



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Escuela Técnica Superior de Ingeniería del Diseño

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería del Diseño

DISEÑO DE UN SISTEMA ELECTRÓNICO PARA LA MEDIDA DE LA CALIDAD DEL AIRE EN UNA OFICINA

TRABAJO FINAL DEL

Grado en Ingeniería Electrónica Industrial y Automática

REALIZADO POR

Antonio Navarro Núñez

TUTORIZADO POR

Enrique Berjano Zanón

Juan José Martínez Pérez

CURSO ACADÉMICO: 2020/2021

Contenidos

Contenidos	1
Índice de figuras	3
Índice de tablas	5
Motivaciones.....	6
Resumen.....	7
Resum.....	7
Abstract	7
MEMORIA.....	8
1. Objetivos	8
2. Herramientas.....	9
2.1. Herramientas humanas.....	9
2.2. Herramientas Hardware.....	9
4.1. Herramientas Software	10
3. Antecedentes	11
3.1. Microcontroladores.....	11
3.1.1. Raspberry Pi.....	12
3.1.2. Tipos de placa de Arduino	13
3.1.2.1. Arduino UNO R3	13
3.1.2.2. Arduino Nano	14
3.1.2.3. Arduino Mega.....	15
3.2. Comparación placas Arduino	16
3.3. Sensores que miden la calidad del aire.....	17
3.2.1. Sensor de Polvo PPD42NS PM2.5.....	17
3.2.2. Sensor de CO2 NDIR MH-Z19C HVAC 5000ppm.....	18
3.2.3. Sensor de CO2 NDIR MH-Z14A Detección de dióxido de Carbono	19
3.2.4. Nova PM Sensor SDS011 – Sensor de calidad del aire de alta precisión láser PM2.5 20	
3.2.5. Air Monitoring HAT para Raspberry Pi – HAT Monitoreo de aire	21
3.2.6. Winson – ZPHS01B MULTI-IN-ONE SENSOR MODULE	22
3.2.7. Comparativa de sensores	23
3.2.8. Sensores de gases de la familia MQ.....	24
4. Descripción de la solución adoptada	26

4.1.	Diagrama de bloques del sistema	26
4.2.	Arduino.....	27
4.2.1.	Microcontrolador Arduino UNO.....	28
4.3.	Sensor MQ135.....	29
4.4.	Interfaz de usuario	31
4.5.	Montaje real.....	33
4.6.	Fuente de alimentación	34
4.7.	Diseño de la fuente de alimentación	36
4.8.	Programa.....	44
5.	Conclusiones.....	50
6.	Referencias.....	51
7.	Anexos.....	52
	Planos	56
1.	Circuito de alimentación y conexiones del sistema	56
-	Circuito de alimentación	56
-	Conexiones del sistema.....	57
	Pliego de Condiciones	60
1.	Objeto.....	60
2.	Herramientas.....	60
2.1.	Ordenador Portátil MSI GE63 Raider RGB 8RF:.....	60
2.2.	Placa Protoboard.....	60
2.3.	Fuente de alimentación	60
2.4.	Placa Arduino UNO.....	60
2.5.	Interfaz de usuario	61
	Presupuesto	63

Índice de figuras

Figura 1. Raspberry Pi.....	12
Figura 2. Arduino UNO R3	13
Figura 3. Arduino Nano	14
Figura 4. Arduino Mega.....	15
Figura 5. Sensor de Polvo PPD42NS PM2.5.....	17
Figura 6. Sensor de CO2 NDIR MH-Z19C HVAC 5000ppm.....	18
Figura 7. Sensor de CO2 NDIR MH-Z14A Detección de dióxido de Carbono	19
Figura 8. Nova PM Sensor SDS011 – Sensor de calidad del aire de alta precisión láser PM2.5 .	20
Figura 9. Air Monitoring HAT para Raspberry Pi – HAT Monitoreo de aire	21
Figura 10. Winson – ZPHS01B MULTI-IN-ONE SENSOR MODULE	22
Figura 11. Familia de sensores MQ	25
Figura 12. Diagrama de bloques del sistema	26
Figura 13. Sensor MQ135.....	30
Figura 14. Pantalla LCD GeeePi IIC/I2C 2004 20x4.....	31
Figura 15. Conexión de los diodos Led a la placa de Arduino	32
Figura 16. Sistema de medición de calidad de aire detectando aire limpio	33
Figura 17. Sistema de medición de calidad de aire detectando gases peligrosos	33
Figura 18. Señal rectificada completamente	34
Figura 19. Diagrama de pines del regulador de tensión 7805	35
Figura 20. Diagrama de la fuente de alimentación del sistema.....	36
Figura 21. Tensión de rizado	37
Figura 22. Datasheet familia de diodos 1N4000	38
Figura 23. Voltaje necesario para calcular la potencia disipada del regulador de tensión.....	39
Figura 24. Datasheet del regulador LM7805.....	40
Figura 25. Esquema a tener en cuenta para calcular el disipador de calor	41
Figura 26. Elección de los datos en el datasheet del regulador LM7805.....	42
Figura 27. Disipador de calor elegido.....	43
Figura 28. Instalación de drivers de Arduino IDE	52
Figura 29. Ventana principal de Arduino IDE	53
Figura 30. Escoger el puerto de la placa en Arduino IDE	53
Figura 31. Pestaña con programas ejemplo en Arduino IDE.....	54

Figura 32. Interfaz Arduino IDE	54
Figura 33. Diseño fuente de alimentación	56
Figura 34. Conexión del sistema	57

Índice de tablas

Tabla 1. Comparación de las placas Arduino	16
Tabla 2. Comparación de los sensores	23
Tabla 3. Sensores de la familia MQ	25
Tabla 4. Presupuesto del personal	63
Tabla 5. Presupuesto herramientas hardware	63
Tabla 6. Presupuesto herramientas software	64
Tabla 7. Suma total de los presupuestos	64
Tabla 8. Presupuesto total del proyecto	64

Motivaciones

Este año ha sido un año duró para todos ya que hemos tenido que convivir con un virus que nos ha hecho cambiar totalmente nuestras formas de vida ya que la totalidad de la población mundial ha pasado mucho tiempo en sus casas y se ha tenido que alejar de sus seres queridos por un largo tiempo.

Por ello, todo este tiempo nos ha servido para darnos cuenta de lo vulnerables que podemos ser con tan solo el hecho de respirar algo que sea perjudicial para nuestra salud. Podemos encontrar multitud de trabajos en nuestra sociedad en lo que es muy importante la calidad del aire que respiran sus trabajadores. Por ejemplo, es habitual conocer fatídicas noticias sobre el fallecimiento de operarios a consecuencia de fugas de gases tóxicos. Todos hemos escuchado accidentes laborales de este tipo en el sector de la minería, de bodegas vitícolas o los relacionados con trabajos portuarios.

Todas estas razones han servido como motivación para la realización de este trabajo y lo importante que puede llegar a ser para todos nosotros el simple hecho de respirar aire en unas buenas condiciones.

Resumen

Este proyecto abarcará el diseño de un sistema electrónico para medir y visualizar la calidad del aire en una pantalla LCD gracias a un sensor y su debida programación en un microcontrolador del tipo Arduino UNO.

Para ello se ha de diseñar y calcular para su correcto funcionamiento el sistema de alimentación de todos los componentes que forman el proyecto. Además, el sistema será montado físicamente y testeado para comprobar su correcto funcionamiento.

Resum

Aquest projecte abastarà el disseny d'un sistema electrònic per mesurar i visualitzar la qualitat de l'aire en una pantalla LCD gràcies a un sensor i la seva deguda programació en un microcontrolador de l'tipus Arduino UN.

Per a això s'ha de dissenyar i calcular per al seu correcte funcionament el sistema d'alimentació de tots els components que formen el projecte. A més, el sistema serà muntat físicament i provat per a comprovar el seu correcte funcionament.

Abstract

This project will cover the design of an electronic system to measure and display air quality on an LCD screen thanks to a sensor and its proper programming in an Arduino UNO-type microcontroller.

For this, the power system of all the components that make up the project must be designed and calculated for its correct operation. In addition, the system will be physically assembled and tested to verify its correct operation.

MEMORIA

1. Objetivos

El objetivo de este proyecto es el diseño y montaje de un sistema electrónico para medir la calidad de aire en oficinas. Para conseguir tal fin, el proyecto está constituido por un microcontrolador Arduino UNO, un sensor para medir la calidad del aire del tipo MQ135 su fabricante es Az-Delivery (Bräugasse, 94469 Deggendorf, Alemania), un sistema de alimentación de 5 voltios, una pantalla LCD 20x4 para visualizar las medidas obtenidas del sensor y una serie de resistencias y leds para saber sobre que rango de calidad de aire se encuentra el ambiente según el color de dichos leds.

La aplicación principal de este proyecto es supervisar el estado en el que se encuentra en aire de una oficina en la que pueda concurrir gente en todo momento y que el aire tenga que estar debidamente controlado y limpio. Cabe mencionar que el sistema también detecta gases peligrosos que podrían causar un incendio como el butano o propano.

A parte de medir la calidad del aire, se puede conectar un purificador de aire que ayude a eliminar los gases tóxicos o nocivos para la salud del ser humano. Por ello, se ha facilitado una salida del microcontrolador.

Tras un breve resumen de las principales características de las que consta la realización del sistema, es importante señalar más detenidamente todos los objetivos que se han marcado para su diseño y su correspondiente verificación práctica:

- A) Diseño y montaje de todo el sistema electrónico.
- B) Controlar los gases nocivos de una oficina.
- C) Prevención ante incendios.
- D) Tener debidamente ventilado el aire de un espacio interior.

2. Herramientas

2.1. Herramientas humanas

Para la correcta realización de todo este proyecto han cooperado dos personas. Por un lado, Antonio Navarro Núñez, como diseñador y desarrollador del proyecto y Enrique Berjano Zanón que ha tenido el papel de ayudante y tutor del proyecto.

2.2. Herramientas Hardware

En este apartado se mostrarán las herramientas *hardware* utilizadas para el proyecto con todas sus características más importantes:

1. ORDENADOR Portátil MSI GE63 Raider RGB 8RF:

- Procesador Intel® Core i7-8750H (2.2 GHz, 9MB).
- Memoria RAM 16GB DDR4.
- Disco duro 1TB (7200 rpm SATA) + 512 GB SSD.
- Controlador gráfico GeForce® GTX 1070 8GB GDDR5.
- Display 15.6" Full HD (1920*1080) 16:9 120Hz Anti-Glare.
- Sistema Operativo: Windows 10, procesador 64 bits.

2. MICROCONTROLADOR ARDUINO UNO:

- Velocidad de reloj: 16 MHz.
- Voltaje de trabajo: 5V.
- Voltaje de entrada: 7,5 a 12 voltios.
- Pinout: 14 pines digitales (6 PWM) y 6 pines analógicos.
- 1 puerto serie por hardware.
- Memoria: 32 KB Flash (0,5 para bootloader), 2KB RAM y 1KB Eeprom.

3. PANTALLA LCD GEEPI IIC/I2C 2004:

- Pantalla de 20x4.
- Compatible con Arduino UNO y Raspberry Pi

4. SENSOR DE GAS MQ135:

- Chip principal: sensor de gas natural MQ-135 y comparador LM393
- Voltaje de trabajo: 5 V
- Corriente :150mA
- Salida digital: TTL 0/1
- Salida analógica: valor analógico de detecta tensión
- Sensibilidad ajustable con el potenciómetro
- Tamaño: 32mm x 20mmx 18mm

4.1. Herramientas Software

En relación a las herramientas, es importante señalar la utilización de los siguientes *softwares*:

1. Arduino IDE:

Es un programa informático compuesto por un conjunto de herramientas de programación. Puede dedicarse en exclusiva a un solo lenguaje de programación o bien puede utilizarse para varios.

Un IDE es un entorno de programación que ha sido empaquetado como un programa de aplicación, es decir, que consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI). El IDE de Arduino va a ser la herramienta de trabajo con Arduino y habrá que conocer su funcionamiento.

Los programas de Arduino están compuestos por un solo fichero con extensión "ino", aunque es posible organizarlo en varios ficheros.

2. Proteus 8 Profesional:

Proteus software también conocido como Proteus Design Suite se trata de uno de los mejores softwares disponibles para la simulación de distintos tipos de diseños relacionados con microcontroladores.

Su popularidad se debe principalmente a que cuenta con casi todos los microcontroladores del mercado, razón por la que se trata de una herramienta muy útil para los aficionados de la electrónica a través de la que pueden probar diseños y programas integrados. Debido a estas razones es uno de los programas para ingenieros más utilizados, especialmente por los electrónicos.

Incluye una captura esquemática, simulación y módulos de diseño de PCB (placa de circuito impreso), por lo que después de simular su circuito en el software Proteus, puede realizar directamente un diseño de PCB.

3. Office 365:

Office 365 es la herramienta creada por el gigante Microsoft que nos permite crear, acceder y compartir documentos online entre distintos usuarios en Word, Excel, PowerPoint y OneNote, entre otros. Para ello, solo necesitas tener acceso a internet y disponer del programa OneDrive.

Office ofrece diferentes paquetes en función del tamaño de la empresa o el número de usuarios que vayan a utilizar estos servicios. Además, los usuarios pueden escoger entre tres tipos de paquetes: para el uso privado, para empresas o para estudiantes o profesores. Cada uno tiene características propias (número de usuarios, programas integrados, espacio...) y, correspondientemente.

