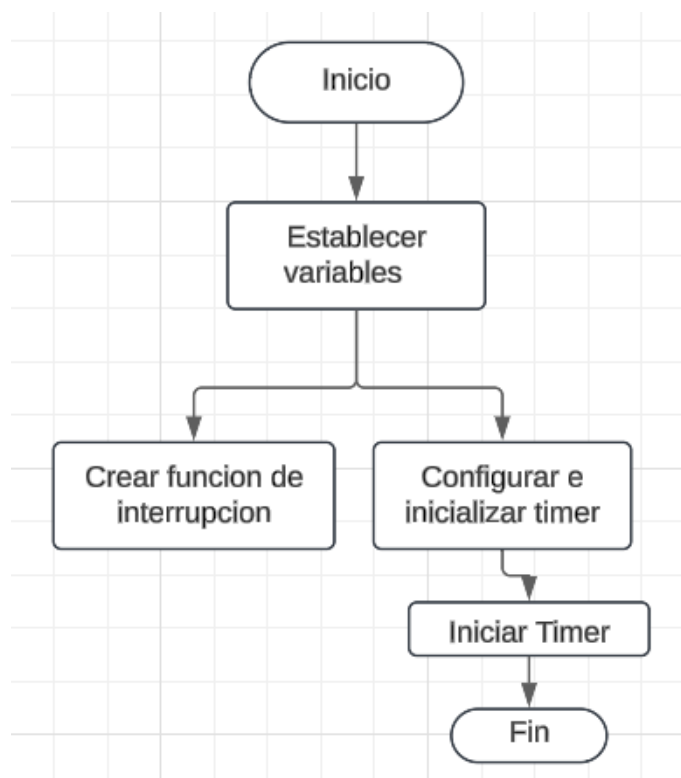



Fig. N°56 -Configuración del timer- Diagrama de flujo [elaboración propia]

Inicialmente se declaran las variables necesarias tanto para la configuración del timer como para la ejecución de la **interrupción**. Luego se crea una función para configurar el tiempo en el que se lanzó la interrupción e inicializar el timer, por último, damos inicio al reloj o timer e inicia la cuenta. Ante la ocurrencia de la **interrupción** se ejecuta la función asignada en su inicialización la cual solamente establece un valor a la bandera e incrementa un contador a utilizar en el “*loop principal*”. Por más detalles sobre código de programación ver anexo.

### 3.6. Comunicación con base de datos

Uno de los desafíos más importante a la hora de desarrollar este proyecto fue, “¿Cómo realizar la lógica de monitoreo IOT?” La respuesta fue, “usando base de

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

*datos*". Sin necesidad de montar un "servidor web" con nuestro **ESP32**, lo cual consumiría muchos recursos que son escasos. La base de datos está situada en una plataforma web gratis llamada "firebase" [16] y será la solución a la comunicación entre nuestro **hardware** y nuestra aplicación móvil. **Firestore** es una plataforma para el desarrollo de aplicaciones móviles y web que pertenece a Google actualmente.

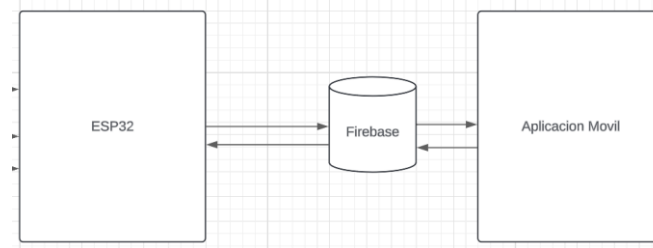


Fig. N°57 -Diagrama ESP32-Aplicación móvil[fuente propia]

Dentro de Firebase utilizamos una **base de datos en tiempo real (RTBD)**, que, a diferencia de una base de datos corriente, se debe mantener la conexión a fin de actualizar sus registros de forma casi inmediata logrando así, el concepto de "tiempo real" de manera que una actualización en un registro desde el microcontrolador se vea reflejada **automáticamente** en la base de datos y en la **aplicación móvil**.

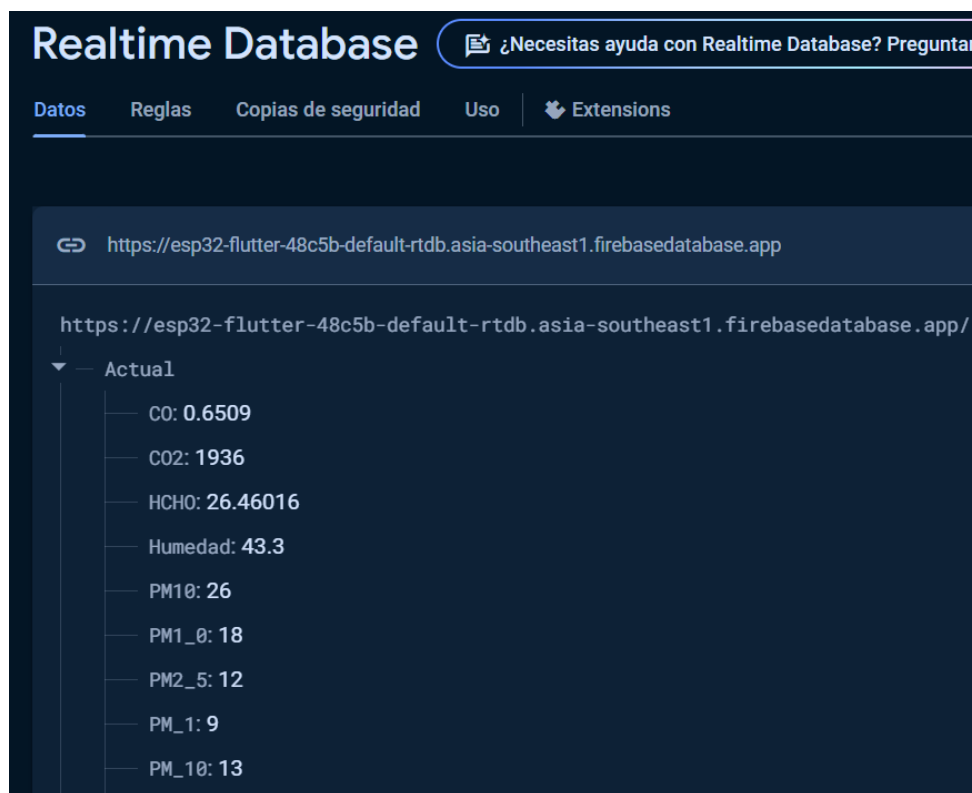



Fig. N°58 -Captura de RTBD – Datos en tiempo real[fuente propia]

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

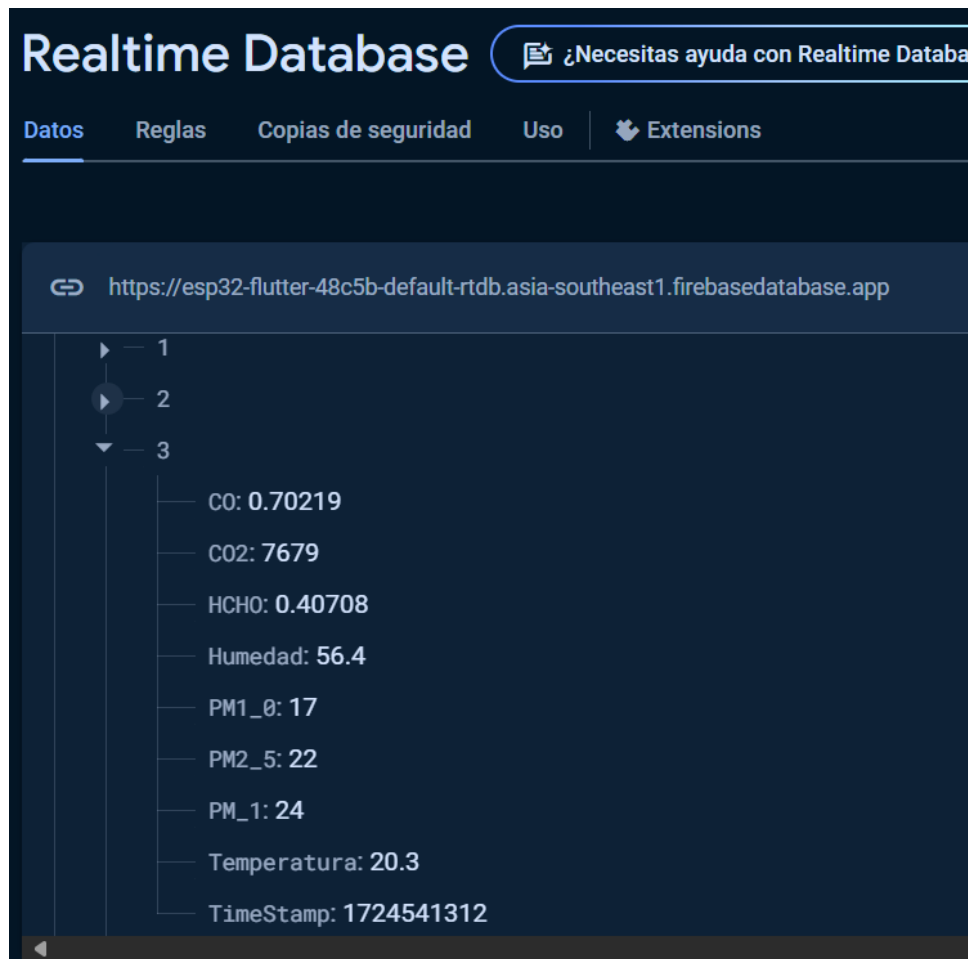


Fig. N°59 -Captura de RTBD – Registros históricos[fuente propia]

### 3.6.1. Establecer conexión

Para crear la comunicación con firebase utilizaremos una librería creada por "Mobizt" y actualizada en este año, llamada "firebaseESP32.h". Incluimos la librería y luego definimos las variables necesarias. A continuación, se presenta un diagrama de flujo que describe como fue programada en nuestra placa **ESP32**.

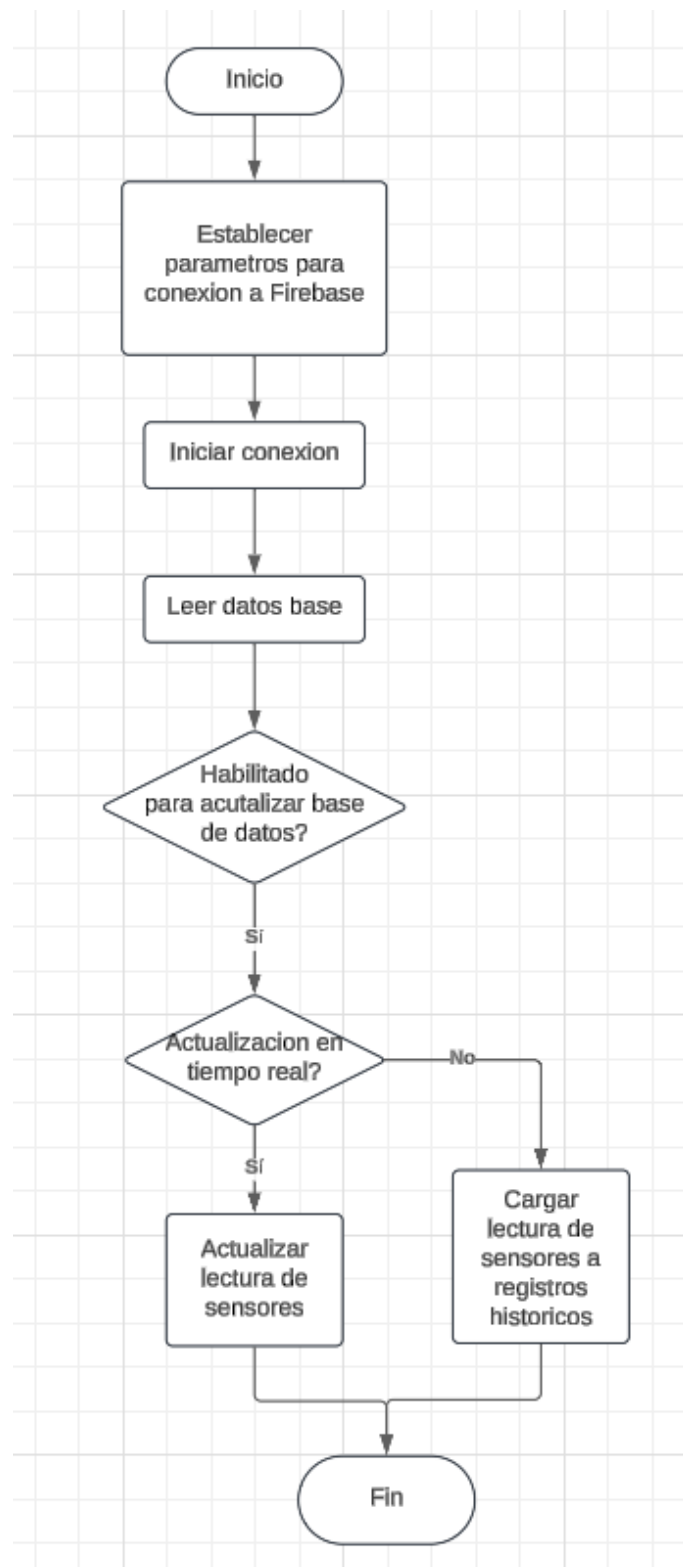



Fig. N°60 -Diagrama de flujo – Conexión a base de datos [fuente propia]

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

Para iniciar, se definen los parámetros necesarios para establecer la conexión a nuestra base de datos. Entre ellos se incluyen la dirección del servidor, la clave de acceso y las credenciales del usuario administrador. A continuación, se crean los objetos correspondientes de la librería utilizada, lo que permite iniciar la comunicación con la base de datos. Una vez que la conexión está activa, se realiza una primera lectura que obtiene dos datos importantes:

- Un valor verdadero o falso que indica si los sensores están calibrados.
- Un contador de registros históricos, útil para identificar cuál fue el último registro cargado.

### 3.6.2. Lectura y escritura de registros

El tipo de dato que gestiona la base de datos es de tipo **JSON**, esto significa que, si solicitamos un dato sin especificar el tipo, nos traerá un dato tipo JSON. Como estamos en una base de datos en tiempo real, la conexión siempre estará activa, esto nos otorga la ventaja de poder “escuchar” cambios en la base de datos, es decir, cuando un dato se actualiza se produce un evento, podremos capturar ese evento y ejecutar una acción en consecuencia.


Por otro lado, se realizan varias escrituras en diferentes direcciones en las cuales se actualiza el dato según se requiera, para las escrituras no se ejecutan/lanzan eventos, solo se actualiza el valor almacenado en la base de datos mediante el método “set”. Dicho método no especifica un tipo de dato, ya que no resulta eficiente porque la base de datos los interpreta y almacena como datos JSON.

La implementación de la fecha se realiza mediante tiempos en formato **timestamp**, este formato nos permite almacenar la fecha, hora, minutos y segundos exactos en un tipo de dato “entero”, es muy utilizado en programación y base de datos para registrar eventos o realizar tareas de sincronización, en nuestro proyecto es muy útil ya que facilita la forma de comunicar el tiempo exacto en el cual se registra cada conjunto de datos.

## 3.7. Aplicación móvil

En este capítulo se abordará el diseño de la aplicación móvil para el monitoreo remoto de nuestro proyecto. No habrá detalle sobre implementación en código de programación, en caso de que se lo requiera se puede consultar en el anexo sección 8.5.2. Se definen los siguientes objetivos a cumplir por la aplicación

- Ingreso por inicio de sesión mediante Google.


 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

- Visualización en tiempo real de las magnitudes CO, CO2, HCHO, PM10, PM2.5, humedad y temperatura.
- Menú desplegable para navegar por las diferentes páginas de la aplicación.
- Sección “registro”. Se visualizan los registros históricos ordenados primeramente por fecha y luego por hora
- Sección “gráficas de registro”. Organizado por fechas
- Sección “información”. Donde el usuario pueda informarse acerca de las magnitudes de las que es posible medir nuestro proyecto
- Sección “acerca de”. Información sobre derechos de autor y alcances del proyecto.
- Posibilidad de cierre de sesión.

Para el desarrollo de la aplicación se tuvieron en cuenta dos opciones de lenguajes de programación, Dart y Kotlin. **Kotlin** es un lenguaje nativo para Android proveniente de java, al ser nativo cuenta con la ventaja de que nuestra aplicación sea altamente personalizable, pero requiere mayor cantidad de código. **Dart** es un lenguaje de programación de código abierto desarrollado por Google, orientado a objetos y admite código multiparadigma. Se opta por el lenguaje Dart por dos razones principales, la compatibilidad sencilla con Google y por su framework, **Flutter**(ver anexo). La aplicación fue desarrollada, depurada y publicada bajo el software “VsCode”.

### 3.7.1. Desarrollo de la aplicación

Para cumplir con los objetivos planteados, la aplicación se compone de cinco “páginas” o “screens” organizadas inicialmente de forma vertical y luego de forma horizontal. Para una mayor comprensión le damos nombres a las páginas y presentamos el siguiente diagrama.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

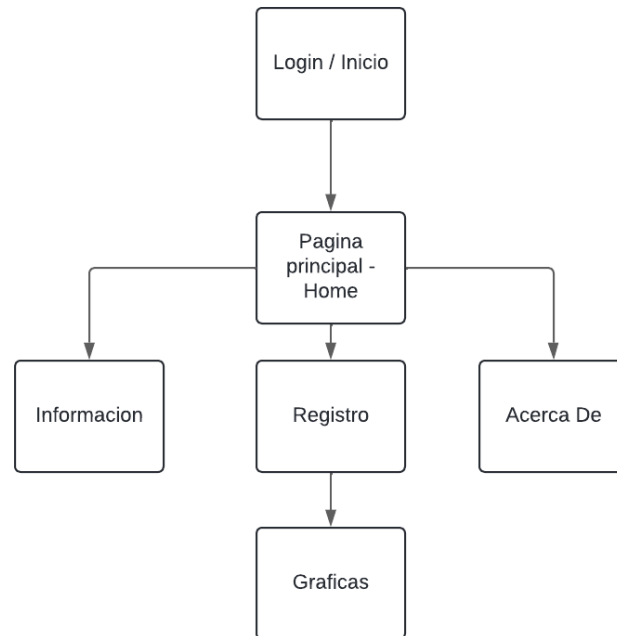



Fig. N°61 -Aplicación móvil – Organización de páginas[fuente propia]

Organización vertical significa que existen niveles superiores e inferiores entre páginas, no se puede acceder a una página sin antes pasar por su superior, ejemplo para acceder a “Home” primero debemos iniciar sesión. Organización horizontal significa que las páginas tienen el mismo nivel de jerarquía, se puede navegar entre páginas sin necesidad de pasar por una página intermedia, desde “home” es posible navegar a “información”, “registro” y “acerca de”.

Existen tres formas de crear una aplicación móvil llamadas, “sin estado”, “con estado” o “mixta” que resulta de la combinación de las dos primeras en una misma aplicación, sería nuestro caso.

Las páginas “con estado” permiten al usuario mantener su sesión, guardar datos y “recordar” su recorrido dentro de la página, permiten que se gestionen múltiples procesos en simultáneo y trabajar de forma asincrónica sin necesidad de una recarga de la página [33].

Las páginas “sin estado” cuentan con la ventaja de ser más económicas en materia de recursos, pero los datos deben ser cargados con anterioridad para ser mostrados, puede realizar procesos, pero no de forma simultánea, para mostrar un nuevo dato se necesita recargar la página [33].

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

A continuación, se irá desarrollando la secuencia de cómo fue planteada cada una de las páginas detallando la interfaz gráfica, los procesos internos que realiza y como se relaciona con las demás.

### 3.7.1.1. Página Inicio- Login


En la página “*inicio*” se realiza los preparativos necesarios para poder ejecutar el resto de la aplicación, lo primero es la conexión con firebase y a la base de datos en tiempo real, luego desarrollar un sistema de autenticación por usuarios con el fin de proporcionar seguridad a la aplicación.

Lograda la conexión a firebase y a la base de datos, se inicia nuestra aplicación donde se muestra un botón de inicio de sesión con el nombre de nuestro proyecto.

En la barra superior está el título de “*inicio de sesión*” en conjunto con el logo de la universidad, el cuerpo es el nombre de nuestro proyecto acompañado del logo de nuestra universidad y un botón que ejecuta el inicio de sesión. Por último, el pie de página son los nombres nuestros, estudiantes que realizaron este proyecto. Las siguientes imágenes ilustran el resultado final.



Fig. N°62 -Aplicación móvil- pantalla de inicio [fuente propia]

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

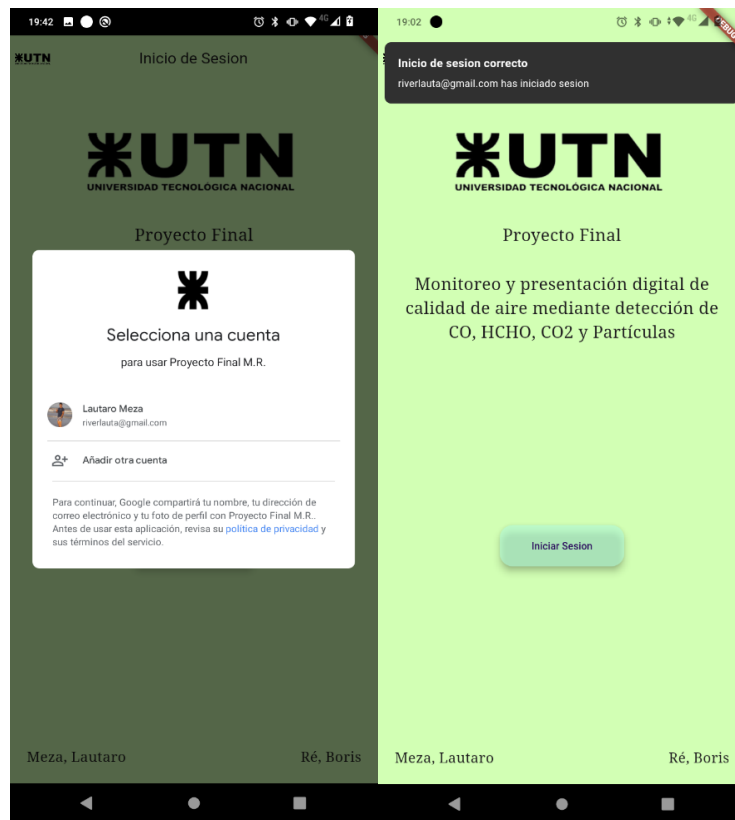


Fig. N°63 -Aplicación móvil – inicio de sesión[fuente propia]

### 3.7.1.2. Página Home

Hasta el momento estamos autenticados con usuario de Google y conectados a la base de datos, el siguiente paso es solicitar la información disponible en la base de datos. Para estas acciones nos trasladamos a la segunda página o screen la que llamamos “home” o “principal”. Los datos de autenticación son trasladados a esta página junto con los datos de conexión a la base de datos, antes de iniciar el lanzamiento de la interfaz gráfica realizamos la **solicitud** de los datos en la base de datos de manera que en la interfaz los datos ya estén cargados.

La organización de la interfaz gráfica fue pensada para que en la primera visualización el usuario pueda observar el **estado general** del sistema, los valores de temperatura y humedad. Luego desplazándose hacia abajo en la aplicación pueda visualizar los valores de cada **magnitud** medida, organizándolos en módulos expandibles donde se muestra el valor en tiempo real, su valor crítico y el estado en el que se encuentra dicha magnitud (nivel bueno, malo, regular, crítico).

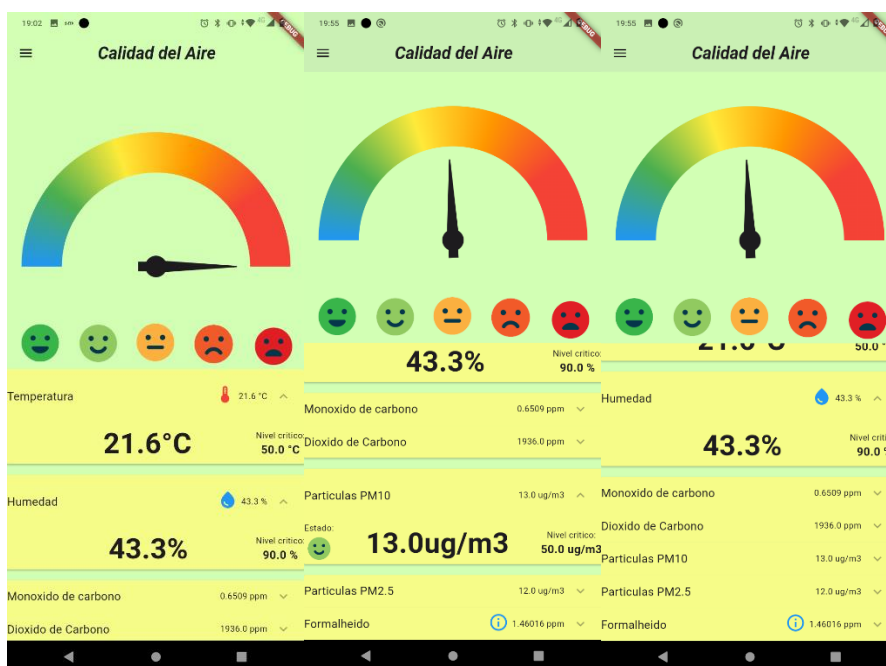


Fig. N°64 -Aplicación móvil – Capturas de página Home[fuente propia]


Para observar el estado general del sistema se creó un gráfico de rangos analógicos basados en una aguja que se mueve basada en la peor situación dentro de todas las magnitudes medidas.

