

SISTEMA DE ADQUISICIÓN DE DATOS METEOROLÓGICOS PARA LA MEDICIÓN DE LA
CALIDAD DE AIRE A TRAVÉS DE ARDUINO.

YERSSON RAMIRO CARRILLO AMADO

UNIVERSIDAD DE PAMPLONA
FACULTAD DE INGENIERIAS Y ARQUITECTURA
MAESTRIA EN INGENIERIA AMBIENTAL PAMPLONA
NORTE DE SANTANDER

2020

SISTEMA DE ADQUISICIÓN DE DATOS METEOROLÓGICOS PARA LA MEDICIÓN DE
LA CALIDAD DE AIRE A TRAVÉS DE ARDUINO.

YERSSON RAMIRO CARRILLO AMADO

Trabajo de grado para optar al título de Magister en Ingeniería Ambiental

Director

Ph. D. QUÍMICA JACIPT ALEXANDER RAMON V.

UNIVERSIDAD DE PAMPLONA
FACULTAD DE INGENIERIAS Y ARQUITECTURA
MAESTRIA EN INGENIERIA AMBIETAL PAMPLONA
NORTE DE SANTANDER

2020

Dedicatoria

Con alegría e inmensa gratitud al DIOS todo poderoso, a Santo Tomas De Aquino que, en su ejemplo de humildad, rectitud de finalidad en el estudio y en el trabajo, me llevan cada día a mejorar como profesional y ser humano, a mis padres, María Evila Amado Soler y Ramiro Carrillo Angarita, a mi esposa Aida Luz Tapias Rodríguez e Hijo Samuel Andrés Carrillo Tapias por su gran aporte emocional para conseguir mis metas, a Jacipt Alexander Ramón por creer en mí y llevarme por la senda del conocimiento.

Agradecimientos

Mi profundo agradecimiento a todas las personas que iniciaron este proceso académico conmigo, por confiar en mí, abrirme las puertas y permitirme realizar todo el proceso investigativo, de igual manera mis agradecimientos a la Universidad de Pamplona, al programa de Maestría en Ingeniería Ambiental, a los profesores quienes con la sabiduría de sus valiosos aportes hicieron que pudiera crecer día a día como profesional, gracias a cada uno de ustedes por su , dedicación, apoyo absoluto y amistad.

Quiero manifestar mi más grande y sincero agradecimiento al Ingeniero Electrónico Miguel Ángel Califa Urquiza, principal colaborador durante todo este proceso, quien, con su dirección, conocimiento, y colaboración permitió el desarrollo de este trabajo, al M.Sc. Camilo Andrés Olarte Quiñonez, por sus amistad y asesoramiento investigativo, y a mis compañeros de clase.

Resumen

El proyecto de la estación meteorológica surgió de una búsqueda exhaustiva de documentos sobre la contaminación atmosférica y la vigilancia en los centros urbanos, dado el gran crecimiento de los automóviles y los parques industriales en los que se convierten en fuentes de contaminación atmosférica, con el propósito de crear un sistema de recopilación de datos meteorológicos para el control de la calidad del aire a través de una plataforma de código abierto llamada Arduino, mediante la aplicación del uso de hardware y software de código abierto en la creación de dispositivos que ayuden a vigilar e inspeccionar la calidad del aire.

La estación meteorológica es de bajo costo y útil porque toma variables atmosféricas y de gases transportados por el aire como temperatura, humedad, presión atmosférica, alcohol, CH₄, NH₄, CO₂, COV, latitud, longitud y fecha.

Para llevar a cabo la recolección de datos de los gases contaminantes, fue necesario utilizar sensores de gas, los cuales no estaban calibrados, por lo que se realizó una calibración de los componentes de los sensores de gas MQ, para ello se creó un modelo matemático lineal a través de la hoja de datos de los fabricantes, creando librerías para la serie MQ, las cuales son competentes e ideales para la recolección de datos sobre la calidad del aire, lo que lleva a la toma de decisiones, para la mejora de la calidad de vida en los centros urbanos.

Summary

The meteorological station project arose from an exhaustive search for documents on air pollution and monitoring in urban centers given the large growth of automobiles and industrial parks in which they become sources of air pollution, with the purpose of making a system for collecting meteorological data for air quality control through an open source platform called Arduino, through the implementation of the use of open source hardware and software in the creation of devices that help monitor and inspect air quality.

The weather station is low cost and helpful because it takes atmospheric and airborne gas variables such as temperature, humidity, air pressure, alcohol, CH₄, NH₄, CO₂, VOC, latitude, longitude and date.

To carry out the data collection of polluting gases, it was necessary to use gas sensors, which were not calibrated, so a calibration of the components of the MQ gas sensors was made, for this a linear mathematical model was created through the manufacturers data sheet, creating libraries for the MQ series, which are competent and ideal for data collection regarding air quality, which leads to decision making, for the improvement of quality of life in urban centers.

Contenido

	pag.
Introducción	17
1. Objetivos	20
1.1 Objetivo General.....	20
1.2 Objetivos Especificos	20
2. Estado del Arte	21
3. Marco Teórico	29
3.1 Marco Referencial.....	29
3.1.1 Calidad del Aire.....	30
3.1.1.1 Efectos de la calidad del aire sobre la salud.	30
3.2 Estaciones Meteorológicas	32
3.2.1 Componentes de un EMA.	32
3.2.2 Datalogger.	32
3.3 Sensores	33
3.4 Tipos de Estaciones Meteorológicas Automáticas	33
3.4.1 Estaciones meteorológicas automáticas de configuración cerrada.....	33
3.4.2 Estaciones meteorológicas automáticas de configuración abierta.....	34
3.5 Datos.....	34
3.6 Arduino	34
3.6.1 Elementos de la placa.	35
Fuente: Ingeniería Mecafenix, 2017.	35
3.6.2 Elementos del microcontrolador (Arduino Pro Mini).	35
3.6.3 Señales analógicas y digitales.	36

3.6.4 Elementos digitales:	37
Función Digital Write	37
Pines de salida	37
3.6.5 El lenguaje y entorno de programación.	38
3.6.6 Estructura general de un programa.	39
Función setup	39
Función loop.....	39
3.6.7 Comandos básicos.....	39
Variables.....	39
3.6.8 Estructuras de control se ven en la siguiente tabla	40
Bucle for.....	40
If (condición) { }	40
If else	40
3.6.9 Funciones.	41
3.6.10 Declaraciones de switch/case.	41
3.7 Sensores Arduino.....	41
3.7.1 Descripción general de BMP180.....	42
3.8 Operación	43
3.8.1 Descripción general.....	43
3.8.2 Funciones generales y esquemas de aplicación.	43
3.9 Sensor de Temperatura DHT11.....	44
3.9.1 Potencia y pin.....	45
3.9.2 Proceso de comunicación: interfaz serial (Single-Wire Two-Way).	45

3.10 Atención de la Solicitud	46
3.10.1 Condiciones de funcionamiento.	46
3.10.2 Efecto de la temperatura.....	46
3.10.3 Afecto de la Luz.....	46
3.10.4 Cables de conexión.....	46
3.10.5 U-blox 6 Módulo GPS.	47
3.10.5.1 Visión de conjunto	47
3.10.6 GPS asistido (A-GPS).....	48
3.10.6.1 Timepulse y referencia de frecuencia.	48
3.10.6.2 Marca de tiempo.	48
3.10.6.3 Navegación por estima.....	49
3.10.7 Posicionamiento preciso del punto.....	49
3.10.8 Antena.	50
Fuente: U-blox, 2011.....	50
3.10.9 Administración de energía.....	50
3.10.10 Modo de rendimiento máximo.	50
3.10.11 Modo eco.	50
3.11 Configuración.....	51
3.11.1 Configuración de arranque	51
3.11.2 Diagramas de tiempo SPI.....	51
Fuente: U-blox, 2011.....	51
Fuente: U-blox, 2011.....	51
3.12 MQ-135 Módulo Sensor de Calidad del Aire	52

3.13 Sensor MQ-4 Detector de Gas Natural y Metano Arduino	53
4. Marco legal.....	55
5. Metodología	57
5.1 Revisión Bibliográfica	58
5.2 Elección de Componentes	58
5.3 Diseño del Dispositivo.....	59
5.4 Elaboración del Dispositivo.....	59
5.5 Programación del Dispositivo	60
5.6 Calibración.....	60
5.7 Toma de Datos	61
6. Analisis y Duscusion de los resultados.....	62
6.1 Diseño y Empleo de Componentes Adecuados para Ejecutar una Estación Meteorológica que Mida la Calidad del aire por Medio del Micro-Controlador Arduino	62
6.2 Arduino Mega 2560 R3.....	64
6.3 Tarjeta SD.....	64
6.4 Sistema de geo Posicionamiento Global	64
6.5 Sensores	65
6.5.1 Sensor de presión bmp180 arduino.....	65
6.5.2 Sensor de temperatura y humedad DTH11	66
6.6 Resistencia	66
6.7 Sensor DHT11	67
6.8 Pruebas de Conexion Sensores y Modulo de GPS	69
6.9 Módulo Versión 2.0.....	75
6.10 Sensor DHT11	75

6.11 Sensor CCS811.....	76
6.12 Sensor BMP180.....	77
6.13 Sensor MQ135.....	77
6.14 Sensor MQ 4.....	78
6.15 Arduino Pro MINI 1.....	79
6.16 Módulo GPS U-blox Neo.....	80
6.17 Wemos D1.....	81
6.18 Nivelador Lógico de VOLTAJE 5v-3.3v Bidireccional.....	82
6.19 Wemos D1 Micro Sd Shield.....	82
6.20 Condensador Electrolítico de 400 UF A 16V.....	83
6.21 Resistencia.....	84
6.22 Prototipo.....	85
6.23 Pruebas de Conexion Sensores y Modulo Wemos d1 Micro sd Shield.....	86
6.23.1 Envío de datos a servidor externo.....	87
6.23.2 Base de datos.....	87
6.23.3 Proceso almacenamiento de datos.....	88
6.24 Programación.....	88
6.24.1 Código de arduino.....	88
6.24.2 Programación Tarjeta de Desarrollo Arduino Pro Mini 1.....	90
6.25 Calibración de Sensores.....	91
6.26 Modelado Matemático.....	93
6.26.1 Montaje de la biblioteca.....	99
6.26.2 Toma de datos confiables.....	100

6.27 PCB..... 101

6.28 Estación Final 103

6.29 Plataforma..... 106

7. Conclusiones 109

8. Recomendaciones 110

Bibliografía 111

Lista de Figuras

	pag.
Figura 1. Problema.....	21
Figura 2. Datalogger.....	33
Figura 3. Logo Arduino	35
Figura 4. Arduino pro mini.....	36
Figura 5. Salidas digitales	38
Figura 6. BMP 180.....	42
Figura 7. Circuito de aplicación típica	44
Figura 8. U-blox 6 Módulo GPS	47
Figura 9. Diagrama de tiempo SPI	51
Figura 10. Sensor MQ-135.....	52
Figura 11. Sensor MQ-4	53
Figura 12. Diagrama de metodología.....	57
Figura 13. Módulo de expansión SD Shield.....	63
Figura 14. Arduino Mega Protoshield.....	63
Figura 15. Arduino Mega 2560 R3.....	64
Figura 16. Tarjeta SD.....	64
Figura 17. GPS.....	65
Figura 18. Sensor BMP 180	65
Figura 19. DTH 11.....	66
Figura 20. Resistencia 4 bandas.....	66
Figura 21. Sensor DHT11	67

Figura 22. Sensor de Presión Bmp180	68
Figura 23. Ublox NEO-6M Módulo GPS	68
Figura 24. Conexión protoboard de sensores, módulo de Gps y placas	69
Figura 25. Sensores y modulo Gps acoplados al Arduino Mega A Protoshield	70
Figura 26. Módulo de expansión SD Shield.....	71
Figura 27. Componentes de la estación	71
Figura 28. Estación terminada	72
Figura 29. Peso de la estación.....	72
Figura 30. Sensor DHT11	76
Figura 31. Sensor CCS811	76
Figura 32. BMP180.....	77
Figura 33. Sensor MQ135	77
Figura 34. Sensor MQ 4.....	78
Figura 35. Arduino pro mini 1	79
Figura 36. Módulo GPS U-blox neo	80
Figura 37. Wemos D1.....	81
Figura 38. Nivelador de voltaje 5V-3.3V	82
Figura 39. 10 MICRO SD shield.....	82
Figura 40. Condensador de 400 UF A 16V	83
Figura 41. Resistencia	84
Figura 42. Prototipo de estacion 2.0	86
Figura 43. Protoboard de conexiones	87
Figura 44. Interfaz Arduino	89

Figura 45. Prueba con gas butano proveniente de un encendedor	91
Figura 46. Prueba con Aceite quemado.....	91
Figura 47. Datos obtenidos sin calibración adecuada	92
Figura 48. Espacios de muestreo de las especificaciones técnicas por fábrica	94
Figura 49. Líneas de tendencia para MQ-309.....	96
Figura 50. PCB placa de circuito impreso	102
Figura 51. PCB con Arduino mini y wemos	102
Figura 52. PCB con componentes de temperatura, GPS y voc.....	103
Figura 53. Estación terminada	104
Figura 54. Datos obtenidos por la estación visualizados en Excel.....	104
Figura 55. Datos tomados por la estación	105
Figura 56. Entorno de la plataforma	107

Lista de Tablas

	pag.
Tabla 1. Niveles máximos permisibles para contaminantes criterio	31
Tabla 2. Elementos de la placa.....	35
Tabla 3. Funciones Microcontrolador	37
Tabla 4. Estructura básica del programa.....	39
Tabla 5. Variables de Arduino	40
Tabla 6. Estructuras de control	40
Tabla 7. Especificaciones de la antena	50
Tabla 8. Tipos de señales del SPI.....	51
Tabla 9. Datos obtenidos por la estación 1.0.....	74
Tabla 10. Sensores MQ de la fábrica de Sparkun utilizados para calibración y pruebas	97
Tabla 11. Sensores de. Pololulu, Haoyuelectronics, Sensorsportal y HANWEI Electronics factories	98

Introducción

La Organización Mundial de la Salud (OMS) define que el aire limpio es un requisito básico para la salud y el bienestar humano, a su vez la OMS hace referencia que la contaminación sigue advirtiéndose una amenaza sustancial para la salud en todo el mundo (Salud, 2006). Se cree que el aire limpio y el medio ambiente son un requisito básico para la salud humana; teniendo en cuenta la introducción de tecnologías limpias en la industria, la producción, el transporte y la energía no han sido suficientes para mantener la calidad del aire.

La contaminación atmosférica sigue siendo un riesgo para la salud, estudios epidemiológicos recientes han demostrado que en Europa cientos de miles de muertes prematuras se atribuyen a la contaminación atmosférica (Organization, 2000).

La contaminación atmosférica, representa un grupo que, con otros factores, produce un gran número de padecimientos agudos y crónicos, asociados especialmente a problemas respiratorios y cardiovasculares, los cuales se desarrollan en la persona en función de su desarrollo orgánico, el tipo de contaminante y el tiempo de manifestación (IDEAM, Informe del Estado de la Calidad del Aire en Colombia, 2017). En el año 2015, sólo 17 de los 33 países de la Región Latinoamericana están registrando datos de la calidad del aire en cierta forma, este tipo de instrumentos de medición se acostumbran a ubicarse en las ciudades principales. Para ese mismo año, las ciudades que poseían los mejores sistemas de monitoreo de la calidad del aire se hallaban en Brasil, Chile, Colombia y México (Chile, 2018).

En Colombia el Instituto de Hidrología, Meteorología y Estudios Ambientales (IDEAM), es quien lleva la tarea de la ejecución de un Método de Vigilancia de Calidad del Aire en los municipios, a conjunto con las Corporaciones Autónomas Regionales y las Autoridades

Ambientales de los grandes asentamientos urbanos, con una organización para determinar el cumplimiento de las normas de calidad del aire y evaluar los riesgos para la salud humana y el ambiente de los colombianos (Sostenible, 2019).

El seguimiento de la calidad del aire se realiza bajo, es una unión de dispositivos de monitoreo de los contaminantes atmosféricos. Su ejecución se efectúa bien sea cuando hay una problemática específica y concerniente con la calidad del aire, en lugares con población por encima de los cincuenta mil habitantes (SIAC, 2018).

Los sistemas de red de sensores inalámbricos han sido utilizados por Raspberry Pi y Arduino para las aplicaciones de monitoreo ambiental; por consiguiente la tecnología de redes de sensores inalámbricos ha venido emergiendo como una solución viable, solución para muchas aplicaciones innovadoras (Ferdoush & Li, 2014). El aumento de estas investigaciones suele ser más pretendidas debido a que hay una gran variedad de sensores que son compatibles con el hardware y software del Arduino, que permiten delimitar las áreas a través de los registros de los datos climáticos presentes en una zona específica (Ferdoush & Li, 2014).

En el desarrollo del sistema de adquisición de datos meteorológicos para el control de la calidad del aire se necesita realizar una búsqueda de información referente a las plataformas de código abierto, específicamente Arduino, considerada como una herramienta libre que se ajusta a las necesidades y requerimientos para el dispositivo que se realizó, a través de la integración de sensores y componentes, los cuales se caracterizan por ser accesibles a la programación, creando librerías entorno a la calibración de los sensores, facilitando que este proyecto sea tenido en cuenta a la hora de tomar datos de la calidad del aire, además de tener bajos costos en los componentes.

Para realizar el dispositivo se tuvo en cuenta unos componentes específicos para ensamblar la estación meteorológica como fueron: el sensor DTH 11, BMP 180, sensores de gases de la serie MQ, GPS, y CSS811; luego se procedió a realizar la calibración de los sensores MQ, para ello se realizó los ajustes y acondicionamientos de los mismos. En el caso de los sensores de gases fue un trabajo dispendioso por qué se debió crear unas librerías de calibración porque existía un fundamento práctico para la calibración de cada sensor, para poder realizarla se tuvo en cuenta las hojas de datos provenientes de los fabricantes y tablas de concentraciones.

Posteriormente se propuso un modelo matemático dónde se hizo una relación de las variables, partes por millón a la relación de $RS/R0$ aportado por el fabricante, para poder realizar la ecuación matemática, donde se hallaron pendientes y líneas de tendencia, arrojando unos valores que permitieron construir las tablas, y por consiguiente fueron introducidos en las librerías que se hicieron y a continuación se llevaron al software de las bibliotecas de Arduino.