En este proyecto se han utilizado los programas Word y PowerPoint. El primer software se ha utilizado para la redacción y explicación de este proyecto. Mientras que el segundo se ha utilizado para realizar la presentación de dicho proyecto a través de diapositivas y defenderlo ante un tribunal.

3. Antecedentes

En este momento del proyecto es importante presentar un estudio sobre el estado del arte en referencia a microcontroladores y sensores comercialmente utilizados para medir la calidad del aire en espacios interiores. Para ello, se presentará una breve descripción de los diferentes sistemas similares a los que se han utilizado en este proyecto.

3.1. Microcontroladores

A continuación, se expondrá un pequeño estudio de mercado sobre microcontroladores que cumplen características necesarias para este proyecto y que se han tenido en cuenta para realizarlo.

Esta búsqueda se ha centrado principalmente en dos marcas de microcontroladores claramente diferenciadas, por un lado, la marca Raspberry y por otro lado la marca de microcontroladores Arduino.

3.1.1. Raspberry Pi

Raspberry Pi se califica como un ordenador del tamaño de una tarjeta de crédito ya que soporta varios componentes de un ordenador común. Raspberry Pi puede ser utilizada para muchas cosas de las que el ordenador de escritorio es capaz como para realizar hojas de cálculo, procesadores de texto, también puede reproducir videos de alta definición y para programar pequeños minijuegos.

La placa posee varios puertos USB, uno de Ethernet y una salida de HDMI, estos puertos permiten conectar el miniordenador a otros dispositivos como teclados ratones o pantallas. También posee un System on Chip que posee un procesador ARM con una velocidad de 700 MHz, un procesador gráfico VideoCore IV y hasta 513 MB de memoria RAM.

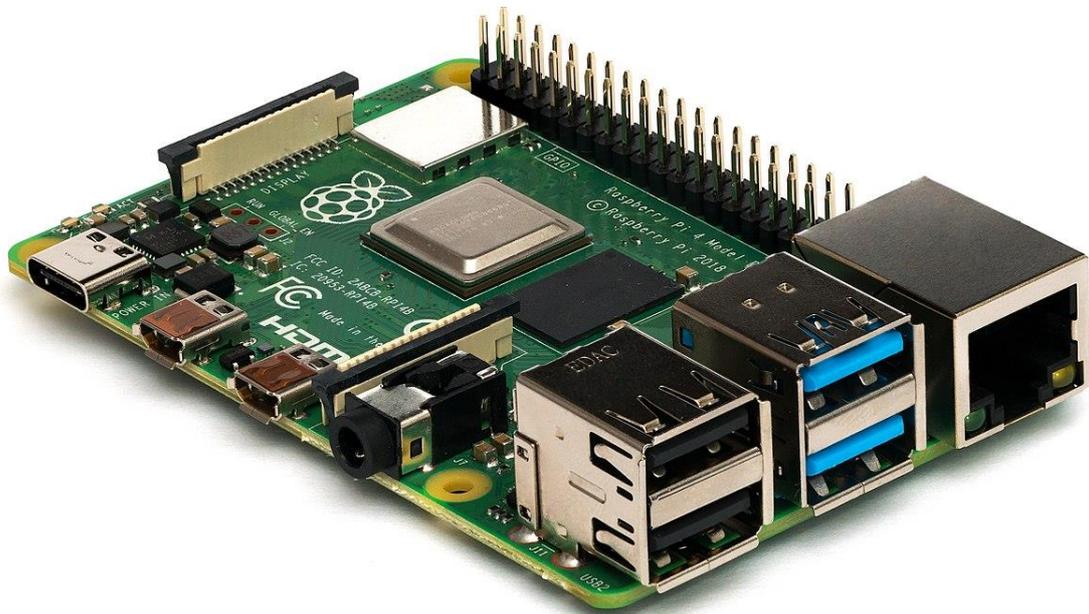


Figura 1. Raspberry Pi

3.1.2. Tipos de placa de Arduino

3.1.2.1. Arduino UNO R3

Esta es la primera placa de microcontrolador de Arduino que salió al mercado. Se trata de una placa de código abierto basado en el microchip ATmega328P. La placa está equipada con conjuntos de pines de entradas y salidas digitales y analógicas que pueden conectarse a varias placas de expansión y otros circuitos.

La placa tiene 14 pines digitales de los cuales 6 de esos pines soportan PWM, 6 pines analógicos y programables con el Arduino IDE (entorno de desarrollo integrado) a través de un cable USB, un oscilador de cristal de 16 MHz, un conector ICSP y un botón de reset. El microcontrolador puede ser alimentado a través del cable USB o por una batería externa.

Sus principales características son:

- Microcontrolador: ATmega328P.
- Velocidad de reloj: 16 MHz.
- Voltaje de trabajo: 5V (DC).
- Voltaje de entrada: 7.5 a 12 voltios (DC).
- Pines: 14 pines digitales (6 PWM) y 6 pines analógicos.
- 1 puerto serie por hardware.
- Memoria: 32 KB Flash (0.5 para bootloader), 2KB RAM y 1KB Eeprom

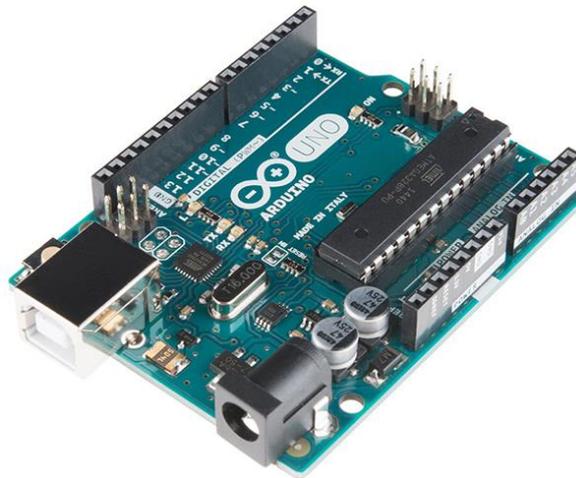


Figura 2. Arduino UNO R3

3.1.2.2. Arduino Nano

Esta placa es de tamaño reducido y de bajo coste que posee prácticamente las mismas características que la placa Arduino UNO, aunque proporciona dos entradas analógicas adicionales. Los pines de esta placa son de tipo macho, diferentes de la placa de Arduino UNO ya que estos son de tipo hembra, lo que le permite un rápido montaje sobre una placa protoboard con el objetivo de reducir los tamaños en los montajes. Otra diferencia es que esta placa se alimenta a través de un cable mini USB que se trata de un acople más pequeño que el de la placa Arduino UNO. Además, esta placa no dispone de conector de alimentación externa.

Sus principales características son:

- Microcontrolador: ATmega328P.
- Velocidad de reloj: 16 MHz.
- Voltaje de trabajo: 5V (DC).
- Voltaje de entrada: 7.5 a 12 voltios (DC).
- Pines: 14 pines digitales (6 PWM) y 8 pines analógicos.
- 1 puerto serie por hardware.
- Memoria: 32 KB Flash (2KB para bootloader), 2KB RAM y 1KB Eeprom

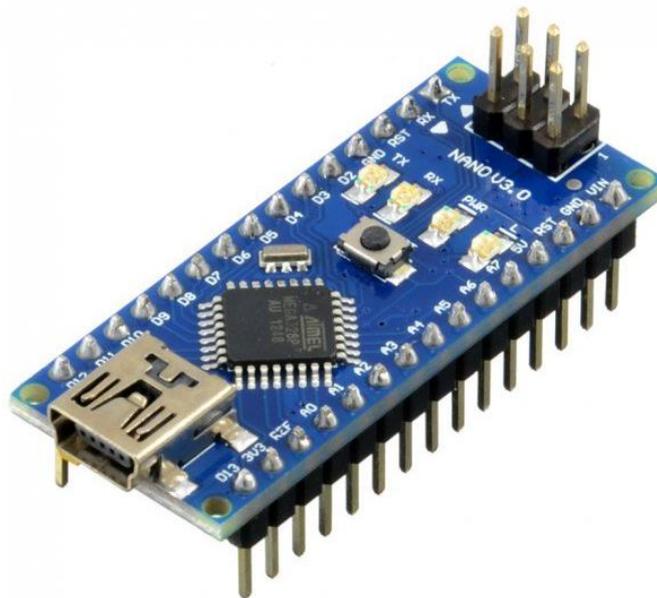


Figura 3. Arduino Nano

3.1.2.3. Arduino Mega

La placa Arduino Mega es prácticamente una versión ampliada de la tarjeta original de Arduino y está basada en el microcontrolador ATmega2560. Es una placa diseñada para proyectos que tengan más complejidad y se necesiten un mayor número de pines para su realización ya que dispone de más entradas y salidas que las otras placas mencionadas anteriormente, además, también dispone de un mayor espacio de memoria. Esta placa ofrece mayor número de posibilidades ya que se pueden implementar con ella hasta proyectos de robótica o incluso proyectos con impresoras 3D.

Sus principales características son:

- Microcontrolador: ATmega2560.
- Velocidad de reloj: 16 MHz.
- Voltaje de trabajo: 5V (DC).
- Voltaje de entrada: 7.5 a 12 voltios (DC).
- Pines: 54 pines digitales (15 PWM) y 16 pines analógicos.
- 3 puertos serie por hardware.
- Memoria: 256 KB Flash (8KB para bootloader), 8KB RAM y 4KB Eeprom.

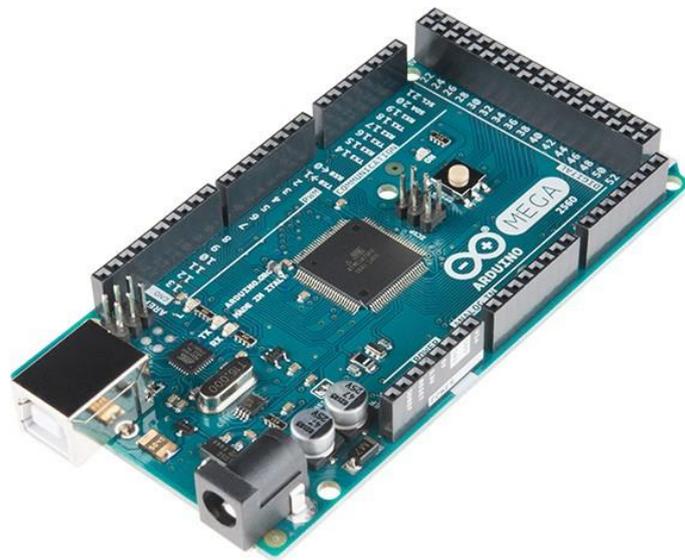


Figura 4. Arduino Mega

3.2. Comparación placas Arduino

En esta tabla se adjuntan las principales propiedades de cada placa de Arduino mencionada anteriormente que son tipo de microcontrolador, velocidad de reloj, voltaje de trabajo, número y tipo de pines, puerto serie y su capacidad de memoria para comparar las placas con mayor facilidad.

Tabla 1. Comparación de las placas Arduino

	Arduino UNO	Arduino NANO	Arduino MEGA
Tipo de microcontrolador	ATMega328P	ATMega328P	ATMega2560
Velocidad de reloj	16 MHz	16 MHz	16 MHz
Voltaje de trabajo	5 V (DC)	5 V (DC)	5 V (DC)
Tipo y número de pines	14 digitales de los cuales 6 PWM y 6 analógicos	14 digitales de los cuales 6 PWM y 8 analógicos	54 digitales de los cuales 15 PWM y 16 analógicos
Puerto Serie	1 puerto serie por hardware	1 puerto serie por hardware	3 puerto serie por hardware
Memoria	32 KB Flash (2 KB para bootloader), 2KB RAM y 1KB Eeprom	32 KB Flash (2 KB para bootloader), 2KB RAM y 1KB Eeprom	256 KB Flash (8 KB para bootloader), 8KB RAM y 4KB Eeprom

Como se puede observar en la Tabla 1, las mejores prestaciones se encuentran en la placa Arduino MEGA ya que es la que más puertos serie y más memoria tiene, además también goza de otro microcontrolador mientras que las otras dos placas tienen prácticamente las mismas características.

3.3. Sensores que miden la calidad del aire

Aquí se añadirá una lista de las diferentes alternativas de mercado que se tuvieron sobre sensores que miden la calidad del aire con sus diferentes características. Al final se adjuntarán una serie de tablas para poder comparar todos los sensores con sus principales propiedades.

3.2.1. Sensor de Polvo PPD42NS PM2.5

Este es un sensor de polvo capaz de detectar micras de partículas de polvo. El sensor tiene dos tipos de salidas, una salida de señal a nivel alto de 4 V que se trataría de un ambiente limpio y una salida a nivel bajo de 0.7 V que se trataría de un ambiente lleno de partículas de polvo. Además, su rango de detección de partículas tiene un máximo de 8000 unidades por 283 ml.