### 3.7.1.3. Página Registro y Gráficas

Si bien es conveniente poder monitorear en tiempo real los gases y partículas presentes en el ambiente sería oportuno poder almacenar las magnitudes en lapsos de 30 minutos de modo que sea posible consultar los valores de concentración que estuvieron presentes en un día específico a un horario específico y también poder ver dichos valores expresados en una **gráfica diaria** que permita visualizar si hubo picos de concentración o no hubo problemas durante el día.

Se creo entonces, una página que permita ver estos datos en forma de lista y en consecuencia otra página que nos muestra las gráficas de concentración del día solicitado. Se planteo contar con 120 registros en total, cada registro se realiza cada 30 minutos lo que nos resulta en la posibilidad de almacenar de manera corrida hasta 60 horas atrás, luego los datos se remplazan desde el más antiguo con nuevos valores.

La página de gráficas depende pura y exclusivamente de la página “registro”, con esto queremos decir que la única forma de navegar hasta ella es por la página “registro” ya que está a cargo de proporcionarle la información para la creación de

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO <sub>2</sub> y Partículas
Ingeniería en Electrónica	3. Desarrollo

las gráficas. Solo tiene la tarea de mostrar los datos de esta manera, por lo que es una página “sin estado”.

La página “registro” tiene estado, de manera que sea posible mantener la sesión del usuario junto con la conexión a la base de datos, sin embargo, la información de los registros se descarga al momento de su creación y luego lanzará la interfaz gráfica, si se agrega un dato al registro no se va a ver reflejado en ella hasta que se recargue la página o se abandone y vuelva a entrar.

Para un mejor entendimiento de la metodología se muestra el resultado, obsérvese la forma en que se enlista. Primero por **fecha** y dentro de cada fecha se enlista por **hora**, luego por cada hora se muestran las mediciones. Si se toca en la fecha se lanzará la página de gráficas donde se mostrarán los mismos datos de la fecha en específico, pero en forma gráfica [34].

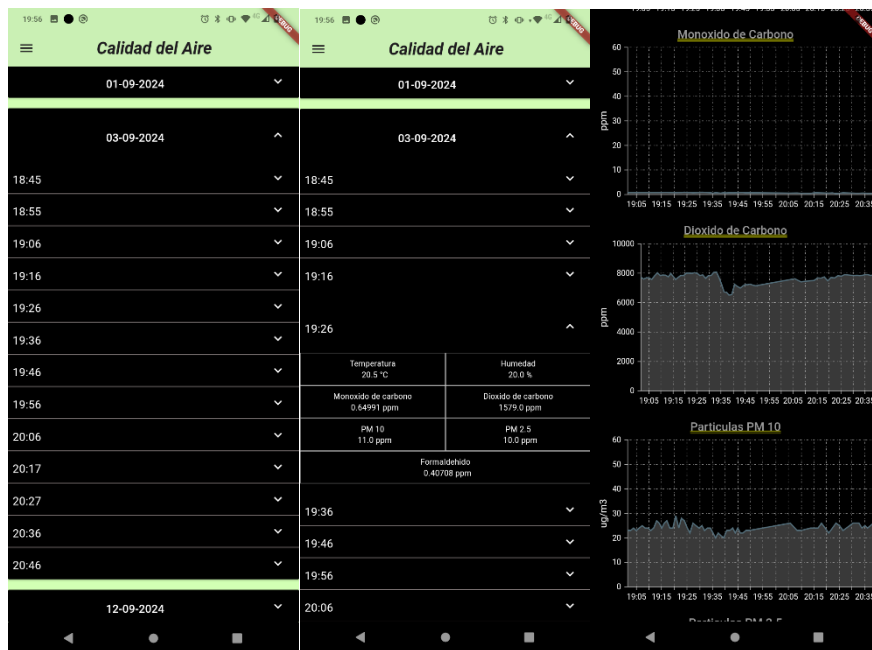



Fig. N°65 -Aplicación móvil- Página Registro [fuente propia]

#### 3.7.1.4. Página Información y Acerca De

A modo de agregado y para más información al usuario se crean las páginas “información” y “acerca de”. Ambas páginas sin estado solo contendrán texto y la presentación del menú desplegable cambiara su diseño y pasará a ubicarse en el pie de la pantalla con el fin de que no generar una distracción visual al lector y la lectura sea más agradable.

 <b>UTN Regional Paraná</b>	<b>Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas</b>
<b>Ingeniería en Electrónica</b>	<b>3. Desarrollo</b>

En la página “*información*” se pondrá a conocimiento la definición de cada magnitud medida explicando la importancia de su **monitoreo**, efectos en la salud y los valores críticos. Mediante técnicas de diseño se ordena el texto, su disposición y se logra formar un párrafo con letra legible y fácil de leer.

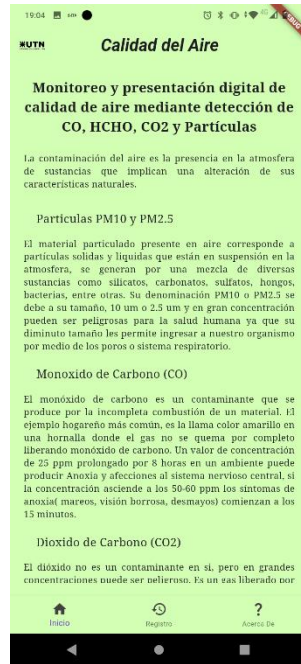



Fig. N°66 -Aplicación Móvil – página Información [fuente propia]

La página “*acerca de*” pondrá en conocimiento al usuario los derechos de autoría de la aplicación, el motivo de realización de esta y los objetivos para la cual fue creada.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

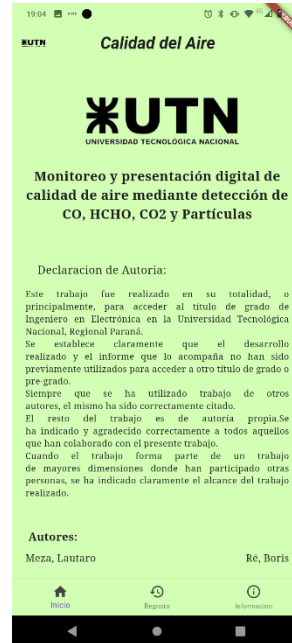


Fig. N°67 -Aplicación móvil – página acerca de [fuente propia]

### 3.8. Diseño de placa

En esta sección se tratará el diseño de la placa PCB que cumplirá la función de realizar el conexionado de las partes. Debemos implementar con circuitos e integrados el siguiente diagrama.

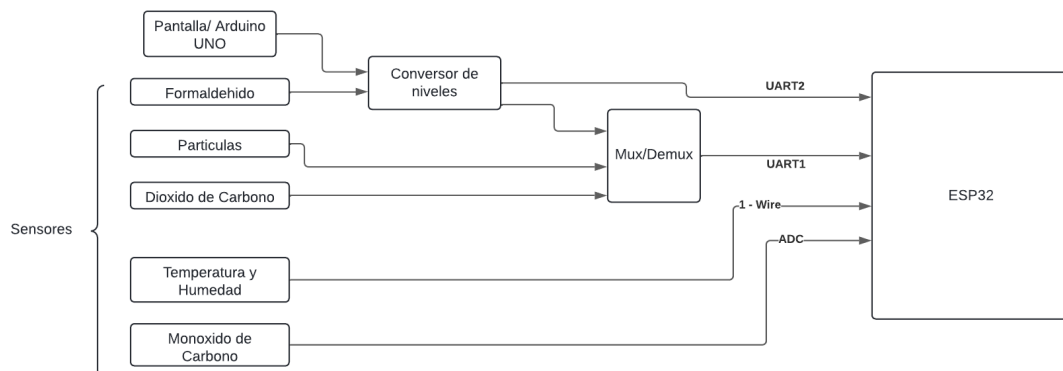


Fig. N°68 -Diseño de PCB – Diagrama circuital [fuente propia]

Los componentes ya fueron mencionados en secciones anteriores, la pantalla será gestionada por un **Arduino UNO** que se comunicará con el **ESP32** por comunicación series **UART**, la particularidad de esta comunicación es que el Arduino trabaja con señales lógicas de 5[V] y el ESP32 con niveles de 3.3[V], la misma situación



se presenta con el sensor del gas Formaldehído **Zp16** y se soluciona mediante el conversor de niveles **TXS0108BE**. Los sensores comunicados por serie UART deben pasar primero por el multiplexor y luego a la entrada de UART del ESP32, entendiéndose el caso particular del sensor Zp16.

El circuito fue desarrollado en el software “*Proteus versión 8.13*”, los sensores se conectan a la placa mediante “*headers*” tipo macho para facilitar el ensamble del conjunto placa y chasis.

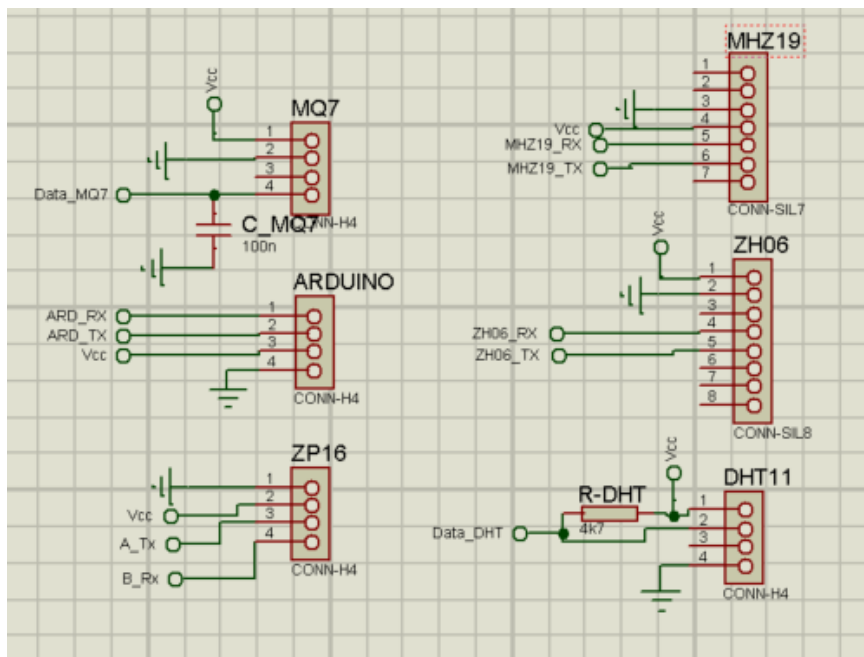


Fig. N°69 -Diseño PCB- Conectores a periféricos [fuente propia]

Se realiza el conexionado de conectores con cada uno de los integrados incluyendo el esp32 y se muestra a continuación.

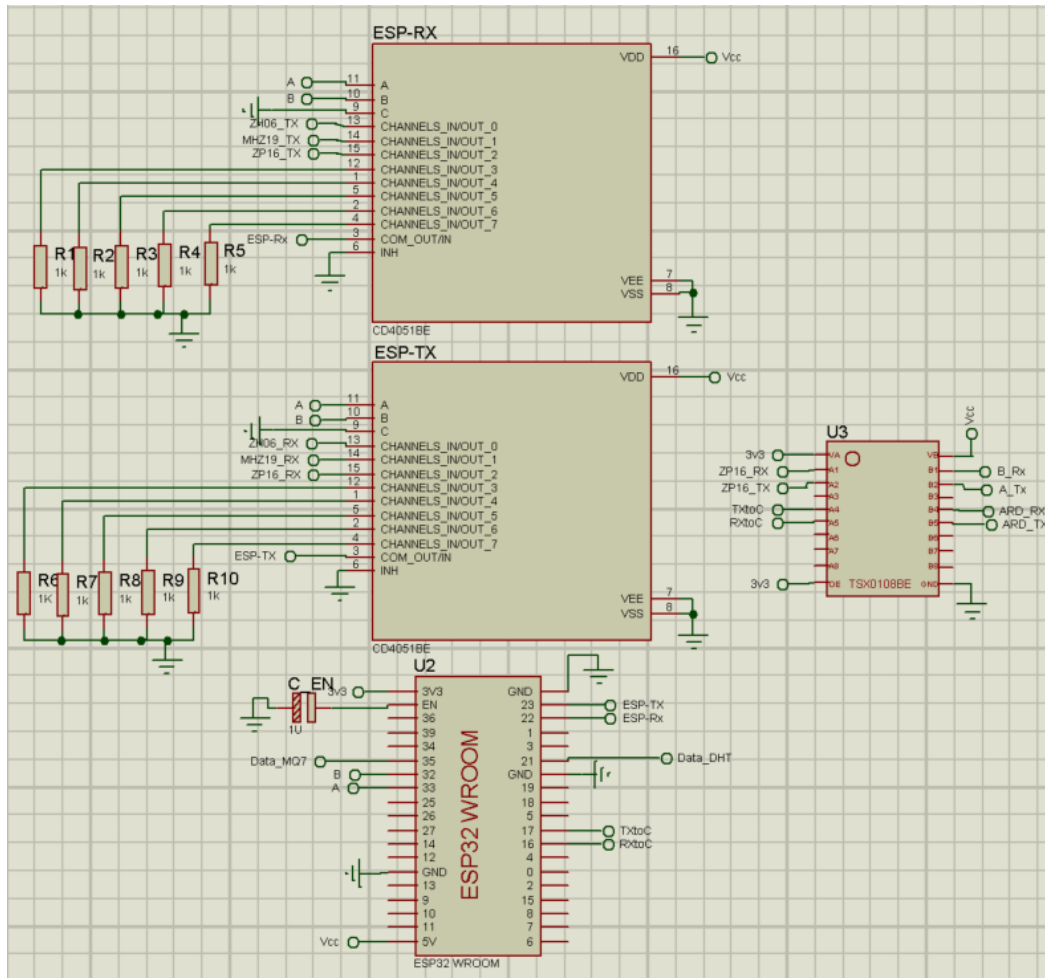


Fig. N°70 -Diseño PCB – Conexión de circuitos integrados [fuente propia]

Analizando cual debería ser la ubicación correcta de los componentes, se concluyó que sería oportuno distribuir a los laterales de la placa los conectores hacia los periféricos de manera que no ocurra un cruce de cables y mantener ordenado el circuito. La alimentación se ubica en el frente de la placa ubicando un conector “Jack” tipo Plug 3.5 [mm] para conectar la alimentación de todo el circuito, adicionalmente se ubica 2 pares de pines “headers” para una conexión de alimentación alternativa (una batería), para poder programar la placa ESP32 sin necesidad de retirarla de la placa PCB, se la ubica en el frente izquierdo. El “layout” de la placa PCB queda de la siguiente manera

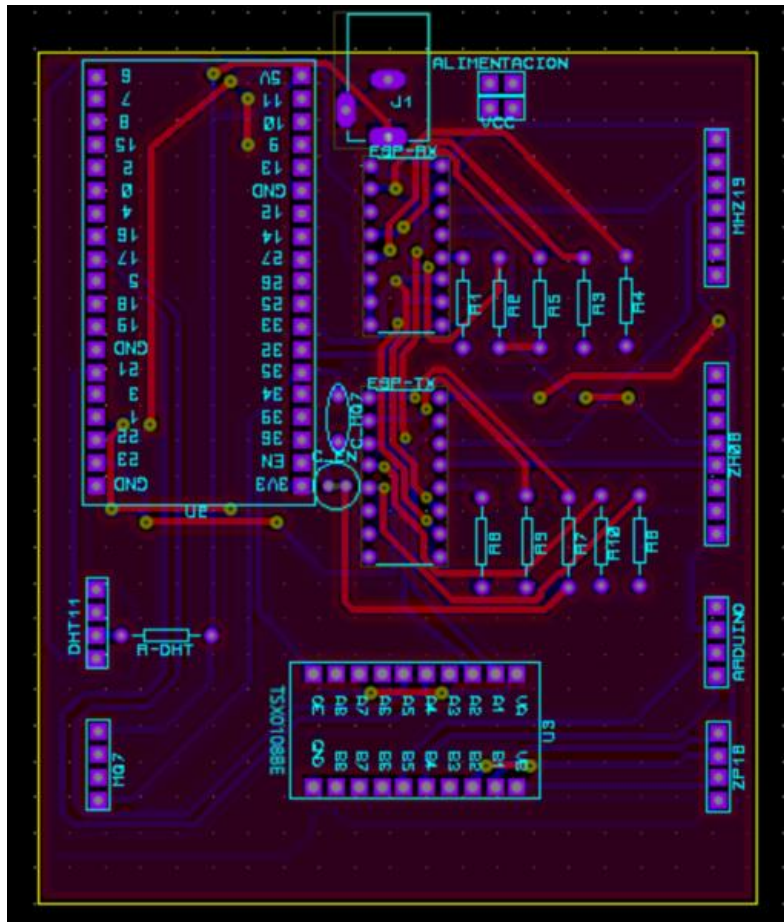


Fig. N°71 -Diseño PCB- Layout completo[fuente propia]

Se planteo un **diseño** de PCB de dos capas, con planos de masa en ambas capas ya que trabajamos con **señales digitales** y debemos reducir el ruido lo mejor posible, el paso de capa en capa se realiza mediante vías de 1[mm] de diámetro. El ruteo se realizó con espesores de 30 [mils] que equivalen a 0.89 [mm], no es necesario trabajar con espesores mayores ya que no trabajamos con señales de potencia y en su mayoría son señales digitales.

Habiendo realizado el conexionado y ruteo en su totalidad pasamos a la impresión de placa y vemos los resultados.

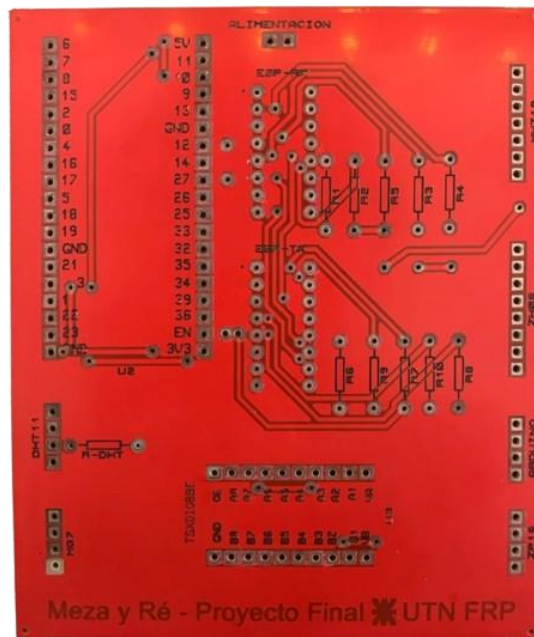


Fig. N°72 -Diseño PCB – Placa impresa [fuente propia]

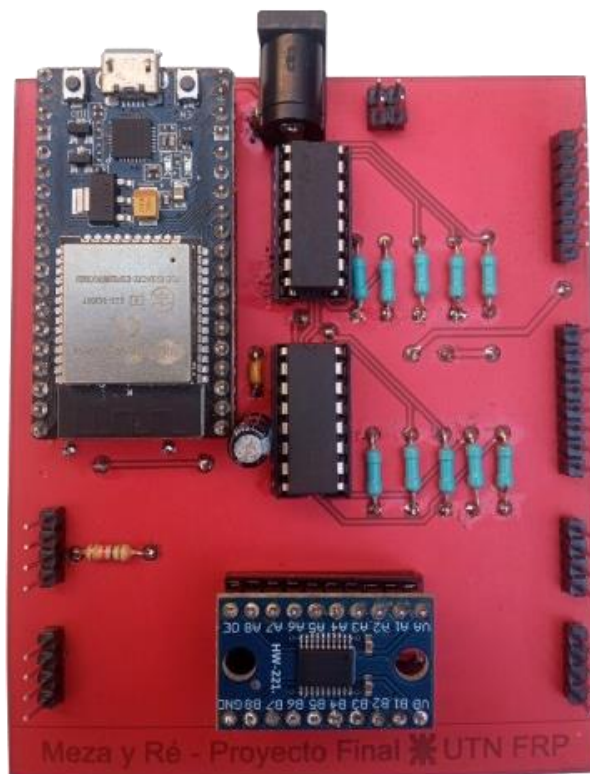


Fig. N°73 -Diseño PCB- Placa finalizada TOP [fuente propia]

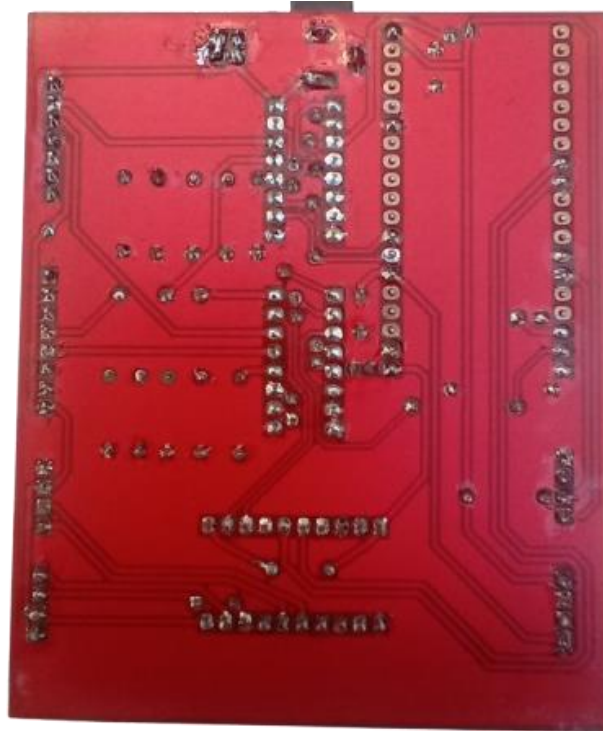


Fig. N°74 -Diseño PCB – Placa finalizada BOTTOM [fuente propia]


### 3.9. Diseño 3D

En este capítulo se describe el proceso de **diseño 3D** empleado en el proyecto, desde la selección de software hasta la estructura y ensamblaje de las piezas. El objetivo es crear una carcasa personalizada que asegure un montaje preciso y funcional de cada una de las piezas, que facilite la construcción y optimice el rendimiento del dispositivo final.

El diseño 3D se realizó con el software “*Tinkercad*”, seleccionado por ser sin licencia y de fácil manejo. Se estructuró en **capas** para permitir la realización de pruebas a medida que se imprime, con el objetivo de minimizar errores tanto en la impresión como en el diseño, cada pieza se diseñó con capacidad de encastre. La **carcasa** se divide en dos partes: una corresponde al habitáculo y la otra a los componentes que se ubican en su interior, los cuales cuentan con sus propios mecanismos de acoplamiento.

#### 3.9.1. Habitáculo

La primera capa del diseño 3D es el habitáculo que, a su vez, se subdivide en dos secciones, una inferior y otra superior, ensambladas entre sí verticalmente mediante un tornillo largo en cada una de sus esquinas.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

### 3.9.1.1. Habitáculo Inferior

Este habitáculo está destinado a alojar la placa PCB en conjunto con el ESP32 y será nuestra base del diseño 3D. Cuenta con dos “*cubiertas*” para alimentación del dispositivo y para el acceso al micro USB del ESP32 para la programación del este. En la siguiente figura se ilustra el diseño final del habitáculo inferior con medidas de 130 x 120 x 40 [mm].

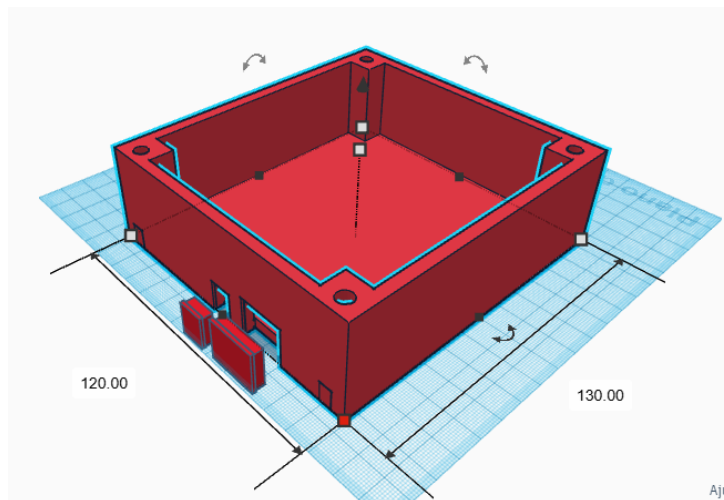


Fig. N°75 -Diseño 3D – Habitáculo inferior[autoría propia]

A fin de asegurar un encastre fijo de la placa PCB en la base del habitáculo, se crearon “*esquineros*” que tendrán la función de contener la placa PCB contra una de las paredes del habitáculo inferior. En adición, se diseñó una placa con el logo y siglas de nuestra facultad para integrarlo al habitáculo.



Fig. N°76 -Diseño 3D – Esquinero + placa con logo UTN[autoría propia]



### 3.9.1.2. Habitáculo superior

Este habitáculo está destinado a ser el cuerpo de la carcasa 3D, está formado por cuatro partes o figuras, la tapa, contendrá a la placa Arduino UNO y la pantalla, una “*cubierta*” para el acceso al puerto serie del Arduino UNO, la “*base o piso*” en la cual se encastran los sensores y la carcasa exterior que contendrá a las tres partes mencionadas anteriormente. En esta última se diseñaron hendiduras en todas sus paredes a fines de permitir la ventilación necesaria para los sensores, contendrá a la tapa y a la abertura mencionada anteriormente, sus dimensiones finales son de 130 x 120 x 60 [mm].