Después de haber resuelto la calibración de los sensores, se procedió a diseñar la PCB, a conjunto el ensamble de la estación y por último realizar un diseño de la caja para acoplar todo el componente electrónico y así tener nuestro dispositivo a la mano.

1. Objetivos

1.1 Objetivo General

Diseñar un sistema de adquisición de datos meteorológicos para la medición de la calidad de aire a través de Arduino, con el fin de registrar los gases contaminantes presentes en el aire.

1.2 Objetivos Especificos

Proponer un sistema de adquisición de datos meteorológicos para la medición de la calidad de aire a través de Arduino, que permita identificar los gases contaminantes presentes en el aire.

Ejecutar una metodología de calibración para los sensores MQ, basado en la información suministrada en la ficha técnica, utilizando una escala logarítmica que genere una ecuación lineal con el fin de calcular las variables de los sensores.

Validar el modelo matemático de calibración y estandarización para las mediciones de la calidad del aire a través de los sensores MQ, utilizando el modelo de regresión matemática lineal para crear las librerías de calibración de los sensores de la serie MQ.

Desarrollar un sistema de adquisición y plataforma de visualización de datos meteorológicos para la medición de la calidad del aire a través de Arduino.

2. Estado del Arte

Para desarrollar esta investigación se tuvo en cuenta referentes bibliográficos al enigma de la contaminación del aire, que permitieron conocer de cómo se lleva a cabo el seguimiento a esta inquietud con el fin de brindar una clara y seria propuesta que conlleve a la mitigación de la problemática de la contaminación del aire (figura 1).

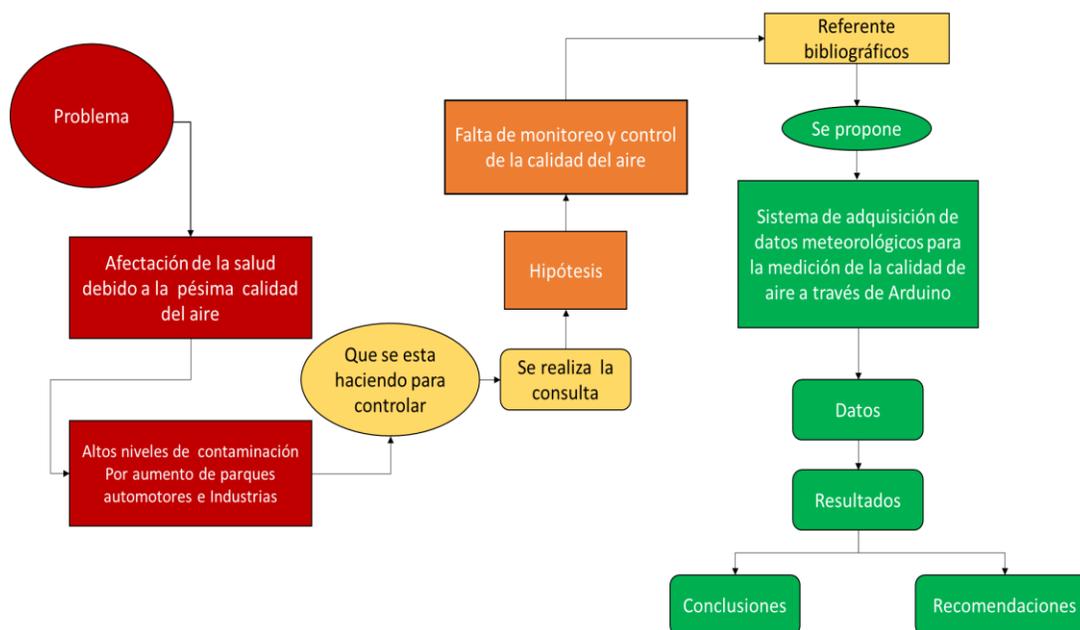


Figura 1. Problema

Fuente: Elaboracion propia.

A nivel internacional se resalta el trabajo realizado por (Romero , Diego, & Alvarez, 2006) donde manifiesta que uno de los grandes problemas de la salud hoy en día es la contaminación del aire, debido a los crecientes niveles de contaminantes en el mismo, a su vez (PNUMA, 2016) propusieron una plataforma donde los países de las regiones Asiáticas intercambian datos e indagaciones sobre sus inconvenientes y mejores prácticas del proceso de sus industrias y crecimiento demográfico, en torno a la solicitud de la asistencia técnica y la implementación de

un acuerdo climático de los países asiáticos entorno al medio ambiente y la calidad del aire.

De acuerdo a los logros y aciertos de esta región basada en la experiencia sobre el acuerdo climático y disminución de sus niveles de contaminación del aire, a través de un centro de coordinación para la inspección de la calidad del aire que facilita la regulación entre los órganos meteorológicos (PNUMA, 2016), como para tratar la contaminación atmosférica e incendios forestales que conllevan a operativos alrededor del monitoreo y evaluación de los riesgos que traen a conjunto la contaminación del aire.

Mientras tanto la (CEPAL, 2015) plantearon que el camino hacia el desarrollo sostenible y destaca que los países de la región de América Latina en su estructura productiva de ventajas comparativas estáticas deben mejorar sus componentes o sus prácticas sociales en torno al manejo y la calidad del aire. Esto es porque la relación entre la diversidad y la sostenibilidad ambiental es un reto que se propone hacia un futuro en América Latina y el Caribe, practicando un análisis de las ciudades en sus dinámicas de crecimientos de población, parques automotores y pequeñas industrias, que presentan un crecimiento de la contaminación de la atmósfera e incrementan las emisiones de contaminantes en el aire, por lo cual se hace vulnerable la salud de los habitantes de esta zona.

El patrón de consumo hace que la expansión de zonas de agricultura o de campo, crezcan de forma desordenada y se conviertan en suburbios, haciendo estas nada sostenibles, ahondado la destrucción del medio ambiente. Donde los medios de transporte son automóviles o motocicletas tradicionales, las cuales aumentan el consumo de los combustibles y por ello una afectación directa a la contaminación atmosférica.

En la actualidad en las grandes ciudades se le atribuyen a que ocurran afecciones en la salud de los seres humanos debido a los agentes contaminantes a causa del crecimiento de la industria petrolera, los servicios, la agroindustria y el incremento de unidades automotoras por lo cual hace un llamado a los países de América Latina, para que a través de un sistema de vigilancia ejerzan control sobre la calidad del aire, que les permita disminuir diferentes problemas de salud.

Para la organización Panamericana de la salud (SALUD, 2020), la contaminación atmosférica genera un importante riesgo para la salud de las Américas, al punto que la contaminación atmosférica ha crecido y hace que sea relevante y cause estragos en la salud, a su vez publica rutas para el manejo y cuidados de la salud, debido al material particulado que emiten pequeñas y grandes empresas, parques automotores y prenden alarmas ya que estas partículas pueden penetrar profundamente en los pulmones, atacando a las células de defensas de los mismos. En las Américas el crecimiento de la contaminación atmosférica ha aumentado las muertes, agravando los riesgos y efectos que han derivado enfermedades a todas las personas sin respetar edades con incidencias en edades previas como en niños menores de 5 años y adultos mayores 75 años que son los más afectados por la contaminación.

Los planteamientos y políticas que tiene la Organización Panamericana de la Salud para reducir la contaminación son oportunas pero los gobiernos internos, han sido débiles en las implementaciones, porque el crecimiento demográfico en América es desordenado y desbordado en los centros urbanos crecen en las periferias causando desorganización y por ende trae consecuencias en la salud.

Organización Mundial de la Salud (2018) plantea una hoja de ruta, estableciendo un programa de la calidad del aire, lanzando una estrategia en la cual la Organización Panamericana

de la Salud y la Organización Mundial de la Salud, entorno a un estudio que genera una relación entre la hidrosfera, la biosfera y la actividad humana, evidenciado el vínculo con los gases de efecto invernadero, los contaminantes primarios y secundarios, los cuales generan un impacto a la salud de los seres humanos, estos signos producen directamente muertes al año, que se le otorgan a la contaminación del aire.

Por su parte en Colombia Pérez, (2017), específicamente en Medellín en el año 2017, a través de un monitoreo realizado para determinar la calidad del aire, lo cual propicio que en Medellín se decretara alerta naranja, para evitar más casos alarmantes y así disminuir el riesgo de enfermedades ocasionadas por la mala calidad del aire. Dicho trabajo se genero debido a una gran carga de contaminantes en el aire que incurre directamente en la salud de las personas, de los animales y las plantas del Valle de Aburra, por lo cual fue necesario hacer un trabajo de monitoreo, realizando seguimientos por un determinado tiempo, arrojando resultados preocupantes para la salud, dado que estas emisiones, creaban deterioros en los pulmones de los habitantes, y aumentaron las patologías, manifestaciones clínicas asociadas a la contaminación del aire de esta zona de Colombia. Este trabajo comprobó que la mala calidad del aire ocasiona enfermedades como las pulmonares crónicas obstructivas (EPOC), cáncer de pulmón y otras afecciones que están directamente relacionadas con la emisión de gases tóxicos al aire.

La contaminación del aire es un asesino silencioso, pero cada vez más personas en distintas ciudades del planeta deciden hacer algo al respecto, por ello crean sistemas de control para los contaminantes atmosféricos (ONU, 2020), que son instalados en un lugares de interés con un intención determinada; estas instalaciones debe estar escoltada de todas las actividades necesarias para su mejor funcionamiento, intrínsecamente las cuales se puede llevar a cabo po personal calificado, un plan de mantenimiento preventivo y correctivo, un sistema de dirección de

información que permita la adecuada validación de los datos, entre otras (SIAC, 2020). Su ejecución se realiza bien sea cuando se muestra una problemática determinada y relacionada con la calidad del aire, o en zonas con población por encima de los cincuenta mil habitantes.

El Instituto de Hidrología, Meteorología y Asuntos Ambientales (IDEAM), a través de las Corporaciones Autónomas Regionales son los encargados de monitorear la calidad del aire, por ello tienen estaciones ubicadas solo en las grandes ciudades del país, lo que ha permitido establecer que hay una mayor contaminación ambiental, determinando que la calidad del aire en Colombia ha disminuido al igual que en el resto del mundo lo cual es preocupante para las condiciones de salud de los habitantes (IDEAM, 2020). A causa de que solo en las grandes ciudades y en puntos estratégicos, están ubicadas las estaciones para monitorear la calidad del aire, quedando así las ciudades pequeñas sin monitoreo, al no haber monitoreo, no hay control y por ende crece la contaminación por el desbordamiento de fuentes contaminantes, en cualquier lugar y dejando por fuera la salud de las personas.

A esta problemática de monitoreo en lugares pequeños, autores como (Ferdoush & Li, 2014), tras años de investigaciones lograron desarrollar una red de sensores inalámbricos, obteniendo una solución viable a través de la integración de hardware y software, siendo las plataformas de código abierto, como Arduino y raspberry PI, las más usadas, obteniendo un alto impacto debido a su bajo costo y gran variedad de sensores, por lo cual hicieron posible el monitoreo ambiental y recolección de datos a pequeña escala.

Para Pérez, Urdaneta & Custodio, (2014) estos autores habla de la incorporación de sensores electrónicos en la elaboración de sistemas de adquisición de datos y proponen métodos que contribuyen a reducir costos de orientación a este tipo de proyectos, brindando una solución más

conveniente al tratar de analizar la implementación independiente a la tecnología de investigación, plantearon un desarrollo de redes de sensores inalámbricos que requieren tecnologías en las áreas de la investigación de la comunicación, lo cual dan resultados que los nodos de sensores coordinados con interfaces gráficas gestionan información importante disminuyendo tiempo y costos de los datos a través de una red de computadoras tomando sistemas existentes para diseñar e implementar redes de sensores inalámbricos fundamentado tecnologías propias, que condiciona la red y los sensores brindada por el fabricante.

Siguiendo con los trabajos y desarrollo de plataformas inalámbricas para toma de datos meteorológicos aparecen (Quezada, Chango, Lopez, Pacheco, & Enciso, 2017), los cuales describen un diseño de sensores inteligentes basados en comunicación inalámbrica con la finalidad de facilitar la implementación, crear espacios inteligentes mediante un monitoreo y supervisión de las variables físicas ambientales, a su vez propusieron que esta red de sensores fuese un sistema autónomo compuesto de diminutos equipos con sensores y con capacidades de procesamiento extraordinario, con un reducido tamaño y capacidad de transmitir sin cables, permitiendo un despliegue rápido y flexible de centenares de dispositivos en el cual la arquitectura del software posee varias unidades de sensores, encaminados hacia la temperatura y la iluminación, estos a su vez llevan a un procesamiento hacia un microcontrolador, que sirve como unidad de comunicación a una radio de frecuencia, llegando a concluir que las tecnologías de redes de sensores inalámbricos puede fomentar el crecimiento y ser aceptadas para implementar en soluciones en la vida real donde se necesiten, aprovechar al máximo las bondades y beneficios que brinda esta tecnología; debido a que son tecnologías de código abierto y los componentes electrónicos se pueden encontrar muy fácilmente, además de ser de bajo costo haciendo que los proyectos sean mucho más fáciles de realizar generando desarrollos libres.

Cuando hablamos de bajo costo es que los componentes para realizar este tipo de proyectos son muy económicos, las tarjetas para el desarrollo como Arduino que es una plataforma de hardware libre, fundada en una placa con un microcontrolador y un ambiente de desarrollo (software), trazada para facilitar el uso de la electrónica en proyectos interdisciplinarios (ARDUINO, 2020). En electrónica se destacan los avances de los autores (Azúa, Vázquez, Arteaga, & Hernández, 2017) con un trabajo donde realizaron un sistema de adquisición de datos de bajo costo por la plataforma Arduino, además describieron la accesibilidad a las diferentes ramas tanto académicas como industriales e hicieron énfasis en los bajos costos de los componentes, por lo cual construyeron un sistema de adquisición con elementos económicos y confiables basados en la plataforma de Arduino.

Describen el dispositivo de placa de desarrollo Arduino Uno, utilizando sensores de temperatura, humedad e incluyeron un módulo de lector de tarjeta SD y memoria SD dónde se pueden consignar los datos, así mismo una serie de componentes necesarios para poder alimentar el sistema, como módulos de alimentación de 3.3V o 5V.

Se resalta que la programación del Arduino no requiere una computadora de altas prestaciones para desarrollar el software y abordan la red de manera gratuita, a su vez los repositorios de librerías dadas por Arduino son de libre acceso. Destacan también que se puede construir un sistema de adquisición de datos de bajo costo basados en la plataforma Arduino con un conocimiento básico de programación y electrónica, utilizando diferentes dispositivos y componentes en el mercado a precios accesibles recalcando el trabajo que se puede desarrollar en varias extensiones por abordar ramas de la investigación donde se tengan interés de monitorear variables climáticas.

A esto sumado de la gran contribución y espacio que ha generado las plataformas de código abierto han permitido tener un gran avance; en Colombia los estudios no se han quedado atrás es así que (Ochoa, Denisse, & Delgado, 2018) preocupados por el mejoramiento de la calidad de vida de las personas, presentan un marco teórico que sustenta un prototipo, descripción y arquitectura de una técnica de medición y monitoreo de ciertas variables ambientales afines a la calidad del aire. Contribuyeron con el soporte de información pertinente para la toma de decisiones gracias a los aportes de los datos de los sensores y mediciones de los dispositivos desarrollados bajo los lineamientos del código abierto teniendo como objetivo establecer acciones, que permitan mejorar la calidad de vida.

Aproximado a nuestra idea (Pinzon, Castillo, Gonzalez, Arauz, & Villareal, 2018) presentan un dispositivo para áreas industriales y domésticas, con el objetivo de usar los sensores para detectar sustancias y partículas contaminantes en el aire o en el medio ambiente, ejecutando las tomas de partes por millón del gas de dióxido de carbono a través del Hardware de Arduino uno y con su propio software, en dicho trabajo además de ocuparse con la plataforma de Arduino utiliza un sensor MQ135, el cual se programa para que detecte gases de dióxido de carbono, su esencia es prevenir o mitigar la contaminación en espacios, exponiendo como conclusión que el sistema detectó las variaciones de índice de CO₂ en un ambiente cerrado siendo así muy útil para detectar niveles peligrosos de dióxido de carbono destacan también la sencillez del diseño y la accesibilidad del dispositivo.

3. Marco Teórico

3.1 Marco Referencial

La necesidad de concebir y comprender el mundo que nos rodea, es un reto de encontrar y buscar herramientas, técnicas, conocimientos, y demás, que, gracias a la investigación, permitan de la forma más vigorosa de extraer y manejar la información real en el tiempo, que sea concreta y medible a la vez (Loza, 2013).

El avance de la tecnología ha favorecido en sobremanera la forma como el hombre experimenta el territorio, de esta manera, la recolección de datos es un instrumento, que surge y se robustece, perfeccionando a otras ciencias, pues proporciona la producción y manejo de la información, sustentado con datos medibles, los estudios y lo cual permite crear una comprensión objetiva del entorno que envuelve a las colectividades.

Si observamos detalladamente el progreso de la meteorología y de las estaciones meteorológicas logramos afirmar sin vacilaciones que estos últimos 20 años ha sido un fortalecimiento de esta especialidad en la cual aún seguimos en aumento. Con relación a la meteorología solo hay que ver como los pronósticos de tiempo se han prolongado en la época y cada vez son más exactos, todo ello gracias al progreso de la tecnología: procesamiento de datos, satélites con mayor eficiencia, perfeccionamiento de modelos más veraces y una red más extensa y más precisa de estaciones meteorológicas dispersas por toda la tierra.

Gracias a la adquisición de datos, hoy en día brindan ayuda para mejorar los pronósticos, comunicar sobre escenarios y fenómenos locales desfavorables, notifican sobre la meteorología a nivel local en tiempo local.