Sus características son las siguientes:

- Tensión de alimentación: 5V (DC)
- Corriente de trabajo: 90 mA
- Sensibilidad: 0.5V/ (0.1 mg/m³)
- Detección de partículas valor mínimo: 1 micras
- Modo de salida: impulsos Hi=4V (ambiente limpio); Lo = 0.7 V (ambiente cargado)
- Temperatura de trabajo: 0-45°C
- Humedad de trabajo: 0-90% RH
- Tamaño de PCB: 59x45x22 mm

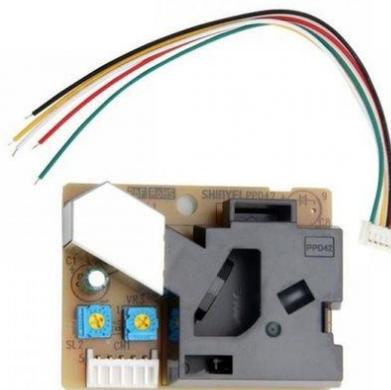


Figura 5. Sensor de Polvo PPD42NS PM2.5

3.2.2. Sensor de CO2 NDIR MH-Z19C HVAC 5000ppm

Se trata de un módulo pequeño que utiliza el principio infrarrojo no dispersivo (NDIR) para detectar la existencia de CO2 en el aire, con una buena selectividad y sin depender del oxígeno que se encuentra en el aire. Su voltaje de trabajo es de alta precisión, además su rango de medición está entre 0 y 5000 ppm. El sensor tiene un bajo consumo energético y consta de salidas UART y PWM.

Puede aplicarse para un sistema de ventilación, un dispositivo que se encargue de la limpieza del aire, para monitorear la calidad del aire en un interior y hasta para una casa inteligente.

Sus características son las siguientes:

- Voltaje de trabajo: $5.0 \pm 0.1V$ (DC) (Margen estrecho a tener en cuenta)
- Corriente promedio < 40 mA (5V fuente de alimentación)
- Corriente máxima: 125 mA (5V fuente de alimentación)
- Nivel de interfaz: 3.3 V (compatible con 5V)
- Rango de detección: 0 a 5000 ppm (opcional) $\pm (50\text{ppm} + 5\% \text{ valor de lectura})$
- Señal de salida: Puerto serie (UART) (TTL nivel 3.3V) / PWM
- Tiempo de precalentamiento: 1 min
- Tiempo de respuesta: $T_{90} < 120$ s
- Temperatura de trabajo: -10°C a 50°
- Humedad de trabajo: 0 a 95% RH (Sin condensación)
- Peso: 5 gramos
- Vida útil > 5 años

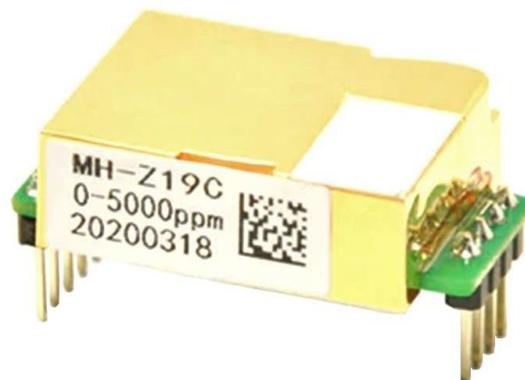


Figura 6. Sensor de CO2 NDIR MH-Z19C HVAC 5000ppm

3.2.4. Nova PM Sensor SDS011 – Sensor de calidad del aire de alta precisión láser PM2.5

Este es un sensor de alta precisión que utiliza la dispersión láser para detectar concentraciones de partículas entre 0,3 y 10 μm en el aire. Se puede inducir la dispersión de la luz cuando las partículas atraviesan el área de detección. La luz dispersa se transforma en señales eléctricas y estas señales serán amplificadas y procesadas. El número y el diámetro de las partículas se pueden obtener mediante un análisis ya que la forma de la onda de la señal del sensor tiene ciertas relaciones con el diámetro de dichas partículas.

Posee salida digital por USB (UART) y un ventilador incorporado. Es un sensor potente y rápido con una respuesta inferior a 10 segundos.

Sus características son las siguientes:

- Salida de medición PM2.5, PM10
- Rango entre 0,0-999,9 $\mu\text{g} / \text{m}^3$
- Voltaje de la fuente de alimentación: 5V (DC)
- Corriente máxima de trabajo: 100 mA
- Corriente de sueño: 2 mA
- Rango de temperatura de funcionamiento entre -20-50 $^{\circ}\text{C}$
- Tiempo de respuesta: 1 segundo
- Frecuencia de salida de datos en serie 1 time/s
- Resolución de diámetro de partículas de menos de 0.3 μm
- Error relativo: 10%
- Tamaño del producto 71x70x23mm



Figura 8. Nova PM Sensor SDS011 – Sensor de calidad del aire de alta precisión láser PM2.5

3.2.5. Air Monitoring HAT para Raspberry Pi – HAT Monitoreo de aire

Este HAT tiene un sensor portátil de código abierto que proporciona el nivel de cantidad de partículas suspendidas con su masa. Air Monitoring HAT dispone de un potente sensor PMSA003 de alta tecnología que proporciona la información de la materia en suspensión en el aire por unidad de volumen, esto lo hace a través de una salida digital con un voltaje que opera a 3.3 V. Esta placa se comunica a través de UART y es solamente compatible con Raspberry Pi. Cabe mencionar que también dispone de una pantalla OLED de 0.91 pulgadas y su tensión de funcionamiento es de 5 voltios DC.

Sus principales características son las siguientes:

Salida de datos	Sensor Digital
Interfaz de comunicación	UART (Serial)
Tipo display	Pantalla OLED 0.91"
Tipo	Biosensor
Dimension placa	65 x 56mm
Rango de medida	0.3 a 1.0; 1.0 a 2.5; 2.5 a 10 micron (μm)
Eficiencia de conteo	50% @ 0.3 μm 98% @ $\geq 0.5 \mu\text{m}$
Tiempo de respuesta	≤ 10 s
Voltaje DC alimentación	5.0V
Corriente máxima de operación	100 mA
Corriente de Standby	$\leq 200 \mu\text{A}$
Nivel de interfaz de datos	L < 0.8 @ 3.3, H > 2.7 @ 3.3 voltios(V)
Rango de temperatura de funcionamiento	10 a +60 Celcius
Rango de humedad de funcionamiento	0 – 99%
Tiempo entre fallos	≥ 3 Años



Figura 9. Air Monitoring HAT para Raspberry Pi – HAT Monitoreo de aire

3.2.6. Winson – ZPHS01B MULTI-IN-ONE SENSOR MODULE

Este es el sensor más completo de esta lista ya que es un sensor todo en uno. Este módulo integra sensor de polvo láser, sensor de dióxido de carbono infrarrojo, sensor de formaldehído electroquímico, sensor de ozono electroquímico, sensor de monóxido de carbono electroquímico, sensor de COV, sensor de dióxido de nitrógeno y sensor de temperatura y humedad. Este puede medir con gran precisión la concentración de varios gases en el aire con la interfaz de comunicación UART. Es un multi-sensor profesional que engloba los principales detectores de gases para monitorear la calidad del aire en un interior con un único módulo. El principal inconveniente es su elevado precio.

Sus aplicaciones serían para ser utilizado en un purificador de aire, sistema de climatización, sistema HVAC, en casas inteligentes, monitorear la calidad del aire, etc.

Sus principales características son las siguientes:

- Gases objetivo: CO₂, PM_{2.5}, CH₂O, O₃, CO, TVOC, NO₂. Además de Temperatura y Humedad
- Tensión de trabajo: 5 V (DC)
- Corriente media < 300 mA
- Corriente pico < 500 mA
- Interfaz física: XH2.54-4P
- Nivel de interfaz: 3 V (compatible con 3.3 V)
- Señal de salida UART (TTL)
- PM_{2.5} Rango de 0 a 1000 µg / m³



Figura 10. Winson – ZPHS01B MULTI-IN-ONE SENSOR MODULE

3.2.7. Comparativa de sensores

En esta tabla se adjuntan las principales propiedades de cada sensor que son voltaje de trabajo, corriente máxima, el tipo de gases y medición que detectan y su temperatura de trabajo para poder comparar todos los sensores con más facilidad.

Tabla 2. Comparación de los sensores

	Voltaje de trabajo	Corriente máxima	Medición	Temperatura de trabajo
Sensor de Polvo PPD42NS PM2.5	5 V (DC)	90 mA	Polvo	De 0 a 45 °C
Sensor de CO2 NDIR MH-Z19C HVAC	5 V (DC)	125 mA	CO2	De -10 a 50 °C
Sensor de CO2 NDIR MH-Z14A	Rango de 4.5 a 5.5 V (DC)	150 mA	CO2	De 0 a 50 °C
Nova PM Sensor SDS011	5 V (DC)	100 mA	Calidad del aire	De -20 a 50 °C
Air Monitoring HAT para Raspberry Pi	5 V (DC)	100 mA	Cantidad de partículas suspendidas en el aire	De -10 a 60 °C
Winson ZPHS01B	5 V (DC)	500 mA	CO2, PM2.5, CH2O, O3, TVOC, NO2, Temperatura y Humedad	De -20 a 65 °C

3.2.8. Sensores de gases de la familia MQ

Los sensores de gases de la familia MQ son dispositivos diseñados para detectar la presencia de diferentes componentes químicos en el aire. Estos sensores se pueden conectar a un autómata o a un microcontrolador.

Existe una gran variedad de sensores MQ y cada modelo está diseñado para poder detectar presencia de una o más sustancias en el aire, es decir, que cada sensor está pensado para un uso específico, como por ejemplo para medir la calidad del aire, detección de gases inflamables o para detectar alcohol en aire inspirado.

Estos sensores de gases suelen proporcionarse en un módulo utilizando un comparador en su circuito y este permite obtener la lectura tanto en valor analógico como en valor digital regulando su sensibilidad a través de un potenciómetro ubicado en su placa. Estos son muy utilizados en proyectos de electrónica con Arduino ya que con ellos se puede encender y apagar un ventilador en función de la calidad de aire que se tenga, hacer un pequeño detector de alcoholemia o una alarma de incendios que suene al detectar humos.

Los sensores de esta familia están compuestos por un sensor electroquímico que varía su resistencia al estar en contacto con las sustancias que hay en el aire. Todos ellos también disponen de un calentador necesario para elevar la temperatura del sensor ya que mientras este no alcance la temperatura de funcionamiento la lectura del sensor no será fiable. El tiempo que necesitan para su calentamiento depende del modelo del sensor, en la mayoría de los modelos es suficiente con unos pocos minutos, aunque algunos modelos necesitan entre 12 y 48 horas para obtener mediciones fiables.

Por otro lado, cada modelo necesita su propia tensión de alimentación para el calentador. En muchos modelos la tensión de alimentación es de 5 V (DC), pero algunos modelos tienen condicionantes especiales para su alimentación. Por esto, el consumo de los sensores MQ puede ser elevado debido a su calentador y su consumo puede llegar a ser hasta de 800 mW. Esta potencia es superior a la potencia que puede suministrar el regulador de algunos microcontroladores, por lo que será necesario una fuente de alimentación externa.

Esta tabla muestra los distintos modelos de sensor MQ:

Tabla 3. Sensores de la familia MQ

Modelo	Sustancias detectadas	Calentador
MQ2	Metano, butano, GLP, humo	5 V (DC)
MQ3	Alcohol, etano, humo	5 V (DC)
MQ303A	Alcohol, etano, humo	0.9 V (DC)
MQ4	Metano, gas natural comprimido (GNP)	5 V (DC)
MQ5	Gas natural, GLP	5 V (DC)
MQ6	Butano, GLP	5 V (DC)
MQ306A	Butano, GLP	0.9 V (DC)
MQ7	Monóxido de Carbono	Alternado 5 y 1.4 V (DC)
MQ307A	Monóxido de Carbono	Alternado 0.2 y 0.9 V (DC)
MQ8	Hidrógeno	5 V (DC)
MQ9	Monóxido de carbono, gases inflamables	Alternado 5 y 1.5 V (DC)
MQ309A	Monóxido de carbono, gases inflamables	Alternado 0.2 y 0.9 V (DC)
MQ131	Ozono	6 V (DC)
MQ135	Benceno, alcohol, dióxido de carbono, humo, calidad de aire	5 V (DC)
MQ136	Ácido sulfhídrico	5 V (DC)
MQ137	Amoniaco	5 V (DC)
MQ138	Benceno, tolueno, alcohol, acetona, propano, formaldehido, hidrógeno	5 V (DC)
MQ214	Metano, gas natural	5 V (DC)
MQ216	Gas natural, gas carbón	6 V (DC)
MQ811	Dióxido de carbono	6 V (DC)

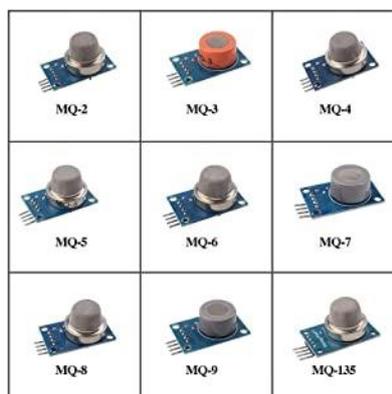


Figura 11. Familia de sensores MQ

4. Descripción de la solución adoptada

En cuanto a los microcontroladores En este apartado se hará una descripción de las soluciones adoptadas para este proyecto y se aportarán también las limitaciones, tanto en tema de sensores como en tema de microcontroladores y distintos lenguajes de programación.

4.1. Diagrama de bloques del sistema

En este apartado se mostrará el diagrama de bloques del sistema que se ha diseñado. Un diagrama de bloques es la representación del funcionamiento del sistema y de la relación que tienen entre sí todos los componentes, además define la organización de todo el proceso interno.