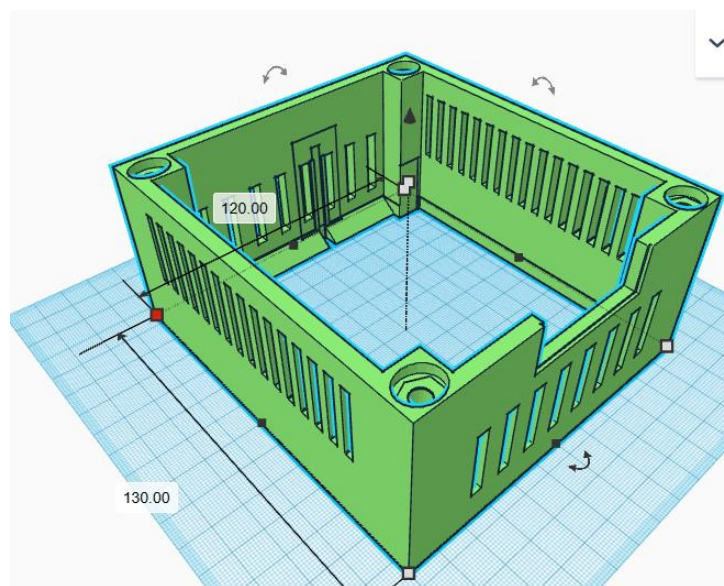


Fig. N°77 - Diseño 3D – Habitáculo sup., carcasa exterior. [autoría propia]

A continuación, se adiciona el piso del **habitáculo superior**. En su diseño se crearon aberturas a los costados que permiten a los sensores acceder a la placa PCB, así como también una guía para los cables que van desde el Arduino hacia la placa y se diseñaron “*huellas*” en las cuales encastran cada una de las bases de los sensores en su correspondiente espacio. Por último, se ilustra como va ubicada la “*cubierta*” del puerto serie del Arduino.

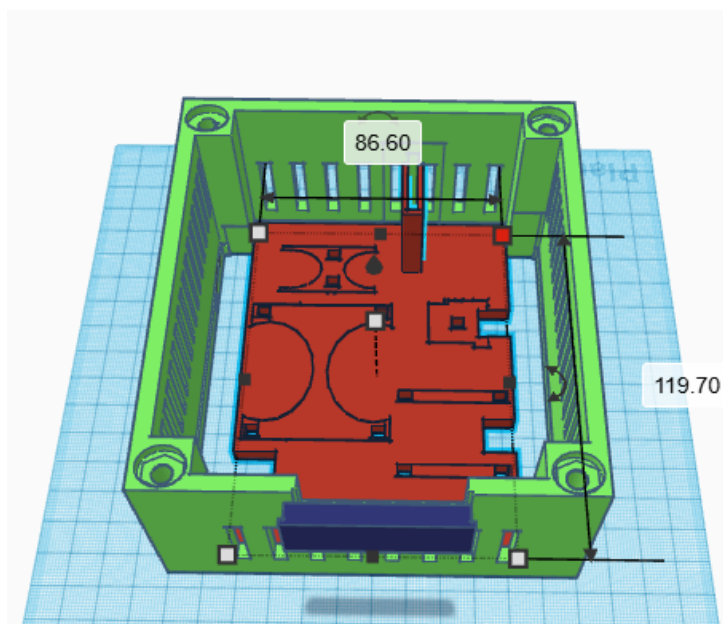


Fig. N°78 -Diseño 3D – Habitáculo superior, base de sensores [autoría propia]

Para finalizar, el diseño de la tapa cuenta con una “*abertura*” para la pantalla y cuatro calces destinados a acoplar la base del Arduino en adición a cuatro calces más en sus esquinas para alinear la tapa a la **carcasa exterior** a fin de lograr un mejor ajuste a esta. Sus dimensiones son de 130 x 1120 [mm].

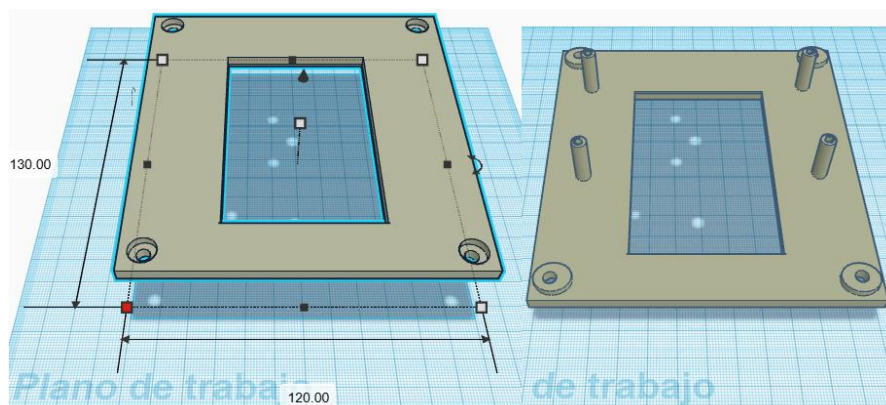



Fig. N°79 -Diseño 3D – Habitáculo superior – tapa [autoría propia]

### 3.9.2. Bases de los dispositivos interiores

Se diseñaron soportes específicos para cada sensor, adaptados a sus espacios ya definidos. En la siguiente imagen se muestran los soportes, en color rojo, del MQ7 (a la derecha) y del ZP16 (a la izquierda). Estos soportes encajan en el piso del

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

**habitáculo superior**, fijando tanto la base como el dispositivo en su lugar logando la firmeza esperada.

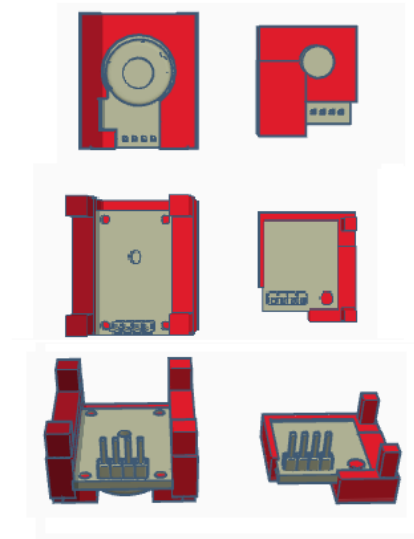


Fig. N°80 -Diseño 3D- Carcasa de sensores 1 [autoría propia]

En la siguiente ilustración se muestran los soportes de los sensores DHT11 (a la derecha) MHZ-19 (al centro) y ZH06 (a la izquierda), al igual que los anteriores, son fijados a la base y se adicionan acoples para asegurar el sensor a su correspondiente “carcasa”.

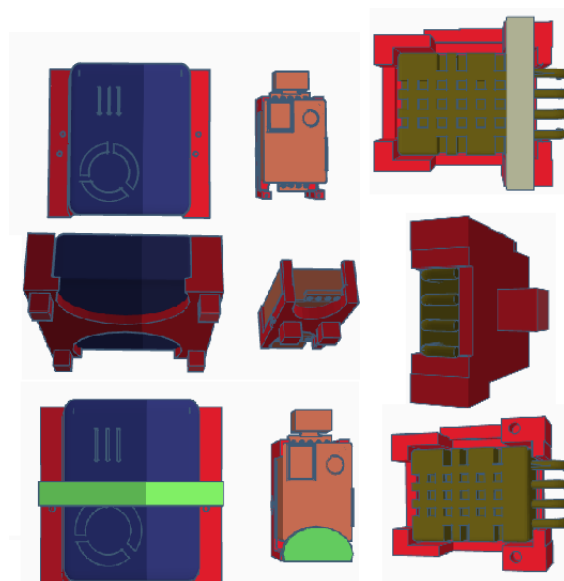



Fig. N°81 -Diseño 3D – Carcasa de sensores 2 [autoría propia]

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

Se diseñó una base soporte con sus respectivos calces para ubicar el **Arduino UNO** justo debajo de la tapa y por encima de los demás sensores existiendo un espacio entre ellos.

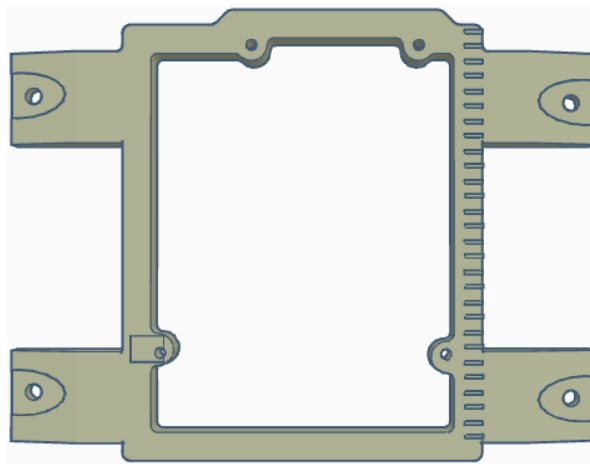


Fig. N°82 - Diseño 3D- Base para Arduino UNO [autoría propia]

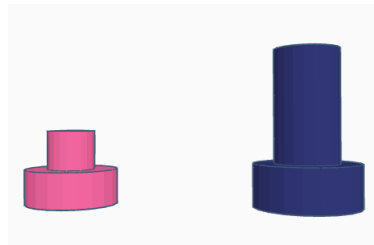


Fig. N°83 - Diseño 3D- Calces para base Arduino UNO [autoría propia]

Por último, para evitar que el piso se desplace una vez colocada la tapa, se diseñó una “*viga*” que se ubica en las cuatro esquinas del piso. Al colocar la tapa, esta comprime el suelo, asegurando que no se mueva. Las dimensiones son 7.2 x 3.53 x 54.5 [mm].



Fig. N°84 -Diseño 3D – Viga Soporte de base de sensores. [autoría propia]



### 3.9.3. Diseño 3D completo

Para finalizar y a modo resumen se presenta el diseño 3D de las partes más importantes del modelo diseñado.

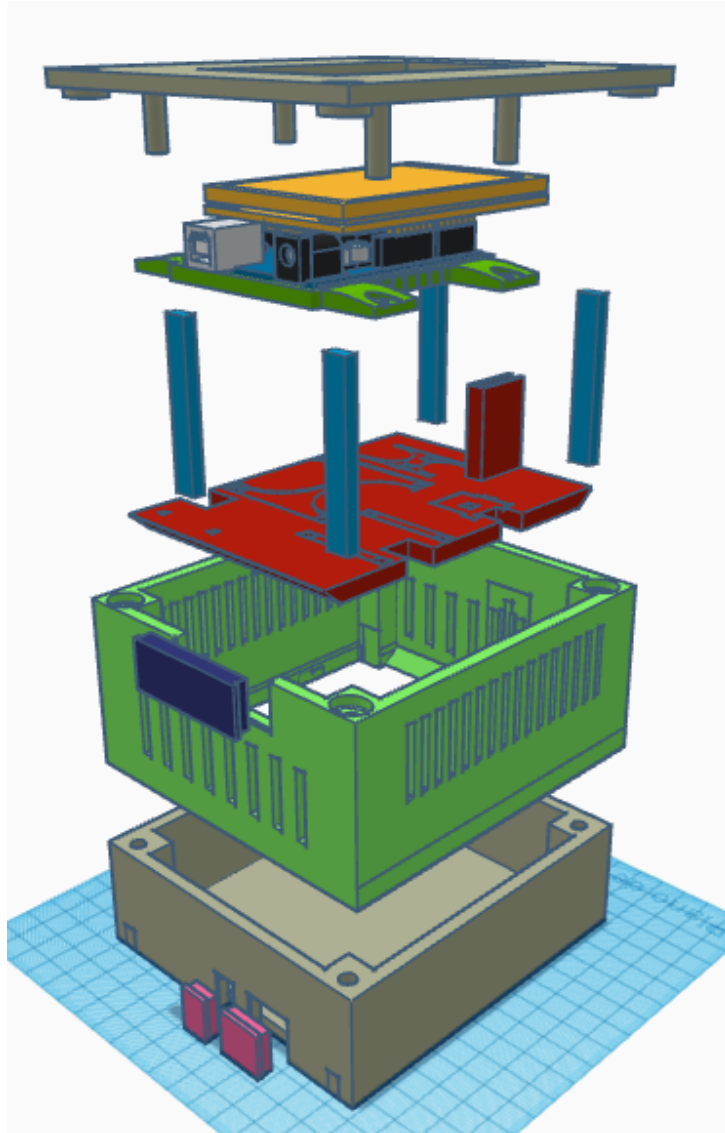



Fig. N°85 - Diseño 3D – Partes principales [autoría propia]

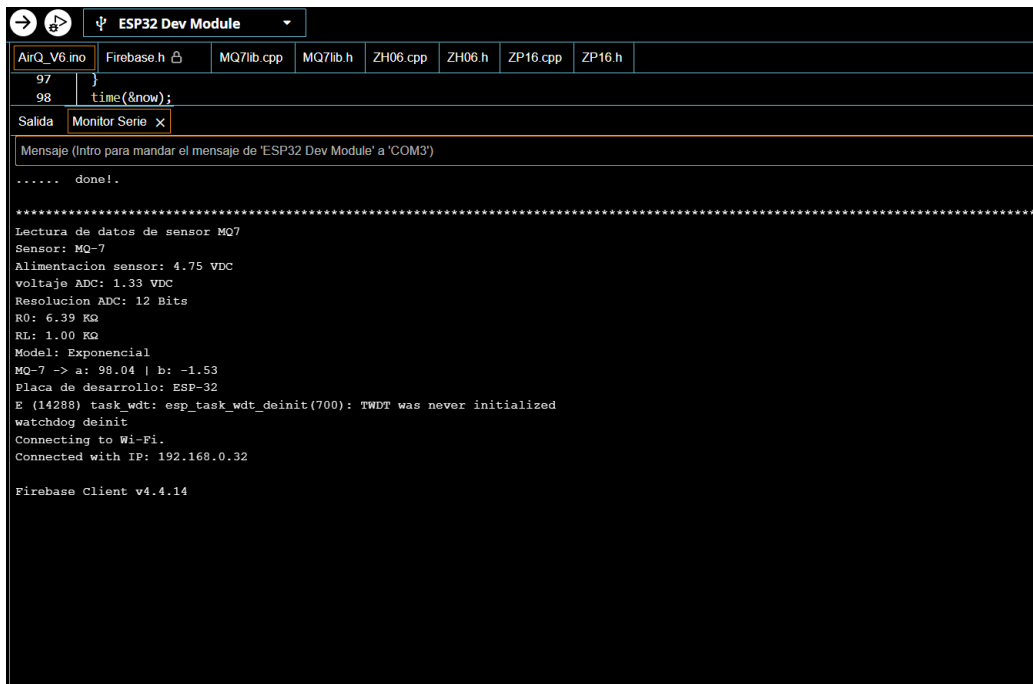
### 3.10. Integración de las partes

Se realizó el análisis de cada uno de los sensores individualmente con su correspondiente **calibración** y **puesta a punto**, se procedió al armado del sistema completo y se evaluaron los resultados.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

Mediante un primer “*debug*” se confirma que el sistema por completo está en funcionamiento sin problemas, todas las tareas del microcontrolador se ejecutan sin problemas.

Cuando el sistema inicia, se realiza la calibración de los sensores, la conexión a Wifi y la conexión a **Firestore** como se muestra en la siguiente imagen “*debug*”.




```

ESP32 Dev Module
AirQ_V6.ino  Firebase.h  MQ7lib.cpp  MQ7lib.h  ZH06.cpp  ZH06.h  ZP16.cpp  ZP16.h
97 }
98   time(&now);
Salida Monitor Serie x
Mensaje (Intro para mandar el mensaje de 'ESP32 Dev Module' a 'COM3')
..... done!.
*****
Lectura de datos de sensor MQ7
Sensor: MQ-7
Alimentacion sensor: 4.75 VDC
voltaje ADC: 1.33 VDC
Resolucion ADC: 12 Bits
R0: 6.39 KQ
RL: 1.00 KQ
Model: Exponencial
MQ-7 -> a: 98.04 | b: -1.53
Placa de desarrollo: ESP-32
E (14288) task_wdt: esp_task_wdt_deinit(700): TWDT was never initialized
watchdog deinit
Connecting to Wi-Fi.
Connected with IP: 192.168.0.32
Firebase Client v4.4.14

```

Fig. N°86 -Debug – Inicio de sistemas [fuente propia]

Una vez iniciado el sistema, se imprime por **monitor serie** los datos que van siendo leídos desde los sensores.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	3. Desarrollo

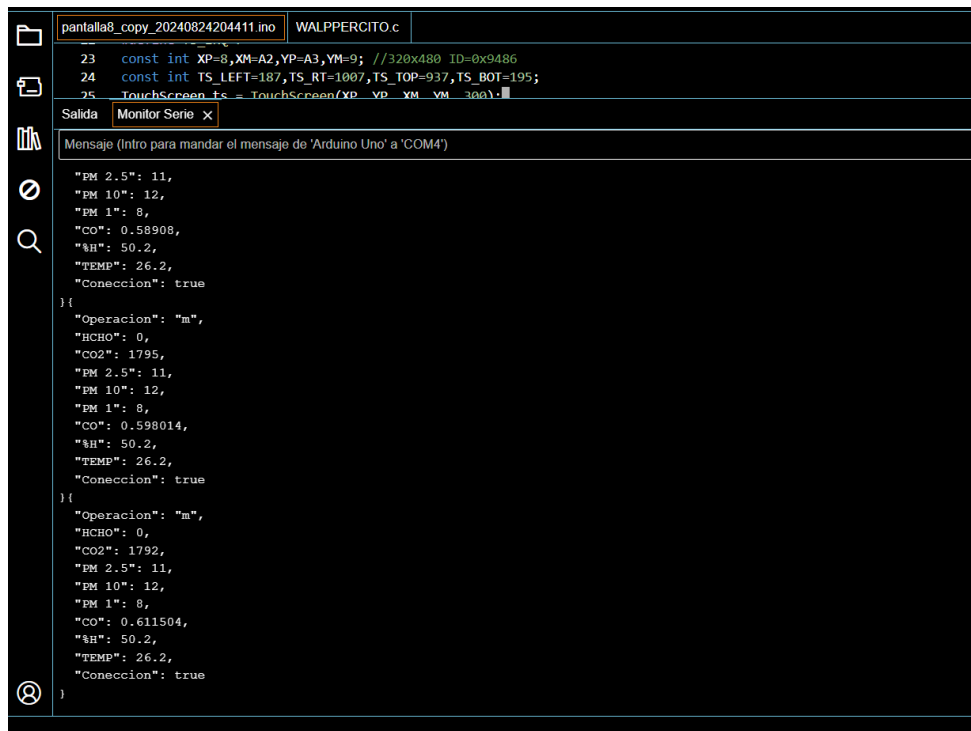
```

HCHO: 0.00ppm
CO2: 3138.00ppm
PM2.5: 15.00ug/m3
PM10: 17.00ug/m3
PM1.0: 12.00ug/m3
CO: 0.03ppm
H: 45.10%
T: 24.40°C
-----
HCHO: 0.00ppm
CO2: 3328.00ppm
PM2.5: 15.00ug/m3
PM10: 17.00ug/m3
PM1.0: 12.00ug/m3
CO: 0.03ppm
H: 45.10%
T: 24.30°C
-----

```

Fig. N°87 -Debug – Lectura de datos [fuente propia]

Luego se corrobora la correcta lectura desde el lado del Arduino y vemos que la conexión es exitosa.



```

panfalla8_copy_20240824204411.ino  WALPPERCITO.c
23  const int XP=8,XM=A2,YP=A3,YM=9; //320x480 ID=0x9486
24  const int TS_LEFT=187,TS_RT=1007,TS_TOP=937,TS_BOT=195;
25  TouchScreen ts = TouchScreen(XP_YP_XM_YM_300);
Salida  Monitor Serie X
Mensaje (Intro para mandar el mensaje de 'Arduino Uno' a 'COM4')
{"PM 2.5": 11,
"PM 10": 12,
"PM 1": 8,
"CO": 0.58908,
"%H": 50.2,
"TEMP": 26.2,
"Conexion": true
}
{"Operacion": "m",
"HCHO": 0,
"CO2": 1795,
"PM 2.5": 11,
"PM 10": 12,
"PM 1": 8,
"CO": 0.598014,
"%H": 50.2,
"TEMP": 26.2,
"Conexion": true
}
{"Operacion": "m",
"HCHO": 0,
"CO2": 1792,
"PM 2.5": 11,
"PM 10": 12,
"PM 1": 8,
"CO": 0.611504,
"%H": 50.2,
"TEMP": 26.2,
"Conexion": true
}

```

Fig. N°88 -Debug – Comunicación Arduino – ESP exitosa [fuente propia]



Se procede entonces a armar el sistema completo con sus respectivas partes y vemos al dispositivo en su faceta final.

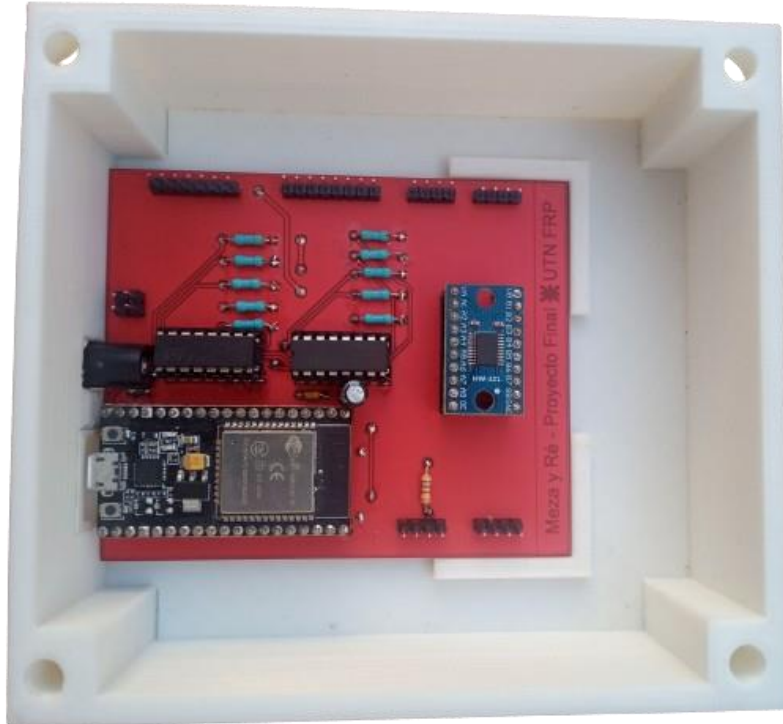


Fig. N°89 -Armado de Sistema – Base [autoría propia]

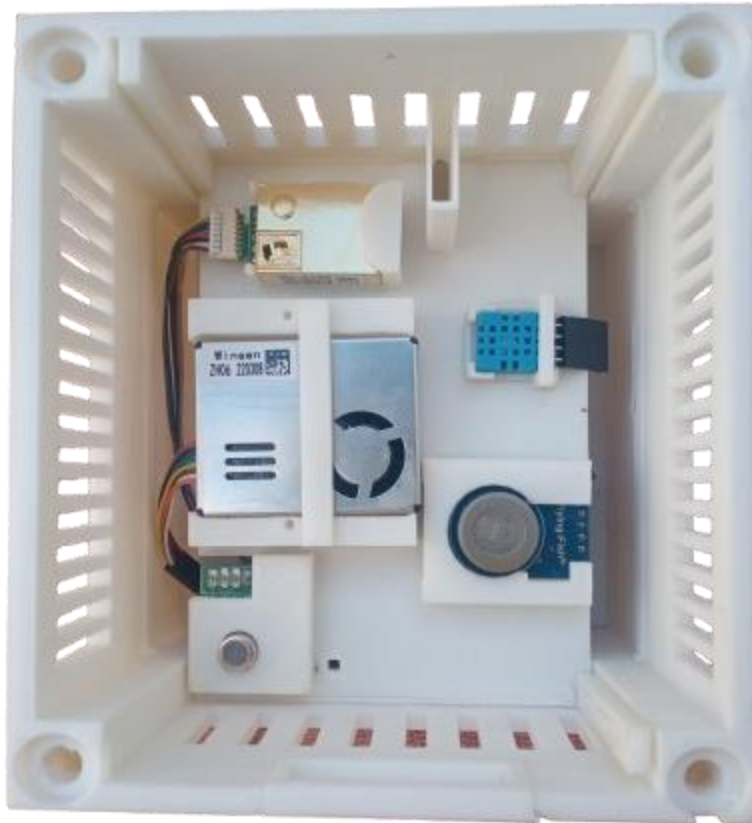


Fig. N°90 - Armado de sistema – Sensores [autoría propia]




Fig. N°91 - Armado de sistema – Tapa [fuente propia]



Fig. N°92 -Sistema armado completo con cables para debug [fuente propia]



Fig. N°93 - Sistema armado completo final [fuente propia]

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO <sub>2</sub> y Partículas
Ingeniería en Electrónica	4. Resultados

## Capítulo 4: Resultados

### 4.1. Análisis de consumo eléctrico

En el planteo de este proyecto se ha presentado la intención de que este dispositivo funcione de manera **inalámbrica** mediante baterías, esto suponía adicionar baterías y un regulador de voltaje para la alimentación.

Mediante información proporcionada por las hojas de datos de todas las partes se recopiló el consumo máximo de cada una de ellas.