3.1.1 Calidad del Aire

La contaminación atmosférica ha aumentado en relevancia, debido a que, como muestra las cifras de la Organización Mundial de la Salud, que, por una de cada ocho fallecimientos ocurridos a nivel mundial, es causada por la contaminación del aire ((OMS), 2014). A nivel, colombiano, el Departamento Nacional de Planeación valoró que, durante el año 2015(DANE), los efectos de esta singularidad se asocio a 10.527 muertes y 67,8 millones de sintomatologías y enfermedades. Opcionalmente, los costos ambientales incorporados a la contaminación atmosférica en Colombia, durante los últimos años han aumentado las enfermedades, pasando de 1,1% del PIB de 2009 (\$5,7 billones de pesos) a 1,59% del PIB de 2014 (\$12 billones de pesos) y del 1,93% del PIB en 2015 (\$15.4 billones de pesos), lo cual sitúa en certeza la necesidad de seguir efectuando estrategias para inspeccionar, evaluar y monitorear estos componetes presentes en el aire (IDEAM, 2015).

3.1.1.1 Efectos de la calidad del aire sobre la salud. El daño de la calidad del aire, ya sea por causas antrópicas o naturales, tiene efectos perjudiciales sobre la salud humana y los ambientes, aumenta la escala global que favorece al cambio climático. Las causas antropogénicas son las que hoy tienen más incidencias negativas elevadose en los últimos períodos del siglo. Aportan a la atmósfera contaminantes como: dióxido de carbono (CO₂), dióxido de azufre (SO₂), monóxido de carbono (CO), ácido sulfhídrico (H₂S), monóxido de nitrógeno (NO), ozono (O₃) troposférico, amoníaco (NH₃), dióxido de nitrógeno (NO₂), y partículas proporcionadas de constitución muy heterogéneas (compuestos inorgánicos, metales, compuestos orgánicos persistentes, compuestos orgánicos volátiles) (Valls, 2017).

En la siguiente tabla (tabla 1) se muestran los niveles válidos para contaminantes en Colombia esta norma se presenta para los niveles máximos legales a condiciones de referencia para contaminantes presentes en el aire , los cuales se deducen con el promedio geométrico para PST y promedio numérico para los restantes contaminantes (IDEAM, 2008).

Tabla 1. Niveles máximos permisibles para contaminantes criterio

CONTAMINANTE	UNIDAD	LÍMITE MÁXIMO PERMISIBLE	TIEMPO DE EXPOSICIÓN
PST	$\mu\text{g}/\text{m}^3$	100	Anual
		300	24 horas
PM ₁₀ *	$\mu\text{g}/\text{m}^3$	70	Anual
		150	24 horas
SO ₂	ppm ($\mu\text{g}/\text{m}^3$)	0.031 (80)	Anual
		0.096 (250)	24 horas
		0.287 (750)	3 horas
NO ₂	ppm ($\mu\text{g}/\text{m}^3$)	0.053 (100)	Anual
		0.08 (150)	24 horas
		0.106 (200)	1 hora
O ₃	ppm ($\mu\text{g}/\text{m}^3$)	0.041 (80)	8 horas
		0.061 (120)	1 hora
CO	ppm (mg/m^3)	8.8 (10)	8 horas
		35 (40)	1 hora

Nota: mg/m^3 o $\mu\text{g}/\text{m}^3$: a las condiciones de 298.15°K y 101.325 KPa. (25 °C y 760 mm Hg).
 * El límite máximo permisible anual de PM₁₀ en el año 2009 será 60 $\mu\text{g}/\text{m}^3$ y en el año 2011 será 50 $\mu\text{g}/\text{m}^3$.

3.1.2 Monitoreo del aire. Se ejecuta con un conjunto de equipos de monitoreo para los contaminantes atmosféricos, se sitúan en lugares de interés con una intención explícita; estos acoplamientos se acompañan de todas las acciones necesarias para su adecuado funcionamiento, dentro de las cuales se logra mencionar la maniobra por el personal calificado, programas de sostenimiento provisorio y correctivo, un sistema de información que permita una correcta certificación de los datos, entre otras (Colombia, 2018).

3.2 Estaciones Meteorológicas

Se concibe como estación meteorológica el área donde se hacen observaciones y mediciones precisas de las diferentes medidas meteorológicas usando instrumentos adecuados, con el fin de seguir el comportamiento atmosférico en las diferentes zonas (IDEAM, 2014).

3.2.1 Componentes de un EMA. Estación meteorológica que por respaldo y el uso de un dispositivo computarizado (datalogger) y sus oportunos sensores electrónicos, registra en forma metódica y programada la toma de antecedentes meteorológicos de acuerdo a la utilidad de su operador y a la capacidad del mecanismo (Ramírez & Ávila, 2017).

3.2.2 Datalogger. Es un sensor de datos (datalogger) es un módulo electrónico que reconoce datos en el tiempo o en relación al sitio por medio de herramientas y sensores propios o acoplados externamente. Casi todos están fundados en microcontroladores. Por lo general son pequeños, con pilas, llevaderos, y provistos de un microprocesador, memoria interna para almacenamiento de datos y sensores. Algunos registradores de datos informan con una computadora personal y utilizan software determinado para activar el registrador de datos, ver y examinar los datos almacenados, mientras que otros poseen un dispositivo de interfaz local (teclado, pantalla LCD) y puede ser manejado como un módulo independiente ver (figura 2) (www.finaltest.com.mx, 2019).



Figura 2. Datalogger

Fuente: Final Test, 2020.

3.3 Sensores

Son aparatos que, de acuerdo a su fabricación y a los elementos electrónicos utilizados, son idóneos para convertir una determinada variable climática en un impulso eléctrico que puede ser registrado interpretado y expuesto como un valor, para cada una de las variables que se quiere monitorear (Ramírez & Ávila, 2017). Las variables que más usualmente se envuelven en estas estaciones son: temperatura, precipitación, humedad relativa, presión barométrica, radiación, dirección y velocidad del viento.

3.4 Tipos de Estaciones Meteorológicas Automáticas

3.4.1 Estaciones meteorológicas automáticas de configuración cerrada. Se incluyen en este grupo a aquellas estaciones que se instalan de un determinado tipo y número de sensores determinados por el constructor de la EMA (Agropecuarias, 2006).

3.4.2 Estaciones meteorológicas automáticas de configuración abierta. Considera aquellas estaciones que reconocen y permiten a los sensores de acuerdo a lo que se desea registrar en variables. Si se dispone de las competencias es posible que el usuario final puede agregar sensores construidos por el mismo de acuerdo a las especificaciones del Datalogger (Agropecuarias, 2006).

3.5 Datos

Corresponde al valor recopilado de los datos y que representa el comportamiento real de la variable que se mide. Para ello es necesario que cada uno de los sensores esté debidamente calibrado y, además, sea chequeado en forma permanente y sistemática (Ministerio de Ambiente, 2008). Solo en esa condición se pueden afirmar que el valor deja de ser un número y se convierte en un dato, de tal manera que cuando el valor de la variable temperatura indica, se puede afirmar que la temperatura medida por el sensor en ese momento es realmente el oportuno.

3.6 Arduino

Arduino es un instrumento que sirve para hacer que los computadores puedan concebir y controlar el mundo físico a través de un computador personal (figura 3). Es una plataforma de adelanto en computación física (physical computing) de código abierto, establecida en una placa con un sencillo microcontrolador y un ambiente de desarrollo para crear software (programas) para la placa. El Arduino se puede usar para crear objetos unidireccionales, leyendo datos de una gran variedad de interruptores, sensores y registrar multitud luces, motores y otros actuadores físicos (ARDUINO, <http://arduino.cl>, 2019). Arduino es una plataforma de código independiente de bajo precio basada en una placa de entrada-salida y en un entorno de perfeccionamiento, el cual se puede usar para desarrollar objetos interactivos automáticos o enlazar al software del

computador (Herrador, 2009).



Figura 3. Logo Arduino

Fuente: Arduino Community, 2020.

3.6.1 Elementos de la placa. Es una placa de circuito impreso donde va situado el la memoria , microprocesador, las conexiones de entrada/salida y la conexión para el puerto USB. En la siguiente tabla se muestra los diferentes elementos e indicaciones para el correcto uso de la placa com se observa en la siguiente tabla se aprecia los elementos y funciones (tabla 2).

Tabla 2. Elementos de la placa

Botón de Reset	Permite resetear el programa y admite cargar uno nuevo.
Puerto USB	A través de él se atribuyen las instrucciones a establecer, el programa que es ejecutado en el entorno de programación de Arduino. Comunicación Arduino-Ordenador.
Microcontrolador	Hace una ejecución (almacenadas en el programa) que le fueron anticipadamente programadas de forma constante. Es un circuito integrado que sujeta muchas de los mismos caracteres que una computadora.
Pines de entrada y salida	Conectan los elementos que transmiten información o realizan una acción (Lectura-Escritura).

Fuente: Ingeniería Mecafenix, 2017.

3.6.2 Elementos del microcontrolador (Arduino Pro Mini). El Arduino Pro Mini es un tablero para microcontroladores establecido en el Atmega328 que es una versión de un

chip microcontrolador que tiene 14 pines de entrada / salida digital (de los cuales 6 se pueden usar como salidas PWM) ver (figura 4), un resonador incorporado, 6 entradas analógicas, un botón de recuperación y orificios para los cabezales de los pines del ensamble. Se puede conectar un cabezal de seis pines a un cable FTDI o a una placa de encendido Sparkfun para suministrar alimentación USB y comunicación a la placa (ARDUINO, store.arduino.cc, 2019).

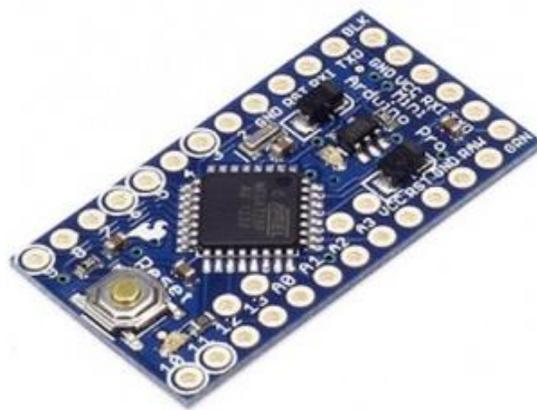


Figura 4. Arduino pro mini

Fuente: SparkFun Electronics, s.f.

3.6.3 Señales analógicas y digitales. Una señal es analoga cuando puede tomar muchos valores entre su valor mínimo y máximo, siendo sus elementos analógicos.

Potenciómetro, ldr,

NTC,

Zumbador,

Motor eléctrico,

En cambio, una señal es digital cuando solo puede tomar 2 valores, el máximo asociado a 1 o a cierto “on” y el mínimo asociado a cero, falso o “off”.

3.6.4 Elementos digitales: Botón, detector de presencia, led, timbre.

En la siguiente tabla se muestran las funciones elementales del microcontrolador ver (tabla 3).

Tabla 3. Funciones Microcontrolador

Función Digital Write	Write proviene del inglés y significa escribir, en este caso escribir en un puerto un valor digital que solo podrá ser 0 (LOW) y 1(HIGH) lógicos.
Pines de salida	High: el microcontrolador traza un 1 en el pin de salida, es decir 5v. El led se prende. Low: el microcontrolador traza un 0 en el pin de salida, es decir 0v. El led no se prende como se muestra en la figura 2.

Fuente: Elaboracion propia.

Con las salidas digitales Arduino Mega, Uno, Leonardo o el que sea, lograremos interactuar con el mundo físico de millares de formas, logramos mandar una señal de dos etapas (1 o 0) es decir, encender o apagar.

Entonces con esto lograremos encender y apagar luces, motores, relés, leds, actuadores, etc. En la siguiente imagen se observa el entorno de la placa ver (figura 5).

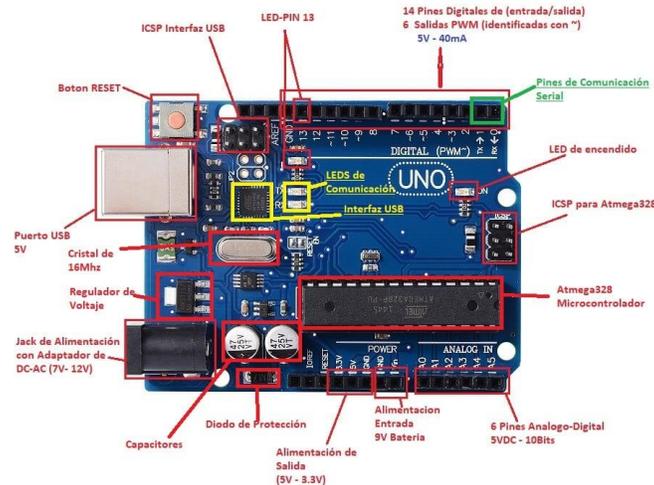


Figura 5. Salidas digitales

Fuente: Control Automatico Educacion, s,f.

3.6.5 El lenguaje y entorno de programación. Para la comunicación con el microcontrolador y la asignación de los programas se usa IDE (Integrated Development Environment), es un método de desarrollo de Arduino, que usa para escribir programas, reunir (y descargar los programas a la placa de Arduino). Un programa es una serie de instrucciones que son realizadas por un computador o un microcontrolador (Pomares, 2009), es una serie específica para realizar un trabajo. Los programas son escritos en prototipos heterogéneos de lenguajes, tales como “C”, Fortran o BASIC. El programa (lenguaje de alto nivel, “humano”) se reúne (se transforma en 1 y 0) y se carga en el microprocesador que lo elabora de forma cíclica. Para programar el microcontrolador se utiliza el IDE de Arduino, en él se escribe un programa que se carga en el microcontrolador mediante el puerto USB (Pomares, 2009).

3.6.6 Estructura general de un programa. En el encabezado se declaran todas las variables que van a ser utilizadas en el programa. El IDE de Arduino nos permite asociar nombres a números, por ello es de gran utilidad a la hora de transformar los programas y además de ello tiene una estructura funcional, así se define en la siguiente (tabla 4).

Tabla 4. Estructura básica del programa

Función setup	
<i>Void setup</i>	{ } es la función de distribución de los pines de Arduino y sólo se hace una vez, para preparar el microcontrolador antes de comenzar la ejecución del programa.
Función loop	
<i>Loop</i>	() hace una y otra vez incluso apague el sistema, o se malgasten las baterías deteniendo el bucle perene en el cual se encuentra el microcontrolador, loop es una palabra de la lengua inglesa que traduce ciclo.

Fuente: SparkFun Electronics, s,f.

3.6.7 Comandos básicos. Como se muestra a continuación:

HIGH - 5V - TRUE (Verdadero) - 1 lógico

LOW - 0V - FALSE (Falso) - 0 lógico

Variables

En la siguiente (tabla 5) se describen las variables de Arduino.

Tabla 5. Variables de Arduino

Char	Este ejemplo de datos char es un tipo rubricado, lo que significa que codifica números desde -128 a 127. Para un tipo de datos sin código, un byte (8 bits), utilice el tipo de datos byte.
Byte	Un byte acumula un número sin signo de 8 bits, de 0 a 255
Int	Enteros son un ejemplo de datos principales que permiten acumular valores numéricos sin fracciones comprendidos en el rango desde -32,768 a 32,767
Unsigned int	Son los mismos que ints en que acumulan un valor de 2 bytes. En término de almacenar números negativos, sin embargo, sólo acumulan valores positivos, dando una clase útil de 0 a 65.535 ($2^{16} - 1$).
Long	El tamaño de variable numérica de ejemplo extendido se refiere a números enteros (tipo de bits) sin decimales que se hallan dentro del rango - 2.147.483.648 a 2.147.483.647.
Unsigned long	Son variables de dimensiones extendidas para el almacenamiento de números y acumulan 32 bits (4 bytes). A diferencia de los longs estándar, los unsigned longs no acumulan números negativos, forjando que su rango sea de 0 a 4.294.967.295 ($2^{32} - 1$).
Float	La forma del dato de tipo “punto flotante” se emplea a los números con decimales. Los números de lugar flotante tienen un mayor valor que los de 32 bits con un rango interpretado $3.4028235E +38$
Double	La aplicación par es exactamente igual que el flotador, sin margen de precisión, ocupa 4 bytes.

Fuente: Arduino, 2017.

3.6.8 Estructuras de control se ven en la siguiente tabla. Como se muestra a continuación en la (tabla 6):

Tabla 6. Estructuras de control

Bucle for	(Condición primera; condición final; incremento) es un orden para repetir la misma maniobra de un cierto número de veces abordando desde la condición inicial, terminando en la condición final e aumentando de acuerdo a la variable de ampliación.
If (condición) {}	Es un comando que sirve para separar si se dio una determinada situación. Las comparaciones son: == igualdad != desigualdad, > mayor que, < menor que, >= mayor o igual que, y <= menor o igual que. Todo lo que conforme entre las llaves será realizado sólo si se da la condición que se halla dentro del paréntesis.
If else	Comprueba si se cumple una situación y elabora lo que está entre llaves, sino se efectúa ejecuta lo que está debajo del else.

While	Hace el conjunto de instrucciones entre llaves mientras se efectúe la condición.
--------------	--

Fuente: <https://elvex.ugr.es/decsai/c/apuntes/control.pdf>

3.6.9 Funciones. El código de segmentación en funciones permite a un programador crear piezas de código modulares que realizan una tarea definida y luego regresan al área de código desde la cual se llamó la función. El caso típico para crear una función es cuando uno necesita realizar la misma acción varias veces en un programa (Vega, Santamaría, & Rivas, 2014).

Hay dos funciones necesarias en un boceto Arduino, `setup ()` y `loop ()`. Otras funciones deben ser creadas fuera de los corchetes de estas dos funciones.

3.6.10 Declaraciones de switch/case. Al igual que si las declaraciones, `switch/case` inspecciona el flujo de los programas, admitiendo a los programadores detallar diferentes códigos que deben ejecutarse en diversos escenarios. En particular, una instrucción `switch` compara el valor de una variable con los valores especificados en los fallos de `case`. Cuando se encuentra una instrucción `case` cuyo valor concuerda con el de la variable, se ejecuta el código en esa instrucción `case`.

El término clave `break` deja la instrucción `switch` y normalmente se usa al final de cada proceso. Sin una instrucción `break`, la sentencia `switch` seguirá haciendo las siguientes expresiones ("falling-through") hasta que se llegue a una interrupción o al final de la instrucción `switch` (Arduino, 2017).

3.7 Sensores Arduino

Un sensor es un módulo capaz de detectar dimensiones físicas o químicas, llamadas variables de instrumentación, y convertir en variables eléctricas.