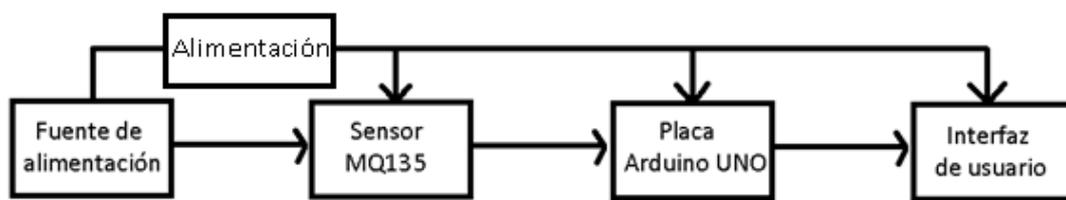


Figura 12. Diagrama de bloques del sistema

En esta figura se puede observar como la fuente de alimentación alimenta a todos los componentes del sistema con 5 V (DC) en paralelo ya que todos los componentes necesitan ese voltaje para su funcionamiento. Seguidamente se tiene el sensor MQ135 que mide la calidad del aire y este se encarga de mandar esa información al microcontrolador Arduino UNO. El microcontrolador, que ya tiene cargado el programa en su microprocesador, se encarga de recoger esa información a través del Pin 2 de señal analógica y clasificar el valor recogido según el nivel de calidad de aire. Así, el microcontrolador pondrá una de sus salidas a las que está conectada la interfaz de usuario a nivel alto y esto hará que se encienda un Led según la calidad del aire en ese momento, también se encarga de mandar la información a la pantalla LCD para que esta muestre el número de partículas en partes por millón de ese entorno y se pueda visualizar claramente.

4.2. Arduino

El proyecto de Arduino nació en 2003, cuando varios estudiantes de un instituto de diseño interactivo quisieron facilitar el acceso y el uso de la electrónica y la programación. Esto se hizo con la finalidad de que los estudiantes de electrónica tuvieran una alternativa más económica ya que por aquel entonces las placas por aquellos tiempos tenían un elevado precio y todos no se lo podían permitir.

Arduino se trata de una plataforma de creación de electrónica de código abierto basada en hardware y software libre, además, esta comunidad diseña y construye placas hardware fáciles de utilizar para sus desarrolladores. Estas placas están compuestas por circuitos impresos que contienen un microcontrolador y un conjunto de componentes electrónicos. Esta plataforma permite crear diferentes tipos de microordenadores con una sola placa.

El término hardware libre se trata de dispositivos cuyas especificaciones y diagramas son de acceso a todo el público. Esto quiere decir que cualquier persona o empresa puede crear sus propias placas, pudiendo ser diferentes entre ellas, pero funcionales a partir de una misma base.

El software libre también son programas a los que a su código puede acceder cualquier persona para que sea quien sea pueda utilizarlo y modificarlo. Arduino también ofrece la plataforma Arduino IDE (Entorno de Desarrollo Integrado), que es un entorno de programación con el que cualquiera puede crear sus propios programas y cuyas instrucciones pueden ser enviadas al microcontrolador mediante comunicación serie por puerto USB entre la placa y un ordenador.

Arduino también ofrece algunas ventajas frente a otras plataformas de microcontroladores ya tienen un precio asequible frente a otras marcas, es multiplataforma, es decir, es compatible con casi todos los sistemas operativos (Windows, IOS, Linux, etc.) frente a otros microprocesadores que están limitados a Windows y además su hardware también es ampliable.

Una vez que conocemos el proyecto Arduino se necesita hacer la descarga y ejecución del Arduino IDE¹.

¹ Véase ANEXO I.

4.2.1. Microcontrolador Arduino UNO

Para la elección del microcontrolador se tuvieron en cuenta dos alternativas principalmente. La primera alternativa que se barajó fue Raspberry Pi, aunque viendo el alcance del proyecto y teniendo en cuenta el lenguaje de programación y la infinidad de posibilidades que da el soporte Arduino, finalmente se eligió el microcontrolador Arduino UNO para la realización de este proyecto. Las razones que han llevado a la elección se pueden clasificar en las siguientes:

- La **flexibilidad que tiene para programarlo** ya que no se requiere de conocimientos muy avanzados para realizar proyectos con este microcontrolador ya que todo el lenguaje es simple y directo ya que está muy simplificado.
- El **software es multiplataforma**, es decir, funciona casi con la totalidad de sistemas operativos mientras que la mayoría de otros entornos para microcontroladores están únicamente limitados a Windows.
- **No se necesita un programador externo o una fuente de energía externa** a la del ordenador con el que se programa.
- **Su coste es bastante bajo** y no es necesario gastar mucho dinero para adquirir la placa.

Después de conocer las ventajas, también es importante conocer sus principales limitaciones:

- Las **carencias de voltaje en los pines de entrada/salida, así como en la corriente de salida**. Si se necesita detectar o medir un voltaje en un pin de entrada digital o analógica hay que intentar asegurar que el voltaje esté siempre entre 0 y 5 voltios. Su corriente de salida además tiene un máximo y es de 500 mA cuando el microcontrolador está alimentado por el puerto USB, aunque si está alimentado de forma externa el fabricante asegura que se puede obtener 1 A.
- El **poder de procesamiento de Arduino UNO tiene menos poder** en comparación con otras opciones más potentes como los STM32. Debido a esta razón, su consumo de energía es más elevado que las últimas placas que hay en el mercado basadas en ARM.

- **Sus pines de entrada/salida del microcontrolador Arduino Uno son bastante limitados**, aunque esto se puede superar utilizando la placa Arduino MEGA.

4.3. Sensor MQ135

En la elección del sensor para medir la calidad del aire se tuvieron en cuenta diversas alternativas (punto 3) teniendo en cuenta principalmente el tipo de gases que medía, su sensibilidad, el consumo del sensor, la alimentación que necesita el sensor y el tipo de salidas que ofrece y sobre todo como de compatible era con el microcontrolador que se eligió para este proyecto.

Este es un sensor que mide la calidad del aire. Utiliza un pequeño calentador en el interior con un sensor electroquímico. Este sensor es sensible para una gama de gases y se utilizan en interiores a temperatura ambiente. Permite detectar algunos gases peligrosos como el amoniaco, dióxido de nitrógeno, alcohol, benceno, dióxido y monóxido de carbono.

El sensor puede detectar concentraciones de gases entre 10 y 1000 ppm y es de mucha utilidad en detección de gases nocivos para la salud humana. Su velocidad de respuesta es bastante alta, por lo que es capaz de activar cualquier dispositivo de manera oportuna.

La presentación de este dispositivo es en un módulo y puede conectarse a un microcontrolador muy fácilmente y se incluye la electrónica básica para realizar la interfaz con el sensor. Este sensor dispone de tanto salida analógica como digital.

El sensor MQ135 es ideal para la detección de gas amoniaco, benceno y alcohol, aunque este sensor también es sensible al humo y otros gases tóxicos como el butano o propano, por lo que puede utilizarse para diseñar dispositivos que alerten sobre cualquier presencia de estos gases en el ambiente.

Por todas estas razones es el sensor elegido ya que el proyecto está destinado a la medición de gases tóxicos en oficinas y centros de trabajo relativamente limpios.

Sus principales limitaciones son que al ser sensible a muchos gases en el aire no es muy posible obtener muy buenos valores de CO2 a partir de él. También tarda un tiempo en ponerse en funcionamiento ya que como este tiene un calentador en su interior necesita sobre unos cinco minutos para proporcionar respuestas fiables.

Las características de este sensor son las siguientes:

- Chip principal: sensor de gas natural MQ-135 y comparador LM393
- Voltaje de trabajo: 5V (DC)
- Corriente: 150 mA
- Salida digital: TTL 0/1
- Salida analógica: valor analógico de tensión
- Sensibilidad: ajustable con el potenciómetro
- Tamaño: 32mm x 20mm x 18mm



Figura 13. Sensor MQ135

4.4. Interfaz de usuario

En este proyecto también se ha utilizado una pantalla LCD Geeepi IIC/I2C 2004 20x4 para visualizar el nivel de calidad de aire que tenemos en el interior del recinto. Esta pantalla ha tenido que ser programada en Arduino IDE para su funcionamiento y está conectada a dos puertos analógicos de la placa, otro pin a alimentación de 5 V (DC) y otro a GND. También se ha combinado con cuatro diodos Leds de distintos colores (verde, azul, amarillo y rojo) para que, dependiendo de la lectura que tenga el sensor se encienda un Led según la calidad de aire que tengamos en el ambiente. Los diodos Leds están montados en una placa protoboard junto con una resistencia en serie para que estos no se dañen. Los diodos Leds están conectados a distintas salidas de la placa de Arduino mediante cablecillos y según la calidad del aire se pondrá una salida a nivel alto para encenderlos.



Figura 14. Pantalla LCD Geeepi IIC/I2C 2004 20x4

Para marcar y distinguir los niveles de calidad de aire, se ha optado por un código visual como diodos Led. La elección de los colores se ha relacionado con la cultura general de peligro en nuestra sociedad. Así, los colores representativos son los siguientes:

- **Diodo verde:** Aire limpio
- **Diodo azul:** Aire con un poco de CO2
- **Diodo amarillo:** Aire con CO2
- **Diodo rojo:** Detección de gases peligrosos

Hay que indicar que estos niveles pueden variar dependiendo de la necesidad del usuario. Se puede llevar cambiándolos en el código del programa.

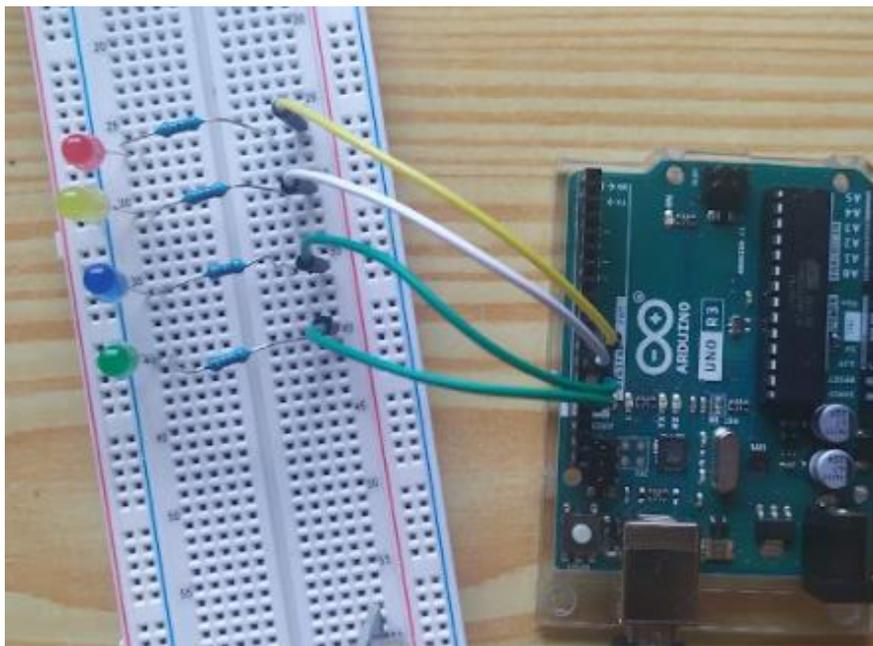


Figura 15. Conexión de los diodos Led a la placa de Arduino

4.5. Montaje real

En este apartado se van a adjuntar dos imágenes para poder visualizar el sistema en funcionamiento en sus dos extremos. Primero con aire limpio y seguidamente se aplicará gas cerca del sensor con un mechero para que este pueda detectar presencia de gases peligrosos que puedan causar un incendio.

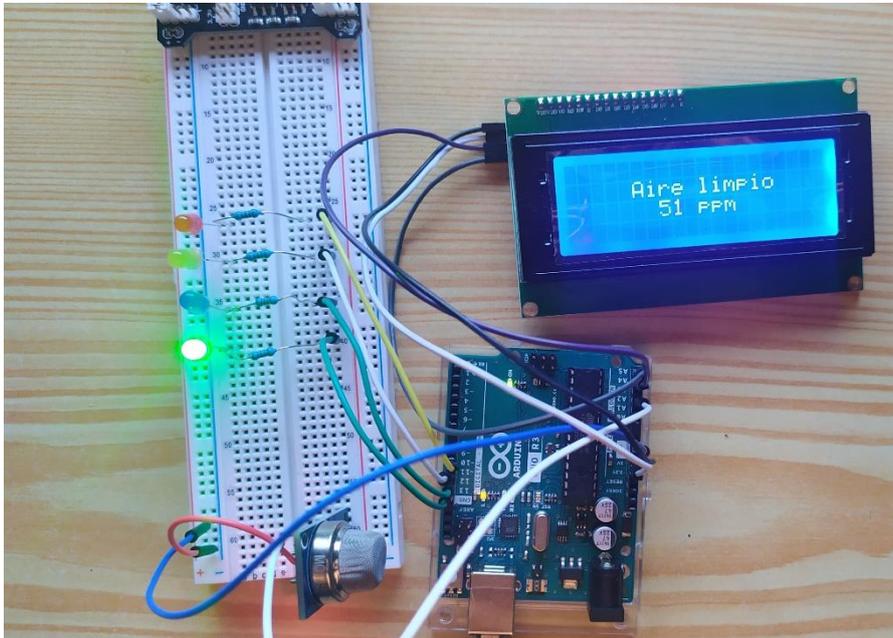


Figura 16. Sistema de medición de calidad de aire detectando aire limpio

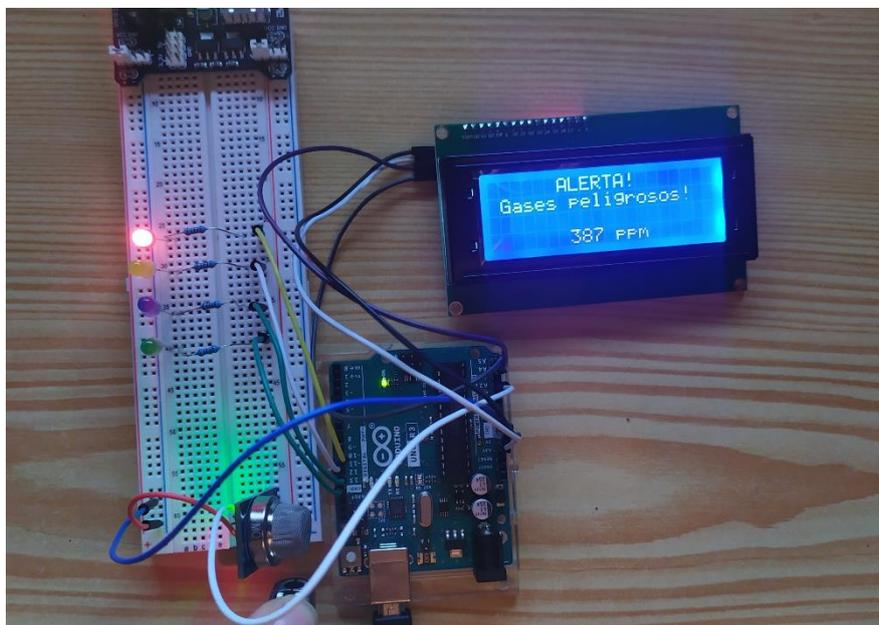


Figura 17. Sistema de medición de calidad de aire detectando gases peligrosos

4.6. Fuente de alimentación

La mayoría de los aparatos electrónicos como televisores, ordenadores, etc. se conectan a la red eléctrica a 230 V de tensión alterna. Pero estos aparatos y sus componentes realmente trabajan en corriente continua y además en tensiones mucho más bajas. Por este motivo estos aparatos llevan una fuente de alimentación en su interior.