Dispositivos	Cantidad	Consumo máximo [A]
CD4051B	2	0.02
DHT11	1	0.001
ESP32	1	0.5
MHZ-19B	1	0.15
MQ7	1	0.07
ZH06	1	0.12
ZP16	1	0.06
TXS108BE	1	0.1
Arduino UNO	1	0.2
TOTAL		1.22

Tabla N°9 - Consumo en Amper de dispositivos

Como se puede observar, el **consumo** máximo teórico de nuestro proyecto resulta en 1,2 [A] lo cual es elevado, para una tensión de trabajo de 5 [V] será necesario una fuente de al menos 6 [W]. Sin embargo, al realizar una prueba de consumo del dispositivo funcionando en condiciones normales el consumo resultó mucho menor.




Fig. N°94 -Prueba de consumo eléctrico. [Fuente propia]

Como indica la figura anterior, el consumo del dispositivo funcionando es en promedio 0.53 [A] con fluctuaciones de +/- 0.02 [A], bajo este valor se requerirá una fuente de alimentación de al menos 1 [A] o 5 [W]. Barajando la posibilidad de alimentarlo mediante **batería** para un funcionamiento estable se requerirá una capacidad de 0.5 [A] por hora para mantener en funcionamiento el dispositivo, es decir, para **4 a 5 horas** de funcionamiento se necesitaría una batería de al menos 3000 [mAh] de capacidad, si se quisiera que funcione por más tiempo deberíamos poder cargar la batería o aumentar su capacidad, pero sería un problema, porque una batería de mayor capacidad es más **costosa** y de **mayor volumen**. En resumen, para un funcionamiento inalámbrico durante un largo periodo de tiempo se requerirá un sistema de alimentación que incluya baterías, regulador de voltaje y un circuito cargador de baterías.

Nuestro dispositivo no posee la capacidad de cargar baterías, por lo cual si quisiéramos portabilidad necesitaríamos una batería externa de mucha capacidad, lo cual nos resulta **inviable**. Sin embargo, este análisis nos permitió conocer el consumo eléctrico del dispositivo y con ello poder conectarle una batería externa para realizar pruebas de campo y medición del dispositivo, el funcionamiento normal y habitual de este será bajo una fuente de alimentación continua de 5 [V] y 2 [A].

#### 4.2. Pruebas de campo

En esta sección se abordarán las pruebas al dispositivo. Donde, a través de un habitáculo muy cerrado le introdujimos **contaminantes** al medidor de calidad de aire

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO <sub>2</sub> y Partículas
Ingeniería en Electrónica	4. Resultados

para analizar la respuesta cualitativa de este, no así su respuesta cuantitativa debido a dificultades en la adquisición de recursos necesarios para este tipo de análisis.

Se introdujeron CO<sub>2</sub> y CO como contaminante para su **análisis**, se buscó determinar la correcta detección de estos por parte del correspondiente sensor y, si existe o no influencia de este en otros sensores que no estén preparados para detectarlo.

#### 4.2.1. Descripción de las pruebas

Se realizaron dos tipos de pruebas, de “*saturación*” y de “*estimación*”. En la saturación se ha introducido concentraciones del contaminante mayores al límite de detección del sensor en unidades de ppm. En la estimación se ha introducido niveles de concentración dentro del rango de detección del sensor.

El CO se introducirá a través de humo de cigarrillo ya que este cuenta con un porcentaje del contaminante. En cuanto al CO<sub>2</sub>, se generará mediante una **reacción química** entre vinagre y bicarbonato.

#### 4.2.2. Descripción de los elementos

El habitáculo cuenta de dos partes:

La primera parte es donde va a estar alojado el medidor de calidad de aire.



Fig. N°95 -Pruebas de campo – Habitáculo contenedor [fuente propia]

La segunda parte es donde se van a agregar los contaminantes o generar reacciones. Donde, van a estar conectado a la primera a través de una manguera de 7 [mm] de diámetro y 49 [cm] de longitud.


 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	4. Resultados

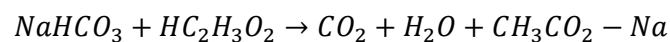


Fig. N°96 - Pruebas de campo – habitáculo de reacción. [fuente propia]

El volumen de ocupación del gas en este sistema es:

- Volumen recipiente:  $5,71639 * 10^{-3}[\text{m}^3]$ .
- Volumen de manguera y frasco de reacción:  $0,27 * 10^{-3}[\text{m}^3]$ .
- Volumen del medidor de calidad de aire:  $1440,18[\text{cm}^3]$ .  $\rightarrow 0,144018 * 10^{-3}[\text{m}^3]$ .
- Volumen de ocupación del gas  $5,71639 * 10^{-3}[\text{m}^3] - 0,144018 * 10^{-3}[\text{m}^3] + 0,27 * 10^{-3}[\text{m}^3] = 4,54545 * 10^{-3}[\text{m}^3]$ .


#### 4.2.3. CO<sub>2</sub>-Reacción vinagre bicarbonato



Ecuación N°.15 - Reacción de vinagre con bicarbonato – Ecuación estequiométrica. [5]

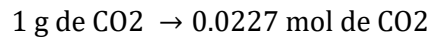
Esta reacción del bicarbonato de sodio con el 5% del ácido acético del vinagre libera dióxido de carbono. En esta reacción, **un mol de bicarbonato** de sodio reacciona con **un mol del ácido acético** del vinagre liberando **un mol de dióxido de carbono**, esto es, una reacción 1 a 1. De esta manera, según la cantidad en gramos de bicarbonato de sodio y de vinagre obtendremos una cantidad en gramos de CO<sub>2</sub>. Para calcularlo necesitaremos algunos datos.

- Masa del bicarbonato:  $84 \left[ \frac{\text{g}}{\text{mol}} \right]$ .
- Masa del dióxido de carbono:  $44,009 \left[ \frac{\text{g}}{\text{mol}} \right]$

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	4. Resultados

- Masa del vinagre:  $60 \left[ \frac{\text{g}}{\text{mol}} \right]$

Ejemplo de cálculo: Generar 1 g de CO<sub>2</sub>



Como la reacción es 1 a 1, se necesitan 0.0227 moles de bicarbonato de sodio con 0.0227 moles de vinagre.

- Bicarbonato de sodio

$$0.0227[\text{mol}] * 84 \left[ \frac{\text{g}}{\text{mol}} \right] = 1.91[\text{g}]$$

Ecuación N°.16 - Ejemplo – Calculo de gramos de bicarbonato

- Vinagre

$$0.0227[\text{mol}] * 60 \left[ \frac{\text{g}}{\text{mol}} \right] = 1.36[\text{g}]$$

Ecuación N°.17 - Ejemplo – Calculo de gramos de vinagre

Como el vinagre tiene un **5% de acidez**. Necesitamos ajustar las cantidades.


$$\text{Vinagre} = \frac{(100\% * 1.36[\text{g}])}{5\%} = 27.24[\text{g}]$$

Ecuación N°.18 - Ejemplo – Ajuste de cantidad de vinagre

En este ejemplo se buscó mostrar como calcular los gramos de vinagre y bicarbonato de sodio necesarios para generar un gramo de dióxido de carbono. Nuestro objetivo es poder calcular estos parámetros, pero partiendo de un valor en **“ppm” deseado** y para ello se necesita contar con el volumen del ambiente, en nuestro caso, el habitáculo.

#### 4.2.3.1. Prueba de Saturación

Para la prueba de **“saturación”**, necesitamos conocer cuáles son los pesos mínimos en [g] de los reactivos para que el sensor llegue a su límite y analizar su comportamiento. Tomando el ejemplo de generar ppm realizamos los cálculos.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	4. Resultados

$$\text{concentracion del gas } \left[ \frac{g}{m^3} \right] = \frac{10000 [ppm] * 44.009 \left[ \frac{g}{mol} \right]}{0.02445 \left[ \frac{m^3}{mol} \right] * 10^6} = 17.99 \left[ \frac{g}{m^3} \right]$$

Ecuación N°.19 - Prueba de saturación – Calculo de concentración  $\left[ \frac{g}{m^3} \right]$

$$CO_2 [mg] = 17995 \left[ \frac{mg}{m^3} \right] * 4.54545 * 10^{-3} [m^3] = 81.79 [mg]$$

Ecuación N°.20 - Prueba de saturación – Calculo de masa CO2

Con tan solo 81.79 [mg] de CO2 se generan 10000 [ppm] de CO2 en el volumen mencionado. A continuación, se hará **saturar** el dispositivo reaccionando 1g de bicarbonato con 15 g de vinagre [5].

Reacción de 1 g de bicarbonato con 15 g de vinagre.

Como se dijo anteriormente como la reacción es 1 mol de bicarbonato a 1 mol de vinagre. Por ende, para sus cálculos dividimos cada uno de los compuestos por su masa molar obtenemos los moles de cada uno.

Las condiciones en las cuales se realizó la prueba fueron las siguientes:

$$T = 36 [^{\circ}C]$$

$$P = 1 atm$$

$$H = 41 \%$$

Nmol Bicarbonato de sodio


$$\frac{1[g]}{84.007 \left[ \frac{g}{mol} \right]} = 1.19 * 10^{-2} [mol]$$

Ecuación N°.21 - Prueba de saturación – Moles de bicarbonato

Nmol Ácido acético

$$\frac{(15 \times 0.05) [g]}{60.05 \left[ \frac{g}{mol} \right]} = 1.24 * 10^{-2} [mol]$$

Ecuación N°.22 - Prueba de saturación – Moles de ácido acético

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	4. Resultados

Como el bicarbonato de sodio es el reactivo limitante tomamos que se va a generar  $1.19 \cdot 10^{-2}$  [mol] de CO<sub>2</sub>, por ende, la cantidad de gramos de CO<sub>2</sub> va a ser:

$$44.009 \left[ \frac{\text{g}}{\text{mol}} \right] * 1.19 * 10^{-2} [\text{mol}] = 0.52 [\text{g}]$$

Ecuación N°.23 - Prueba de saturación – gramos de CO<sub>2</sub>

$$\frac{\text{Gramos de CO}_2}{\text{Volumen de ocupación efectiva del gas}} = \frac{0.52 [\text{g}]}{4.54545 * 10^{-3} [\text{m}^3]} = 114.4 \left[ \frac{\text{g}}{\text{m}^3} \right]$$

Ecuación N°.24 - Prueba de saturación – concentración de CO<sub>2</sub> en  $\left[ \frac{\text{g}}{\text{m}^3} \right]$

Pasando la concentración PPM

$$\text{CO}_2 \text{ppm} = \frac{0.02445 \left[ \frac{\text{m}^3}{\text{mol}} \right] * 114.4 \left[ \frac{\text{g}}{\text{m}^3} \right]}{44.01 \left[ \frac{\text{g}}{\text{mol}} \right]} * 10^6 = 63555.55 [\text{ppm}]$$

Ecuación N°.25 - Prueba de saturación – concentración de CO<sub>2</sub> en [ppm]


En las pruebas realizadas se comenzó con una concentración de contaminante de 2410 [ppm] y le llevó 4 minutos alcanzar los 10000 [ppm], que es el límite del sensor del dióxido de carbono. Las gráficas de los datos sacados de la prueba se muestran a continuación.

#### 4.2.3.2. Pruebas de estimación

En estas pruebas hicimos un análisis **cuantitativo** de la respuesta del sensor. Es importante destacar que no se esperó un resultado exacto ni muy cercano a lo calculado de forma analítica, ya que las condiciones en las que se realizaron las pruebas no fueron óptimas, pero pudimos analizar la respuesta del dispositivo y estimar la exactitud y precisión del dispositivo.

Generar 2000 [ppm]

Se realizaron los cálculos teóricos a fines de obtener una concentración de 2000 [ppm] dentro del habitáculo cuyo volumen fue mencionado anteriormente. El objetivo final de estos cálculos es obtener la masa mínima en gramos de vinagre y bicarbonato de sodio necesaria para generar la concentración deseada.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	4. Resultados

En primer lugar, debemos hacer la conversión de partes por millón (ppm) a  $\left[\frac{mg}{m^3}\right]$ . La ecuación es la siguiente

$$concentracion\ del\ gas\ \left[\frac{g}{m^3}\right] = \frac{2000\ [ppm] * 44.009\ \left[\frac{g}{mol}\right]}{24.45\ \left[\frac{l}{mol}\right]}$$

Ecuación N°.26 - Prueba de estimación 2000ppm – Cálculo de concentración CO2  $\left[\frac{g}{m^3}\right]$

El valor  $24.45\left[\frac{l}{mol}\right]$  es el volumen molar de un gas ideal, el cual surge de la ecuación de gases ideales  $P.V = nRT$ . Debemos tener en cuenta que ppm significa partes por millón y para la conversión a  $\left[\frac{g}{m^3}\right]$  debemos dividir por  $10^6$ , es decir, quitar el millón. Además, debemos pasar de litros a  $m^3$ . Entonces nos queda:

$$concentracion\ del\ gas\ \left[\frac{g}{m^3}\right] = \frac{2000\ [ppm] * 44.009\ \left[\frac{g}{mol}\right]}{0.02445\ \left[\frac{m^3}{mol}\right] * 10^6} = 3.598\ \left[\frac{g}{m^3}\right]$$

Ecuación N°.27 - Prueba de estimación 2000 ppm – Calculo de concentración CO2  $\left[\frac{g}{m^3}\right]$

2000 ppm son equivalentes a  $3598\ \left[\frac{mg}{m^3}\right]$ . Sabiendo el volumen del habitáculo, obtenemos la masa de CO2.

$$CO2\ [mg] = 3598\ \left[\frac{mg}{m^3}\right] * 4.54545 * 10^{-3}\ [m^3] = 16.35\ [mg]$$


Ecuación N°.28 - Prueba de estimación 2000ppm – Calculo de masa CO2

Ahora bien, sabemos que  $16.35\ [mg]$  en un volumen de  $4.54545 * 10^{-3}\ [m^3]$  la **concentración** es de 2000 ppm de CO2. Con este dato podremos calcular la masa de vinagre y bicarbonato.

Primero obtenemos los moles

$$\frac{16.35\ [mg]}{44009\ \left[\frac{mg}{mol}\right]} = 3.715 * 10^{-4}\ [mol]$$

Ecuación N°.29 - Prueba de estimación 2000ppm – Calculo de moles CO2

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	4. Resultados

$$\text{acido acetico [g]} = 60 \left[ \frac{\text{g}}{\text{mol}} \right] * 3.715 * 10^{-4} [\text{mol}] = 22.29 [\text{mg}]$$

Ecuación N°.30 - Prueba de estimación 2000ppm – Calculo de masa de ácido acético

Como la reacción se da con la acidez del 5% del vinagre, calculamos la masa de vinagre necesaria.

$$\text{Vinagre[g]} = \frac{(100\% * 22.29[\text{mg}])}{5\%} = 445.81[\text{mg}]$$

Ecuación N°.31 - Prueba de estimación 2000ppm – cálculo de masa de vinagre

$$\text{Bicarbonato de sodio [g]} = 84 \left[ \frac{\text{g}}{\text{mol}} \right] * 3.715 * 10^{-4} [\text{mol}] = 31.207 [\text{mg}]$$

Ecuación N°.32 - Prueba de estimación 2000ppm – cálculo de masa de bicarbonato de sodio

En resumen, se necesitan 445.81[mg] de vinagre y 31.207 [mg] de bicarbonato de sodio para generar una concentración de 2000 [ppm] dentro del habitáculo.

En el primer ensayo se hizo reaccionar 450[mg] de vinagre con 940 [mg] de bicarbonato de sodio. Recalculamos:

$$\frac{0,94[\text{g}]}{84.007 \left[ \frac{\text{g}}{\text{mol}} \right]} = 11.189 * 10^{-3} [\text{mol}]$$

Ecuación N°.33 - Primer ensayo 2000ppm – moles bicarbonato


$$\frac{0.45[\text{g}] * 5\%}{60 \left[ \frac{\text{g}}{\text{mol}} \right] * 100\%} = 0.375 * 10^{-3} [\text{mol}]$$

Ecuación N°.34 - Primer ensayo 2000ppm – moles vinagre

En esta reacción el vinagre será nuestro reactivo limitante.

$$\text{CO}_2[\text{g}] = 44.009 \left[ \frac{\text{g}}{\text{mol}} \right] * 0.375 * 10^{-3} [\text{mol}] = 16.5[\text{mg}]$$

Ecuación N°.35 - Primer ensayo 2000ppm – masa CO2 generada

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	4. Resultados

$$CO_2 \left[ \frac{mg}{m^3} \right] = \frac{16.5[mg]}{4.54545 * 10^{-3}[m^3]} = 3630.74 \left[ \frac{mg}{m^3} \right]$$

Ecuación N°.36 - Primer ensayo 2000ppm – concentración  $\left[ \frac{mg}{m^3} \right]$  CO2 generada

Pasando a ppm en condiciones normales

$$CO_2[ppm] = \frac{0.02445 \left[ \frac{m^3}{mol} \right] * 3.63074 \left[ \frac{g}{m^3} \right]}{44.01 \left[ \frac{g}{mol} \right]} * 10^6 = 2017.12 [ppm]$$

Ecuación N°.37 - Primer ensayo 2000ppm – concentración ppm generados

En el segundo ensayo se hizo reaccionar 450[mg] de vinagre con 100 [mg] de bicarbonato de sodio. Recalculamos:

$$\frac{0,1[g]}{84.007 \left[ \frac{g}{mol} \right]} = 1.19 * 10^{-3}[mol]$$

Ecuación N°.38 - Segundo ensayo 2000ppm – moles de bicarbonato

$$\frac{0.45[g] * 5\%}{60 \left[ \frac{g}{mol} \right] * 100\%} = 0.375 * 10^{-3}[mol]$$


Ecuación N°.39 - Segundo ensayo 2000ppm – moles de ácido acético

Se observa que el vinagre sigue siendo nuestro reactivo limitante.

$$CO_2[g] = 44.009 \left[ \frac{g}{mol} \right] * 0.375 * 10^{-3}[mol] = 16.5[mg]$$

Ecuación N°.40 - Segundo ensayo 2000ppm – masa de CO2

$$CO_2 \left[ \frac{mg}{m^3} \right] = \frac{16.5[mg]}{4.54545 * 10^{-3}[m^3]} = 3630.74 \left[ \frac{mg}{m^3} \right]$$

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	4. Resultados

Ecuación N°.41 - Segundo ensayo 2000ppm - concentración  $\left[\frac{mg}{m^3}\right]$  CO2 generada

Pasando a ppm en condiciones normales

$$CO2[ppm] = \frac{0.02445 \left[\frac{m^3}{mol}\right] * 3.63074 \left[\frac{g}{m^3}\right]}{44.01 \left[\frac{g}{mol}\right]} * 10^6 = 2017.12 [ppm]$$

Ecuación N°.42 - Segundo ensayo 2000ppm – concentración ppm de CO2 generada

Teniendo en cuenta las condiciones en las cuales se realizaron los ensayos, podemos realizar una **corrección de ppm** y verificar el resultado con exactitud [35].

Partiendo de la ecuación de los gases ideales

$$P * V = n * R * T$$

Ecuación N°.43 - Ecuación de los gases ideales [5]

Nos interesa recalcular el volumen molar

$$\frac{V \left[\frac{m^3}{mol}\right]}{n \left[\frac{mol}{mol}\right]} = \frac{R \left[\frac{atm * l}{mol * K}\right] * T[K]}{P[atm]}$$

Ecuación N°.44 - Formula - Calculo de volumen molar

$$R = 0.08205745 \left[\frac{atm * l}{mol * K}\right] = 82.05745 * 10^{-6} \left[\frac{atm * m^3}{mol * K}\right]$$

Condiciones del primer ensayo

$$T = 35.3 [^{\circ}C] = 308.45 [K]$$

$$P = 1 atm$$

$$V = 3.736 * 10^{-3} [m^3]$$

$$\frac{V \left[\frac{m^3}{mol}\right]}{n \left[\frac{mol}{mol}\right]} = \frac{82.05745 * 10^{-6} \left[\frac{atm * m^3}{mol * K}\right] * 308.45[K]}{1[atm]} = 0.02531 \left[\frac{m^3}{mol}\right]$$



Ecuación N°.45 - Primer ensayo 2000 ppm - Ajuste de volumen molar según condiciones

Recalculo de ppm

$$CO_2 \left[ \frac{mg}{m^3} \right] = \frac{16.5 [mg]}{3.736 * 10^{-3} [m^3]} = 4416.48 \left[ \frac{mg}{m^3} \right]$$

Ecuación N°.46 - Primer ensayo 2000 ppm – Ajuste de concentración  $\left[ \frac{mg}{m^3} \right]$  CO<sub>2</sub> generada

$$CO_2 [ppm] = \frac{0.02531 \left[ \frac{m^3}{mol} \right] * 4.41648 \left[ \frac{g}{m^3} \right]}{44.01 \left[ \frac{g}{mol} \right]} * 10^6 = 2540 [ppm]$$

Ecuación N°.47 - Primer ensayo 2000 ppm – Ajuste de concentración ppm CO<sub>2</sub> generada

Para el segundo ensayo se realizó el mismo procedimiento y resultaron en una concentración de 2550 [ppm].

Generar 4000 ppm


Se realizaron los cálculos teóricos a fines de obtener una concentración de 4000[ppm] dentro del habitáculo cuyo volumen fue mencionado anteriormente. El objetivo final de estos cálculos es obtener la **masa mínima** en gramos de vinagre y bicarbonato de sodio necesaria para generar la concentración deseada.

Ya que los cálculos son similares al ejemplo con 2000 ppm, nos enfocaremos en mostrar formula y resultado.

$$concentracion\ del\ gas \left[ \frac{g}{m^3} \right] = \frac{4000 [ppm] * 44.009 \left[ \frac{g}{mol} \right]}{0.02445 \left[ \frac{m^3}{mol} \right] * 10^6} = 7196 \left[ \frac{mg}{m^3} \right]$$

Ecuación N°.48 - Prueba de estimación 4000 ppm- Concentración  $\left[ \frac{mg}{m^3} \right]$  CO<sub>2</sub> generada

$$CO_2 [mg] = 7196 \left[ \frac{mg}{m^3} \right] * 4.54545 * 10^{-3} [m^3] = 32.70 [mg]$$

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	4. Resultados

Ecuación N°.49 - Prueba de estimación 4000 ppm- Masa de CO2 generada

Sabemos que 16.35 [mg] en un volumen de  $4.54545 * 10^{-3} [m^3]$  la concentración es de 4000 ppm de CO2. Con este dato podremos calcular la masa de vinagre y bicarbonato.

Primero obtenemos los moles y luego la masa

$$\frac{32.7[mg]}{44009 \left[ \frac{mg}{mol} \right]} = 7.4303 * 10^{-4} [mol]$$

Ecuación N°.50 - Prueba de estimación 4000 ppm – Moles de ácido acético

$$\text{acido acetico [g]} = 60 \left[ \frac{g}{mol} \right] * 7.4303 * 10^{-4} [mol] = 44.58 [mg]$$

Ecuación N°.51 - Prueba de estimación 4000 ppm – Masa de ácido acético

Como la reacción se da con la acidez del **5% del vinagre**, calculamos la masa de vinagre necesaria.

$$\text{Vinagre [g]} = \frac{(100\% * 22.29 [mg])}{5\%} = 891.63 [mg]$$

Ecuación N°.52 - Prueba de estimación 4000 ppm – Masa de vinagre


$$\text{Bicarbonato de sodio [g]} = 84 \left[ \frac{g}{mol} \right] * 7.4303 * 10^{-4} [mol] = 62.414 [mg]$$

Ecuación N°.53 - Prueba de estimación 4000 ppm – Masa de bicarbonato

En resumen, se necesitan 891.63 [mg] de vinagre y 62.414 [mg] de bicarbonato de sodio para generar una concentración de 4000[ppm] dentro del habitáculo.