Las variables pueden ser, por ejemplo: temperatura, distancia, presión, fuerza, intensidad, lumínica, aceleración, humedad, movimiento, inclinación, desplazamiento, torsión, pH, etc.

Una dimensión eléctrica puede ser una resistencia eléctrica (como en una RTD), una capacidad eléctrica (como en un sensor de humedad o un sensor capacitivo), una tensión eléctrica (como en un termopar), una corriente eléctrica (como en un fototransistor), (Aprendinedo, 2018).

Los sensores se pueden clasificar en función de los datos de salida en:

3.7.1 Descripción general de BMP180. El BMP180 es un sensor compatible a su antecesor de la BMP085 y es una nueva generación de sensores digitales de alta precisión para aplicaciones de consumo (ver figura 6).

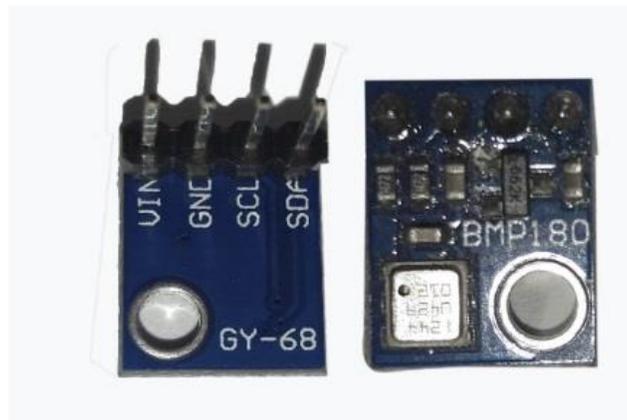


Figura 6. BMP 180

Fuente: Elaboracion propia.

El sensor BMP180 está optimizado para su uso en teléfonos móviles, gracias a su ultra-baja potencia, también aplicable a dispositivos de navegación GPS y equipos al aire libre. Con un bajo nivel de ruido. Además, el BMP180 ofrece un rendimiento superior y la interfaz de

comunicaciones I2C permite integración de sistemas con un microcontrolador (Sensortec, 2013).

El BMP180 está basado en tecnología piezo-resistiva para robustez EMC, alta precisión y linealidad como, así como la estabilidad a largo plazo; Robert Bosch es líder en el mercado mundial de sensores de presión en aplicaciones automotrices. Bosch posee experiencia de más de 400 millones de sensores de presión en el campo y el BMP180 continúa siendo un ejemplo de sensores de presión micro-mecanizados.

3.8 Operación

3.8.1 Descripción general. El BMP180 está planteado para conectarse directamente a un microcontrolador por medio de los pines de comunicación que sigan el protocolo I2C, los datos de presión y temperatura deben ser transmitidos desde la memoria EEPROM integrada en el sensor BMP180.

3.8.2 Funciones generales y esquemas de aplicación. El BMP180 consta de un sensor piezo-resistivo, un convertidor analógico a la digital y una unidad de control Con EEPROM y una interfaz serie I2C. El BMP180 entrega el valor no compensado de la presión y la temperatura. La EEPROM ha almacenado 176 bits de datos de calibración individuales pregrabados. Esto se utiliza para compensar el desplazamiento, la dependencia de la temperatura y otros parámetros del sensor como se muestra en la (figura 7).

UP = datos de presión (16 a 19 bits)

UT = datos de temperatura (16 bits)

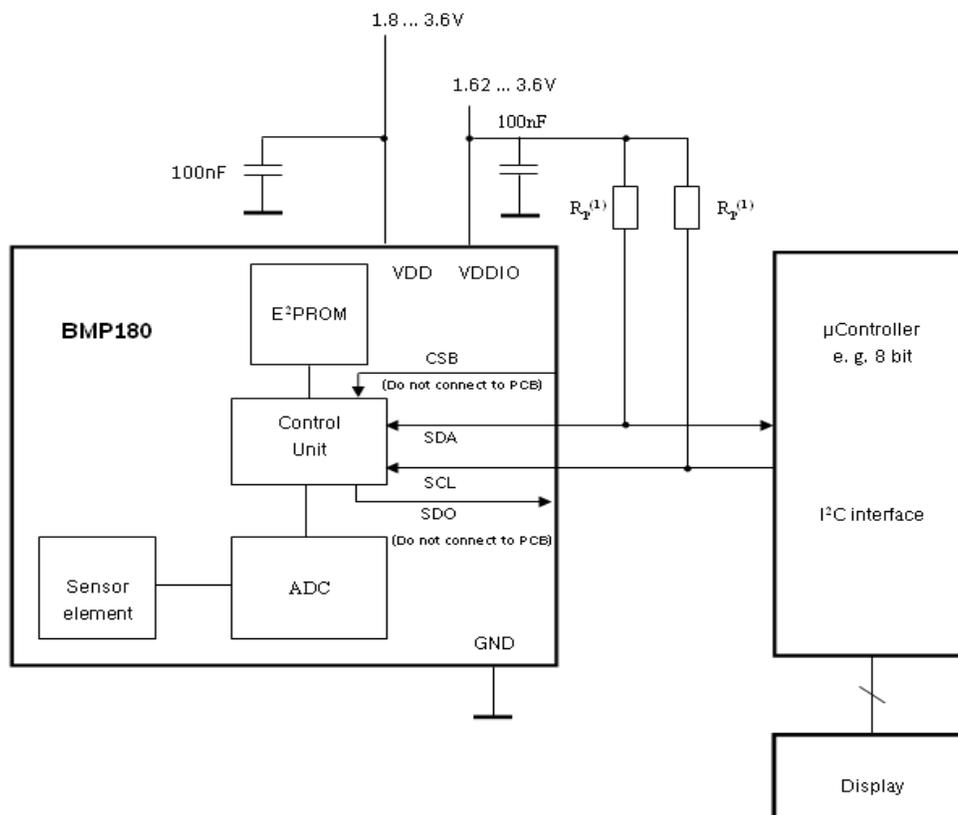


Figura 7. Circuito de aplicación típica

Fuente: BMP180 Data sheet, s,f.

3.9 Sensor de Temperatura DHT11

Cada elemento DHT11 está ajustadamente graduado en el laboratorio que es extremadamente preciso en la calibración de humedad. Los factores de calibración se almacenan como programas en la memoria OTP, que son manejados por el proceso de detección de señal interna del sensor (UK, 2010). La interfaz en cadena de un solo hilo hace que la integración del sistema sea rápida y fácil. Su pequeño tamaño, bajo consumo de energía y transmisión de señal de hasta 20 metros lo convierten en la mejor opción para múltiples aplicaciones, incluso las más exigentes como se observa en la figura 6. El componente es de 4 pines de una sola hilera de paquetes.

Cuando el cable de conexión es de menos de 20 metros, se recomienda una resistencia de 5K pull-up; Cuando el cable de conexión tenga más de 20 metros, elija una resistencia pull-up apropiada según sea necesario, en la siguiente imagen se observa el diagrama de aplicación típica ver (figura 8).

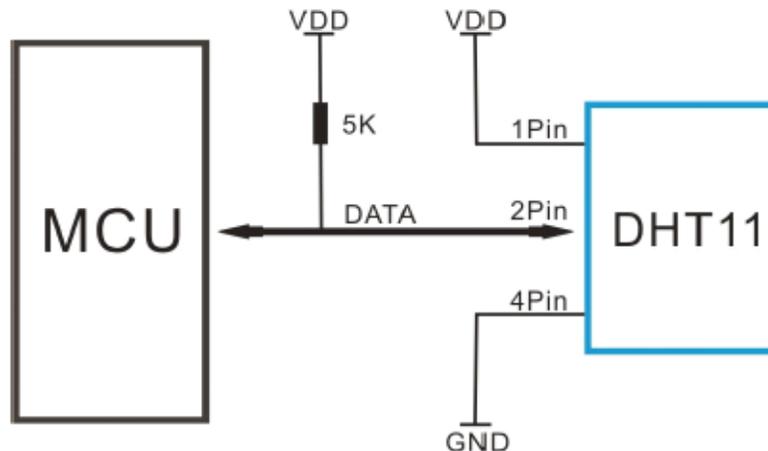


Figura 8. Aplicación típica

Fuente: DHT11 Humidity & Temperature Sensor, s.f.

3.9.1 Potencia y pin. La fuente de alimentación del DHT11 es 3-5.5V DC, inmediatamente se suministra energía al sensor se debe esperar alrededor de un segundo para entrar en el estado estable, un condensador con valor de 100nF (nano Faradios) es conveniente conectar entre VDD y GND para el filtrado de la señal.

3.9.2 Proceso de comunicación: interfaz serial (Single-Wire Two-Way). El formato de datos de un solo cable se utiliza para la comunicación y sincronización entre el sensor MCU y DHT11. Un proceso de comunicación es de 4 ms (milisegundos). Los datos consisten en partes decimales y partes integrales. Una transmisión de datos completa es de 40 bits, y el sensor envía

un bit de datos más alto primero también llamado BIT de parada.

Formato de datos: 8 bits datos RH integrales + 8 bits decimales RH datos + 8 bits datos T integrales + 8 bits decimales T datos + 8 bits de verificación suma. Si la transmisión de datos es correcta, la suma de comprobación debe ser la última 8bit de "8bit datos RH integrales + 8bit decimal RH datos + 8bit integral T datos + 8bit decimales T datos = 40 bits".

3.10 Atención de la Solicitud

3.10.1 Condiciones de funcionamiento. La aplicación del sensor DHT11 más allá de su rango de trabajo indicado en esta hoja de datos puede dar como resultado un cambio/discrepancia de señal de 3% de HR. El sensor DHT11 puede recuperar gradualmente el estado calibrado cuando vuelve a la condición de funcionamiento normal y trabaja dentro de su rango. Se debe tener en cuenta que el funcionamiento del sensor DHT11 en condiciones de trabajo no normales acelerará el proceso de envejecimiento del sensor.

3.10.2 Efecto de la temperatura. La humedad relativa depende en gran medida de la temperatura. Aunque la tecnología de compensación de temperatura se utiliza para asegurar una medición precisa de la humedad relativa, se recomienda mantener los sensores de humedad y temperatura trabajando bajo la misma temperatura. DHT11 debe ser montado en el lugar lo más lejos posible de las partes que pueden generar calor.

3.10.3 Afecto de la Luz. La exposición prolongada a luz solar fuerte y ultravioleta puede degradar el rendimiento de DHT.

3.10.4 Cables de conexión. La calidad de los cables de conexión afectará la calidad y la distancia de la comunicación y se recomienda un cable blindado de alta calidad.

3.10.5 U-blox 6 Módulo GPS. Como se muestra a continuación:

3.10.5.1 Visión de conjunto. La serie de módulos NEO-6 es un conjunto de receptores GPS independientes con el alto rendimiento u-blox. Estos receptores flexibles y rentables ofrecen cuantiosas opciones de conectividad de tamaño reducido, sus medidas oscilan de 16 x 12,2 x 2,4 mm. Su diseño compacto y sus opciones de alimentación y memoria hacen que los módulos NEO-6 ideal para módulos móviles con baterías con restricciones de costes y espacio muy estrictas como se muestra en la (figura 9).

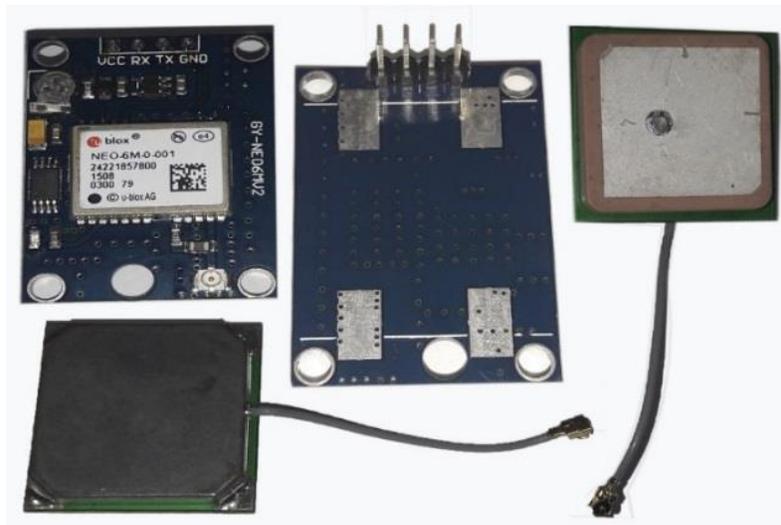


Figura 9. U-blox 6 Módulo GPS

Fuente: Elaboración propia.

El motor de posicionamiento u-blox 6 de 50 canales cuenta con un TTFF (Time-To-First-Fix) de menos de 1 segundo. El dedicado Motor de ganancia, con 2 millones de correladores, es capaz de búsquedas en espacio paralelo en tiempo/frecuencia, Lo que le permite hallar satélites al instante (Ublox, 2011). El diseño y tecnología innovadores eliminan las fuentes de interferencia y aminora los efectos de trayectos múltiples, dando a los receptores GPS NEO-6 un excelente

rendimiento de navegación en la mayoría de los

3.10.6 GPS asistido (A-GPS). El GPS asistido suministra información de ayuda como efemérides, almanaque, última posición, hora, estado del satélite y una señal de sincronización de tiempo opcional reducirá significativamente el tiempo de primera fijación y mejorará la adquisición. Todos los módulos NEO-6 soportan los servicios on-line y AssistNow Offline A-GPS de u-blox 11 y son compatibles con OMA SUPL.

U-blox Beneficios autónomos de AssistNow son:

Fijación más rápida de la posición.

No requiere conectividad.

Complementario con los servicios de Online y Offline de AssistNow.

Sin esfuerzo de integración, los cálculos se hacen en segundo plano.

3.10.6.1 Timepulse y referencia de frecuencia. NEO-6T viene con una salida timepulse que puede ser configurada desde 0,25 Hz hasta 10 MHz. El tiempo de sincronización (es decir, 1 pulso por segundo) o bien como frecuencia de referencia en el intervalo de MHz. Un Timepulse en el rango de MHz ofrece excelente precisión a largo plazo y la estabilidad.

3.10.6.2 Marca de tiempo. NEO-6T puede utilizarse para mediciones de tiempo precisas con resolución de sub-microsegundos con la interrupción externa (EXTINT0). Los bordes ascendentes y descendentes de estas señales están marcados con hora en el GPS o en la hora UTC y contados. La funcionalidad de marca de tiempo se puede habilitar con el mensaje UBX-CFG-TM2.

3.10.6.3 Navegación por estima. Automotive Dead Reckoning (ADR) es u-blox de la industria de la solución de recuento Dead-Recopilación probada para el nivel uno Clientes automotrices. La solución ADR de u-blox combina GPS y datos digitales de sensores usando un Filtro de Kalman, los cuales mejoran la precisión de posición durante períodos de señal GPS no o degradada.

El NEO-6V proporciona funcionalidad ADR a través de su interfaz de sensor de software. Una variedad de sensores (tales como Garrapatas y giroscopio), con los datos del sensor recibidos a través de mensajes UBX de la aplicación procesador. Esto permite una fácil integración y una sencilla interfaz de hardware, reduciendo los costes. Mediante el uso de datos del sensor disponibles en el bus del vehículo, los costes de hardware se minimizan ya que no se requieren Funcionalidad Dead Reckoning. ADR está diseñado para la integración simple y fácil configuración de diferentes sensores opciones (por ejemplo, con o sin giroscopio) y variantes de vehículo, y se auto calibra completamente.

3.10.7 Posicionamiento preciso del punto. El algoritmo PPP probado en la industria de u-blox proporciona niveles extremadamente altos de precisión de posición en estática y lenta moviendo aplicaciones y hace del NEO-6P una solución ideal para una variedad de aplicaciones de alta precisión como encuestas, cartografía, actividades marinas, agrícolas o de ocio.

Las correcciones ionosféricas tales como las recibidas de los satélites geoestacionarios locales SBAS12 (WAAS, EGNOS, MSAS o desde GPS permiten la mayor precisión de posicionamiento con el algoritmo PPP. La mejora máxima de la precisión de posicionamiento se alcanza con PPP + SBAS y sólo se puede esperar en un entorno sin obstáculos cielo durante un período en el orden de los minutos.

3.10.8 Antena. Los módulos NEO-6 están diseñados para su uso con antenas pasivas y activas como se puede observar en la (tabla 7).

Tabla 7. Especificaciones de la antena

Parámetro	Especificación	
Tipo de antena		Antena pasiva y activa
Recomendaciones de Antena Activa	Ganancia mínima	15 dB (para compensar la pérdida de señal en el cable de RF)
	Ganancia máxima	50 dB
	Máxima figura de ruido	1.5 dB

Fuente: U-blox, 2011.

3.10.9 Administración de energía. Los receptores u-blox admiten diferentes modos de alimentación. Estos modos representan estrategias de cómo controlar la adquisición y seguimiento de motores con el fin de lograr el mejor rendimiento posible o un buen rendimiento con consumo de energía reducido.

3.10.10 Modo de rendimiento máximo. Durante un arranque en frío, un receptor en el modo de rendimiento máximo despliega continuamente el motor de adquisición para búsqueda de todos los satélites. Una vez que el receptor tenga una posición fija (o si se dispone de información del motor de adquisición sigue siendo utilizado para buscar todos los satélites visibles que no están siendo rastreados).

3.10.11 Modo eco. Durante un arranque en frío, un receptor en modo eco funciona exactamente como en el modo de máximo rendimiento. Una vez una posición se puede calcular y se está siguiendo un número suficiente de satélites, el motor de adquisición se apaga, lo que resulta en un ahorro de energía significativo. El motor de seguimiento sigue de forma continua los satélites adquiridos y adquiere otros satélites disponibles o emergentes.

3.11 Configuración

3.11.1 Configuración de arranque. Los módulos NEO-6 proporcionan pines de configuración para el arranque. Éstas se hacen efectivas inmediatamente después de la puesta en marcha. Una vez que el módulo ha ejecutado, los ajustes de alineación se pueden modificar con la configuración UBX Mensajes. Los arreglos rectificadas permanecen activos hasta que se apague o reinicie. Si se han almacenado estos ajustes En la memoria RAM de reserva de la batería, entonces la configuración modificada será retenida, mientras la fuente de la batería de reserva no sea interrumpida.