Una fuente de alimentación se encarga de transformar la tensión alterna en tensión continua y regula o cambia la tensión de salida a un valor determinado.

Una fuente de alimentación está compuesta primero por un transformador que lo que hace es disminuir la tensión a un valor mucho menor al que se encuentra en la corriente eléctrica. En este proyecto se elegirá un transformador de 230 V a 12 V.

A continuación, como la corriente alterna tiene tanto parte positiva como negativa en tensión, lo que se necesitará es rectificar la señal. Esto se realizará mediante un puente de diodos que convierten una señal con partes positivas y negativas en una señal que tiene solo partes positivas. Esto se conoce como rectificador de onda completa.

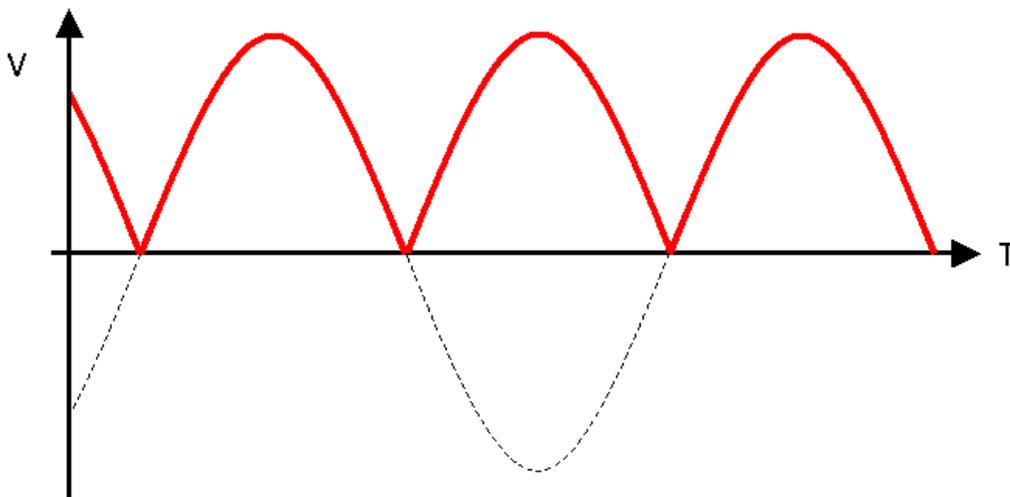


Figura 18. Señal rectificada completamente

Si se utilizara un solo diodo, este lo que haría es quedarse con la parte positiva, es decir, la parte negativa de la tensión desaparecería por completo. Pero utilizando el puente podemos aprovechar también la parte negativa ya que el puente lo que hace es convertir la parte negativa en positiva también.

El puente se combina con un condensador de rizado y también con un diodo Zener principalmente. Esto se hace para limitar la tensión y convertir la corriente alterna en continua. La misión del puente de diodos es hacer que la tensión vaya en un solo sentido, mientras que el resto de los componentes (condensador y diodo Zener) tienen la función de estabilizar la señal.

Para la alimentación de todo el circuito de este proyecto, sensor, placa de Arduino UNO, la pantalla LCD y los diodos Led, se ha diseñado una fuente de alimentación teniendo en cuenta el consumo máximo de todos los componentes de este proyecto. Pero, por otro lado, en este proyecto no se ha utilizado un diodo Zener después del condensador de filtro. Se ha utilizado un regulador de tensión LM7805 que es un circuito integrado cuya finalidad es mantener un voltaje estable en su pin de salida independientemente de lo que se haya aplicado en su pin de entrada. Este dispositivo consta de tres pines uno de entrada otro de salida y otro terminal de masa.

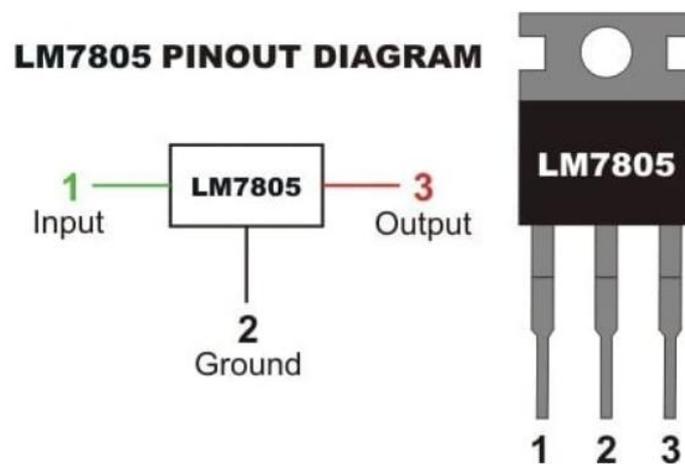


Figura 19. Diagrama de pines del regulador de tensión 7805

4.7. Diseño de la fuente de alimentación

Para alimentar todos los componentes de este proyecto necesitamos que la fuente de alimentación nos proporcione 5 V (DC) ya que todos los componentes necesitan ese voltaje de alimentación.

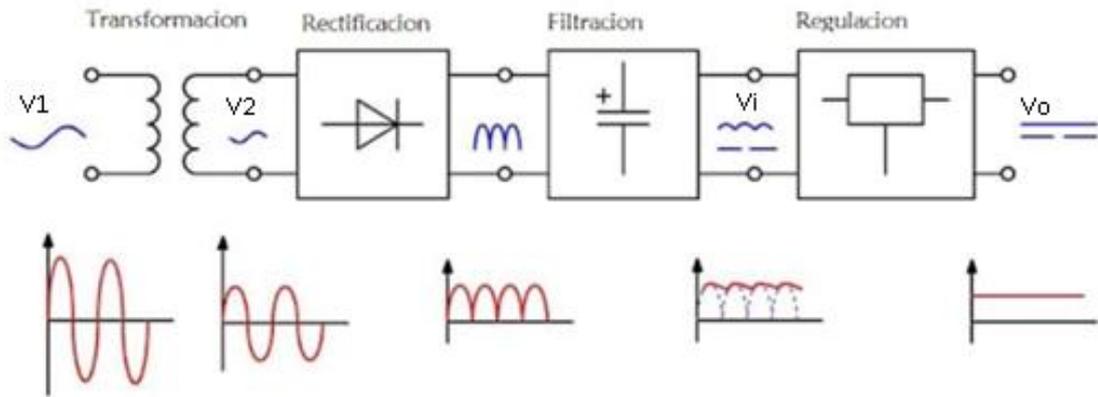


Figura 20. Diagrama de la fuente de alimentación del sistema

El regulador de tensión 7805 necesita como mínimo 3 V más a la entrada que los que entrega a la salida según el fabricante. Por lo tanto:

$$V_o = 5 V$$

$$V_i = V_o + 3 V = 8 V$$

Como la tensión de salida del regulador es de 5 V necesitamos 8 V como mínimo a la entrada de este. Para ello elegiremos un transformador de 230 V a 12 V que es el transformador más cercano que hay en el mercado al voltaje que se quiere convertir, cuya relación de transformación será la siguiente:

$$\frac{V_2}{V_1} = \frac{N_2}{N_1} \rightarrow \frac{V_2}{V_1} = \frac{12}{230} \rightarrow \frac{N_2}{N_1} = 0.052$$

$$V_{Riz P-P} = 15.6 V - 12 V = 3.6 V$$

$$V_{Riz P-P} = \frac{I_o}{2 \times F_{red} \times C} \rightarrow C = \frac{I_o}{2 \times F_{red} \times V_{Riz P-P}} =$$

$$= \frac{0.5 A}{2 \times 50 Hz \times 3.6 V} = 1388 \mu F$$

Con este resultado se elegirá un condensador de 2200 μF , que es mayor al del valor obtenido en los cálculos teóricos, pero esto servirá para que el rizado de la tensión sea menor.

En cuanto al puente de diodos, los seleccionaremos según la corriente promedio que circulará por ellos. Como conducen dos diodos en cada semiciclo cada diodo aportará la mitad de corriente de salida, es decir, 0.25 A. Con esto se puede decir que valdrá cualquier diodo de la familia 1N4000, ya que tienen la capacidad de conducir una corriente hasta de 1 A.

Ahora, también habrá que tener en cuenta la tensión de entrada que son 230 V. Con esto se elegirá el diodo 1N4004 que es capaz de conducir hasta 280 V.

Maximum Ratings and Electrical Characteristics @ $T_A = 25^\circ C$ unless otherwise specified									
Single phase, half wave, 60Hz, resistive or inductive load. For capacitive load, derate current by 20%.									
Characteristic	Symbol	1N4001 G/GL	1N4002 G/GL	1N4003 G/GL	1N4004 G/GL	1N4005 G/GL	1N4006 G/GL	1N4007 G/GL	Unit
Peak Repetitive Reverse Voltage Working Peak Reverse Voltage DC Blocking Voltage	V_{RRM} V_{RWM} V_R	50	100	200	400	600	800	1000	V
RMS Reverse Voltage	$V_{R(RMS)}$	35	70	140	280	420	560	700	V
Average Rectified Output Current (Note 1) @ $T_A = 75^\circ C$	I_o	1.0							A
Non-Repetitive Peak Forward Surge Current 8.3ms single half sine-wave superimposed on rated load (JEDEC Method)	I_{FSM}	30							A
Forward Voltage @ $I_F = 1.0A$	V_{FM}	1.0							V
Peak Reverse Current at Rated DC Blocking Voltage @ $T_A = 25^\circ C$ @ $T_A = 125^\circ C$	I_{RM}	5.0 50							μA
Reverse Recovery Time (Note 3)	t_{rr}	2.0							μs
Typical Junction Capacitance (Note 2)	C_j	8.0							pF
Typical Thermal Resistance Junction to Ambient	$R_{\theta JA}$	100							K/W
Operating and Storage Temperature Range	T_j, T_{STG}	-65 to +175							$^\circ C$

Figura 22. Datasheet familia de diodos 1N4000

Seguidamente se va a proceder a calcular si el regulador de tensión 7805 necesita un disipador de calor. Para ello, se calculará la potencia que el regulador de tensión tiene que disipar.

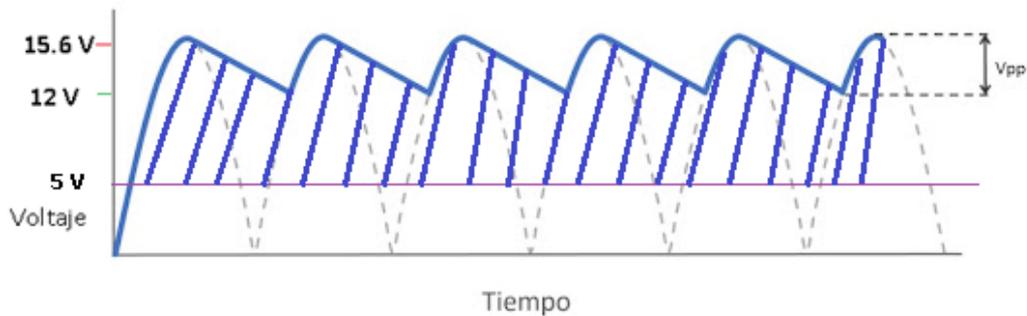


Figura 23. Voltaje necesario para calcular la potencia disipada del regulador de tensión

Para calcular la potencia disipada por el regulador de tensión se necesita saber el voltaje sombreado de la figura 24. Para ello se hará una aproximación y se tendrá en cuenta el voltaje máximo a su entrada, el voltaje mínimo a su entrada y el voltaje a la salida del regulador.