En el primer ensayo se hizo reaccionar 890[mg] de vinagre con 510 [mg] de bicarbonato de sodio. Recalculamos:

$$\text{moles bicarbonato} = \frac{0,51[g]}{84.007 \left[ \frac{g}{mol} \right]} = 6.0714 * 10^{-3} [mol]$$

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	4. Resultados

Ecuación N°.54 - Primer ensayo 4000 ppm- moles de bicarbonato

$$\text{moles vinagre} = \frac{0.89[\text{g}] * 5\%}{60 \left[ \frac{\text{g}}{\text{mol}} \right] * 100\%} = 0.7416 * 10^{-3} [\text{mol}]$$

Ecuación N°.55 - Primer ensayo 4000 ppm- moles de vinagre

En esta reacción el vinagre será nuestro reactivo limitante.

$$\text{CO}_2[\text{g}] = 44.009 \left[ \frac{\text{g}}{\text{mol}} \right] * 0.7416 * 10^{-3} [\text{mol}] = 32.64 [\text{mg}]$$

Ecuación N°.56 - Primer ensayo 4000 ppm- Masa de CO2 calculada

$$\text{CO}_2 \left[ \frac{\text{mg}}{\text{m}^3} \right] = \frac{16.5 [\text{mg}]}{4.54545 * 10^{-3} [\text{m}^3]} = 7180.82 \left[ \frac{\text{mg}}{\text{m}^3} \right]$$

Ecuación N°.57 - Primer ensayo 4000ppm- concentración CO2 calculada

Pasando a ppm en condiciones normales

$$\text{CO}_2[\text{ppm}] = \frac{0.02445 \left[ \frac{\text{m}^3}{\text{mol}} \right] * 7.18082 \left[ \frac{\text{g}}{\text{m}^3} \right]}{44.01 \left[ \frac{\text{g}}{\text{mol}} \right]} * 10^6 = 3989.34 [\text{ppm}]$$


Ecuación N°.58 - Primer ensayo 4000ppm- concentración CO2 [ppm] calculada

En el segundo ensayo se hizo reaccionar 900[mg] de vinagre con 70 [mg] de bicarbonato de sodio. Recalculamos:

Numero de moles:

$$\text{moles bicarbonato} = \frac{0,07[\text{g}]}{84.007 \left[ \frac{\text{g}}{\text{mol}} \right]} = 8.333 * 10^{-4} [\text{mol}]$$

Ecuación N°.59 - Segundo ensayo 4000ppm- Moles bicarbonato

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	4. Resultados

$$\text{moles vinagre} = \frac{0.90[\text{g}] * 5\%}{60 \left[ \frac{\text{g}}{\text{mol}} \right] * 100\%} = 0.75 * 10^{-3} [\text{mol}]$$

Ecuación N°.60 - Segundo ensayo 4000ppm – Moles de vinagre

Gramos de CO2:

$$\text{CO}_2[\text{g}] = 44.009 \left[ \frac{\text{g}}{\text{mol}} \right] * 0.7416 * 10^{-3} [\text{mol}] = 32.64 [\text{mg}]$$

Ecuación N°.61 - Segundo ensayo 4000ppm – Masa de CO2 generada

$$\text{CO}_2 \left[ \frac{\text{mg}}{\text{m}^3} \right] = \frac{16.5 [\text{mg}]}{4.54545 * 10^{-3} [\text{m}^3]} = 7180.82 \left[ \frac{\text{mg}}{\text{m}^3} \right]$$

Ecuación N°.62 - Segundo ensayo 4000ppm – Concentración de CO2 generada

$$8068 \left[ \frac{\text{mg}}{\text{m}^3} \right] \rightarrow 3989.34 [\text{ppm}]$$

Realizamos los ajustes de ppm según las condiciones del ensayo.

Condiciones del primer ensayo

$$T = 36 [^{\circ}\text{C}] = 309.15 [\text{K}]$$


$$P = 1 \text{ atm}$$

$$V = 3.736 * 10^{-3} [\text{m}^3]$$

$$\frac{V \left[ \frac{\text{m}^3}{\text{mol}} \right]}{n} = \frac{82.05745 * 10^{-6} \left[ \frac{\text{atm} * \text{m}^3}{\text{mol} * \text{K}} \right] * 309.15 [\text{K}]}{1 [\text{atm}]} = 0.02536 \left[ \frac{\text{m}^3}{\text{mol}} \right]$$

Ecuación N°.63 - Primer ensayo 4000ppm – Ajuste de volumen molar

Recalculo de ppm

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	4. Resultados

$$CO_2 \left[ \frac{mg}{m^3} \right] = \frac{32.64[mg]}{3.736 * 10^{-3}[m^3]} = 8736.61 \left[ \frac{mg}{m^3} \right]$$

Ecuación N°.64 - Primer ensayo 4000ppm – Ajuste de concentración CO2

$$CO_2[ppm] = \frac{0.02536 \left[ \frac{m^3}{mol} \right] * 8.73661 \left[ \frac{g}{m^3} \right]}{44.01 \left[ \frac{g}{mol} \right]} * 10^6 = 5036 [ppm]$$

Ecuación N°.65 - Primer ensayo 4000ppm – Ajuste de concentración ppm CO2

En el segundo ensayo, el único cambio fue la temperatura registrada.

$$T = 37.1 [^{\circ}C] = 310.25 [K]$$

$$CO_2[ppm] = 5053 [ppm].$$

El aumento en 1.1 [°C] significó un aumento de 17 [ppm].

#### 4.2.4. CO – Combustión incompleta de papel y humo de cigarrillo

En estas pruebas, nos propusimos como objetivo la **correcta detección** del gas monóxido de carbono mediante su inyección, primero a través de la **combustión incompleta** de una servilleta de papel y luego mediante la inyección de humo de tabaco.

En la primera prueba, se colocó un fragmento de servilleta en la abertura de una botella previamente comprimida. A continuación, se procedió a encender el papel, generando una fuente de humo. Posteriormente, se liberó la presión ejercida sobre la botella, lo que permitió que recuperara su forma original y, en consecuencia, absorbiera el humo generado por la combustión. La llama fue extinguida de forma intencional y la botella se selló con una tapa perforada, a la cual se conectó una manguera. Al presionar la botella, el humo fue inyectado progresivamente en el habitáculo, lo que permitió medir la **concentración** de monóxido de carbono en su interior.

En la segunda prueba, se replicó el procedimiento anterior, pero en esta ocasión se utilizó un cigarrillo en lugar del papel. La botella permaneció cerrada con la tapa perforada, en cuyo orificio se insertó el cigarrillo. Se procedió a encenderlo y, posteriormente, se liberó la presión ejercida sobre la botella, permitiendo que esta recuperara su forma original y atrapara el humo generado por la **combustión** del tabaco en su interior.



La calibración del sensor fue realizada de manera teórica basados en diferentes fuentes bibliográficas [36] [37] [38]. Bajo la misma metodología utilizada en las pruebas anteriores se obtuvieron datos de voltaje analógico medido por la placa y **valores en ppm** de las concentraciones del contaminante.

#### 4.2.5. Resultados

En este apartado ilustraremos los resultados de las pruebas, se hará un análisis y conclusión de lo obtenido.

##### 4.2.5.1. CO<sub>2</sub>- Prueba de saturación

Esta prueba tuvo como objetivo conocer que sucedía al invadir el ambiente de dióxido de carbono en una concentración mayor al límite del sensor.

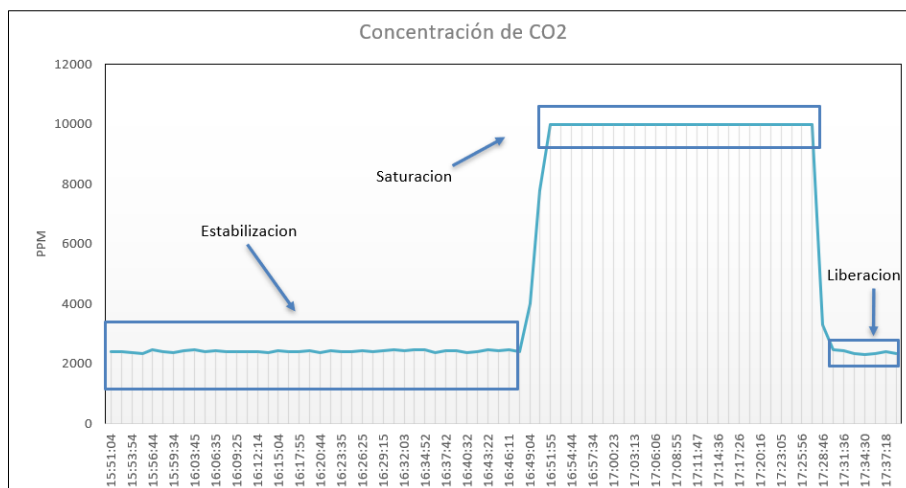


Fig. N°97 - Prueba de saturación – Grafica de datos

En esta prueba podemos destacar 3 etapas. En la primera, llamada “*estabilización*” se ubicó el sensor dentro del habitáculo sin agregado de contaminantes para que, durante una hora, tome lecturas y se estabilice. Los resultados de esta primera etapa arrojaron un promedio de 2414 [ppm] con “*offset*” de 86 [ppm] por encima del promedio y 74 [ppm] por debajo del promedio. Basados en la hoja de datos del sensor [21], que especifica un +/- 50[ppm], en las lecturas, determinamos un error máximo de 36 [ppm] lo cual no resulta significativo ya que en un ambiente normal la concentración oscila entre 2300 [ppm] y 2500 [ppm].

En la segunda etapa, llamada “*saturación*” se inundó el habitáculo con una concentración de 63555.55 [ppm]. Se puede observar la rapidez con la que el sensor



detecta y llega a su límite máximo de 10000 [ppm] y quedándose en dicha lectura durante toda etapa en la que el habitáculo permaneció cerrado, la etapa tuvo una duración de casi una hora y luego se abrió una hendidija para que salga el CO<sub>2</sub>, de manera que la concentración disminuya e inicia la tercera etapa llamada “liberación”. En esta última etapa se liberó todo el gas del habitáculo y las lecturas volvieron a sus valores normales.

#### 4.2.5.2. CO<sub>2</sub>- Pruebas de estimación

Estas pruebas tuvieron como objetivo analizar la precisión y exactitud del dispositivo, teniendo en cuenta las condiciones en las que se realizan los ensayos. Para estas pruebas, el volumen se tuvo que reducir debido a que el tiempo de espera hasta la detección era muy alto utilizando la unidad externa para la reacción. Entonces, se procedió a quitar la unidad externa y realizar la reacción dentro del habitáculo mayor.

#### 4.2.5.3. CO<sub>2</sub>- Prueba número uno

La primera prueba la cual consta de poner 2000 [ppm] dentro del habitáculo donde se ubica el dispositivo, se realizó mediante **dos ensayos** en dos días diferentes para un mejor resultado. Bajo las condiciones reales, los resultados analíticos arrojaron una concentración de 2540 [ppm] y 2550 [ppm] para el primer y segundo ensayo respectivamente.

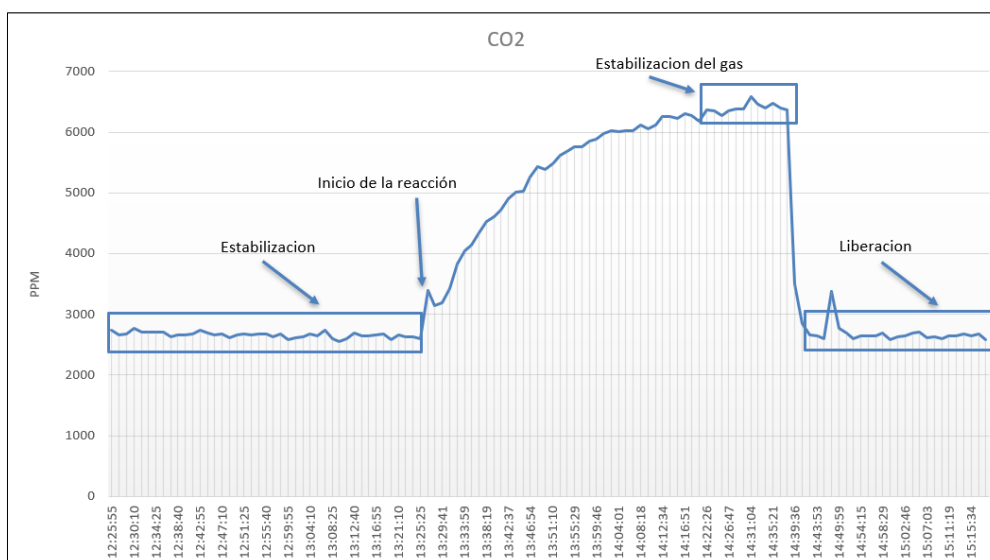


Fig. N°98 - Prueba uno de estimación – Grafica primer ensayo

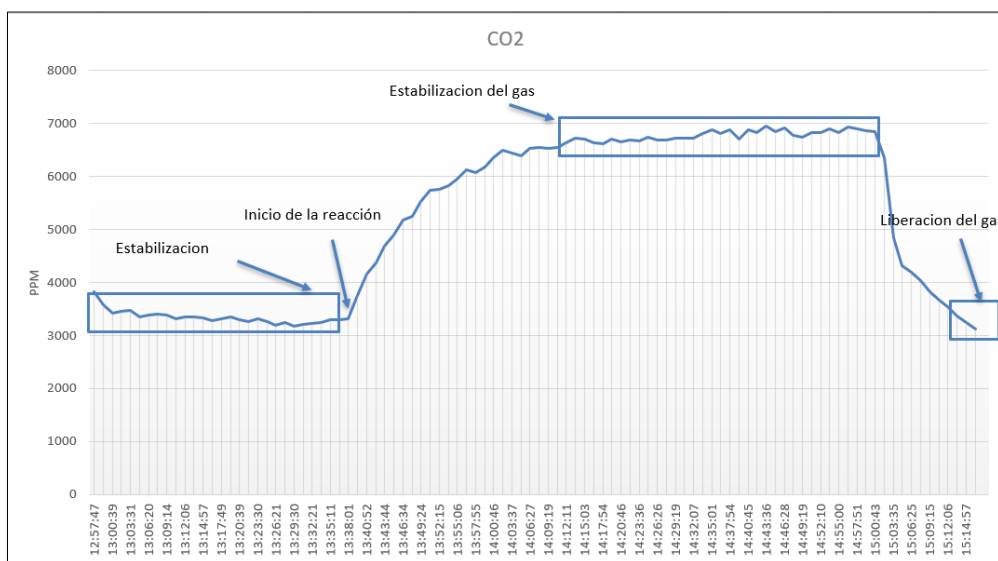


Fig. N°99 - Prueba uno de estimación – Grafica segundo ensayo

Como se observa, en ambos ensayos hubo un tiempo de estabilización del dispositivo de al menos una hora dentro del habitáculo. En la “*estabilización*” del primer ensayo el registro “normal” de ppm fue de 2640 [ppm] en promedio con máximos de 62 [ppm] por encima del promedio y 61 [ppm] (todo valor es truncado a entero). Sabiendo que el sensor tiene una exactitud de +/- 50 [ppm] fácilmente vemos que en el sistema se agrega un error de aproximadamente +/-12 [ppm]. En el segundo ensayo la concentración promedio fue cercana a los 3318[ppm], con picos de 155 [ppm] por encima del promedio y 142 [ppm]. En este caso la oscilación de valores fue mayor y por ende el error también, sin embargo, es destacar que no ubo **lecturas atípicas** durante esta primera etapa, tal vez solamente faltó tiempo de estabilización en este segundo ensayo.

Al momento de iniciar la reacción pudimos observar el mismo fenómeno en ambas pruebas, la **concentración** del gas fue elevándose, dibujando una curva logarítmica hasta acercarse al valor real de concentración del gas. Vemos que tarda al menos unos 30 minutos en llegar a un nivel que podríamos considerar estable, pero esto es relativo ya que, según los ensayos, dependió de la temperatura en el sistema y de la cercanía de la reacción al dispositivo (más temperatura y mayor reacción significan mayor velocidad de propagación del gas y detección más rápida).

Dentro del rango donde consideramos una concentración estable del gas contaminante pudimos extraer la siguiente información.

- En el primer ensayo se calcularon 2540 [ppm] de CO<sub>2</sub>, la concentración inicial fue de 2641[ppm] en promedio lo que nos resulta que en teoría debería haberse logrado una concentración de 5180 [ppm] dentro del



habitáculo. El resultado obtenido por nuestro dispositivo fue de 6398 [ppm] en promedio. Resulta en una diferencia de 1218 [ppm].

- En el segundo ensayo se calcularon 2550 [ppm] de CO<sub>2</sub>, la concentración inicial fue de 3318 [ppm] en promedio, en teoría debería haberse logrado una concentración de 5868 [ppm]. El resultado obtenido por nuestro dispositivo fue de 6768[ppm] en promedio. Resulta una diferencia de 900[ppm].

Por último, para la “*liberación*” del gas (abrir el habitáculo) se optaron por dos métodos diferentes en cada ensayo. En el primero se abrió por completo el habitáculo y resulto en una brusca caída de la concentración del gas hasta su valor “normal”. En el segundo, solo se abrió parcialmente el habitáculo de manera que el contaminante se libere progresivamente, y resulto en una curva del tipo exponencial decreciente.

#### 4.2.5.4. CO<sub>2</sub>- Prueba numero dos

La prueba numero dos consta de elevar la concentración del contaminante dentro del habitáculo, en este caso se introdujeron 4000 [ppm]. El procedimiento fue el mismo que en la prueba número uno, dos ensayos en dos días diferentes. La **concentración** del gas calculada para ambos ensayos fue de 5036 [ppm] y 5053 [ppm] respectivamente.

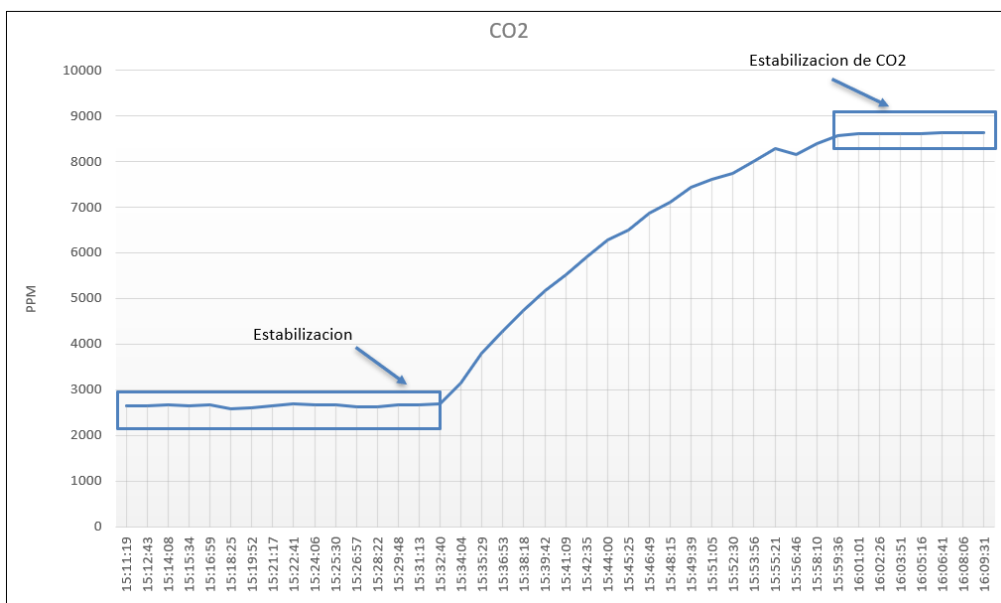


Fig. N°100 - Prueba dos de estimación – Grafica de primer ensayo

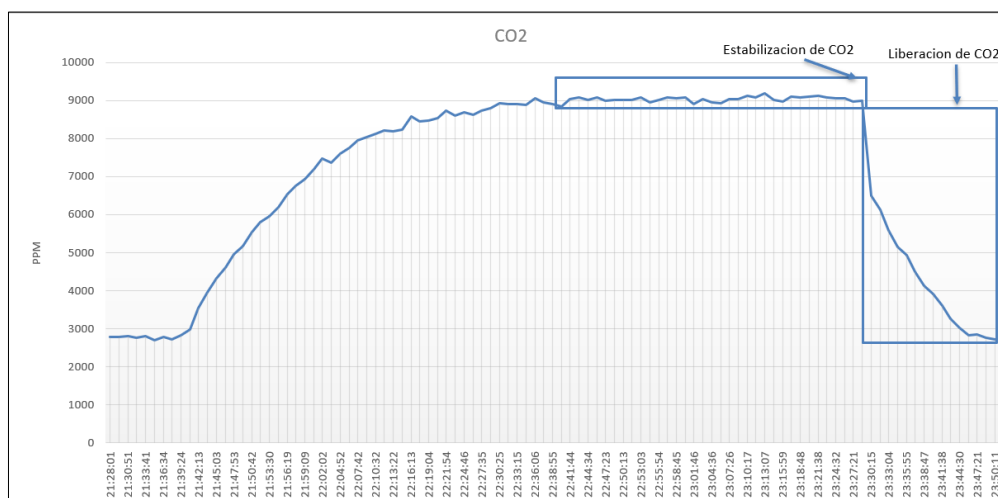


Fig. N°101 - Prueba dos de estimación – Grafica de segundo ensayo

Se dejó al dispositivo dentro del habitáculo un tiempo de estabilización sin contaminantes, luego se inicia la reacción dentro de este. El comportamiento fue el mismo en ambos ensayos, incluyendo la prueba número uno. La ascensión de la concentración del contaminante fue en forma logarítmica hasta llegar a un valor estable de concentración.

En esta prueba, se alteró el **tiempo de exposición** sin contaminantes dentro del habitáculo entre ambos ensayos.

El primer ensayo tuvo un tiempo prolongado obteniendo un promedio de 2653 [ppm] de concentración con picos de 38 [ppm] por encima del promedio y 70 [ppm] por debajo del promedio.

En el segundo ensayo solo fueron unos minutos, y se obtuvo una diferencia de 155[ppm] por encima del promedio.

Luego de producida la reacción y de inyectar el contaminante al habitáculo. Las lecturas logran **estabilizarse** y oscilar entre un valor promedio concentración en ppm y obtuvimos la siguiente información.

- Para el primer ensayo, se obtuvo un **resultado analítico** de 5036 [ppm]. Previo a la inyección del contaminante la concentración obtenida por el sensor fue de 2653[ppm], sumando ambos resulta en una concentración teórica de 7689[ppm] dentro del habitáculo. Como **resultado empírico** obtuvimos una concentración promedio de 8470[ppm] lo que resulta en una diferencia de 780[ppm].

Para el segundo ensayo, se obtuvo un **resultado analítico** de 5053 [ppm]. Previo a la inyección de contaminante, la concentración obtenida por el sensor fue de 2799 [ppm], sumando ambos resulta en una concentración teórica de 7852 [ppm]



dentro del habitáculo. Como **resultado empírico** obtuvimos una concentración promedio de 9013 [ppm] lo que resulta en una diferencia de 1161 [ppm].

#### 4.2.5.5. CO- Combustión incompleta

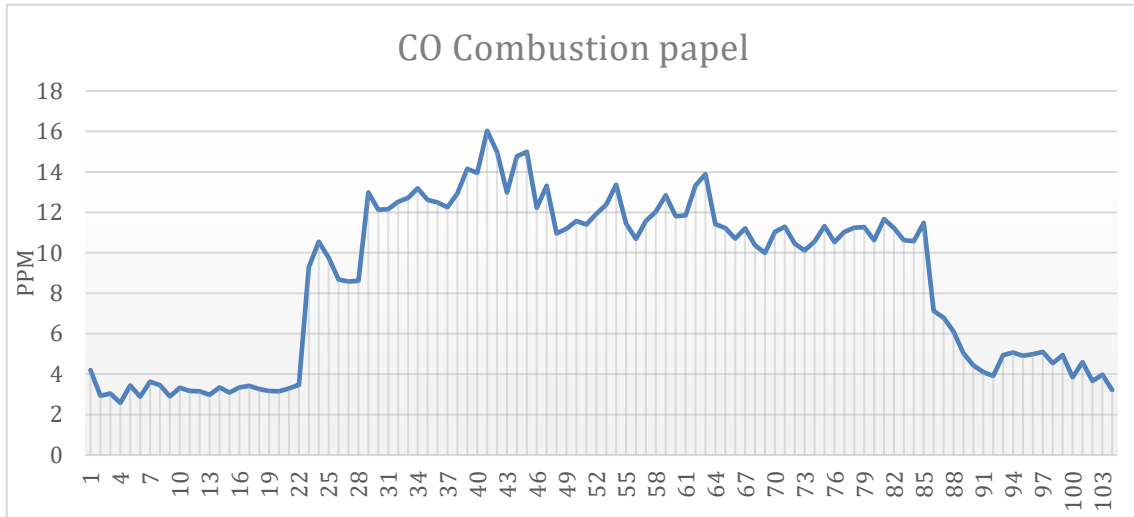


Fig. N°102 - Prueba de papel servilleta – Grafica de concentración ppm

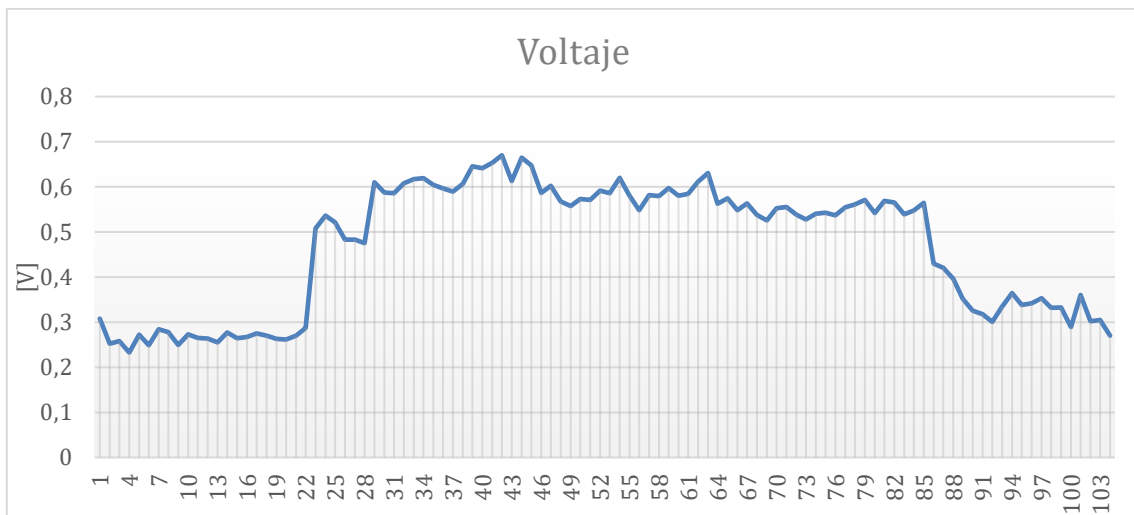


Fig. N°103 - Prueba de papel servilleta – Grafica de voltaje leído

En la prueba de **combustión incompleta** del papel servilleta, se procuró primeramente lograr lecturas de concentración estables, para luego introducir el contaminante al habitáculo. A partir de la lectura número 22 se introduce el contaminante y es detectado casi en su **inmediatez**, ya en su próxima lectura (luego de 85 segundos) se ve un brusco aumento de la concentración y lo mismo se replica en la gráfica



de voltaje, luego la concentración se mantiene elevada marcando algunos altibajos, pero nunca decayendo al valor de inicio, efectivamente se logra ver la detección del gas acompañado del aumento en las lecturas ppm. Se observan grandes fluctuaciones en la gráfica ppm, pero no así en la gráfica de voltaje, esto es porque las concentraciones ppm están presentadas sin **base logarítmica**, así como lo está en la hoja de datos del sensor. Si hiciéramos la conversión logarítmica de los valores ppm obtendríamos una gráfica muy similar su forma a la de voltaje. A pesar de haber ingresado una gran cantidad de humo, solo se observó un pico máximo de 16[ppm] y un pico mínimo de ~ 10 [ppm] manteniéndose dentro de estos límites. Para finalizar la prueba, se liberó el gas dentro del habitáculo hasta volver a la concentración de inicio.