3.11.2 Diagramas de tiempo SPI. Para evitar un uso defectuoso del SPI, el usuario debe cumplir con ciertas condiciones de temporización. Las siguientes señales deben ser consideradas para las restricciones de tiempo (ver tabla 8 y figura 9).

Tabla 8. Tipos de señales del SPI

<i>Símbolo</i>	<i>Descripción</i>
<i>SS_N</i>	Señal de selección de esclavo
<i>SCK</i>	Señal de reloj esclavo

Fuente: U-blox, 2011.

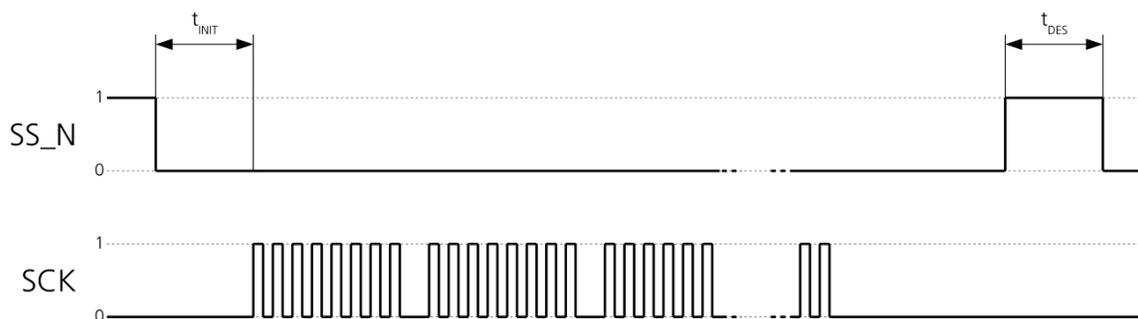


Figura 8. Diagrama de tiempo SPI

Fuente: U-blox, 2011.

3.12 MQ-135 Módulo Sensor de Calidad del Aire

El MQ-135 ver (figura 10) es un sensor de gases nocivos usado para el control de la calidad del aire y es apropiado para la detección de NH₃ (amoníaco), NO_x, alcohol, benceno, humo, CO₂, etc.

Este sensor no suministra valores absolutos, sino que absolutamente proporciona una salida analógica que debe ser monitoreado y se comparada con los productos de umbral.



Figura 9. Sensor MQ-135

Fuente: Olimex, s.f.

Características:

Voltaje de trabajo: 5V.

Pin Definición: 1_salida, 2_GND, 3_VCC.

Características de respuesta rápida.

Con una larga vida y estabilidad fiable.

Chip principal: Sensor MQ-135 Air Quality Sensor.

Doble salida: Salida analógica y salida de nivel TTL.

Alta sensibilidad al Amoníaco (NH_3), Óxidos de nitrógeno (NO_x), Alcohol, Sulfuros, Benceno (C_6H_6), Monóxido de carbono (CO), humo y otros gases nocivos.

Rango de detección: 10-1000ppm.

Buena sensibilidad a los gases dañinos en amplia gama

Alta sensibilidad a Amoníaco, Sulfuro y Benceno

3.13 Sensor MQ-4 Detector de Gas Natural y Metano Arduino



Figura 10. Sensor MQ-4

Fuente: Sparkfun, s.f.

El MQ-4 ver (figura 11) se usa en la detección de escapes de equipos de gas en los mercados de consumo y la manufactura, este sensor es apropiado para la detección de gas CH_4 , natural, GNL, impedir la exposición al alcohol, vapores y humo de cigarrillo. La sensibilidad puede ser ajustada por el potenciómetro.

Características:

Tensión de suministro: 5V

interfaz: Analógico.

Definición: 1-salida, 2 GND, 3-VCC

sensibilidad a CH₄, gas natural.

Invisible al alcohol, el humo.

Rápida respuesta.

4. Marco legal

Para el año 1982, a través del Decreto 02 se acogió la primera norma que reguló la emisión y concentración de contaminantes en la atmósfera, año en el cual se ampararon los estándares de calidad del aire y de emisión por fuentes fijas.

Para el año 1991, queda establecido en el capítulo 3 de los derechos colectivos y del ambiente, en los artículos 79 y 80 consagra la constitución política el derecho colectivo a regocijarse de un ambiente sano y es obligación del estado de proteger la pluralidad e integridad del ambiente planear el manejo y beneficio de los recursos naturales a fin de garantizar su desarrollo sostenible su mantenimiento restauración o sustitución y prevenir los componentes del quebranto ambiental.

Para el año 1995, a través del Decreto 948 , el Decreto contiene el Reglamento de Protección y Control de la Calidad del Aire, de importancia general y aplicable en todo el territorio nacional, mediante el cual se instauran las normas y manuales generales para la protección atmosférica, los componentes de prevención, control y atención de episodios por contaminación del aire, generada por fuentes contaminantes fijas y móviles, las normas y competencias para la fijación de las normas de calidad del aire o niveles de inspiración, las normas básicas para la fijación de los estándares de emisión y descarga de contaminantes a la atmósfera, las de emisión de ruido y olores ofensivos, se regulan el otorgamiento de permisos de emisión, los instrumentos y medios de control y vigilancia, el régimen de sanciones por la comisión de infracciones y la participación ciudadana en el control de la contaminación atmosférica. El presente decreto tiene por esencia definir el marco de las acciones y los elementos administrativos de que disponen las autoridades ambientales para mejorar y preservar la calidad del aire, y evitar y reducir el deterioro del medio

ambiente, los recursos naturales renovables y la salud humana ocasionados por la emisión de contaminantes químicos y físicos al aire; a fin de optimizar la calidad de vida de la población y gestionar su bienestar bajo el principio del impulso sostenible.

Para el año 2008, se dicta la resolución 909, establece las normas y los estándares de emisión admisibles de contaminantes al aire para fuentes fijas, acoge los procedimientos de medición de emisiones para fuentes fijas y reglamenta los convenios de reconversión a tecnologías limpia.

Para el 2008 a través de la resolución 910, establece los niveles máximos permisibles de emisión de contaminantes que deben cumplir las fuentes móviles terrestres, regula los requisitos y certificaciones a las que están sujetos los vehículos y demás fuentes móviles, sean importadas o de fabricación nacional, y se adoptan otras disposiciones.

Para el 2009, a través de la resolución 2604, determinar los combustibles limpios teniendo en cuenta como criterio fundamental el contenido de sus componentes y reglamentar los límites máximos permisibles en pruebas dinámicas que se vinculen a la prestación del servicio público de transporte terrestre de pasajeros y para motocarros que se vinculen a la prestación del servicio público de transporte terrestre de automotor mixto.

Para el año 2013, a través de la resolución 1541, establece las reglas para la recepción de quejas, los niveles permisibles de la calidad del aire o de emisión y la evaluación de las emisiones de los olores ofensivos.

Para el año 2017, a través de la Resolución 2854, establece la norma de la calidad del aire nivel de emisión y adopta disposiciones para la gestión del recurso del aire en el territorio nacional con el objeto de garantizar un ambiente sano y minimizar el riesgo sobre la salud humana que puede ser causado por la exposición los contaminantes en la atmósfera.

5. Metodología

Esta investigación, parte del punto de vista cuantitativo, con diseños experimentales, para ello se llevó a cabo un proceso sistemático y ordenado, debido a que se realizó un análisis documental respecto al problema planteado, siguiendo determinados pasos de planeación en la cual consistió en el compromiso de acuerdo a la estructura lógica en la toma de decisiones y con una estrategia orientado a la obtención de respuestas adecuadas a los problemas de indagación propuesto.

La metodología con énfasis en desarrollo tecnológico enmarco la elaboración del sistema de adquisición de datos meteorológicos como temperatura, humedad, geoposicionamiento; y censado de gases para la medición de la calidad del aire a través de Arduino, lo cual se planteó y realizó una exhaustiva revisión bibliográfica con referentes a los sistemas de adquisición de datos de plataformas de código libre, se decidió realizar algunos diseños para poder determinar cuál sería el más adecuado y pertinente para la fabricación de este sistema para esta metodología se crea un mapa de flujo ver (figura 12) .



Figura 11. Diagrama de metodología

Fuente: Elaboración propia.

5.1 Revisión Bibliográfica

Se realizó una exploración profunda donde se resalta los trabajos con las placas Arduino, que son placas de código abierto y en constante crecimiento en la investigación, por tener características que lo definen como (Microcontrolador, Voltaje Operativo, Voltaje de Entrada, Pines de Entradas/Salidas Digital, Pines de Entradas Análogas, Memoria Flash, entre otras). Se trata de los tipos de las placas más notorios del universo maker (cultura hacedora), la cual se diferencia de otras placas por que no poseen un magnífico modelo, sino que ofrece plataformas de hardware abierto para que otros constructores puedan crear sus propias placas.

Como fue de interés hacia las estaciones meteorológicas, rescatamos los trabajos de (Ferdoush & Li, 2014), donde plantean una red de sensores inalámbricos utilizados por Raspberry Pi y Arduino para las aplicaciones de monitoreo ambiental; por consecuente la tecnología de redes de sensores inalámbricos ha venido emergiendo como una solución viable, solución para muchas aplicaciones.

Con la finalidad de aportar un sistema de adquisición para monitoreo de la calidad del aire, estableciendo un conjunto de componentes en un solo dispositivo con la plataforma Arduino tanto software, como hardware de monitoreo de los contaminantes atmosféricos.

5.2 Elección de Componentes

Si bien se tomaron todas las opciones de componentes para la creación del sistema de adquisición donde se inició con la placa de Mega 2560 R3, se decidió por obtener sensores de temperatura, humedad, sensores de gases serie MQ, para ello se dispuso de una protoboard para ensayos y así aprovechar los componentes que tenemos sin desoldarlos, con lo cual se redujo

tiempo y se pudo modificar los circuitos con mayor rapidez. Los componentes sufren menos desgaste y así no deseamos el circuito ya montado incluyendo sus componentes, con la pérdida implícita de los mismos, con este tipo de placa temporal, no definitivo, en el que se puede improvisar cambios y sustituciones de componentes, podemos realizar todo tipo de medidas.

El montaje y desmontaje es sencillo (aunque hay que tener una precaución especial con cortocircuitos y conexiones accidentales). Además, como no se suelda ni se cortan, el aprovechamiento de los componentes es máximo.

5.3 Diseño del Dispositivo

Una gran apuesta que se tuvo en cuenta fue respecto al tamaño del dispositivo, las conexiones a conjunto de las características que se tuvieron al momento de tomar decisiones para determinar la producción del prototipo del sistema de adquisición de datos meteorológicos. Se debe realizar pruebas al artefacto construido para saber si corresponde a su finalidad, cumpliendo las condiciones inicialmente establecidas.

5.4 Elaboración del Dispositivo

Teniendo la tarjeta Arduino atmega, la protoboard y los componentes se dio el inicio de construir un prototipo de dispositivo teniendo en cuenta la plataforma de software y hardware de Arduino, su diseño está preparado para facilitar la utilización de la electrónica en diversos proyectos multidisciplinarios, buscando lo ideal para el proyecto y la gran variedad de placas, no hay sola una placa sino una gran familia que tienen ciertas similitudes y diferencias en función del proyecto que quieras realizar con ella, de esta forma se inició el trabajo en el cual se fueron descartando y mejorando las estructuras, componentes electrónicos del proyecto definiendo e

indicando las conexiones entre Arduino y los sensores.

5.5 Programación del Dispositivo

La Implementación del código para cada componente se realizó mediante las librerías, las cuales contienen el código que permiten hacer cosas comunes desde la elección de cada componente, como leer la temperatura actual, la presión, las coordenadas, y funciones establecidas para la toma de datos meteorológicos, los cuales son necesarios para la recepción de datos de los sensores en Arduino y la transmisión vía wifi. Junto al código base de Arduino, se implementa una librería(C++), de esta forma se evita un código demasiado amplio en Arduino.

5.6 Calibración

Una de las apuestas más atrayentes en este proyecto fue la calibración de los sensores de la serie MQ, que son una familia de dispositivos diseñados para detectar la presencia de distintos componentes químicos en el aire. Conseguimos enlazar estos dispositivos a un autómata o procesador como Arduino, estos sensores de gases conviene ser calibrados antes de obtener una medida precisa. Aun graduados estos sensores no disponen de la garantía necesaria para formar parte de un sistema de seguridad.

5.6.1 Testeo. Teniendo un sistema más sólido donde las conexiones eléctricas con los componentes y previamente cargadas las librerías se procede a realizar un testeo sobre el sistema de adquisición para monitoreo de la calidad del aire para comprobar su eficiencia y si por ende funciona correctamente el sistema.

5.7 Toma de Datos

El dispositivo será capaz de tomar una serie de variables meteorológicas y de gases presentes en el aire, las cuales fueron previamente programadas, calibradas, que se encuentran guardadas en una memoria, en formato .CSV y se pueden visualizar en plataforma, que permiten realizar análisis del punto específico tomado por el dispositivo y así poder tomar determinaciones apropiadas.

6. Analisis y Duscusion de los resultados

6.1 Diseño y Empleo de Componentes Adecuados para Ejecutar una Estación

Meteorológica que Mida la Calidad del aire por Medio del Micro-Controlador Arduino

A partir de la revisión de los referentes teóricos consultados se realizó un diseño experimental para la adquisición de datos meteorológicos a través de la plataforma de Arduino, después de analizar la información existente se llegó a la conclusión que se debían utilizar los siguientes materiales:

El sistema consta de un módulo de expansión SD Shield ver (figura 13), Arduino Mega Protoshield ver (figura 14), Arduino Mega 2560 R3 ver (figura 15), sensor Bmp 180 ver (figura 18), sensor de temperatura DTH 11 ver (figura 19), GPS ver (figura 17), y tarjeta SD ver (figura 16), ver (Figura 20). Resistencia 4 bandas, que en conjunto constituyeron la estación meteorológica.

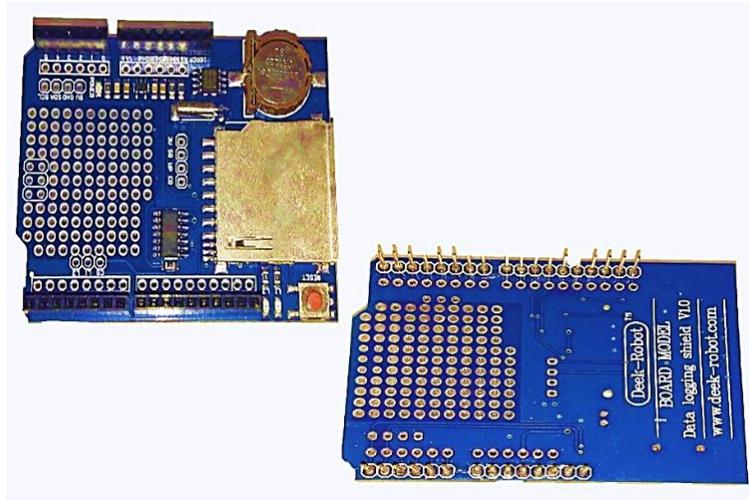


Figura 12. Módulo de expansión SD Shield

Fuente: Elaboracion propia.

Arduino Mega Protoshield:

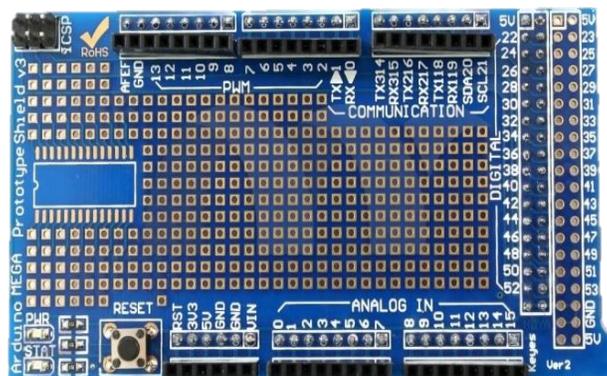


Figura 13. Arduino Mega Protoshield

Fuente: Elaboracion propia.

6.2 Arduino Mega 2560 R3



Figura 14. Arduino Mega 2560 R3

Fuente: Elaboracion propia.

6.3 Tarjeta SD



Figura 15. Tarjeta SD

Fuente: Elaboracion propia.

6.4 Sistema de geo Posicionamiento Global

Ublox NEO-6M Módulo GPS con EEPROM para CMM/GY-NEO6MV2 aeroQuad con la Antena para control de vuelo.

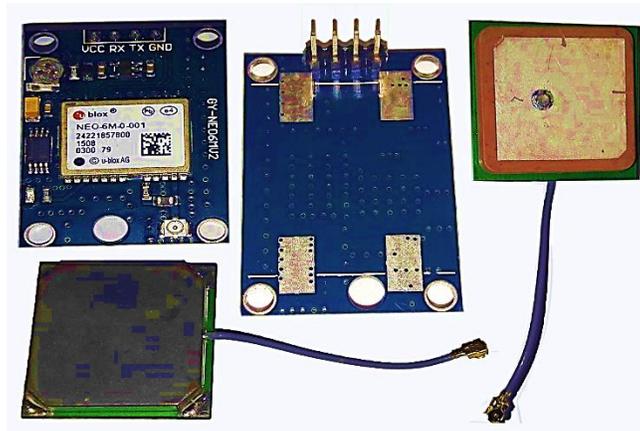


Figura 16. GPS

Fuente: Elaboracion propia.

6.5 Sensores

6.5.1 Sensor de presión bmp180 arduino. Como se muestra a continuacion:

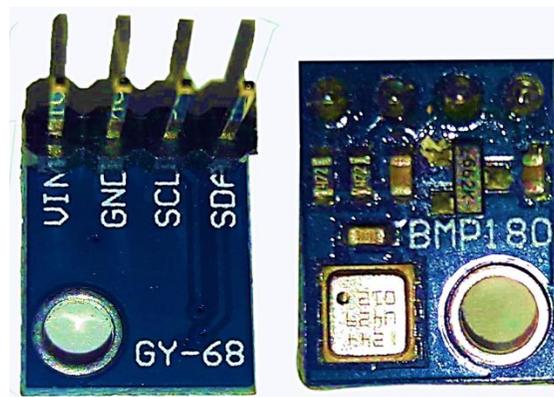


Figura 17. Sensor BMP 180

Fuente: Elaboracion propia.