Por lo tanto:

$$V_{i_{max}} = 15.6 V \quad V_{i_{min}} = 12 V \quad V_o = 5 V$$

La aproximación se hará de la siguiente manera. Se restará el voltaje máximo a la entrada menos el voltaje mínimo a la entrada y este resultado se dividirá entre dos ya que la tensión varía con tiempo. Por otro lado, se restará el voltaje mínimo a la entrada por el voltaje de salida y estos dos resultados obtenidos se sumarán.

$$V_1 = \frac{15.6 V - 12 V}{2} = 1.8 V \approx 2 V$$

$$V_2 = 12 V - 5 V = 7 V$$

$$V_{aprox} = 2 V + 7 V = 9 V$$

Mediante el voltaje de aproximación calculado y la corriente de salida del regulador de tensión 7805 ya se puede calcular la potencia disipada por este.

$$V_{aprox} = 9 V \quad I_o = 0.5 A$$

$$P = V \times I$$

$$P = 9 V \times 0.5 A = 4.5 W$$

A continuación, mediante la potencia disipada calculada, la resistencia térmica entre la unión y el ambiente del regulador de tensión, y suponiendo una temperatura ambiente de 20 °C se sacará la temperatura que tiene la unión y con esta se podrá saber si es necesario aplicarle un disipador de calor al regulador.

Por lo tanto:

$$T_j = T_a + P \times R_{th_{j-a}}$$

Donde T_j es la temperatura de la unión, T_a es la temperatura ambiente, P es la potencia que tiene que disipar el regulador y $R_{th_{j-a}}$ es la resistencia térmica entre la unión y el ambiente del regulador y esta se encuentra en el datasheet del fabricante.

Absolute Maximum Ratings

Parameter	Symbol	Value	Unit
Input Voltage (for $V_O = 5V$ to $18V$)	V_I	35	V
(for $V_O = 24V$)	V_I	40	V
Thermal Resistance Junction-Cases (TO-220)	$R_{\theta JC}$	5	°C/W
Thermal Resistance Junction-Air (TO-220)	$R_{\theta JA}$	65	°C/W
Operating Temperature Range	TOPR	0 ~ +125	°C
Storage Temperature Range	TSTG	-65 ~ +150	°C

Figura 24. Datasheet del regulador LM7805

$$T_j = 20\text{ }^\circ\text{C} + 4.5\text{ W} \times 65\text{ }^\circ\frac{\text{C}}{\text{W}} = 312.5\text{ }^\circ\text{C}$$

Así con este resultado es muy fácil de ver que el regulador de tensión LM7805 necesita urgentemente un disipador de calor ya que, si no este se fundiría, ya que, según el datasheet del fabricante, la temperatura de este no puede sobrepasar los 125 °C.

Para saber que disipador sería el mejor para el regulador se tienen que hacer también unos cálculos teniendo en cuenta el siguiente esquema:

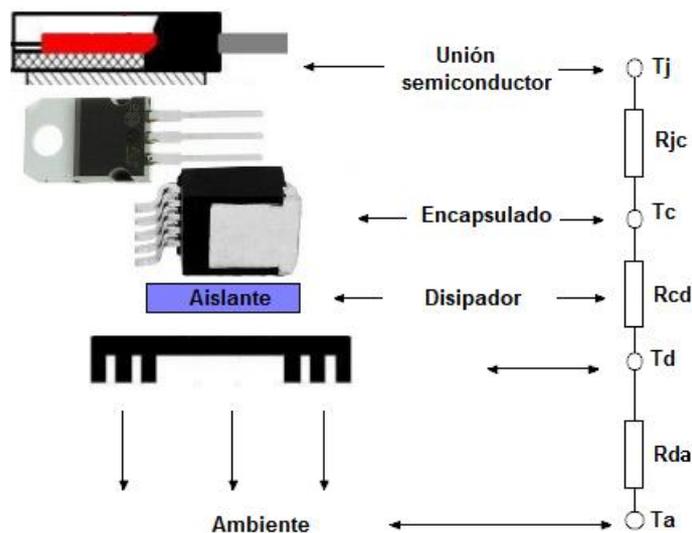


Figura 25. Esquema a tener en cuenta para calcular el disipador de calor

En este esquema se tienen diferentes resistencias y temperaturas. Por un lado, tenemos la temperatura de la unión del semiconductor (T_j), seguidamente tenemos una resistencia térmica que se trata de la resistencia entre la unión del semiconductor y el encapsulado de este. Seguidamente tenemos la temperatura del encapsulado (T_c) que será menor a la temperatura de la unión (T_j). Como entre el encapsulado y el disipador se va a colar una pasta térmica para facilitar la transmisión de calor entre estos también hay que considerar la resistencia térmica de esta pasta térmica (R_{cd}) que normalmente suele tener un valor pequeño. Y por último se tiene la resistencia térmica entre el disipador y el ambiente (R_{da}) que es la resistencia térmica que se necesita saber su valor para elegir un tipo de disipador.

Para calcular esta resistencia térmica se necesitan los siguientes valores, algunos los proporciona el fabricante en el datasheet del componente electrónico y otros se han calculado previamente.

$$T_{j_{max}} = 125 \text{ }^{\circ}\text{C} \quad P = 4.5 \text{ W} \quad T_a = 20 \text{ }^{\circ}\text{C}$$

$$R_{th_{j-c}} = 5 \frac{\text{ }^{\circ}\text{C}}{\text{W}} \quad R_{th_{c-d}} = 1 \frac{\text{ }^{\circ}\text{C}}{\text{W}}$$

El valor de la temperatura de la unión máxima del componente cuando este está en funcionamiento ($T_{j_{max}}$) se puede obtener directamente del datasheet del fabricante, junto con la resistencia térmica entre la unión y el encapsulado ($R_{th_{j-c}}$). La resistencia térmica de cualquier pasta térmica ($R_{th_{c-d}}$) se considera siempre del valor de $1 \text{ }^{\circ}\text{C}/\text{W}$. Por otro lado, la potencia que disipa el regulador (P) ya se ha calculado anteriormente y la temperatura ambiente (T_a) también se ha supuesto previamente de $20 \text{ }^{\circ}\text{C}$.

Absolute Maximum Ratings

Parameter	Symbol	Value	Unit
Input Voltage (for $V_O = 5\text{V}$ to 18V)	V_I	35	V
(for $V_O = 24\text{V}$)	V_I	40	V
Thermal Resistance Junction-Cases (TO-220)	$R_{\theta JC}$	5	$^{\circ}\text{C}/\text{W}$
Thermal Resistance Junction-Air (TO-220)	$R_{\theta JA}$	65	$^{\circ}\text{C}/\text{W}$
Operating Temperature Range	TOPR	0 ~ +125	$^{\circ}\text{C}$
Storage Temperature Range	TSTG	-65 ~ +150	$^{\circ}\text{C}$

Figura 26. Elección de los datos en el datasheet del regulador LM7805

Con estos valores y con la siguiente fórmula se conseguirá el valor de la resistencia entre el disipador y el ambiente ($R_{th_{d-a}}$):

$$R_{th_{d-a}} = \frac{0.7 \times T_{j_{max}} - T_a}{P} - R_{th_{j-c}} - R_{th_{c-d}}$$

El factor 0.7 se utiliza por motivos de seguridad ya que, aunque el fabricante aconseje en el datasheet que la temperatura no sobrepase los 125 °C cuando este está en funcionamiento.

Así sustituyendo en la fórmula anterior:

$$R_{th_{d-a}} = \frac{0.7 \times 125 \text{ }^{\circ}\text{C} - 20 \text{ }^{\circ}\text{C}}{4.5 \text{ W}} - 5 \frac{\text{ }^{\circ}\text{C}}{\text{W}} - 1 \frac{\text{ }^{\circ}\text{C}}{\text{W}} = 8.2 \frac{\text{ }^{\circ}\text{C}}{\text{W}}$$

Con este dato ya se puede elegir el disipador. Para ello se va a utilizar el catálogo de disipadores de la empresa Disipa Heat S.L. donde se encontrarán diferentes tipos de disipadores con sus respectivas resistencias térmicas y dimensiones.

Para este diseño se necesita un disipador que tenga la misma resistencia térmica o menor que la calculada para que la transmisión de calor entre el disipador y el ambiente sea lo más eficiente posible.

Así con los datos obtenidos y con el catálogo de la empresa Disipa se elegirá el siguiente disipador:

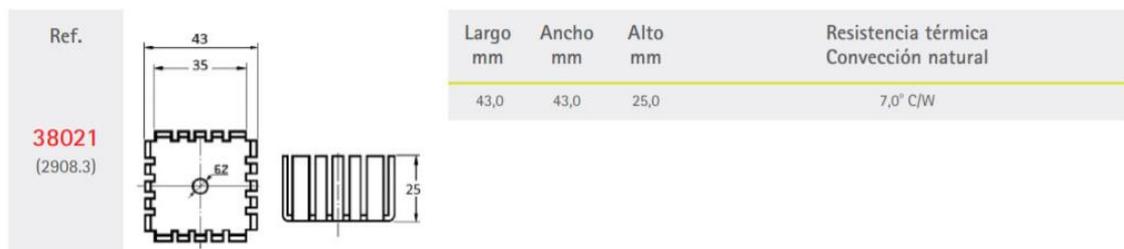


Figura 27. Disipador de calor elegido

Este es el disipador de calor elegido. Como se puede observar la resistencia térmica por convección natural, es decir, que no necesita convección forzada añadiendo algún ventilador, es menor a la que se ha calculado, por lo tanto, al ser la resistencia menor, tendrá más facilidad a disipar el calor del componente. El catálogo también disponía de otro disipador justo con las mismas características, pero se ha elegido este por el simple hecho que el regulador de tensión dispone de una sujeción a la que se le puede acoplar un tornillo y es justamente compatible con este disipador por lo que el regulador de tensión estará debidamente sujeto.

4.8. Programa

En este apartado se va a proporcionar el programa que se ha realizado en Arduino IDE para descargarlo en la placa y que esta siga las instrucciones que se le proporcionan. El programa está debidamente explicado paso por paso lo que va a hacer el microcontrolador en cada momento.

Cada comentario que indica lo que hace cada instrucción comienza con “//”:

```
//Se inicializan las sistema
#include <LiquidCrystal_I2C.h> //Librería para la pantalla LCD

#include <Wire.h> //Librería para comunicarse con la pantalla
LCD

//Se inicializan las variables
int s_analogica_mq135=0; //Variable para la señal analógica. Se
igual a 0

int limpio = 13; //Variable para el aire limpio en el PIN 13

int aire = 12; //Variable para el aire con un poco de CO2 en el
PIN 12

int CO2 = 11; //Variable para el aire con CO2 considerable en el
PIN 11

int gas = 10; //Variable para gases peligrosos en el aire en el
PIN 10

LiquidCrystal_I2C lcd(0x27,20,4); //Puerto en el que está
conectado la pantalla y su tamaño
```

```
//Primera función a ejecutar, donde se configuran las variables
void setup() {

    Serial.begin(9600); //Inicializar pantalla de Arduino IDE en
    el PC

    //Declaración de los pines de salida
    pinMode(aire, OUTPUT);

    pinMode(CO2, OUTPUT);

    pinMode(gas, OUTPUT);

    pinMode(limpio, OUTPUT);

    //Se inicializan los pines de salida en modo bajo (0 V)
    digitalWrite(aire, LOW);

    digitalWrite(CO2, LOW);

    digitalWrite(gas, LOW);

    digitalWrite(limpio, LOW);

    //Se inicializa la pantalla
    lcd.init();
    lcd.init();

    lcd.backlight(); //Se apaga la pantalla por un momento para
    verificar que funciona

}
```

```

//Inicio de la función principal del programa
void loop() {

  s_analogica_mq135 = analogRead(2); //Se lee el PIN analógico 2
de la placa de Arduino que es donde está conectado el sensor

  Serial.println(s_analogica_mq135, DEC); //Aparece en la
pantalla de Arduino IDE el número de partículas

  Serial.println(" ppm"); //Se pone PPM ya que son las
46istema46a46 por partes por millón en el aire

  delay(250); //Se hace una pausa para que el valor se actualice
cada cuarto de segundo

  if (s_analogica_mq135<=55) //Si el valor es menor de 55 el aire
será limpio
  {

    Serial.println("Aire limpio"); //Aparece en la pantalla de
Arduino IDE "Aire Limpio"

    digitalWrite(limpio,HIGH); //La salida 13 de aire limpio pasa
a valor alto

    digitalWrite(aire,LOW); //La salida 12 de aire con un poco de
CO2 sigue en valor bajo

    digitalWrite(CO2,LOW); //La salida 11 de aire con CO2 sigue en
valor bajo

    digitalWrite(gas,LOW); //La salida 10 de aire con gases
peligrosos sigue en valor bajo

    lcd.clear(); //Se borra todo lo que hay en la pantalla LCD

    lcd.setCursor(5,1); //Se pone el cursor de la pantalla LCD en
la posición (5,1)

    lcd.print("Aire limpio"); //Se escribe en la pantalla LCD
"Aire Limpio"

    lcd.setCursor(7,2); //Se pone el cursor de la pantalla LCD en
la posición (7,2)
    lcd.print(s_analogica_mq135, DEC); //Se escribe en la pantalla
LCD el número de partículas
    lcd.print(" ppm"); //Se escribe en la pantalla LCD PPM (partes
por millón)

    delay(50); //Pausa de 5 milisegundos  }