#### 4.2.5.6. CO – Cigarrillo

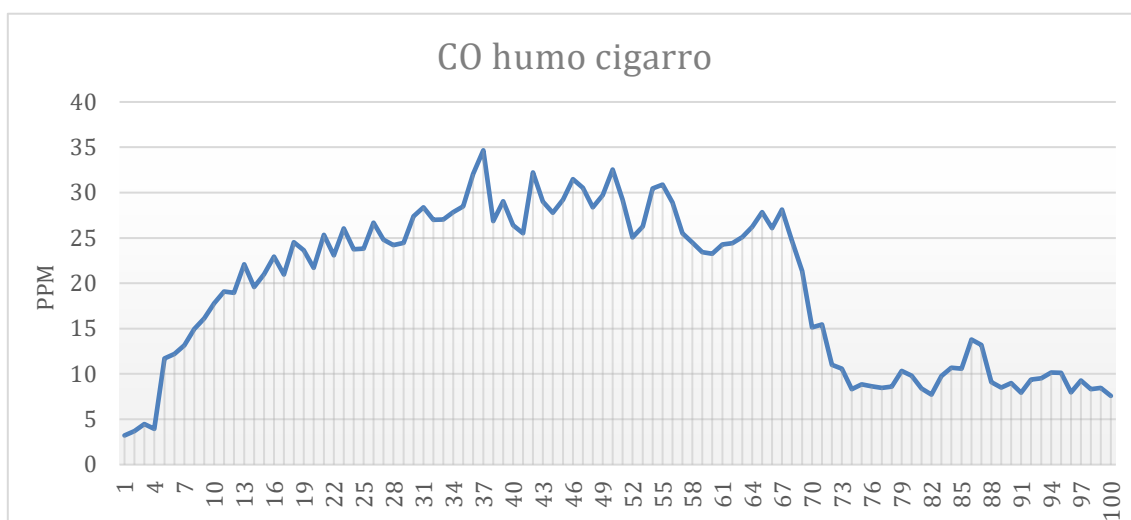


Fig. N°104 - Prueba de Cigarrillo – Grafica de concentración ppm

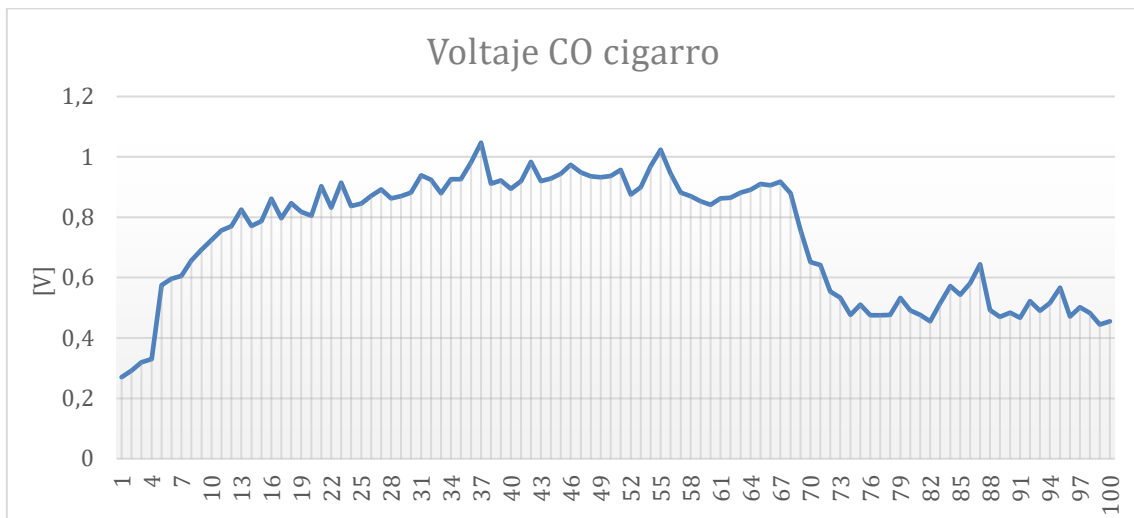


Fig. N°105 - Prueba de Cigarrillo – Grafica de voltaje leído

En la prueba de cigarrillo, se consumió la mitad de este atrapando su humo y luego inyectándolo en el habitáculo. La **detección** no fue brusca como en la prueba anterior, se observa un crecimiento de **forma gradual** con fluctuaciones como era de esperarse, se obtuvo un pico máximo de 35 [ppm] y un pico mínimo de 22 [ppm] en presencia del contaminante.


Luego de una hora de lectura con el humo encerrado, en la muestra número 68 se comienza a liberarlo dejando una pequeña hendidura abierta lo cual comienza a notarse en las gráficas.

#### 4.2.6. Conclusiones de las pruebas

Se realizaron **cálculos analíticos** en condiciones normales, ajustes por condiciones reales y varios ensayos para obtener el mejor resultado posible, a pesar de que el método utilizado no es estrictamente exacto debido a que depende de muchas variables y fue desarrollado en un ambiente doméstico no controlado. La elección del método para generar CO<sub>2</sub> radica en su sencillez de implementación y en que no expone a los usuarios a gases nocivos.

En la prueba de “*saturación*”, se observó la precisión del sensor dentro del sistema. Durante la etapa de “*estabilización*”, se registró un error de 36 [ppm] por encima de lo especificado en el datasheet, lo cual puede atribuirse a las condiciones en las que se desarrolló la prueba. También se determinó que, al superar una concentración por encima del límite, el sensor solo muestra su lectura máxima. Por lo tanto, podemos concluir que el sensor es **fiable** para lecturas menores a 10000 [ppm], pero no para lecturas superiores a su límite.

En las pruebas de “*estimación*”, el comportamiento del dispositivo fue, en términos generales, **consistente**, lo cual refleja la estabilidad que ofrece. Desde un punto

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO <sub>2</sub> y Partículas
Ingeniería en Electrónica	4. Resultados


de vista cualitativo, podemos afirmar que el dispositivo se comporta de manera estable y precisa, sin lecturas atípicas y dentro de los rangos esperados. Desde un punto de vista **cuantitativo**, no se esperaba un resultado perfecto debido a las condiciones del entorno, pero se puede destacar que el dispositivo tuvo una respuesta aceptable. La diferencia más significativa se obtuvo en el primer ensayo de la primera prueba (1218 [ppm]), mientras que la menor diferencia se registró en el primer ensayo de la segunda prueba (780 [ppm]).

Dada la tolerancia asociada a este gas, un error de esta magnitud representa un riesgo moderado. Sin embargo, es importante mencionar que las reacciones realizadas no fueron perfectas. Aunque se hicieron los **cálculos correctos**, no se contó con un elemento patrón para determinar si la **concentración calculada** correspondía con la **concentración real** lograda. Si se realizaran ensayos en un ambiente controlado o una calibración en un entorno libre de contaminantes, podrían obtenerse resultados más precisos en las lecturas.

En las pruebas realizadas para la medición y detección de monóxido de carbono, no se llevaron a cabo cálculos matemáticos, ya que cualquier método para estimar la concentración del gas en estas condiciones resultaría **inexacto**. Si bien no fue posible determinar la precisión y exactitud del dispositivo en la medición de este gas debido a la falta de un instrumento patrón para contrastar las mediciones, se pudo comprobar que su respuesta es **coherente**, ya que detectó la presencia del gas y mostró un aumento en la concentración a medida que se inyectaba humo en el habitáculo en ambas pruebas.

Se observó una diferencia entre ambas pruebas. En la prueba con papel de servilleta, la detección del gas fue más **brusca**, con un incremento repentino en la concentración de monóxido de carbono. En contraste, en la prueba con el cigarrillo, se obtuvo un valor de concentración en ppm mayor que en la prueba con papel, pero el incremento fue más **gradual**, lo que sugiere una emisión más sostenida del gas en el tiempo.

En conclusión, a pesar de las limitaciones experimentales y la falta de un estándar de referencia para la **calibración**, el dispositivo demostró un desempeño acorde a lo esperado. Su respuesta fue **consistente** con la presencia y variación de la concentración de gases en el entorno, lo que sugiere que puede ser una herramienta fiable dentro de los rangos evaluados. Para mejorar su precisión, se recomienda realizar pruebas en un **ambiente controlado** con instrumentos de referencia.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	5. Análisis de costos

## Capítulo 5: Análisis de costos

### 5.1. Análisis


En este capítulo se abordará el análisis de costos, incluyendo coste de materiales, mano de obra de terceros y mano de obra de personal especializado, en este caso, ingenieros. A continuación, se detallan los costes fijos que estuvieron presentes en este proyecto, donde se incluye costes de materiales, el coste de realización de la placa PCB y de la impresión 3D.

Producto	Descripción	Cantidad	Valor Unitario	Total ARS	Valor USD	Total USD
ZH06	Sensor de partículas	1	\$ 13.800,00	\$ 13.800,00	29,55	29,55
MHZ-19	Sensor de Dioxido de Carbono	1	\$ 8.237,00	\$ 8.237,00	17,64	17,64
ESP32	Microcontrolador	1	\$ 3.300,00	\$ 3.300,00	7,07	7,07
MQ7	Sensor de Monoxido de Carbono	2	\$ 1.143,50	\$ 2.287,00	2,45	4,9
DHT11	Sensor de Humedad y Temperatura	1	\$ 735,00	\$ 735,00	1,57	1,57
ZP16	Sensor de Formaldehido	1	\$ 600,00	\$ 600,00	1,28	1,28
ILI9488	Pantalla Touch	1	\$ 7.213,00	\$ 7.213,00	15,45	15,45
Zocalos Ic	Zocalos para encastrar IC	4	\$ 200,00	\$ 800,00	0,43	1,71
CD4051	Multiplexor 8 canales	2	\$ 1.700,00	\$ 3.400,00	3,64	7,28
Resistencias	Resistores	11	\$ 100,00	\$ 1.100,00	0,21	2,36
1 uF	Capacitor Electroлитico	3	\$ 100,00	\$ 300,00	0,21	0,64
10nF	Capacitor ceramico	3	\$ 80,00	\$ 240,00	0,17	0,51
TSX01108BE	Adaptador de niveles Logicos	2	\$ 3.800,00	\$ 7.600,00	8,14	16,27
Conector Plug	Ficha de alimentacion pcb	1	\$ 500,00	\$ 500,00	1,07	1,07
Headers	Tira de pines macho x40	4	\$ 2.000,00	\$ 8.000,00	4,28	17,13
Cables dupont	Cables dupont 1.27mm x8	0,5	\$ 5.000,00	\$ 2.500,00	10,71	5,35
Housing	Pines hembra x40 para armado de conectores	2	\$ 2.500,00	\$ 5.000,00	5,35	10,71
Crimp	Crimps metalicos para armado de conectores x40	2	\$ 2.500,00	\$ 5.000,00	5,35	10,71
Placa PCB	Placa PCB	3	\$ 23.000,00	\$ 69.000,00	49,25	147,75
Impresión 3D	Materiales + Mano de obra	1	\$ 19.050,00	\$ 19.050,00	40,79	40,79
Tornillos	Bulones Alem x11cm	4	\$ 900,00	\$ 3.600,00	1,93	7,71
Logistica	Envio a domicilio de comprar por internet	1	\$ 3.184,00	\$ 3.184,00	6,82	6,82
<b>TOTAL</b>			<b>\$ 99.642,50</b>	<b>\$ 165.446,00</b>	<b>\$ 213,36</b>	<b>\$ 354,27</b>

Tabla N°10 - Costos Fijos

De la tabla se observa, los valores expresados en las últimas dos columnas están representados en moneda extranjera a valor dólar Libre Blue a la fecha en la cual se realizó la compra de los materiales y/o pago de los servicios prestados por terceros siempre y cuando la compra se haya realizado en Argentina. Los sensores, en su mayoría, fueron importados desde china a valor dólar tarjeta y se encuentran expresados al tipo de cambio correspondiente.

A continuación, se expresa el valor costo de mano de obra. Se tiene en cuenta el sueldo inicial de dos Ingenieros electrónicos con experiencia laboral menor a un año de trabajo.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	5. Análisis de costos

Sueldo mensual de un ingeniero	Días mensuales	Horas diarias	Horas totales	Valor hora	Valor hora USD
\$ 1.500.000,00	20	8	160	\$ 9.375,00	\$ 7,68

Tabla N°11 - Costo Hora hombre

La duración en bruto de este proyecto fue de un año y seis meses. Teniendo en cuenta que el trabajo en este proyecto fue en promedio cinco horas diarias, tres días a la semana y, solamente tomando los días de trabajo, calculamos en términos de valores netos el costo de mano de obra ingenieril expresando el detalle de las horas trabajadas.

Trabajo	Horas
Puesta a punto de sensores	60
Diseño de PCB	60
Montaje PCB	15
Diseño de 3D	120
Programacion	305
Pruebas de Campo	40
<b>Total Horas</b>	<b>600</b>

Tabla N°12 - Detalle horas trabajadas

Meses de trabajo neto	Días por mes	Horas de trabajo diarias netas	Horas de trabajo netas totales	Valor Mano de obra ARS	Valor mano de obra USD
10	12	5	600	\$ 5.625.000,00	\$ 4.608,00

Tabla N°13 - Costo Neto mano de obra


Debe comprenderse que, si bien se necesitaron 18 meses de trabajo para la realización de este proyecto, solo se trabajó en 10 meses debido a razones externas. Se puede observar que el costo por ingeniero en dólares al día 21/10/2024 es de USD \$4608 que distribuidos en el tiempo no es una suma significativa.

Para resumir, se presenta el costo total del proyecto y se realizan conclusiones sobre este.



Costos Fijos ARS	Costos fijos USD	Mano de obra ARS	Mano de obra USD
\$ 165.446,00	\$ 354,27	\$ 11.250.000,00	\$ 9.216,00
		<b>TOTAL ARS</b>	\$ 11.415.446,00
		<b>TOTAL USD</b>	\$ 9.570,27

Tabla N°14 - Presupuesto Total

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO <sub>2</sub> y Partículas
Ingeniería en Electrónica	6. Discusión y conclusión

## Capítulo 6: Discusión y conclusión


Este proyecto surge a partir del interés en conocer la calidad del aire del ambiente en el que respiramos y nos desarrollamos diariamente, así como también sus efectos en la salud, con el objetivo de explicar ciertos síntomas, en su mayoría respiratorios, que afectan a las personas en ambientes cerrados y con **escasa ventilación**. A lo largo de este proyecto, se ha analizado qué sucede cuando no se mantiene una habitación ventilada y qué gases son responsables de estos síntomas. Repasando los objetivos planteados, se ha demostrado que el **dispositivo es capaz** de medir las concentraciones de gases o partículas en niveles potencialmente nocivos durante un periodo de tiempo, así como también medir magnitudes físicas adicionales, como la temperatura y la humedad.

El dispositivo permite visualizar los valores de las magnitudes físicas en tiempo real con un intervalo de cinco segundos entre lecturas, con un retardo menor a un segundo entre la medición y su visualización. En caso de que alguna magnitud supere el umbral definido, el sistema es capaz de notificar al usuario tanto de manera local como remota. Localmente, la alerta se genera mediante indicadores en pantalla y señales sonoras, mientras que, a distancia, las notificaciones se envían a un dispositivo móvil mediante la aplicación, de forma clara y concisa. De este modo, el dispositivo mantiene informado al usuario en todo momento sobre la calidad del aire.

Las **pruebas de campo** realizadas reflejaron la respuesta del sensor de CO<sub>2</sub> bajo condiciones de saturación y con dos concentraciones diferentes, medidas en partes por millón (ppm). El objetivo principal era verificar **la correcta detección del gas** y que la respuesta del sensor fuera proporcional a la concentración introducida en el habitáculo. Los resultados nos mostraron que el comportamiento fue el esperado.

En cuanto al monóxido de carbono (CO), las pruebas realizadas confirmaron la capacidad del dispositivo para **detectar su presencia** y variación en la concentración. Aunque no se llevaron a cabo cálculos matemáticos para determinar la concentración exacta del gas debido a la ausencia de un instrumento de referencia, se observó que el sensor respondió de manera coherente ante la inyección de humo en el habitáculo. Se evidenció un incremento de la concentración de CO a medida que se añadía más humo, lo que valida la funcionalidad del sistema para la detección de este gas. Además, se identificó una diferencia en la respuesta del sensor dependiendo de la **fuerza de emisión**.

Por último, en cuanto a los costos notamos que la primera inversión para el desarrollo del dispositivo es elevada (valuada en el contexto argentino con inflación) debido al tiempo invertido en investigación y desarrollo. Sin embargo, en una segunda unidad los costos se reducirían casi en un 70% ya que no habrá **etapas de diseño, investigación y desarrollo**. El costo estará mayormente marcado por la compra de insumos y el ensamble de partes, si existiese la posibilidad de una pro-

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	6. Discusión y conclusión

ducción a gran escala los **costos** reducirían significativamente. En resumen, podemos decir que nuestro dispositivo producido en grandes cantidades sería de fácil acceso económico.

Concluimos que nuestros **objetivos** han sido **alcanzados**. Este dispositivo se diferencia de otros medidores de calidad del aire en dos aspectos clave: en primer lugar, la integración de la medición de material particulado, y, en segundo lugar, la incorporación del **paradigma IoT**, permitiendo el monitoreo en tiempo real y a distancia mediante una aplicación móvil. A pesar de las limitaciones experimentales, el sistema ha demostrado un desempeño acorde a lo esperado, respondiendo de manera fiable y consistente dentro de los rangos evaluados.

### 6.1. Mejoras

Para cerrar, podemos sugerir mejoras en el dispositivo a fines de optimizar su performance en varios aspectos que detallaremos a continuación.


Adición de una batería en conjunto con un regulador de voltaje transformando el dispositivo de alámbrico a inalámbrico permitiendo la portabilidad.

Posibilidad de conectarse a una red wifi por medio de la pantalla, esta característica mejoraría notablemente al dispositivo.

Adición de funciones en la aplicación móvil para la gestión del dispositivo. La aplicación móvil solo cumple la función de visualizar los datos obtenidos desde el dispositivo, sería interesante que desde esta sea posible gestionarlo y que este actúe en consecuencia.


Reversionar placa PCB. Si bien funciona correctamente, es posible mejorar la placa PCB haciéndola más robusta y con mayor inmunidad al ruido o a posibles errores de falsos contactos.

Cambio de conectores de interconexión, con esto nos referimos a mejorar la interconexión entre los diferentes dispositivos optando por conectores más robustos evitando errores de falso contacto debido a vibraciones o caídas.


 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO <sub>2</sub> y Partículas
Ingeniería en Electrónica	7. Bibliografía

## Capítulo 7: Bibliografía

- [1] Editorial Etecé. (2021, Julio) Concepto.de. [Online]. <https://concepto.de/aire/>
- [2] Erica Yanina Sanchez Andrés Porta, *Calidad de aire, monitoreo y modelado de contaminantes atmosféricos. Efectos en la salud pública*, 1st ed., Universidad de la plata, Ed. La Plata, Argentina: Editorial de la Universidad de La Plata, 2018.
- [3] Carlos Fito. (2024, Noviembre) ITENE. [Online]. <https://itene.com/soluciones/calidad-aire-industrial-exterior-urbano-interior/>
- [4] Organizacion mundial de la salud. who. [Online]. [https://www.who.int/es/health-topics/air-pollution#tab=tab\\_1](https://www.who.int/es/health-topics/air-pollution#tab=tab_1)
- [5] Ente Nacional Regulador de la Electricidad. (2018) Enre.gov.ar. [Online]. <https://www.enre.gov.ar/web/bibliotd.nsf/58d19f48e1cdebd503256759004e862f/13c4a17de5155b1f03256de50070020e?OpenDocument>
- [6] Gobierno de Entre Ríos. (1978, Noviembre) entrerios. [Online]. [https://www.entrerios.gov.ar/industria/userfiles/files/Ley\\_6260.pdf](https://www.entrerios.gov.ar/industria/userfiles/files/Ley_6260.pdf)
- [7] Terrie K. Boguski, "Understanding Units of Measurement," Manhattan, 2006. [Online]. [https://cfpub.epa.gov/ncer\\_abstracts/index.cfm/fuseaction/display.files/fileid/14285](https://cfpub.epa.gov/ncer_abstracts/index.cfm/fuseaction/display.files/fileid/14285)
- [8] National Institute of Biomedical Imaging and Bioengineering. (2022, Abril) nibib.nih.gov. [Online]. <https://www.nibib.nih.gov/espanol/temas-cientificos/sensores#:~:text=%C2%BFQu%C3%A9%20son%20los%20sensores%3F,cantidades%20y%20caracter%C3%ADsticas%20que%20detectan>
- [9] Jeff Shepard. (2022, Octubre) digikey.com. [Online]. <https://www.digikey.com/es/articles/how-to-use-smart-air-quality-sensors-for-environmental-monitoring>
- [10] MRU-Instruments. (2022) MRU- Instruments. [Online]. <https://mru-instruments.com/understanding-electrochemical-sensors-and-their-shelf-life-a-deep-dive/#:~:text=Los%20sensores%20electroqu%C3%ADmicos%20son%20dispositivos,de%20la%20calidad%20del%20aire.>
- [11] AsahiKasei. (2024, May) akm.com. [Online]. <https://www.akm.com/global/en/products/co2-sensor/tutorial/what-is-ndir/>
- [12] Co2Meter. (2024, Sep.) Co2Meter.com. [Online]. <https://www.co2meter.com/blogs/news/how-does-an-ndir-c>
- [13] Beckam solutions. (2020) Beckam.es. [Online]. <https://www.beckman.es/resources/technologies/laser-diffraction>


 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO <sub>2</sub> y Partículas
Ingeniería en Electrónica	7. Bibliografía

- [14] Espressif. [Online]. <https://www.espressif.com/en/products/socs/esp32>
- [15] Arduino. [Online]. <https://www.arduino.cc/>
- [16] Firebase. [Online]. <https://firebase.google.com/>
- [17] Hanwei Electronics CO. Technical Data MQ7 Gas Sensor. Datasheet. [Online]. <https://www.flyingfish.com/file/0J313/MQ7.pdf>
- [18] Jaycon. (2023, September) Gas Sensor. WebSite. [Online]. <https://www.jaycon.com/understanding-a-gas-sensor/>
- [19] WebPlotDigitizer. Automers. [Online]. <https://automeris.io/>
- [20] Web Plot Digitizer. automeris. [Online]. <https://apps.automeris.io/wpd4/>
- [21] Winsen Sensor. (2020) MH-Z19 Datasheet. Datasheet. [Online]. <https://www.winsen-sensor.com/d/files/PDF>
- [22] Winsen Sensor. (2021, Aug.) ZH06- Laser dust sensor. Datasheet. [Online]. <https://www.winsen-sensor.com/d/files/manual/zh06-iv.pdf?searchid=4722>
- [23] Winsen Sensors. (202) Zp16 Datasheet. Datasheet. [Online]. <https://www.winsen-sensor.com/product/zp01-mp503.html>
- [24] Mouser. (2020) DHT11 Temperature & Humidity Sensor. Datasheet. [Online]. <https://www.mouser.com/datasheet/2/758/dht11-technical-data-sheet-translated-version-1143054.pdf>
- [25] Spressif Systems. (2021, Mar.) ESP32-Datasheet. Datasheet. [Online]. [https://www.espressif.com/sites/default/files/documentation/esp32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf)
- [26] Spressif Systems. Spressif. [Online]. <https://www.espressif.com/en/products/devkits>
- [27] ILITEK. (2019, June) HPinfotech.ro. [Online]. <https://www.hpinfotech.ro/ILI9486.pdf>
- [28] (2020, Mar.) engineersgarage.com. [Online]. <https://www.engineersgarage.com/arduino-ili9486-driver-3-5-inch-tft-lcd-touch-screen>
- [29] Amazon. (2024, Oct.) Amazon. [Online]. <https://www.amazon.com/-/es/DIYmalls-pantalla-pulgadas-resistiva-Mega-2560/dp/B0BFDYCX94>
- [30] Texas Instruments. (2023, September) CD405xB CMOS Single 8-Channel Analog Multiplexer or Demultiplexer With Logic-Level Conversion. Datasheet.
- [31] Texas Instruments. (2022, June) TXS0108E. Datasheet. [Online]. [https://www.ti.com/lit/ds/symlink/txs0108e.pdf?ts=1727337857409&ref\\_url=https%253A%2](https://www.ti.com/lit/ds/symlink/txs0108e.pdf?ts=1727337857409&ref_url=https%253A%2)