6.5.2 Sensor de temperatura y humedad DTH11. Como se muestra a continuacion:

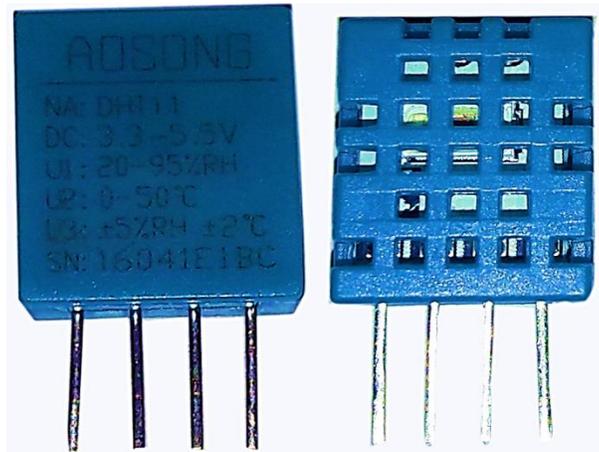


Figura 18. DTH 11

Fuente: Elaboracion propia.

6.6 Resistencia



Figura 19. Resistencia 4 bandas

Fuente: Elaboracion propia.

El empleo de los materiales mencionados anteriormente constituyó la fuente para ejecutar el sistema sensorial para la gestión y toma de datos meteorológicos, registrando los datos que midan

la temperatura, la humedad, la presión atmosférica y la geolocalización (latitud, longitud en la cual se encuentra el dispositivo).

Para medir la temperatura y humedad se utilizó el sensor DHT11, el cual mediante la conexión física a la tarjeta Arduino permitió leer datos de temperatura y humedad, (para comparar con el termómetro de uso cotidiano) ver (figura 21).

6.7 Sensor DHT11

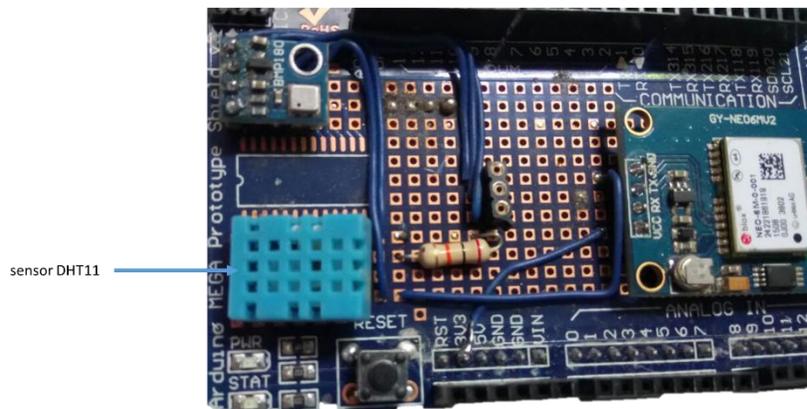


Figura 20. Sensor DHT11

Fuente: Elaboración propia.

Al conectar el Arduino al computador se creó un puerto virtual (COMXX) que se empleó para cargar la programación encargada de realizar la lectura y almacenar los datos en la memoria SD.

Para medir la presión atmosférica se usó el sensor de presión Bmp180 conectado a la placa Arduino Mega Protoshield ver (figura 22), el cual a su vez se conecta al Arduino Mega en los pines de comunicación I2C, este sensor le suministra los datos al Arduino los cuales son almacenados en la memoria SD.

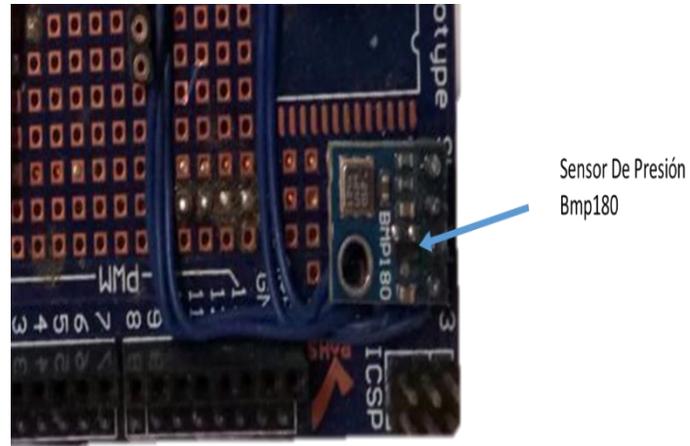


Figura 21. Sensor de Presión Bmp180

Fuente: Elaborcion propia.

Para medir la geolocalización se empleó el módulo Ublox NEO-6M o GPS conectado a una antena externa de alta ganancia quien en conjunto con el Arduino Mega es capaz de obtener la geolocalización en tiempo real almacenándolos en la memoria SD ver (figura 23).

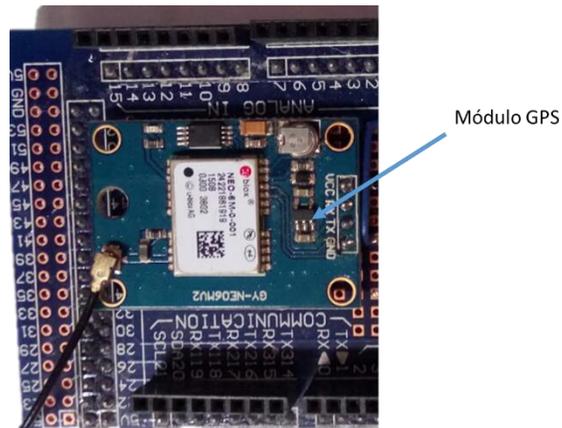


Figura 22. Ublox NEO-6M Módulo GPS

Fuente: Elaborcion propia.

6.8 Pruebas de Conexión Sensores y Módulo de GPS

Se realizó el uso de protoboard para generar conexiones temporales para realizar las respectivas pruebas de enlace entre los sensores, módulo de Gps y demás placas, después de haber pasado por rigurosas pruebas que determinaron estar listos para ensamblar al Módulo de expansión SD Shield vcer (figura 24). Sensores y modulo Gps acoplados al Arduino Mega A Protoshield ver (figura 25).

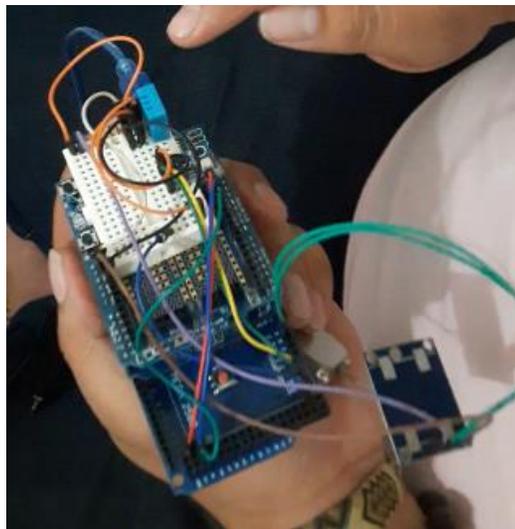


Figura 23. Conexión protoboard de sensores, módulo de Gps y placas

Fuente: Elaboración propia.

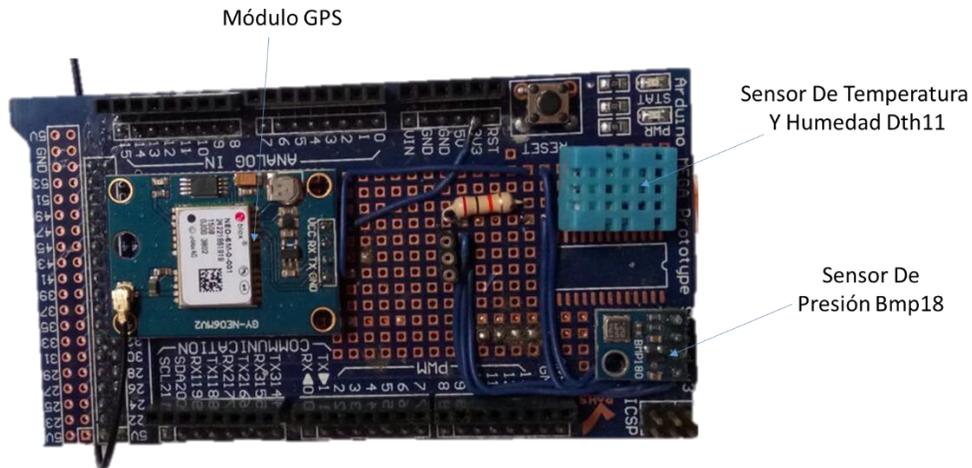


Figura 24. Sensores y modulo Gps acoplados al Arduino Mega A Protoshield

Fuente: Elaboracion propia.

A partir del ensamble de los sensores se necesitó tener un dispositivo para guardar los datos que se registraban, en este caso se optó por una placa de Módulo de expansión SD Shieldver (figura 26), la cual con una programación establecida y unida a la placa Arduino Mega Protoshield, se logró que pudiese guardar los datos en la SD, para luego así llevarlos a Excel, tener un conglomerado de datos metrológicos, para luego comparar con medidores de temperatura local.

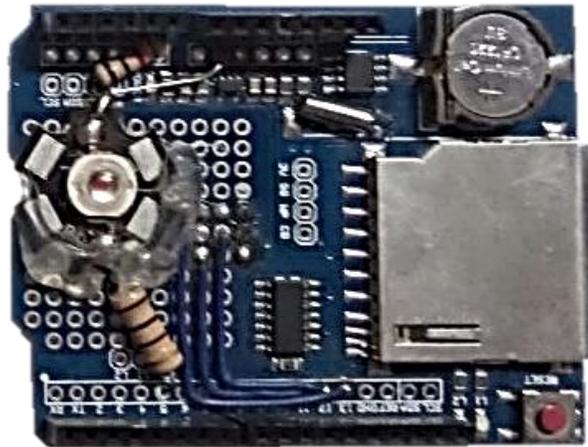


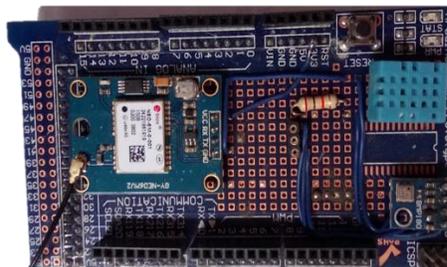
Figura 25. Módulo de expansión SD Shield

Fuente: Elaboracion propia.

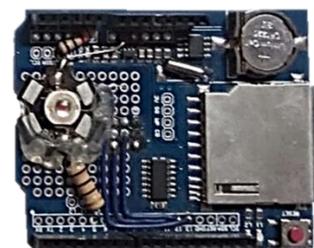
El ensamble de la estación consto de tres partes: la primera de ellas fue la placa de Arduino Mega que es la controladora de los sensores y regulación de corriente, la segunda es la placa Protoshield donde van anclados los sensores y el modulo GPS, y la última parte para guardar los datos obtenidos por los sensores se estableció un módulo SD, donde almacena los datos obtenidos por los sensores y modulo GPS ver (figura 27). Estación terminada ver (figura 28). dando un peso de 102 gramos, con un peso muy liviano ver (figura 29).



Arduino Mega



Protoshield



Modulo SD

Figura 26. Componentes de la estación

Fuente: Elaboracion propia.

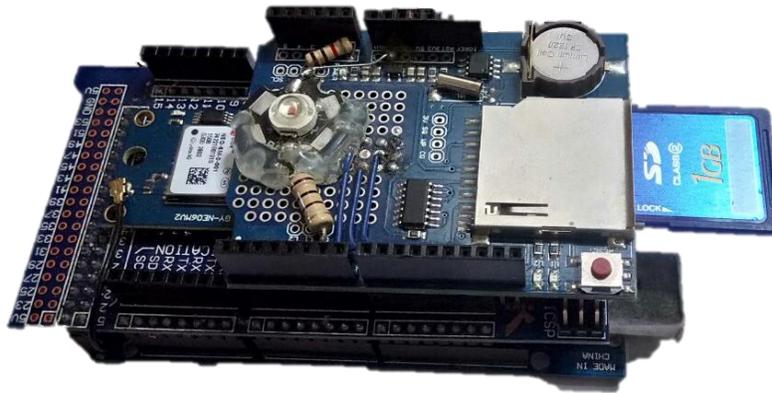


Figura 27. Estación terminada

Fuente: Elaboracion propia.



Figura 28. Peso de la estación

Fuente: Elaboracion propia.

La estación tiene un peso de 102 gramos, la cual se convierte en una estación portátil, pero de gran precisión y ayuda a la hora de obtener datos, lo cual es propicio para dar uso en zonas especiales que necesiten la ayuda de una estación para la toma de datos atmosféricos; a su vez está diseñada para adaptársela a un dron o vehículo no tripulado, para que al tiempo que realice los vuelos pueda obtener datos atmosféricos de la zona sobrevolada, para así poder tener antecedentes específicos atmosféricos y geo posicionamiento ver tabla en Excel (tabla 9).

Tabla 9. Datos obtenidos por la estación 1.0



	A	B	C	D	E	F	G	H
1	Temperatutra	Humedad	Presion Absoluta	Indice de calor	Latitud	Longitud	Altitud	Fecha y Hora GMT
2	32.08	42.00	973.59	31.21	7.891.137	-72.490.097	368.00	04/26/2017 19:18:23
3	32.07	42.00	973.49	31.21	7.891.142	-72.490.097	369.20	04/26/2017 19:18:24
4	32.09	42.00	973.56	31.21	7.891.142	-72.490.097	369.20	04/26/2017 19:18:24
5	32.09	42.00	973.58	31.21	7.891.142	-72.490.097	369.20	04/26/2017 19:18:24
6	32.08	42.00	973.57	31.21	7.891.142	-72.490.097	369.10	04/26/2017 19:18:25
7	32.09	42.00	973.56	31.21	7.891.143	-72.490.097	369.10	04/26/2017 19:18:26
8	32.09	42.00	973.59	31.21	7.891.143	-72.490.097	369.10	04/26/2017 19:18:26
9	32.09	42.00	973.58	31.21	7.891.145	-72.490.097	370.10	04/26/2017 19:18:27
10	32.09	42.00	973.57	31.21	7.891.145	-72.490.097	370.10	04/26/2017 19:18:27
11	32.10	42.00	973.59	31.21	7.891.145	-72.490.097	370.30	04/26/2017 19:18:28
12	32.09	42.00	973.66	31.21	7.891.157	-72.490.127	369.80	04/26/2017 19:18:29
13	32.10	42.00	973.55	31.21	7.891.155	-72.490.127	369.80	04/26/2017 19:18:29
14	32.10	42.00	973.59	31.21	7.891.155	-72.490.127	369.80	04/26/2017 19:18:29
15	32.11	42.00	973.60	31.21	7.891.155	-72.490.127	369.80	04/26/2017 19:18:29
16	32.09	42.00	973.57	31.21	7.891.155	-72.490.127	370.50	04/26/2017 19:18:30
17	32.11	42.00	973.58	31.21	7.891.155	-72.490.127	370.50	04/26/2017 19:18:30
18	32.11	42.00	973.57	31.21	7.891.160	-72.490.142	371.50	04/26/2017 19:18:33
19	32.12	42.00	973.63	31.21	7.891.160	-72.490.142	371.50	04/26/2017 19:18:33
20	32.13	42.00	973.64	31.21	7.891.160	-72.490.142	371.50	04/26/2017 19:18:33
21	32.13	42.00	973.60	31.21	7.891.160	-72.490.142	369.90	04/26/2017 19:18:34
22	32.12	42.00	973.59	31.21	7.891.160	-72.490.142	369.90	04/26/2017 19:18:34
23	32.14	42.00	973.55	31.21	7.891.160	-72.490.142	369.90	04/26/2017 19:18:34

Fuente: Elaboracion propia.

6.9 Módulo Versión 2.0

Con las suficientes revisiones bibliográfica sobre las plataformas de Arduino, este dispositivo de hardware libre, basada en una placa con un microcontrolador y un entorno de desarrollo (software), planteada para suministrar el uso de la electrónica en propósitos multidisciplinarios, después de analizar la información existente se llegó a la decisión que se debían utilizar los siguientes materiales:

El componente de la estación meteorológica cual mide la calidad del aire fue ensamblado a partir de sensores de Arduino, lo cual consto de sensores dt11, ccs811, bmp180, mq135 mq4, GPS, Arduino mini Y tarjeta Wmos wi-fi.

Se realizó la planeación de realizar una estación meteorológica que midiera la calidad del aire, la cual tuviera sensores que tomen datos de temperatura, humedad, presión atmosférica, dióxido de carbono, compuestos volátiles orgánicos, gas metano y alcoholes.

A partir de la revisión de los referentes teóricos consultados se realizó un diseño experimental para la adquisición de datos meteorológicos a través de la plataforma de Arduino.

6.10 Sensor DHT11

El sensor DHT11ver (figura 30) nos permitio medir la temperatura y humedad con Arduino. Una de las prioridades que ofrece el DHT11, además de medir la temperatura y la humedad, es que es digital. Este sensor maneja un pin digital para enviar la información y, por lo tanto, se está protegido frente al ruido electrónico.

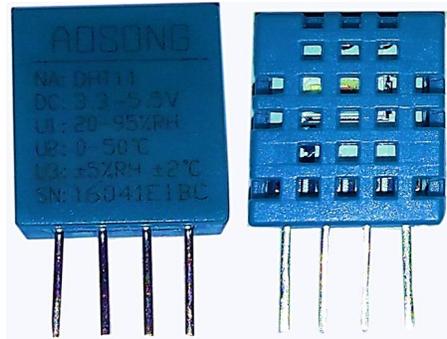


Figura 29. Sensor DHT11

Fuente: Elaboracion propia.

6.11 Sensor CCS811

Este sensor ver (figura 31) de gas puede detectar una amplia gama de compuestos orgánicos volátiles (COV) y está diseñado para el monitoreo de la calidad del aire interior, y registra la concentración de CO₂ (equivalente de dióxido de carbono calculado) dentro de un rango de 400 a 8192 partes por millón (ppm), y la concentración de TVOC (Compuesto orgánico volátil total) dentro de un rango de 0 a 1187 partes por billón (ppb).



Figura 30. Sensor CCS811

Fuente: Adafruit, s.f.