```

```

//aire con un poco de dióxido de carbono
if (s_analogica_mq135>=56 && s_analogica_mq135<=65) //Si el
valor está entre 56 y 65 el aire tendrá un poco de CO2
{

    Serial.println("Aire con CO2"); //Aparece en la pantalla de
Arduino IDE "Aire con CO2"

    digitalWrite(limpio,LOW); //La salida 13 de aire limpio sigue
a valor bajo

    digitalWrite(aire,HIGH); //La salida 12 de aire con un poco de
CO2 pasa a valor alto

    digitalWrite(CO2,LOW); //La salida 11 de aire con CO2 sigue a
valor bajo

    digitalWrite(gas,LOW); //La salida 10 de aire con gases
peligrosos sigue a valor bajo

    lcd.clear(); //Se borra todo lo que hay en la pantalla LCD

    lcd.setCursor(5,1); //Se pone el cursor de la pantalla LCD en
la posición (5,1)

    lcd.print("Aire con CO2"); //Se escribe en la pantalla LCD
"Aire con CO2"

    lcd.setCursor(7,2); //Se pone el cursor de la pantalla LCD en
la posición (7,2)

    lcd.print(s_analogica_mq135, DEC); //Se escribe en la pantalla
LCD el número de partículas

    lcd.print(" ppm"); //Se escribe en la pantalla LCD PPM (partes
por millón)

    delay(50); //Pausa de 5 milisegundos
}

```

```

//aire con dióxido de carbono
if (s_analogica_mq135>=74 && s_analogica_mq135<=230) //Si el
valor está entre 74 y 230 habrá CO2 en el aire
{

  Serial.println("CO2!!"); //Aparece en la pantalla de Arduino
IDE "CO2!!"

  digitalWrite(limpio,LOW); //La salida 13 de aire limpio sigue
a valor bajo

  digitalWrite(aire,LOW); //La salida 12 de aire con un poco de
CO2 sigue a valor bajo

  digitalWrite(CO2,HIGH); //La salida 11 de aire con CO2 pasa a
valor alto

  digitalWrite(gas,LOW); //La salida 10 de aire con gases
peligrosos sigue a valor bajo

  lcd.clear(); //Se borra todo lo que hay en la pantalla LCD

  lcd.setCursor(8,1); //Se pone el cursor de la pantalla LCD en
la posición (8,1)

  lcd.print("CO2!!"); //Se escribe en la pantalla LCD "CO2!!"

  lcd.setCursor(7,2); //Se pone el cursor de la pantalla LCD en
la posición (7,2)

  lcd.print(s_analogica_mq135, DEC); //Se escribe en la pantalla
LCD el número de partículas

  lcd.print(" ppm"); //Se escribe en la pantalla LCD PPM (partes
por millón)

  delay(50); //Pausa de 5 milisegundos
}

```

```

//gas propano y butano
if (s_analogica_mq135>=240) //Si el valor está por encima de
240 habrán gases peligrosos
{

    Serial.println("Alerta!! Gases peligrosos"); //Aparece en la
pantalla de Arduino IDE "Alerta!! Gases peligrosos"

    digitalWrite(limpio,LOW); //La salida 13 de aire limpio sigue
a valor bajo

    digitalWrite(aire,LOW); //La salida 12 de aire con un poco de
CO2 sigue a valor bajo

    digitalWrite(CO2,LOW); //La salida 11 de aire con CO2 sigue a
valor bajo

    digitalWrite(gas,HIGH); //La salida 10 de aire con gases
peligrosos pasa a valor alto

    lcd.clear(); //Se borra todo lo que hay en la pantalla LCD

    lcd.setCursor(6,0); //Se pone el cursor de la pantalla LCD en
la posición (6,0)

    lcd.print("ALERTA!"); //Se escribe en la pantalla LCD
"ALERTA!"

    lcd.setCursor(1,1); //Se pone el cursor de la pantalla LCD en
la posición (1,1)

    lcd.print("Gases peligrosos!"); //Se escribe en la pantalla
LCD "Gases peligrosos!"

    lcd.setCursor(7,3); //Se pone el cursor de la pantalla LCD en
la posición (7,3)

    lcd.print(s_analogica_mq135, DEC); //Se escribe en la pantalla
LCD el número de partículas

    lcd.print(" ppm"); //Se escribe en la pantalla LCD PPM (partes
por millón)

    delay(50); //Pausa de 5 milisegundos

}} //Fin del programa

```

5. Conclusiones

Al comienzo de este proyecto se planteaba una problemática existente en nuestra sociedad, la calidad del aire que respiramos. Concretamente, se proponía como punto de partida mejorarla en espacios cerrados, como pueden ser las oficinas. Por ello, el proyecto giraba en torno al diseño y montaje de un sistema para el control de la calidad del aire.

Los objetivos que debía cumplir el proyecto se cerraron, principalmente, en cuatro:

- E) Diseño y montaje de todo el sistema electrónico.
- F) Controlar los gases nocivos de una oficina.
- G) Prevención ante incendios.
- H) Tener debidamente ventilado el aire de un espacio interior.

Si se siguen los pasos llevados a cabo y la relación con los objetivos propuestos, se puede decir que se cumplen en su mayoría. Así, en primer lugar, se puso en marcha el diseño y montaje de todo el sistema (objetivo A). Una vez realizado todo esto, se verificó el control de gases nocivos procedentes de una fuente externa al sistema (objetivo B). Por lo que se puede anticipar y prevenir incendios en espacios cerrado, ya que se puede detectar el gas inflamable con anterioridad a algún suceso fatídico posterior (objetivo C). Ahora bien, sería importante para evitar este tipo de sucesos con mayor eficacia añadir al sistema un purificador de aire controlado. Por lo tanto, es importante también hacer mención a las limitaciones que plantea el sistema.

Como limitaciones se deben señalar, además de la ya indicada (incluir un purificador de aire controlado), las siguientes:

1. El sensor de gases MQ135 tarda en calentarse sobre unos cinco minutos. En consecuencia, sería necesario que el sistema se activara con anterioridad a su uso porque las mediciones dentro de ese tiempo no podrían ser las correctas.
2. Otra limitación, que entra en relación con la mencionada, sería la relación que se tiene que establecer entre lo que tarda el gas en ser detectado por un único sensor. Así, sería importante realizar estudios posteriores dirigidos a establecer la relación entre la magnitud del espacio que puede abarcar un sensor para detectar un gas inflamable a tiempo.

Como punto final y conclusión de todo lo expuesto es importante indicar que se trata de sistema que puede ser de bastante utilidad en espacios interiores, ya sean oficinas, minas, bodegas, etc., ayudando a prevenir accidentes laborales graves. Pero para poder llevarlo a cabo será necesario actuar sobre todas las limitaciones planteadas y que serían necesarias para conseguir el fin último del diseño de este sistema, mejorar la calidad del aire que respiramos.

6. Referencias

- Información relacionada con los sensores de calidad de aire:
<https://solectroshop.com/es/> (accedido el 7/07/2021)
- Información sobre Arduino y sus microcontroladores, Proteus 8 Profesional, Office 365 y Raspberry Pi:
<https://es.wikipedia.org/> (accedido el 10/07/2021)
- Información de los datasheet del regulador de tensión LM7805 y de la familia de sensores 1N4000:
<https://www.alldatasheet.es/> (accedido el 21/07/2021)
- Más información sobre Arduino y su instalación:
<https://arduino.cl/> (accedido el 24/07/2021)
- Información sobre fuentes de alimentación:
<https://www.areatecnologia.com/> (accedido el 14/07/2021)
- Información y elección del disipador:
<https://disipa.net/> (accedido el 28/07/2021)
- Información sobre la familia de sensores MQ:
<https://www.luisllamas.es/> (accedido el 02/08/2021)
- Información sobre ORDENADOR Portátil MSI GE63 Raider RGB 8RF:
<https://www.pccomponentes.com/> (accedido el 10/08/2021)

Las imágenes de este proyecto se han adquirido de la web o han sido editadas:

- Imágenes:
<https://www.google.es/imghp?hl=es> (accedido el 15/08/2021)
- Imágenes sobre regulador de tensión LM7805:
<https://www.hwlibre.com/> (accedido el 15/08/2021)

7. Anexos

ANEXO I

Descarga y ejecución de Arduino IDE

Lo primero de todo es ingresar a la web oficial de Arduino y acceder a la pestaña de “Software” que se encuentra en la parte superior de la página web. A través de esta, en un menú desplegable se encontrará una opción que indica “Descargas” o “Downloads” y desde allí, se podrá obtener el archivo de descarga del programa. A continuación, hay que elegir el sistema operativo que se tenga en el ordenador.

Una vez se haya descargado el software de Arduino, se tiene que entrar en la carpeta de Descargas del ordenador (o en la ubicación en la que se ha escogido para guardar la descarga del archivo) y hacer doble clic en el instalador del programa. Así, se iniciará el proceso de instalación.

Es importante que se mantengan marcados los componentes de instalación predeterminados. Uno de los más relevantes es el “Install USB driver” o “Instalar el controlador USB” para poder enviar los programas a la placa de Arduino. Cuando esto se complete se pulsará el botón “Siguiete” o “Next” en la parte inferior derecha y después se dejará como carpeta de destino la que dice como predeterminada que es “Program Files (x86) / Arduino”. Con esto, solo falta hacer clic en el botón “Instalar” o “Install”.

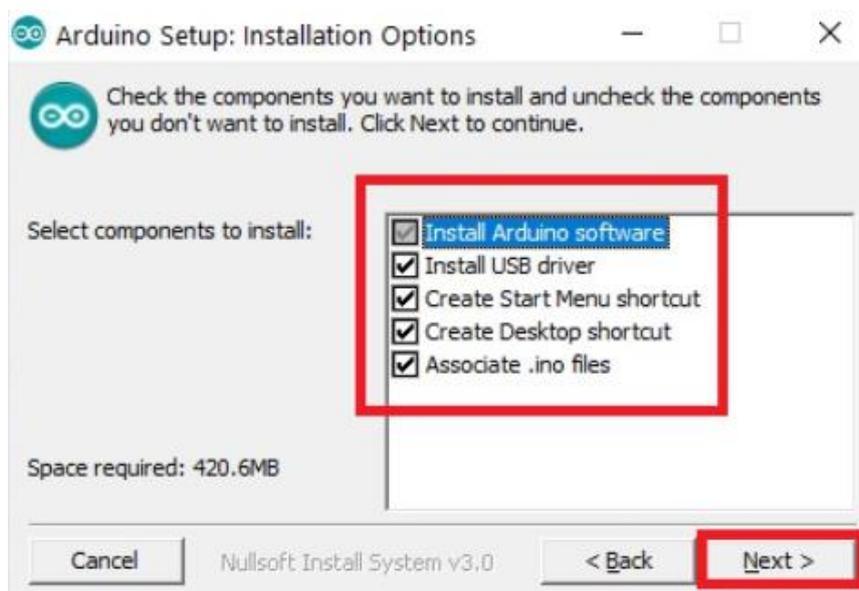


Figura 28. Instalación de drivers de Arduino IDE

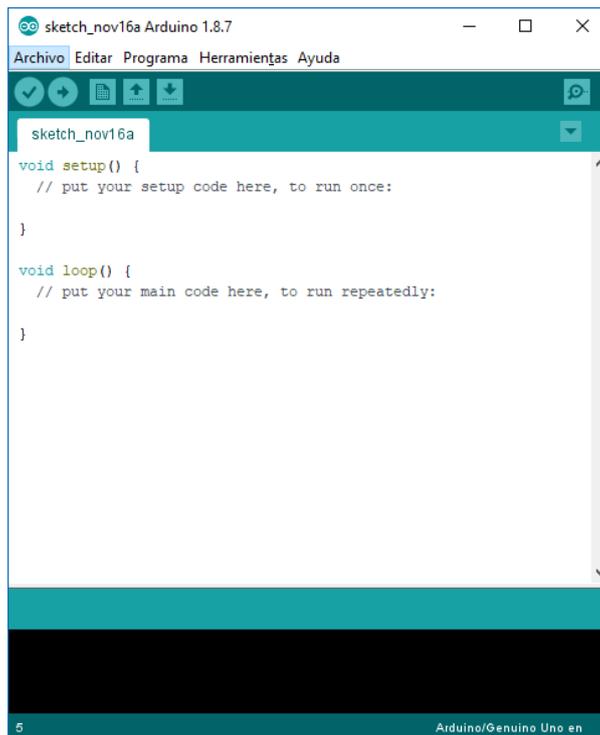


Figura 29. Ventana principal de Arduino IDE

Cuando se haya iniciado el Arduino IDE, es necesario realizar algunos ajustes. El primero es elegir el puerto de comunicación a través de la pestaña “Herramientas” de la parte superior y desde la opción de “Puerto”, se desplegará un menú en el que permitirá escoger el puerto en el que se haya conectado la placa de Arduino para que el ordenador pueda comunicarse con ella.