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	7. Bibliografía

[52F%252Fwww.ti.com%252Fproduct%252FTXS0108E%253Futm\\_source%253Dgoogle%2526utm\\_medium%253Dcpc%2526utm\\_campaign%253Dasc-null-null-GPN\\_EN-cpc-pf-google-eu%2526utm\\_content%253](https://www.ti.com/product/FTXS0108E?utm_source=google&utm_medium=dcpc&utm_campaign=asc-null-null-GPN_EN-cpc-pf-google-eu&utm_content=52)

- [32] Fernán García de Zuñiga. (2024, Apr.) Arsys. [Online]. [1. https://www.arsys.es/blog/archivo-json-que-es-y-para-que-sirve#:~:text=Los%20archivos%20JSON%20son%20simples,programaci%C3%B3n%20como%20para%20las%20personas](https://www.arsys.es/blog/archivo-json-que-es-y-para-que-sirve#:~:text=Los%20archivos%20JSON%20son%20simples,programaci%C3%B3n%20como%20para%20las%20personas)
- [33] RedHat. (2020, Marzo) Red Hat. [Online]. <https://www.redhat.com/es/topics/cloud-native-apps/stateful-vs-stateless>
- [34] Dart. (2024, November) Sync Fusion flutter charts. Web Page. [Online]. [https://pub.dev/packages/syncfusion\\_flutter\\_charts](https://pub.dev/packages/syncfusion_flutter_charts)
- [35] (2025, Feb) Meteored.com. [Online]. [https://www.meteored.com.ar/tiempo-en\\_Parana-America+Sur-Argentina-Entre+Rios-SAAP-1-16889.html](https://www.meteored.com.ar/tiempo-en_Parana-America+Sur-Argentina-Entre+Rios-SAAP-1-16889.html)
- [36] Nisal Kobbekaduwa. (2021, September) Calibration and Implementation of Heat Cycle requirement of MQ7 Semiconductor Sensor for detection of carbon monoxide concentrations.
- [37] Yersson Ramiro Carrillo-Amado, "Calibration and standardization of air quality measurements using MQ sensors," *Journal of Engineering Sciences*, vol. 1, p. 8, April 2020.
- [38] Laurent Michelle Easterline, "Smart Air Monitoring with IoT-based MQ2, MQ7, MQ8 and MQ135 sensors using NodeMcu ESP32," Bina Nusantara University, Jakarta, Indonesia, Investigation work ICCSCI 2024, 2024.
- [39] Flutter Docs. (2023, Mar.) docs.flutter.dev. [Online]. <https://docs.flutter.dev/resources/architectural-overview>
- [40] naylamp. (2024, Oct.) naylampmechatronics. [Online]. <https://naylampmechatronics.com/ardusystem-shields/360-shield-display-lcd-tft-35-tactil-ili9486ili9488.html>
- [41] (2022, Mar.) Json. [Online]. <https://www.json.org/json-es.html>
- [42] Vanessa Graham. (2018, August) Molekule. [Online]. <https://molekule.com/blogs/all/back-to-school-thought-does-air-quality-matter-for-students>
- [43] Samuel Torres de Carvalho Mera, "La calidad del aire en la UPM y su relacion con la huella de carbono," Universidad Politecnica de Madrid, Madrid, Tesis 2021.
- [44] DifferentAutors. (2023, Dec.) StackOverflow. [Online]. <https://stackoverflow.com/questions/12346349/why-is-json-more-lightweight-than-xml>

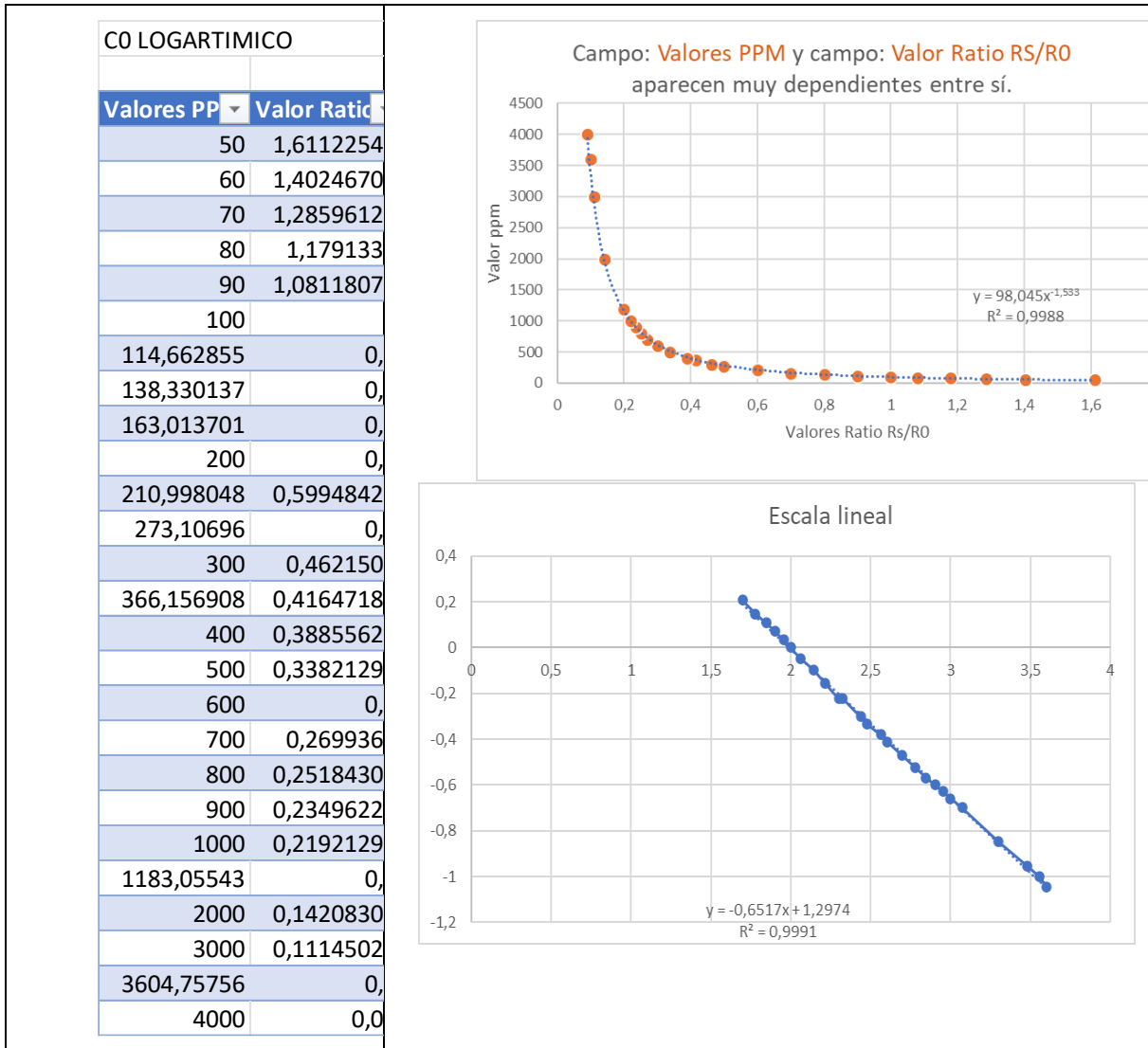
 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	7. Bibliografía

[45] KINNERA BHARATH KUMAR, "IOT BASED AIR QUALITY MONITORING SYSTEM USING MQ135 AND MQ7 WITH MACHINE LEARNING ANALYSIS," The PNG University of Technology, Papue, nueva Guinea, Investigation 1895-1767, 2019.



### Capitulo 8: Anexo


#### 8.1. Tabla de datos de curva MQ7



#### 8.2. Código Aplicación Móvil

En el siguiente link está disponible el repositorio de la aplicación móvil.

[LautaroMeza/ProyectoFinal](https://github.com/LautaroMeza/ProyectoFinal)

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	8. Anexo

### 8.3. Código Microcontrolador ESP32

En el siguiente link está disponible el repositorio del código desarrollado en la placa ESP32.

[LautaroMeza/ESP32AirQ](https://github.com/LautaroMeza/ESP32AirQ)

### 8.4. ESP32 – Características utilizadas



Fig. N°106 - ESP32-DevKit [26]

Esta placa de desarrollo nos permite acceder a la totalidad de salidas utilizables que tiene el microcontrolador. Dentro de sus características más importantes destacamos:

- 8. Placa Wifi, 802.11n (2.4 GHz), hasta 150 Mbps
- 9. Bluetooth v4.2
- 10. CPU 240 MHz
- 11. 3 interfaces UARTs
- 12. 4 Timers. Interno a cristal de 8 [MHz], Interno RC ajustable, interno RTC, y RTC Watchdog
- 13. 34 GPIOs programables
- 14. 2 DAC de 8 bits

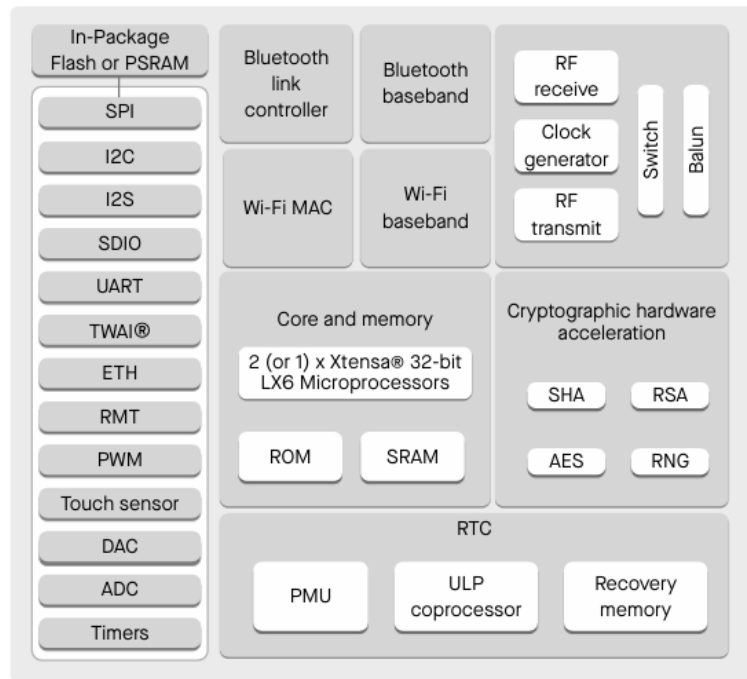


Fig. N°107 - ESP32- Funcional Block diagram [25]

En la siguiente imagen se puede ver con más detalles los GPIOs utilizables que tiene la placa.

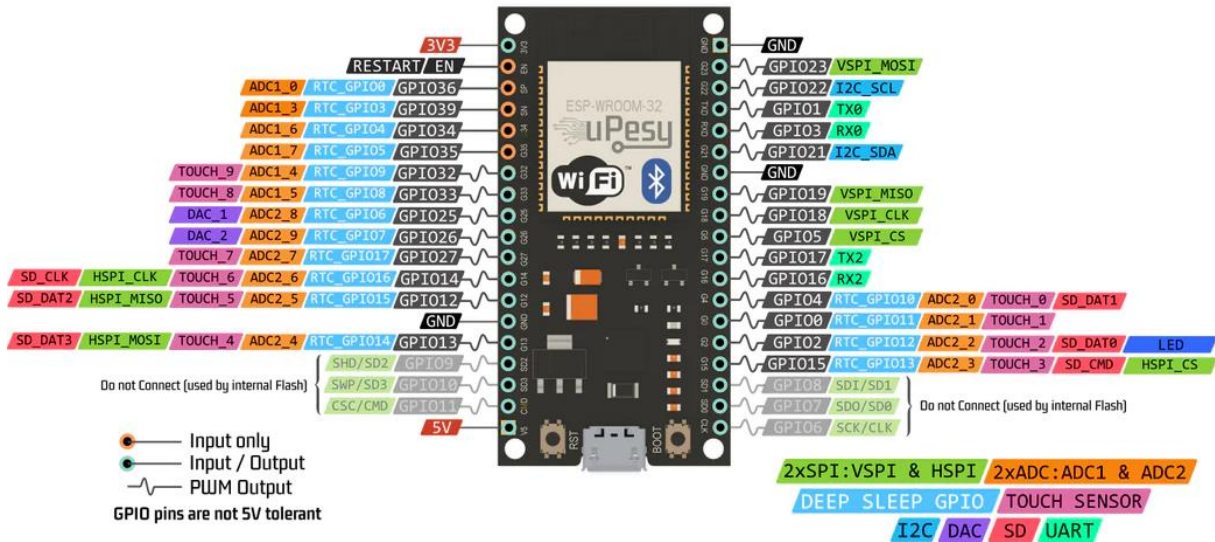


Fig. N°108 - ESP32-DevKit Pinout [14]

A continuación, se realizará una breve descripción de las características del ESP32 de las que haremos uso.



### 8.4.1.1. ADC

ESP32 integra dos SAR-ADC de 12 bits y soporta mediciones en 18 canales, 8 canales en ADC\_1 y 10 canales en ADC\_2, la palabra "SAR" refiere al término aproximaciones sucesivas el cual es el método utilizado por el microcontrolador para la conversión analógica-digital. Ambos ADC son gestionados por 5 controladores de los cuales dos son utilizados para multicanal, dos son utilizados durante el modo "deep-sleep" del esp32 y el último es utilizado para el módulo Wifi. Es importante saber, que, si el módulo Wifi está siendo utilizado, este último controlador no puede ser utilizado para lecturas ADC, en este caso todo el ADC 1 quedara bloqueado para uso externo. La siguiente imagen muestra la asignación de controladores para ambos ADC.

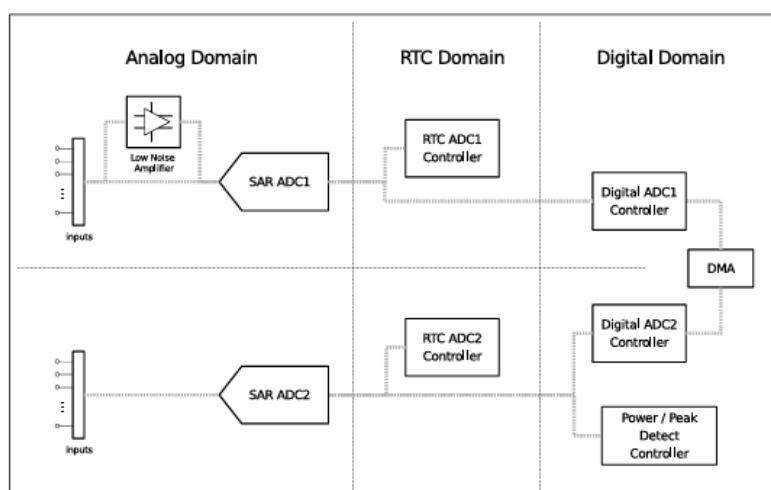


Fig. N°109 - ESP32- SAR ADC controllers [14]

Esp32 trabaja con tensiones de hasta 3.3V en sus pines ADC, con 4096 valores posibles, lo que resulta en una resolución de 0.8[v] por valor. En términos reales, existen no linealidades en la lectura, la siguiente imagen tomada del datasheet [25] indica el error máximo que tiene la lectura ADC.

Parameter	Description	Min	Max	Unit
DNL (Differential nonlinearity)	RTC controller; ADC connected to an external 100 nF capacitor; DC signal input;	-7	7	LSB
INL (Integral nonlinearity)	ambient temperature at 25 °C; Wi-Fi&Bluetooth off	-12	12	LSB
Sampling rate	RTC controller	-	200	ksps
	DIG controller	-	2	MSPS


 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	8. Anexo

Fig. N°110 - ESP32- características ADC [14]

Esta imagen indica que por cada lectura que realice el ADC tendrá un error de +/- 7 en el bit de menor peso. Para cada medición, se compara el valor a medir con una tensión de referencia por defecto de 1.1[V], si se quisiera medir con mayor resolución, el ADC provee funcionalidad de ganancia variable, esto es un circuito de atenuación para medir rangos de tensión más acotados, con mayor resolución y una significativa reducción del error, a continuación, se ilustra una tabla con error según la atenuación.

Parameter	Description	Min	Max	Unit
Total error	Atten = 0, effective measurement range of 100 ~ 950 mV	-23	23	mV
	Atten = 1, effective measurement range of 100 ~ 1250 mV	-30	30	mV
	Atten = 2, effective measurement range of 150 ~ 1750 mV	-40	40	mV
	Atten = 3, effective measurement range of 150 ~ 2450 mV	-60	60	mV

Fig. N°111 - ESP32- Error del ADC [14]

Por defecto, el ADC trabaja en rango de 0 – 3.3 [V] pero por no linealidades el rango real es de 150 – 2450 [mV], las cuales provocan un error de +/- 60[mV] lo cual resulta crítico e importante a la hora de realizar las lecturas desde el ADC.

#### 8.4.1.2. UART

El transmisor-receptor asíncrono universal (UART) incluido en el chip esp32 facilita la comunicación asíncrona de datos entre el chip y cualquier dispositivo compatible con UART. El microcontrolador cuenta con tres UARTs (UART0, UART1 y UART2) independientes entre sí, es decir, es posible utilizar todas en simultáneo, aunque no es recomendable utilizar la UART0 ya que es la encargada de establecer la comunicación con la PC para la programación del microcontrolador. La trama típica de UART inicia con un bit de START, seguido del dato en bits, luego el bit de paridad opcional para detección de errores y por último el bit de STOP. Cada UART cuenta con las siguientes características:

- a. Programable baud rate
- b. RAM compartida entre Rx FIFOs y Tx FIFOs
- c. Soporta varias longitudes de datos (6/7/8 bits)
- d. Soporta bit de stop
- e. Soporte de bit de paridad



- f. Soporte de protocolos RS232, RS485 y IrDA
- g. Soporte de control por hardware y por software

Trama de datos

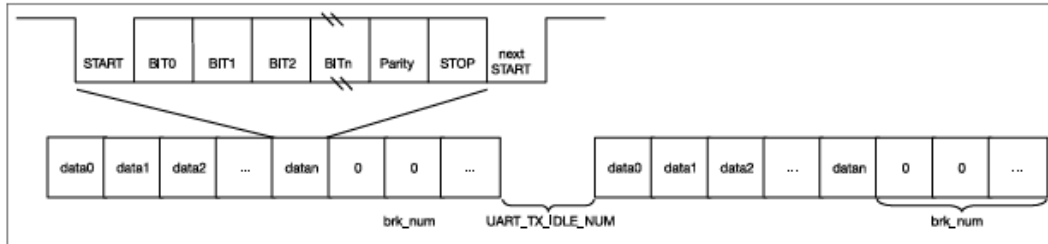


Fig. N°112 - ESP32-UART trama de datos [14]

La trama comienza con un solo bit de START en estado bajo y puede finalizar con 1, 1.5 ó 2 bits de STOP en estado alto. Continúa con el dato, la longitud del dato debe ser configurada y puede ser de 5 a 8 bits, luego sigue el bit de paridad que puede ser seteado o no.

a) Controlador UART

Al ser UARTs independientes, habrá tres controladores UART, uno por cada una. Todos ellos cuentan con idénticos set de registros para una documentación más eficiente y facilidad de programación, a continuación, se realiza una breve explicación del funcionamiento del controlador UART que es igual para las tres.

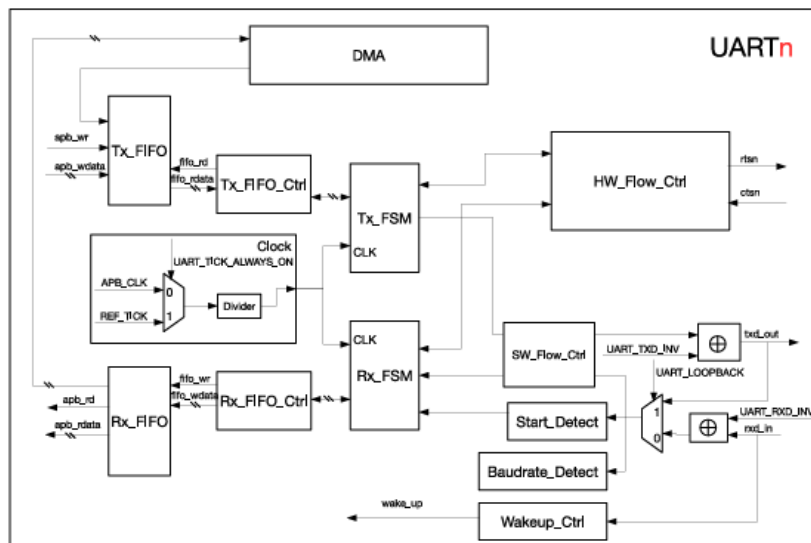



Fig. N°113 - ESP32- UART Controller [14]

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	8. Anexo

Del diagrama UART controller se observa que la fuente de reloj de sincronía puede provenir desde APB\_CLK, que es un clock de 80 MHz determinado por el CPU\_CLK o desde REF\_TICK el cual es utilizado para conversiones más lentas. Por defecto se utiliza APB\_CLK, pero es posible cambiar la fuente de reloj en la configuración de UART. Luego de la fuente de reloj se observa el bloque “divider” el cual se encarga de generar las señales de clock para el control del módulo UART [14].

El bloque de transmisión contiene un buffer FIFO transmisor, el cual aguardan ser transmitido. El software puede escribir TX\_FIFO vía APB y transmitir el dato dentro de TX\_FIFO vía DMA. Tx\_FIFO\_Ctrl es utilizado para controlar el acceso de escritura-lectura de TX\_FIFO. Cuando el registro no es nulo, TX\_FSM lee el dato a través de Tx\_FIFO\_Ctrl y transmite el dato fuera de acuerdo con el formato de trama seteado.


El bloque de recepción contiene un buffer FIFO receptor, el cual espera a ser procesado. El bit de entrada de flujo se conecta al controlador UART. El bloque “baudrate\_detection” se encarga de medir la velocidad de tasa de bits de la señal de entrada midiendo el menor ancho de pulso del bit de entrada de flujo. “Start\_detect” es utilizado para detectar el bit de START en la trama del dato recibido, luego de detectar el bit START, RX\_FSM almacena el dato recuperado desde la trama recibida dentro de Rx\_FIFO a través de Rx\_FIFO\_Ctrl [14].

#### 8.4.1.3. Timers

La placa ESP32 cuenta con dos módulos timers, cada uno con dos temporizadores ó timers, en total se disponen de cuatro temporizadores/timers de propósito general. Todos son de 64-bits basados en preescalers de 16-bits, el contador de timer puede ser de incremento o de decremento, ante un desborde del contador se produce una interrupción que puede ser controlada o también mediante el valor de un registro se puede comparar valores y lanzar una interrupción. En ambos casos, el contador vuelve a recargarse, es automático.

##### a) 16-bit preescaler

Cada timer utiliza APB\_clk (subreloj que resulta de dividir por un factor tres al CPU\_clk) de 80 [MHz] como fuente de tiempo. Este reloj es luego dividido mediante el preescaler de 16-bit lo cual genera la base de tiempo del contador del timer (TB\_clk) y cada ciclo de TB\_clk causa un incremento o decremento de dicho contador en uno. El preescaler puede dividir a APB\_clk en un factor desde 2 a 65536, específicamente si el preescaler toma el valor 1 o 2, el divisor se establece en 2, si el preescaler toma el valor 0, el divisor se establece en 65536 y para cualquier otro

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	8. Anexo

valor el divisor toma el valor del preescaler. La configuración del preescaler debe realizarse con el timer desactivado.

#### b) Contador 64-bit

El contador del timer es el encargado de llevar la cuenta de los ciclos que van ocurriendo desde el inicio del timer, tiene una longitud de 64-bit y al llegar a su valor máximo vuelve a cero, es utilizado para realizar funciones en tiempos determinados y certeros. Es posible que su incremento sea positivo o negativo, que se pueda habilitar o deshabilitar la cuenta (se pone en pausa el timer), establecer alarmas dependiendo eventos (contador llega a un valor determinado o se desborda) y es posible que inicie la cuenta desde un valor determinado (en caso de overflow se recarga en el valor seteado).

#### c) Generación de alarmas


Como se había introducido, el timer puede generar alarmas las cuales causan recargas y/o interrupciones. Las alarmas pueden ser habilitadas o deshabilitadas según se requiera y ocurren cuando el contador del timer alcanza un valor establecido en un registro específico, cuando el contador se desborda (incremento) o cuando el contador llega a cero (decremento). Una vez ocurrida la alarma se desactiva, es de una sola ocurrencia, pero puede volver a activarse y los registros del timer retoman los valores establecidos en su última configuración.

#### 8.4.1.4. WatchDog

El watchdog es un timer específico utilizado para recuperar al sistema ante una falla. El microcontrolador cuenta con tres Watchdog timers, uno en cada módulo timer sistema principal watchdog timer con sus registros asociados (WDT) y uno en el módulo RTC (RWDT). Existen casos en que el contador de programa sufre una alteración y se pierde la secuencia normal o queda en un bucle infinito, y es en estos casos donde entra en acción el watchdog reiniciando el programa.