6.12 Sensor BMP180

El sensor de presión barométrica BMP180 ver (figura 32) está planteado para leer la presión atmosférica y de esta forma estimar indirectamente la altura sobre el nivel del mar. El sensor BMP180 registra la presión absoluta (Barométrica) y la temperatura; al registrar la temperatura se puede compensar su predominio en la presión y así establecer con mayor exactitud la altitud.

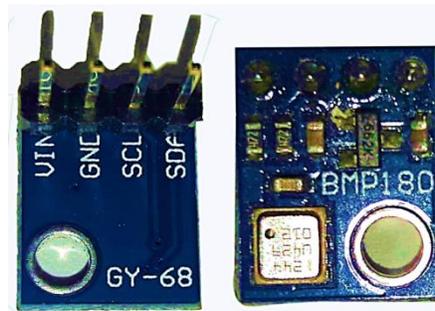


Figura 31. BMP180

Fuente: Elaboracion propia.

6.13 Sensor MQ135



Figura 32. Sensor MQ135

Fuente: Olimex, s.f.

Este sensor ver (figura 33) de inspección de calidad de aire es usado para la detección de contaminación en el medio ambiente, por lo general se incluye en circuitos de control como alarmas en las casas, sitios donde se desea advertir altos niveles de contaminación a nivel aeróbico como manufacturas que manejan compuestos químicos que pueden ser perjudiciales para la salud, fundamentalmente en equipos supervisores de calidad de aire en edificios y oficinas.

6.14 Sensor MQ 4



Figura 33. Sensor MQ 4

Fuente: Sparkfun, s.f.

Este sensor ver (figura 34) es utilizado medir las concentraciones de gas natural en el aire (compuesto principalmente de gas metano [CH₄]). El MQ-4 puede detectar concentraciones de gas natural desde 200 a 10000ppm (partes por millón) lo cual es muy confiable para la toma de metano. Este sensor tiene una alta sensibilidad y de respuesta muy rapida

6.15 Arduino Pro MINI 1



Figura 34. Arduino pro mini 1

Fuente: Agspecinfo, s.f.

El Arduino Pro Mini ver (figura 35) este entorno de microcontroladores fundado en el Atmega328. Tiene 14 pines d e los cuales 6 se pueden usar como salidas PWM), 6 entradas analógicas, un resonador compuesto, un botón de recuperación y orificios para los cabezales de los pines de montaje. El Arduino Pro Mini está destinado a la instalación semipermanente en objetos. Viene sin cabezales premontados, permitiendo el uso de varios tipos de terminales o la soldadura directa de cables. es simultáneo con el Arduino Mini.

6.16 Módulo GPS U-blox Neo



Figura 35. Módulo GPS U-blox neo

Fuente: Bastelgarage, 2020.

El módulo GPS en su modelo GY-GPS6MV2 ver (figura 36) viene con un módulo de serie U-Blox NEO 6M equipado en el PCB, una EEPROM con configuración de fábrica, una batería de botón para salvaguardar los datos de distribución en la memoria EEPROM, un indicador LED y una antena cerámica. Asimismo, posee los pines o conectores Vcc, Rx, Tx y Gnd por el que se puede enlazar a algún microcontrolador mediante una interfaz serial. Para que nuestro módulo GPS desempeñe a la perfección se invita hacer las pruebas en un ambiente abierto o cercano a la ventana para una correcta recepción de la señal.

6.17 Wemos D1

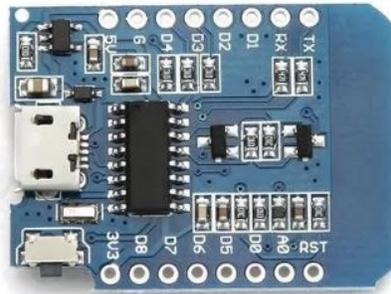


Figura 36. Wemos D1

Fuente: Prometec, s.f.

La placa de desarrollo Wemos D1 ver (figura 37) se basa en el notorio chip que revolucionó el Wifi en sistemas embebidos: específicamente el ESP8266. Con este sencillo modelo se puede ejecutar el prototipo de cualquier sistema para el IoT (internet of things) en cuestión de horas. El concepto es excelente, no requiere de otros anexos para tener conexión a internet como el accesorio ethernet shield o wifi shield de arduino oficial. Con el Wemos D1 se obtiene todo: wifi y un potente ordenador para potenciar tu estudio.

6.18 Nivelador Lógico de VOLTAJE 5v-3.3v Bidireccional

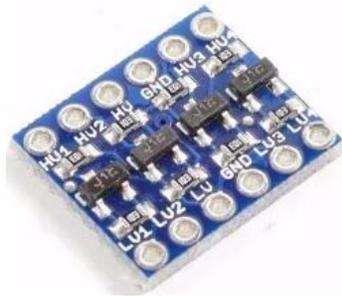


Figura 37. Nivelador de voltaje 5V-3.3V

Fuente: Llamas, 2008.

Este conversor ver (figura 38) de niveles de 2 canales bidireccional es un pequeño dispositivo que con seguridad convierte niveles de 5 V a 3.3 V y pasa de 3.3 V a 5 V al mismo tiempo.

6.19 Wemos D1 Micro Sd Shield



Figura 38. 10 MICRO SD shield

Fuente: Naylamp Mechatronics SAC, 2016.

Este escudo ver (figura 39) está diseñado para su uso con las placas de desarrollo WeMos D1 mini y mini Pro. El escudo proporciona un medio para permitir que un WeMos D1 mini se comunique directamente con la mayoría de los tipos de tarjetas microSD.

6.20 Condensador Electrolítico de 400 UF A 16V



Figura 39. Condensador de 400 UF A 16V

Fuente: Electro Componentes, 2016.

Es un condensador eléctrico ver (figura 40), es un terminal pasivo, utilizado en electricidad y electrónica, capaz de acumular energía sustentando un campo eléctrico. Desarrollado un par de superficies conductoras, habitualmente en forma de láminas o placas, en situación de atribución integral.

6.21 Resistencia



Figura 40. Resistencia

Fuente: Codigo Electronica, s,f.

La resistencia eléctrica ver (figura 41) se emplea para emparejar la corriente al desplazarse a través de un conductor. lo más común es utilizar un condensador acoplado en paralelo al pulsador. El condensador esperará cierto tiempo en cargarse y una vez que esté lleno, la señal de salida será semejante a la señal de entrada.

El empleo de los materiales mencionados anteriormente constituyó la fuente para ejecutar la estación meteorológica, para la medición de la calidad del aire, registrando los datos que tomen datos de temperatura, humedad, presión atmosférica, dióxido de carbono, compuestos volátiles orgánicos, gas metano y alcoholes (latitud, longitud en la cual se encuentra el dispositivo).

Para el montaje de la estación, se programó el Arduino Pro Mini 1, adhiriéndole cada uno de los sensores, además de la tarjeta de almacenamiento (Wemos D1). La estación portátil fue programada para que realizara mediciones cada 5 segundos. A continuación, se muestra la estación finalmente construida.

6.22 Prototipo

1. Para medir la presión atmosférica se usó el sensor de **presión Bmp180** conectado a la Arduino Pro Mini 1, este sensor le suministra los datos al Arduino los cuales son almacenados en la memoria micro SD.

2. Para medir la temperatura y humedad se usó el **sensor DHT11**, este sensor le suministra los datos al Arduino Pro Mini 1 los cuales son almacenados en la memoria micro SD.

3. Para medir la geolocalización se empleó el módulo Ublox NEO-6M o GPS conectado a una antena externa de alta ganancia quien en conjunto con Arduino Pro Mini 1, es capaz de obtener la geolocalización en tiempo real almacenándolos en la memoria SD.

4. Para medir los compuestos orgánicos volátiles (COV) se utilizó el sensor ccs811 de gas, que puede detectar una amplia gama COV, enviando datos los cuales son almacenados en la memoria micro SD.

5. Para medir la calidad de aire y particularmente CO₂ y alcoholes se usó el sensor MQ 135, este sensor le suministra los datos al Arduino Pro Mini 1 los cuales son almacenados en la memoria micro SD.

6. Para medir las concentraciones Gas Metano (Gas Natural) se el sensor MQ4 suministrando los datos al Arduino Pro Mini 1 los cuales son almacenados en la memoria micro SD.

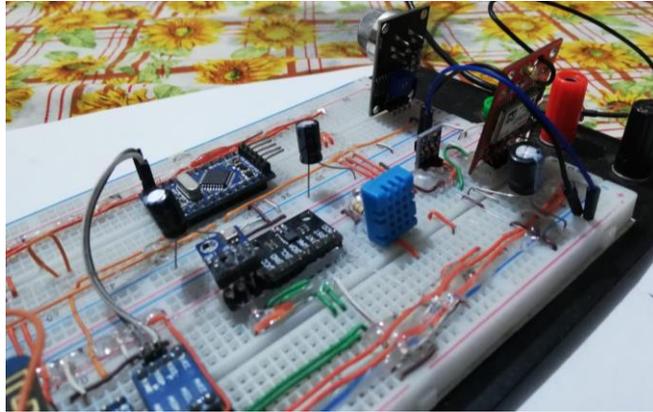


Figura 41. Prototipo de estacion 2.0

Fuente: Elaboracion propia.

El ensamble de la estación consto de tres partes: la primera de ellas fue la Arduino Pro Mini 1 que es la controladora de los sensores y regulación de corriente, la segunda es la placa Protoshield donde se anclaron los sensores, el modulo GPS, y la última parte para guardar los datos obtenidos por los sensores se estableció un wemos d1 micro sd shield, donde se almacena los datos obtenidos por los sensores y modulo GPS ver (figura 42).

6.23 Pruebas de Conexion Sensores y Modulo Wemos d1 Micro sd Shield

Se realizó el uso de una protoboard para generar conexiones temporales y de esa manera ejecutar las respectivas pruebas de enlace entre los sensores, módulo de Gps, junto con los otros dispositivos. Posteriormente de haber efectuado rigurosas pruebas que permitieron determinar que los sensores se encontraban dispuestos para ajustarlos al Módulo Wemos D1 Micro Sd Shield.

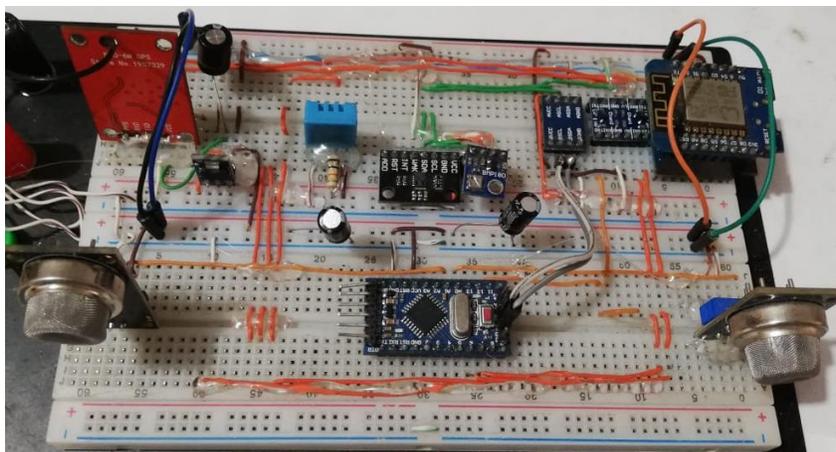


Figura 42. Protoboard de conexiones

Fuente: Elaboracion propia.

A partir del acople de los sensores se necesitó un dispositivo que guardara los datos que se registraban, por lo cual se optó por una placa de Módulo de Wemos D1 Micro Sd Shield, la cual con una programación establecida y unida a la placa Arduino Pro Mini 1, se logró que pudiese guardar los datos en la micro SD, para luego así llevarlos a Excel, y obtener una serie de datos meteorológicos ver (figura 43).

6.23.1 Envío de datos a servidor externo. Se empleó un servidor en tiempo real, para el envío de datos se utilizó el módulo Wemos D1 y se conectó a un servidor para visualizar la información desde cualquier parte, donde se debe contar con internet.

6.23.2 Base de datos. Principalmente, el objetivo es acceder a la información que ha recibido la estación desde cualquier punto de manera remota, es decir sin visualización física de la estación meteorológica de calidad del aire, aunque se cuenta con almacenamiento de información en el sitio donde se encuentra la estación, tan sólo es un respaldo en caso de que ocurra alguna falla en la transmisión vía Wi-Fi.

6.23.3 Proceso almacenamiento de datos. Los datos se pueden almacenar en la memoria del sistema, así como permite descargarlos en archivos .csv, para su fácil procesamiento, los cuales son documentos que permite representar datos en una tabla en donde las columnas son separadas por comas y las filas por saltos de líneas.

Como respaldo, se utiliza la tarjeta de memoria SD, en donde se almacena la información en archivos .csv. Desde el código de programación en Arduino se crean los archivos, se dan los títulos para las columnas y se ubican los valores según las condiciones de fecha y hora dadas. Por cada día se crea un archivo y por cada mes se crea una carpeta.

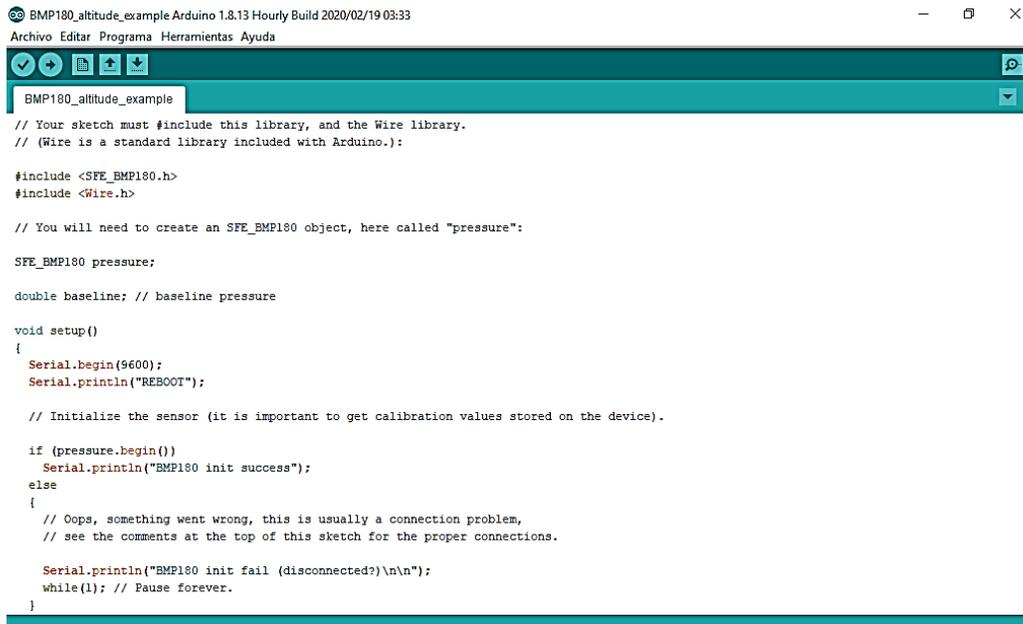
Se verifican las columnas separadas correctamente y se aceptan todas las ventanas hasta finalizar la exportación. Allí se podrán graficar y analizar los datos como se requieren.

La estación se convierte en una estación meteorológica pequeña y portátil, pero de gran precisión, ayuda a la hora de obtener datos, lo cual es propicio para dar uso en zonas especiales que necesiten la asistencia de una estación para la toma de datos atmosféricos, a su vez está puede adaptarse a un drone o vehículo no tripulado, para que al tiempo que realice los vuelos pueda obtener datos atmosféricos de la zona sobrevolada, para así poder tener antecedentes específicos atmosféricos y geo posicionamiento.

6.24 Programación

6.24.1 Código de arduino. El Símbolo principal del sistema, de este código es el que posteriormente manda a la placa Arduino Pro Mini 1 a hacer las operaciones importantes para que el sistema marche correctamente (main en programación). Antes de explicar el código se expone cómo instaurar un proyecto en Arduino, importar librerías a este y su distribución usual.

Para que el Arduino se disponga a funcionar, la primera salida es ejecutar un programa conocido como “sketch” ver (figura 44). Una vez creado, se efectuó el código, se compilo y vuelve en la memoria del microcontrolador.



```

BMP180_altitude_example Arduino 1.8.13 Hourly Build 2020/02/19 03:33
Archivo Editar Programa Herramientas Ayuda

BMP180_altitude_example
// Your sketch must #include this library, and the Wire library.
// (Wire is a standard library included with Arduino.):

#include <SFE_BMP180.h>
#include <Wire.h>

// You will need to create an SFE_BMP180 object, here called "pressure":

SFE_BMP180 pressure;

double baseline; // baseline pressure

void setup()
{
  Serial.begin(9600);
  Serial.println("REBOOT");

  // Initialize the sensor (it is important to get calibration values stored on the device).

  if (pressure.begin())
    Serial.println("BMP180 init success");
  else
  {
    // Oops, something went wrong, this is usually a connection problem,
    // see the comments at the top of this sketch for the proper connections.

    Serial.println("BMP180 init fail (disconnected?)\n\n");
    while(1); // Pause forever.
  }
}

```

Figura 43. Interfaz Arduino

Fuente: Elaboracion propia.

Podemos observar que se muestra por defecto un comando setup y un comando loop. El setup tiene como destino de inicializar las variables, librerías, pines, etc. Es decir, es un apartado de proporción el cual solo se hace una vez en el instante de reinicio o carga del programa a Arduino. Mientras que loop indica que es usado para controlar mediante instrucciones relevantes a Arduino.

6.24.2 Programación Tarjeta de Desarrollo Arduino Pro Mini 1. Para realizar el código de la Tarjeta de Desarrollo Arduino Pro Mini 1 se empleó el software Arduino IDE, basado en lenguaje Java. Las librerías que se emplearon fueron:

```
# include < SoftwareSerial.h >
```

```
# include < TinyGPS ++. h >
```

```
# incluye < DHT.h >
```

```
# include < SFE_BMP180.h >
```

```
# incluye " SparkFunCCS811.h "
```

```
# include < MQ135.h > // Biblioteca de sensor de aire MQ135
```

```
# include < GAS_MQ4.h > // Biblioteca de sensor de aire MQ4
```

Librerías que están en repositorio abierto GitHub que es un medio de gestión de proyectos y control de transcripciones de código, es una plataforma de red social diseñada para desarrolladores de software libre, que accede trabajar en colaboración con otras personas de todo el mundo, para planear proyectos y realizar un seguimiento del trabajo, allí está consignado todo el código de la estación, programación, ejemplos y seguimientos que hacen otros desarrolladores respecto a la plataforma y componentes de Arduino.