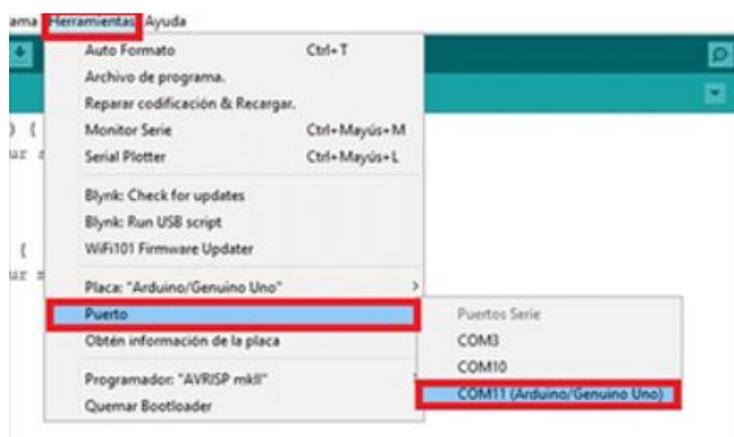


Figura 30. Escoger el puerto de la placa en Arduino IDE

Después de realizar todos estos pasos, el software de Arduino está lista para empezar a programar y construir proyectos con Arduino. Por ejemplo, si se es principiante con este software se puede acceder a la pestaña “Archivo” y elegir la opción “Ejemplos”. Por medio de ella, el software proporcionará un listado de códigos en forma de ejemplos para que cada uno se pueda guiar y estimular su aprendizaje y así poder practicar programación en Arduino desde cero.

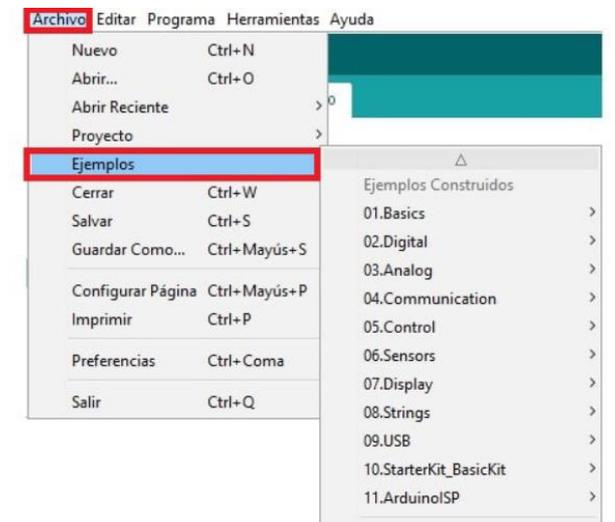


Figura 31. Pestaña con programas ejemplo en Arduino IDE

Para cargar el programa en la placa, es necesario pulsar sobre el botón “Verificar” y lo que se hará es una compilación del código y se informará si este tiene errores de programación. En el caso de que no existan errores, en la parte inferior de la pantalla del ordenador, se notificará los recursos de memoria usados. Después de esto, hay que hacer clic en el botón “Subir”, con esto el programa se descargará en la placa del microcontrolador.

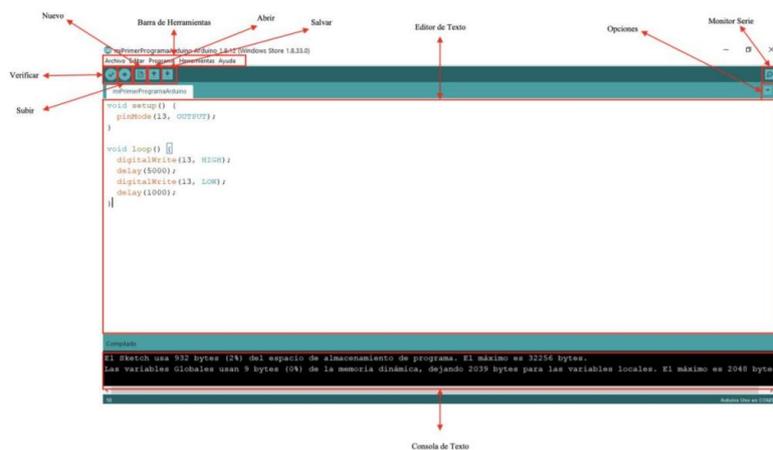


Figura 32. Interfaz Arduino IDE

(Esta página se deja en blanco a propósito)

- **Conexiones del sistema**

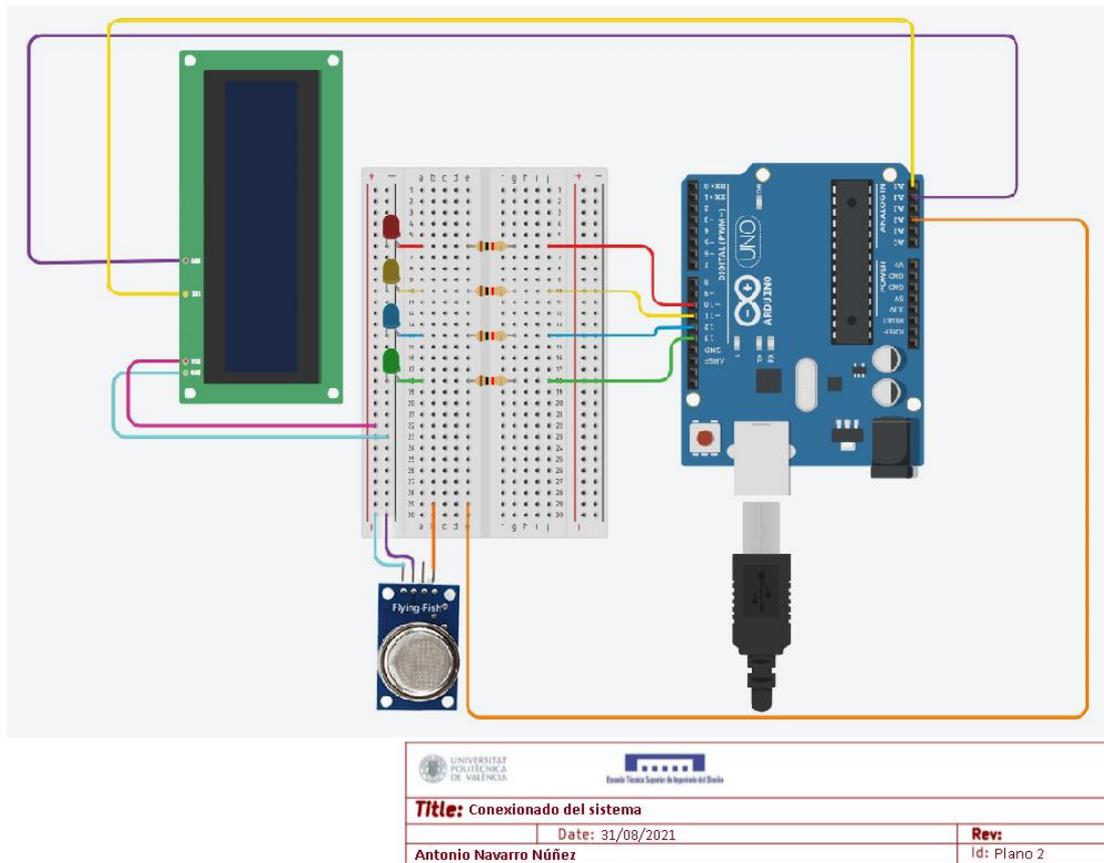


Figura 34. Conexionado del sistema

La figura 34 muestra todo el conexionado del sistema (Placa Arduino UNO, sensor MQ135 e interfaz de usuario). Como se puede observar se ha utilizado una placa protoboard para que las conexiones entre todos los componentes sean más fáciles de realizar.

Si se empieza a analizar la figura por la izquierda se encuentra la interfaz de usuario (pantalla LCD y diodos Led). La pantalla LCD posee cuatro terminales para su conexión. Dos terminales son los de alimentación (VCC y GND) que estos están conectados a la alimentación de todo el sistema que se encuentra en las dos columnas de la izquierda de la placa protoboard (línea roja positivo de la tensión 5 V DC y línea azul GND). Los otros dos terminales (SCL y SDA) son los terminales de comunicación de la pantalla LCD y estos se encuentran conectados en los pines analógicos 4 y 5 de la placa Arduino UNO. Los diodos Led están conectados en serie con una resistencia de 1 K Ω para que estos no sufran ningún daño. De ahí se conectan a las salidas 10, 11, 12 y 13 de la placa Arduino para tener una referencia visual según la calidad del aire que se tenga en el ambiente.

En la parte inferior se encuentra el sensor MQ135 que contiene cuatro pines, dos de alimentación (VCC y GND) y los otros dos son los pines de salida, una salida es de señal digital (D0) que no se utiliza en este proyecto ya que lo que interesa es saber el número de partículas que se tienen en el aire y la otra es una salida de señal analógica (A0), que es la que está conectada a la placa Arduino. El sensor está conectado a la placa protoboard y de ahí mediante cablecillos se conecta debidamente a la alimentación del sistema. De ahí también se conecta la salida de señal analógica del sensor al pin analógico 2 de la placa Arduino para su comunicación.

(Esta página se deja en blanco a propósito)

Pliego de Condiciones

1. Objeto

Este proyecto trata sobre el diseño e implementación física de un dispositivo el cuál se encarga de medir la calidad del aire en el interior de una oficina o un recinto cerrado.

En este Pliego de condiciones se darán una serie de especificaciones sobre todas las herramientas y componentes que han sido utilizadas en el montaje de todo el sistema explicando la función que tienen dentro de este la función que tienen dentro de este.

2. Herramientas

2.1. Ordenador Portátil MSI GE63 Raider RGB 8RF:

La función del ordenador portátil en este proyecto ha sido la de buscar información por Internet y poder almacenarla en él, así como para la redacción de esta memoria.

También se ha utilizado para realizar la escritura de todo el programa que se tiene que mandar a la placa de Arduino UNO, además, se ha realizado en él el diseño de la fuente de alimentación para poder simular si el funcionamiento era el correcto.

2.2. Placa Protoboard

La función de la placa protoboard es únicamente que los componentes estén debidamente colocados y que sea más fácil conectar todos entre sí para el correcto funcionamiento de todo el sistema ya que una mala conexión puede hacer que el sensor haga una mala lectura de la calidad del aire o que todo el sistema falle por una conexión indebida.

2.3. Fuente de alimentación

La fuente de alimentación se encarga de tener el sistema debidamente alimentado en todo momento que este esté en marcha. La fuente de alimentación está diseñada para repartir 5 V (DC) a todos y cada uno de los componentes sin importar el consumo de cada uno de estos, ya que ha sido diseñada para que nunca le falte potencia al sistema.

2.4. Placa Arduino UNO

Esta placa es el corazón de todo el proyecto. Se encarga de recibir toda la información del sensor de calidad de aire y además de interpretarla. Una vez que la placa recibe la información del sensor esta se encarga de mandar esta información a la interfaz de usuario para poder tener información visual sobre la calidad que se tiene donde el sistema esté colocado.

2.5. Interfaz de usuario

La interfaz de usuario es la encargada de dar una información visual de la calidad del aire a través de la pantalla LCD y de los diodos Led.

En la pantalla LCD se tendrá la información de lo cargado que está el aire de partículas ya que aparecerá un número en la pantalla según lo cargado que esté el ambiente.

Los diodos Led darán información a través de cuatro colores que son verde, azul, amarillo y rojo. Según la calidad del aire se encenderá uno u otro siendo el verde el mejor estado del aire, es decir, un nivel de aire limpio y el rojo se encenderá cuando el nivel del aire esté en las peores condiciones.

(Esta página se deja en blanco a propósito)

Presupuesto

En esta sección se va a detallar el presupuesto de todo el proyecto. Para ello se van a tener en cuenta diversos factores, por un lado, el gasto en personal, gastos en recursos hardware, gastos en recursos software y los gastos de todos los componentes del sistema.

Para calcular el gasto en el personal, se hará una estimación de las horas que se han gastado en cada parte del proyecto, así como el precio por hora que cobrará cada uno de los ingenieros que se necesitan para hacer todo el sistema. Para este proyecto se necesita un ingeniero electrónico que diseñe todo el sistema de alimentación y conexionado de los componentes. Por otra parte, también haría falta un ingeniero informático capaz de programar la placa Arduino para que funcione de la mejor manera posible.

Tabla 4. Presupuesto del personal

Personal	Horas	Coste (€/h)	Coste total (€)
Ingeniero electrónico	30	15	450
Ingeniero informático	40	14	560
TOTAL			1010

Por otro lado, hay que tener en cuenta el coste de todas las herramientas hardware que se han utilizado para este proyecto.

Tabla 5. Presupuesto herramientas hardware

Hardware	Unidades	Uso (%)	Precio Unidad (€)	Coste Total (€)
Ordenador Portátil MSI GE63 Raider RGB 8RF:	1	20	1500	300
Arduino UNO	1	100	25	25
Pantalla LCD	1	100	11	11
Sensor MQ135	1	100	6	6
Placa protoboard	1	100	10	10
Diodos Led y Resistencias	8	100	0.5	4
TOTAL				356

Ahora se va a calcular el coste de todas las herramientas software que se han empleado teniendo en cuenta el coste de las licencias de cada programa.

Tabla 6. Presupuesto herramientas software

Software	Coste licencia (€)	Coste total (€)
Arduino IDE	Gratuito	0
Proteus 8 Profesional	200	200
Office 365	69	69
TOTAL		269

Con todo esto ya se puede calcular el coste total de todo el proyecto.

Tabla 7. Suma total de los presupuestos

	Coste total (€)
Presupuesto del personal	1010
Presupuesto herramientas hardware	356
Presupuesto herramientas software	269
TOTAL	1635 €

A continuación, se mostrará una tabla del presupuesto final, en la cual se añade el coste del IVA y un beneficio del 10 %.

Tabla 8. Presupuesto total del proyecto

		Coste total (€)
Suma total de los presupuestos		1635
Beneficio	10 %	163.50
IVA	21 %	343.35
TOTAL		2141.85

Por lo tanto, el coste total del proyecto será de:

DOS MIL CIENTO CUARENTA Y UN EUROS CON OCHENTA Y CINCO CÉNTIMOS