Un temporizador watchdog tiene cuatro etapas. Cada etapa puede tomar una de tres o cuatro acciones al expirar un periodo de tiempo programado para esta etapa, a menos que el watchdog sea alimentado o deshabilitado. Las acciones son: interrupción, reinicio de la CPU, reinicio del núcleo y reinicio del sistema. Sólo el RWDT puede activar el reinicio del sistema, y es capaz de reiniciar todo el chip y el sistema principal, incluido el propio RTC. Se puede establecer un valor de tiempo de espera para cada etapa individualmente.

Durante el arranque flash, la RWDT y la primera MWDT se inician automáticamente para detectar y recuperarse de los problemas de arranque.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	8. Anexo

Es vital que se realice correctamente la calibración inicial de los sensores ya que son los dispositivos principales del proyecto, si no se realiza correctamente debe volver a intentarse, aquí es donde entra en acción el watchdog timer.

#### 8.4.1.5. Wifi

Una de las principales características por la cual se opta por usar este microcontrolador es la inclusión de un módulo wifi en placa. ESP32 está equipado con el módulo radio Wifi el cual consiste en emisor-receptor de 2.4 [GHz] bias y reguladores, balun de RF y emisor-receptor switches y un generador de reloj.

##### a) Receptor 2.4 [GHz]

El receptor de 2,4 GHz demodula la señal de RF de 2,4 GHz a señales de banda base en cuadratura y las convierte al dominio digital con dos ADC de alta resolución y velocidad, haciendo uso del ADC1 del microcontrolador. Para adaptarse a las condiciones variables del canal de la señal, el chip integra filtros RF, control automático de ganancia (AGC), circuitos de cancelación de offset CC y filtros de banda base [25].


##### b) Transmisor 2.4 [GHz]

El transmisor de 2,4 GHz modula las señales de banda base en cuadratura a la señal de RF de 2,4 GHz y alimenta la antena con un amplificador de potencia de semiconductor de óxido metálico complementario (CMOS) de alta potencia. El uso de la calibración digital mejora aún más la linealidad del amplificador de potencia, lo que permite un rendimiento de vanguardia en la entrega de hasta +20,5 dBm de potencia para una transmisión 802.11b y +18 dBm para una transmisión 802.11n [25].

##### c) Radio y banda base

ESP32 implementa protocolo TCP/IP y full 802.11 b/g/n Wifi MAC. Soporta “basic service set (BSS)”, STA y operaciones softAP bajo funciones de control distribuido (DCF). ESP32 Wifi radio y banda base ofrece las siguientes características:

1. 802.11b/g/n
2. 802.11 MCS0-7 en ambos anchos de banda 20 [MHz] y 40 [MHz]
3. Velocidad de 150 Mbps
4. 20.5 dBm de potencia de transmisión
5. Potencia de transmisión ajustable


 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	8. Anexo

## 8.5. Flutter

Flutter es un framework de Dart, es orientado a Widgets, es altamente responsivo y permite crear aplicaciones multiplataforma con un único código. Orientado a widgets implica que sin importar el dispositivo donde se ejecute la aplicación, la interfaz de usuario no se verá afectada y no habrá problemas de alineación de elementos.

Durante el desarrollo, las aplicaciones de Flutter se ejecutan en una máquina virtual que ofrece una recarga activa de cambios con estado sin necesidad de una re-compilación completa. Para su lanzamiento, las aplicaciones de Flutter se compilan directamente en código máquina, ya sean instrucciones Intel x64 o ARM, o en JavaScript si se dirigen a la web. El marco es de código abierto, con una licencia BSD permisiva, y tiene un próspero ecosistema de paquetes de terceros que complementan la funcionalidad principal de la biblioteca [39].

Flutter está diseñado como un sistema extensible y en capas. Existe como una serie de bibliotecas independientes que dependen cada una de la capa subyacente. Ninguna capa tiene acceso privilegiado a la capa inferior, y cada parte del nivel del marco está diseñada para ser opcional y reemplazable [39].

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	8. Anexo

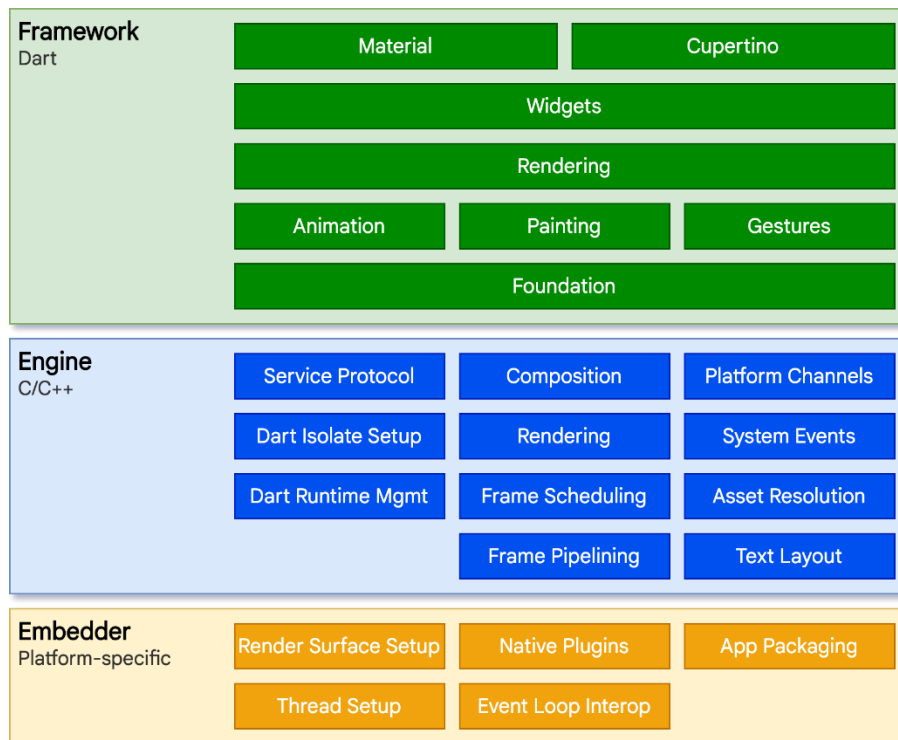


Fig. N°114 - Arquitectura de Flutter [39]

El desarrollo de la aplicación se realiza sobre la capa Framework, luego se traduce a lenguaje nativo dependiendo la plataforma, java y C++ para Android, Objective-C/Objective-C++ para iOS y macOS, y C++ para Windows y Linux.

### 8.5.1. Anatomía de una aplicación

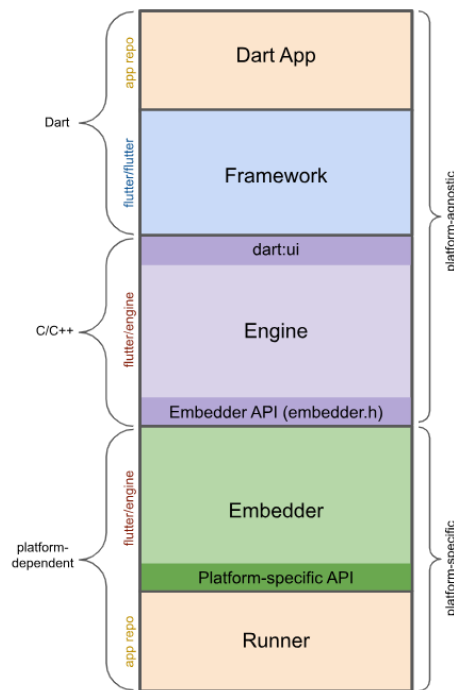



Fig. N°115 - Anatomía de una aplicación - Flutter [39]

La aplicación Dart compone los widgets en la interfaz de usuario deseada, implementa la lógica de la aplicación y es propiedad del desarrollador de la aplicación. El framework proporciona una API de nivel superior para crear aplicaciones de alta calidad y compone el árbol de widgets de la aplicación en una escena. El motor (engine) proporciona una implementación de bajo nivel de las API principales de Flutter y se integra con una plataforma específica utilizando la API embedder del motor. El embedder se encarga de coordinar con el sistema operativo subyacente para el acceso a servicios como superficies de representación, accesibilidad y entrada, administra el bucle de eventos y expone la API específica de la plataforma para integrar el embedder en las aplicaciones, las piezas de dicha API son compuestas por el Runner en un paquete de aplicación que se puede ejecutar en la plataforma de destino.

### 8.5.2. Desarrollo de la aplicación

#### 8.5.2.1. Página Inicio- Login

En la página de inicio de la aplicación se realizarán los preparativos necesarios para poder ejecutar el resto de la aplicación, lo primero es la conexión con firebase y a la base de datos en tiempo real, luego desarrollar un sistema de autenticación por usuarios con el fin de proporcionar seguridad a la aplicación.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	8. Anexo

Lo primero a ejecutarse en nuestra aplicación es la conexión a la base de datos, se requieren algunos parámetros desde nuestro proyecto en firebase, para ello debemos agregar nuestro proyecto Flutter a nuestro proyecto firebase y esto nos generará un archivo llamado “Google-services.json” que se encargará de configurar nuestro proyecto para la conexión a firebase. Una vez realizada esta tarea, desde nuestro proyecto firebase extraemos las variables necesarias e iniciamos la conexión, se muestra la siguiente figura para más detalles.

```

void main() async {
  WidgetsFlutterBinding.ensureInitialized();

  //for web
  if(kIsWeb){
    await Firebase.initializeApp(
      options: const FirebaseOptions(
        apiKey: "AIzaSyDtHvpkrqijGy5m5vRHeYZtNwfkveRsgxo",
        appId: "1:892165958267:web:4933db1de1a56d05e610de",
        messagingSenderId: "892165958267",
        projectId: "esp32-flutter-48c5b",
        storageBucket: "esp32-flutter-48c5b.appspot.com",
        databaseURL: "https://esp32-flutter-48c5b-default-rtdb.asia-southeast1.firebaseio.com",
      ));
  } else if(Platform.isAndroid){
    await Firebase.initializeApp(
      options: const FirebaseOptions(
        apiKey: "AIzaSyDtHvpkrqijGy5m5vRHeYZtNwfkveRsgxo",
        appId: "1:892165958267:android:4933db1de1a56d05e610de",
        messagingSenderId: "892165958267",
        projectId: "esp32-flutter-48c5b",
        storageBucket: "esp32-flutter-48c5b.appspot.com",
        databaseURL: "https://esp32-flutter-48c5b-default-rtdb.asia-southeast1.firebaseio.com",
      ));
  } else {
    await Firebase.initializeApp();
  }
  await FirebaseApi().initNotifications();
  runApp(app: const MyApp());
}


```

Fig. N°116 - Desarrollo de aplicación – Conexión a firebase[fuente propia]

Lograda la conexión a firebase y a la base de datos, se inicia nuestra aplicación donde se mostrará un botón de inicio de sesión con el nombre de nuestro proyecto. La estructura de la pantalla de inicio se va armando mediante un árbol de widgets, donde el primero es el denominado contenedor padre que dentro del mismo pondremos nuestra barra superior, un cuerpo y un pie de página.

En la barra superior pondremos el título “inicio de sesión” en conjunto con el logo de la universidad, el cuerpo será el nombre de nuestro proyecto acompañado del logo de nuestra universidad y un botón que ejecutará el inicio de sesión. Por último, el pie de página serán los nombres nuestros, estudiantes que realizaron el proyecto.

La acción del botón “iniciar sesión” conlleva la ejecución de una API de Google, la cual muestra una ventana emergente propia de Google para el inicio de sesión,

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	8. Anexo

se ponen los datos necesarios y luego se redirecciona para comparar los usuarios permitidos dentro de nuestro proyecto firebase. Para ello la Api hace un llamado a una función para la conexión a firebase. Se trasladan las credenciales del usuario a la base de datos y se continúa con la ejecución de la aplicación. En caso de inicio de sesión incorrecto, se regresa a la página de inicio.

```

void _handleFirebase() async{
  GoogleSignInAuthentication? googleAuth = await _currentUser?.authentication;
  final AuthCredential credential = GoogleAuthProvider.credential(
    | idToken: googleAuth?.idToken , accessToken: googleAuth?.accessToken
  );
  final UserCredential credUser = await firebaseAuth.signInWithCredential(credential: credential);
  if(credUser.user != null){
    | firebaseuser = credUser.user!;
    | setState(fn: () {
    | | Navigator.of(context).pushReplacementNamed<Object?, Object?>(routeName: '/home');
    | | // Navigator.of(context).pushNamedAndRemoveUntil('/home', (Route<dynamic> route) => false);
    | | });
  }
}

```

Fig. N°117 - Aplicación móvil - Validación con firebase[fuente propia]

En caso de haber iniciado sesión y luego cerrado la aplicación, es posible realizar un inicio de sesión automático debido a que la sesión del usuario permanece durante un tiempo y estos datos se mantienen, de manera que al ingresar nuevamente a la aplicación realiza un inicio de sesión automático.

#### 8.5.2.2. Página Home

Hasta el momento estamos autenticados con usuario de Google y conectados a la base de datos, el siguiente paso es solicitar la información que anteriormente fue subida. Para estas acciones nos trasladamos a la segunda página o screen la que llamaremos “home” o “principal”. Los datos de autenticación son trasladados a esta página junto con los datos de conexión a la base de datos, antes de iniciar el lanzamiento de la interfaz gráfica realizamos la solicitud de los datos en la base de datos primero solicitamos información de una sola escucha, es decir, solamente se extrae la información en la ruta solicitada una sola vez, luego extraemos información de las mediciones de los gases con escucha activa, esta vez la aplicación va estar atenta ante el cambio de algún valor en la base de datos y poder actualizar el dato en la interfaz gráfica logrando así el concepto de “monitoreo en tiempo real”.




```
class Dashboard extends StatefulWidget {  
  const Dashboard({super.key});  
  
  @override  
  State<Dashboard> createState() => _DashboardState();  
}  
  
class _DashboardState extends State<Dashboard>  
  with TickerProviderStateMixin {  
  bool isLoading= false;  
  double ratiomax =0;  
  StreamSubscription<DatabaseEvent>? _subscription;  
  final GoogleSignIn googleSignIn = GoogleSignIn();  
  final databaseReference = FirebaseDatabase.instance.ref();  
  late List<ExpansionItem> lista;  
  late List<bool> oldExpandState;  
  | | int data =0 ;  
  late Map<dynamic, dynamic> jsonData; // valores de sensores en tiempo real  
  late Map<dynamic, dynamic> jsonDataMax; // valores Maximos permisibles de sensores. Refiere a valores nocivos  
  @override  
  void initState(){  
    super.initState();  
    BackButtonInterceptor.add(myInterceptor, context: context);  
    _getMaxValues();  
    subscription = databaseReference  
      .child('Actual')  
      .onValue.listen((event) {
```

Fig. N°118 - Aplicación móvil- declaración de página Home[fuente propia]

En la declaración de nuestra página home se crea el estado de la página, se solicitan los datos de sesión y se inicia una instancia a la base de datos. El método “initState” da inicio al estado de la página, aquí es donde le diremos a nuestra página que debe estar atenta a los cambios en la base de datos e informarlos para actualizar la interfaz gráfica.

La organización de la interfaz gráfica fue pensada para que en la primera visualización el usuario pueda observar el estado general del sistema, los valores de temperatura y humedad. Luego desplazándose hacia abajo en la app pueda visualizar los valores de cada magnitud medida, organizándolos en módulos expandibles donde se muestra el valor en tiempo real, su valor crítico y el estado en el que se encuentra dicha magnitud (nivel bueno, malo, regular, crítico).

Cada dato recibido desde la base de datos se almacena en una clase la cual contendrá información específica de cada magnitud en particular como su estado, id, nombre de la magnitud, su valor corriente, su valor crítico, su unidad y su símbolo correspondiente), declarar una clase nos permite representar a cada magnitud como un objeto y mediante la misma función crear la interfaz gráfica de cada magnitud (nótese en la imagen los bloques de cada magnitud, son iguales). Para observar el estado general del sistema se creó un gráfico de rangos con una aguja que se moverá basado en la peor situación dentro de todas las magnitudes medidas.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO2 y Partículas
Ingeniería en Electrónica	8. Anexo

### 8.5.2.3. Menú desplegable

La navegación entre páginas está planeada mediante un menú desplegable en el cual el usuario puede navegar libremente entre las páginas creadas, tal y como lo mostramos al inicio de este capítulo. Dicho menú se replica en cada una de las páginas y sólo en páginas “información” y “acerca de” se presenta de manera diferente para dar una facilidad al usuario debido a la disposición de estas páginas.

Dicho menú se encuentra oculto y puede ser desplegado tocando en su icono a la izquierda de la pantalla o deslizando el dedo hacia la derecha. Se podrá navegar hacia “registros”, “información”, “home”, “acerca de” y tendrá un icono extra para finalizar la sesión y volver a la pantalla inicial de inicio de sesión. Se empleo un sistema de navegación por rutas y control de botón “atrás” para que la aplicación “recuerde” en que página se encontraba anteriormente.

```

routes: {
  '/':(context)=> const LoginScreen(title:'Inicio de Sesion'),
  '/home':(context) => const Dashboard(),
  '/registro':(context)=> const DataControl(),
  '/informacion':(context)=> const InformationPage(),
  '/acercaDe':(context)=> const AcercaDe(),
},
); // MaterialApp
}


```

Fig. N°119 - Aplicación móvil – Rutas a las páginas[fuente propia]

### 8.5.2.4. Página Registro y Gráficas

Si bien es conveniente poder monitorear en tiempo real los gases y partículas presentes en el ambiente sería oportuno poder almacenar las magnitudes en lapsos de 30 minutos de modo que sea posible consultar los valores de concentración que estuvieron presentes en un día específico a un horario específico y también poder ver dichos valores expresados en una gráfica diaria que permite visualizar si hubo picos de concentración o no hubo problemas durante el día. Para ello, a través del microcontrolador cada 30 almacenamos en la base de datos la concentración de los gases y partículas de manera que luego en nuestra aplicación podamos mostrar la lista de valores de concentración ordenados por fecha y hora.

Se creo entonces, una página que permita ver estos datos en forma de lista y en consecuencia otra página que nos muestra las gráficas de concentración del día solicitado. Se planteo contar con 120 registros en total, cada registro se realiza cada 30 minutos lo que nos resulta en la posibilidad de almacenar de manera corrida hasta dos días y media atrás, luego los datos se remplazan desde el más antiguo con nuevos valores.

 <b>UTN Regional Paraná</b>	Monitoreo y presentación digital de calidad de aire mediante detección de CO, HCHO, CO <sub>2</sub> y Partículas
Ingeniería en Electrónica	8. Anexo

La página de gráficas dependerá pura y exclusivamente de la página registro, con esto queremos decir que la única forma de navegar hasta ella es por la página registro ya que está a cargo de proporcionarle la información para la creación de las gráficas. Solo tiene la tarea de mostrar los datos de esta manera, por lo que es una página sin estado.

La página registros tendrá estado, de manera que sea posible mantener la sesión del usuario junto con la conexión a la base de datos, sin embargo, la información de los registros se descarga al momento de su creación y luego lanzará la interfaz gráfica, si se agrega un dato al registro no se va a ver reflejado en ella hasta que se recargue la página o se abandone y vuelva a entrar. Se crea una clase llamada "expansionRegistro" que contiene la información sobre cada registro al igual como se hizo en la página "home" con el agregado de que se agregan dos argumentos más que son la fecha y la hora en que se tomaron esas medidas y luego se arma una lista de esta clase. En resumen, tenemos una lista desde la base de datos y se traslada a otra lista de la clase nombrada con un tratamiento de datos de por medio. Además de esto se crea otra lista de las diferentes fechas que existan dentro de la lista de registros para luego poder ir armando diferentes sublistas por cada fecha. Habiendo realizado el tratamiento de la información es posible lanzar la interfaz gráfica y mostrarlos.

## 8.6. Pantalla touch

Para visualizar los datos, en este proyecto, se decidió usar pantallas con tecnologías Transistor de Película Fina (TFT), que son usadas para distintos dispositivos. En el mercado hay muchos tamaños, como 2,4", 2,8", 3,2" y 3,5" [28]. Se optó por la pantalla TFT ILI9486 de 3,5" de 420x380 pixeles. Esta pantalla cuenta con una Interfaz Periférica Serial (SPI), que se usa para la comunicación entre la pantalla y el microcontrolador [27].

Si bien esta pantalla goza de la comunicación SPI, esta no es usada en el proyecto, ya que esta se encarga de la memoria SD, la cual no se utiliza.

Para programar la pantalla se usa la librería MCUFRIEND\_kbv que para ser usada solo necesita la creación de un objeto TFT para comenzar a programarla [27]. A continuación, se detallan otras librerías. necesarias para trabajar con esta pantalla [28].

- TouchScreen.h: Para utilizar la pantalla táctil.
- bitmap\_mono.h: Para mostrar una imagen de mapa de bits de la biblioteca.
- bitmap\_RGB.h: Para mostrar una imagen de mapa de bits de la biblioteca.
- SPI.h y SD.h: Para usar la tarjeta SD.



Para trabajar con esta pantalla primero se la debe calibrar y la librería MCUFRIEND\_kbv tiene un programa para esto. Una vez cargada la librería, el diseñador tiene que abrir y cargar el sketch de Ejemplo -> MCUFRIEND\_kbv -> TouchScreen\_Calibr\_native y una vez cargado el ejemplo en el microcontrolador, se debe seguir los pasos que se indica en la pantalla y copiar los parámetros de calibración dados [40]. Estos parámetros se cargan en un código que use la pantalla touch para poder usar la pantalla correctamente. En la parte resaltada, del siguiente código, se muestran los parámetros de calibración usados en un ejemplo como también el bloque de código que se necesita al inicio para usar la pantalla táctil.

```
13 #define MINPRESSURE 700//presion minima que va a tomar como toque en la pantalla
14 #define MAXPRESSURE 1000//presion maxima que va a tomar como toque en la pantalla
15 int pixel_x, pixel_y;// variables de las cordendas calculadas de la presion
16 // Pinout para la pantalla táctil
17 #define TS_CS 8
18 #define TS_IRQ 7
19 const int XP=8, XM=A2, YP=A3, YM=9; //320x480 ID=0x9486
20 //parametros de calibracion
21 const int TS_LEFT=187, TS_RT=1007, TS_TOP=937, TS_BOT=195;
22 //objeto utilizado para obtener los datos X, Y y presion de un toque de la pantalla con la funcion getPoint()
23 TouchScreen ts = TouchScreen(XP, YP, XM, YM, 300);
```

Fig. N°120 - Código Arduino – Parámetros de calibración [fuente propia]

Para comenzar a programar esta pantalla se tiene que crear un objeto de la clase MCUFRIEND\_kbv para poder crear una comunicación entre la pantalla y el Arduino.

- MCUFRIEND\_kbv tft;

Luego es necesario leer su ID y suministrarlo al método begin() en setup().

- uint16\_t ID = tft.readID().
- tft.begin(ID);

A continuación, se muestran algunas funciones para dibujar patrones o hacer determinadas acciones en la pantalla luego de haber hecho todos los pasos necesarios para inicializar el código [27].

- tft.drawPixel(x, y, t): Dibuja un píxel en la posición (x, y) con el color t.
- tft.readPixel(x, y): Lee el color de un píxel en la posición (x, y) y devuelve el color en formato uint16\_t.
- tft.drawFastVLine(x, y, h, t): Dibuja una línea vertical en la posición (x, y), con una longitud de h píxeles y el color t.
- tft.drawFastHLine(x, y, w, t): Dibuja una línea horizontal en la posición (x, y), con una longitud de w píxeles y el color t.
- tft.drawLine(xi, yi, xj, yj, t): Dibuja una línea desde (xi, yi) hasta (xj, yj) con el color t.
- ft.drawTriangle(x1, y1, x2, y2, x3, y3, t): Dibuja un triángulo con vértices en (x1, y1), (x2, y2), y (x3, y3), y con el color t.



- tft.fillTriangle(x1, y1, x2, y2, x3, y3, t): Dibuja un triángulo relleno con vértices en (x1, y1), (x2, y2), y (x3, y3), y con el color t.
- tft.invertDisplay(i): Invierte los colores de la pantalla. i es un valor booleano donde true invierte los colores y false la muestra de manera normal.
- tft.reset(): Reinicia la pantalla y borra su contenido

### 8.7. Estructura de un archivo JSON

Los archivos JSON se ordenan en pares clave/valor. Esto permite poder trabajar fácilmente con los datos tanto como en la generación como en la lectura [41]. En la siguiente figura, se indica el archivo estructurado, en Arduino, que se manda por el puerto serial, como se nota primero va la clave y se la iguala al valor. Por ejemplo "json["CO"]=random(0, 5);" la clave es CO y el valor el número que genera la función Random(0,5).

```
20  
21 json["Operacion"]="m";  
22 json["CO"]=random(0, 5);  
23 json["CO2"] = random(5, 15);  
24 json["t"] = random(2, 3);  
25 json["%H"] = random(2, 3);  
26 json["PM 2.5"] = random(1, 1);  
27 json["PM 10"] = random(1, 5);  
28 json["PM 1"] = random(1,2);  
29 json["HCHO"] = random(0,3);  
30 json["Coneccion"] =true;  
31
```

Fig. N°121 - Código Arduino – Estructura clave/valor [fuente propia]

Además, este tipo de archivos permite organizar los datos en estructuras más complejas, para transmitir información con grado de complejidad tanto como el desarrollador necesite [32]. Este formato de archivo es ligero y fácil de interpretar y generar para las computadoras. Aunque está basado en un subconjunto del lenguaje de programación JavaScript, JSON es independiente del lenguaje y utiliza convenciones conocidas por programadores de otros lenguajes como C, C++, C#, Java y Python, entre otros [28].