Para la obtención de la programación de la placa, los sensores y demás componentes del proyecto de programación pueden dirigirse a <https://github.com/Yercar18/Dronefenix>, allí se encuentra a disposición la programación de la estación.

6.25 Calibración de Sensores

Uno de los grandes problemas que surgieron sobre la calibración de los sensores, es que no se tenían referentes de calibración y esto hacía de alguna otra forma que los sensores no dieran la confiabilidad a la hora de censar los datos, estos sensores son de bajo costo, pero eficaces, y ayudan en la toma de decisiones, cuando se adquieren no están calibrados, se formaron laboratorios de censado ver (figura 45 y 46) en diferentes materiales contaminantes obteniendo valores no deseados.



Figura 44. Prueba con gas butano proveniente de un encendedor

Fuente: Elaboracion propia.



Figura 45. Prueba con Aceite quemado

Fuente: Elaboracion propia.

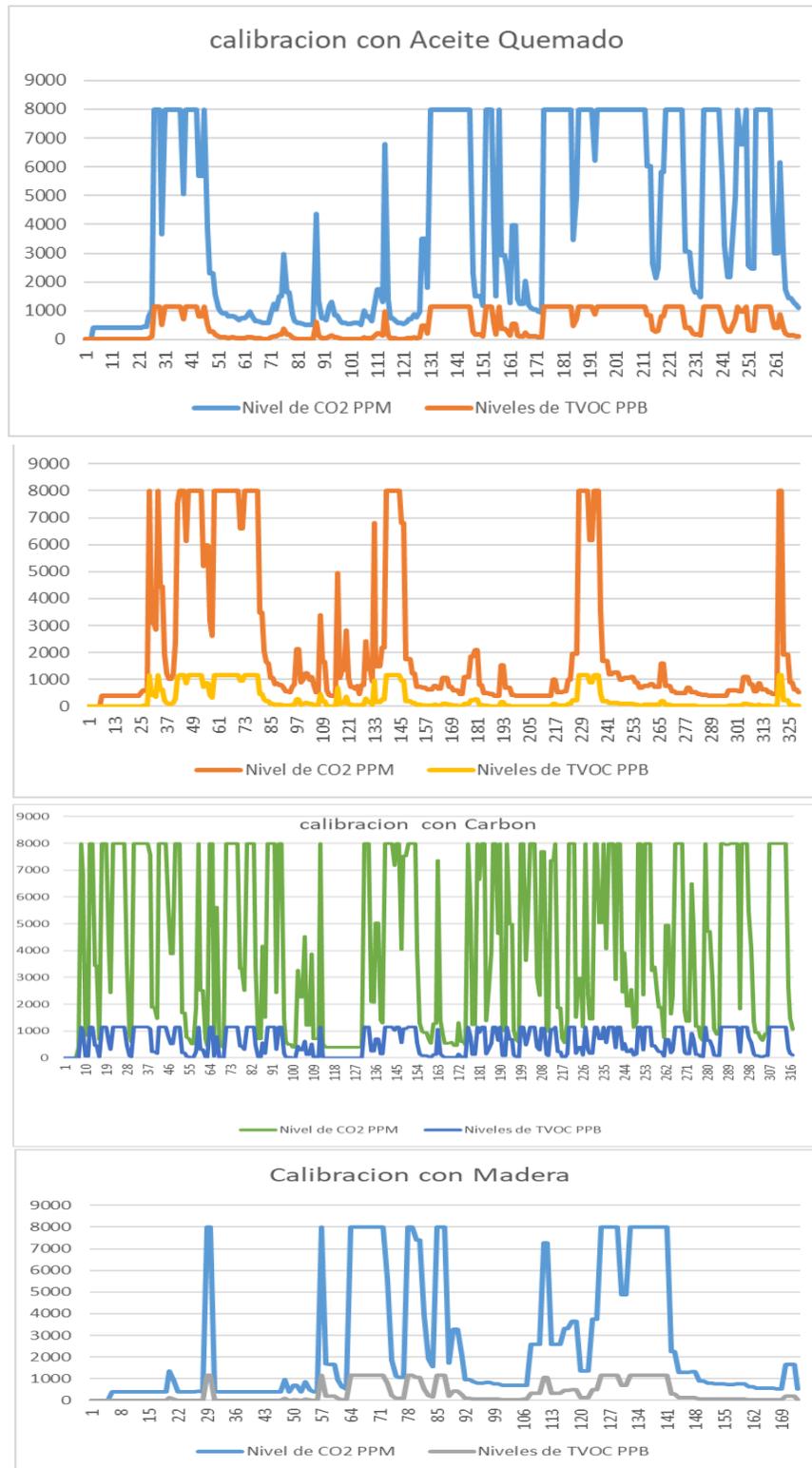


Figura 46. Datos obtenidos sin calibración adecuada

Fuente: Elaboracion propia.

Esto nos llevó a concluir que debíamos averiguar y consultar mucho mejor sobre estos sensores ver (figura 47) para ello se planteó entonces un modelo matemático para calibrar donde se establecen rutas con los las hojas de datos de los fabricantes, obteniendo la información, los detalles en los manuales de instrucciones de estos componentes de registros de los sensores; esos documentos traen sus funcionamientos, características y ejemplos de la alimentación; todo ello es electrónico y había que tener en cuenta de que ellos traen unas especificaciones de los símbolos y las características sensibles, a su vez presentan una gráfica con los valores mínimos y máximos en partes por millón, es una resistencia a las variables en las concentraciones de gases, de ahí se tomaron los valores y se realizaron en hojas de Excel tomando datos.

Para este trabajo se registraron en la serie de todos los sensores MQ existentes para la línea de gases.

6.26 Modelado Matemático

En la hoja de datos de los sensores MQ se describe la relación PPM utilizando los términos R_s/R_o , donde R_s es la resistencia interna del sensor MQ a tiempo, y R_o es el valor de R_s en aire limpio. Las especificaciones tienen gráficos de comportamiento del sensor MQ para algunos gases utilizando la relación R_s/R_o , permitiendo obtener información sobre cómo medir cada tipo de gas, así en la (Figura 48) se muestra el gráfico de gases legible para MQ-6 a partir de las especificaciones técnicas sobre la relación R_s/R_o y utilizando escala logarítmica.

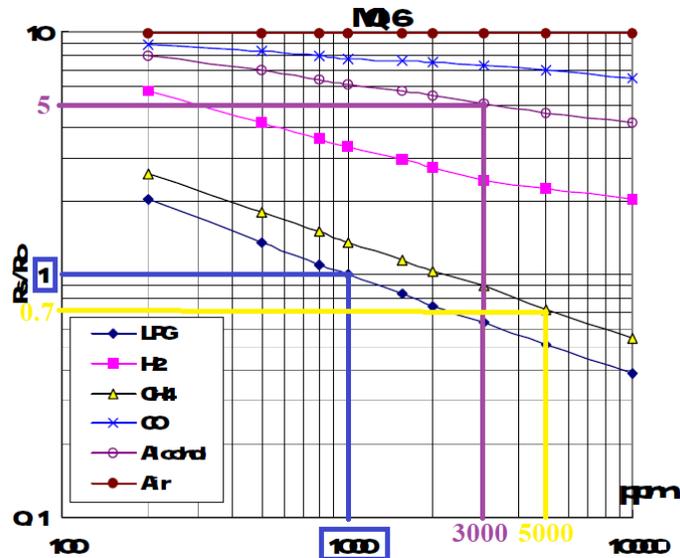


Figura 47. Espacios de muestreo de las especificaciones técnicas por fábrica

Fuente: Sparkfun, s.f.

En efecto, la Fig. 1 no sólo ofrece las especificaciones de la hoja de datos, sino que las líneas azules del eje indican una forma de indicar el valor de las variables PPM y Rs/Ro para el gas GLP, en este caso es suficiente el valor del eje de lectura para el punto seleccionado, CH4 y Air no es legible directamente desde la escala, pero es posible completar los valores del eje para saber cómo es el valor. Para cada gas se obtuvieron 3 puntos en los espacios evitando valores medios donde se requiera una aproximación externa utilizando la ecuación (1).

$$p_m = \sqrt{p_{inf} p_{sup}} \quad (1)$$

Dónde:

pm es el punto medio requerido

pins es el punto inferior usando la escala logarítmica (a la izquierda del punto)

psup es el punto superior usando la escala logarítmica (a la derecha del punto)

Este procedimiento se aplica a todo el gas de cada sensor y se traslada a un libro de Excel para obtener ecuaciones lineales utilizando el procedimiento de cálculo lineal de regresión simple en la ecuación (2), donde se presenta los datos necesarios para el cálculo y se aplica automáticamente la ecuación de Excel cuando el usuario solicita una línea de tendencia.

$$m = \frac{1}{n-1} \left(\sum_{i=1}^{n-1} \frac{(y_{i+1} - y_i)}{(x_{i+1} - x_i)} \right), \quad b = \frac{1}{n} \left(\sum_{i=1}^n y_i - m \sum_{i=1}^n x_i \right) \quad (2)$$

Dónde:

m son las pendientes de la línea de tendencia

n es el número de muestras utilizadas para el cálculo

y_i es un i-ésimo término para los datos que representan un valor para la relación R_s/R_o

x_i es un i-ésimo término para los datos que representan el valor PPM

b es la intercepción y cuando $x=0$

En la Figura 2 se puede ilustrar un ejemplo de cálculo de la línea de tendencia para MQ-309A de HANWEI, esta línea de tendencia fue graficada con algunas muestras y la puntuación R2 se añade a la leyenda con el fin de cuantificar el error presente para estos datos.

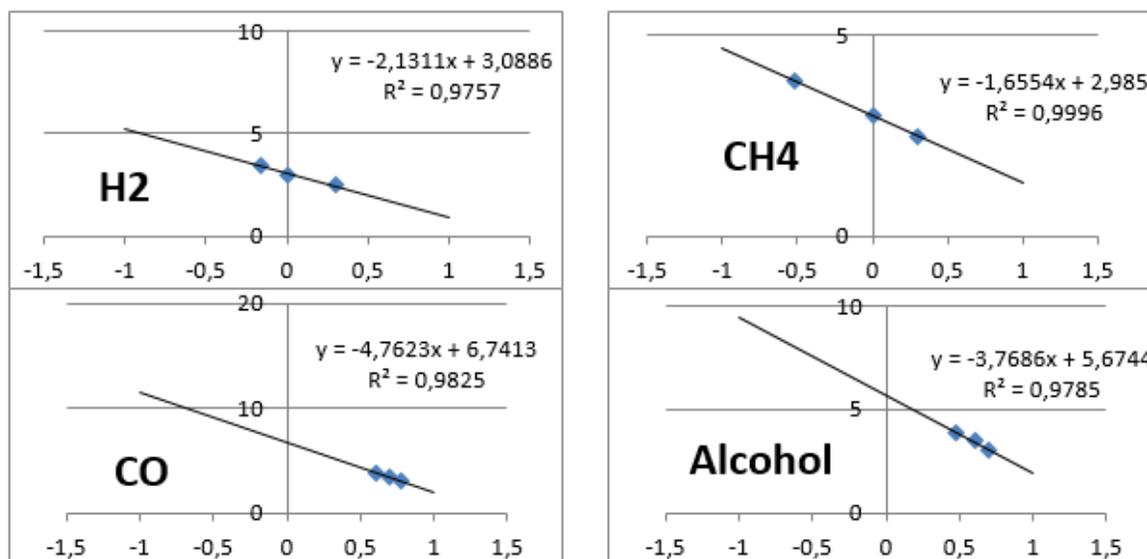


Figura 48. Líneas de tendencia para MQ-309

Fuente: Elaboracion propia.

MQ-309A solo mide PPM para 4 gases, pero otro sensor como el MQ-2 puede medir casi 6 gases, para ello las Tablas I y II describen las líneas de tendencia obtenidas para todos los sensores MQ tomados en este trabajo, donde hay 6 de la fábrica Sparkun, excepto para el MQ 5 que es de fábrica de paralaje de forma, pero utilizando el máximo espacio cómodamente se presentan en la (figura 49). Por otro lado, la (Tabla 10 y 11) incluye sensores con fábricas de Pololulu, Sensorsportal y HANWEI Electronics (CARRILLO AMADO, CALIFA URQUIZA, & RAMON VALENCIA , 2020)

Tabla 10. Sensores MQ de la fábrica de Sparkun utilizados para calibración y pruebas

GASES	MQ-3		MQ-4		MQ-5		MQ-6		MQ-7		MQ-8	
	m	b	m	b	m	b	m	b	m	b	m	b
H2	-----	-----	-----	-----	-4.368	2.9667	-3.6775	5.0286	-1.3290	1.8864	-0.7152	2.9891
LPG	-3.1851	4.7048	-2.5818	3.6303	-2.5723	1.8943	-1.6567	2.8775	-7.8626	9.1056	-3.4190	7.3513
CH4	-17.5310	28.7850	0.9873	2.6386	-2.4438	2.3044	-1.0000	3.3010	-54878	8.8387	-7.5609	15.2430
CO	-4.3390	6.4432	-5.5945	5.6693	-4.8188	5.2023	-12.791	14.523	-1.4065	2.0162	-7.0753	15.3960
Alcohol	-1.4350	0.4103	-11.8900	9.0375	-4.4190	4.8044	-5.8057	7.5292	-6.3219	9.9240	-1.7459	4.7575
Benzine	-2.7009	0.632	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----
Hexane	-2.7268	3.6299										
Smoke			-11.189	9.0375								

Fuente: Elaboracion propia.

Tabla 11. Sensores de. Pololulu, Haoyuelectronics, Sensorsportal y HANWEI Electronics factories

GASES	MQ-2		MQ-9		MQ-131		MQ-135		MQ-303A		MQ-309A	
	m	b	m	b	m	b	m	b	m	b	m	b
H2	- 2.2459	2.9845	-----	-----	-----	-----	-----	-----	-----	-----	- 2.1311	3.0886
LPG	- 2.2879	- 2.7901	- 2.2535	2.9855	-----	-----	-----	-----	-----	-----	-----	-----
CH4	- 2.6208	3.6075	- 1.3012	3.1476	-----	-----	-----	-----	-----	-----	- 1.6554	2.9850
CO	- 3.1157	4.5134	- 1.7490	2.8270	-----	-----	- 2.7268	2.3010	-----	-----	- 4.7623	6.7413
Alcohol	- 2.7028	3.5595	-----	-----	-----	-----	- 2.8608	1.8627	-----	-----	- 3.7686	5.6744
Propane	- 2.2879	2.7901	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----
NOX	-----	-----	-----	-----	- 2.7245	3.3004	-----	-----	-----	-----	-----	-----
Cl₂	-----	-----	-----	-----	- 1.0333	1.7117	-----	-----	-----	-----	-----	-----
O₃	-----	-----	-----	-----	- 1.2307	1.455	-----	-----	-----	-----	-----	-----
CO₂	-----	-----	-----	-----	-----	-----	- 3.2819	1.9903	-----	-----	-----	-----
Toluene	-----	-----	-----	-----	-----	-----	5.7015	1.1612	-----	-----	-----	-----
NH₄	-----	-----	-----	-----	-----	-----	2.2119	2.0473	-----	-----	-----	-----
Acetone	-----	-----	-----	-----	-----	-----	5.9682	1.0175	-----	-----	-----	-----
ISO-butane	-----	-----	-----	-----	-----	-----	-----	-----	2.3543	1.1440	-----	-----
Hydrogen	-----	-----	-----	-----	-----	-----	-----	-----	2.4338	0.7558	-----	-----
Ethanol	-----	-----	-----	-----	-----	-----	-----	-----	2.5597	0.4436	-----	-----

Fuente: Elaboracion propia.

6.26.1 Montaje de la biblioteca. Con todos los modelos obtenidos de la sección anterior, el siguiente paso fue construir la librería para un hardware embebido con el fin de codificar ecuaciones en un dispositivo programable para la adquisición y monitorización de datos. La librería contiene sólo una clase descrita en la Figura 3, donde se especifican los métodos y variables globales para cada sensor MQ, este diagrama de clases fue realizado utilizando el lenguaje de programación Arduino, que parece estructurar C++ respetando la manipulación de variables y sentencias principales (tabla 12).

Tabla 12. Diagrama de clase para actuadores de control

MQUnifiedsensor
<pre> -kind: enum -pin: int int _pin, _type, _R0, _lecturePosInArray; String _nameLectureRequeired; int _VOLT_RESOLUTION = 5.0; // if 3.3v use 3.3 int _RLValue = 10; //Value in KiloOhms int _PPM; float _ratioInCleanAir, _sensor_volt, RS_air, _m, _b, _ratio; setKind(newKind) setPin(newPin) getKind(newKind) getPin(newPin) MQUnifiedsensor(int pin, int type); void inicializar(); void setR0(double R0); void setVoltResolution(float voltaje); void setSensorCharacteristics(String nameLectureRequeired, bool print); void setDefaultGas(); int readSensor(String nameLectureRequeired = "", bool print = false); int readPPM(int m, int b); long calibrate(boolean print = false); double getVoltage(int read = true); double stringToDouble(String & str); String getnameLecture(); </pre>

Fuente: Elaborcion propia.

La primera versión de código tuvo un consumo masivo de almacenamiento sobre el microcontrolador Atmega328, debido a las variables globales y su tipo de datos, probando arrays, listas, y otros, finalmente fue seleccionada como variables del compilador usando la instrucción `#define`, esto gasta 25% para las placas de desarrollo Arduino Mega 2560 y 45% para Arduino UNO. Por otro lado, la codificación toma en cuenta el paradigma OOP, esta es una estructura estándar desde los paradigmas básicos de programación como son las estructuras o procesos y los paradigmas concurrentes. Finalmente, para hacer escalable y repetible esta librería, los autores han liberado código en la plataforma GitHub bajo la integración de licencias del MIT, permitiendo al público en general modificar, descargar, usar y distribuir la librería reconociendo el trabajo de los autores sobre ella (Califa, Contreras & Carrillo, 2019).

6.26.2 Toma de datos confiables. En esta tabla se tomaron alrededor de 1500 valores, que, tomado hacia una columna analógica, estos valores retornan del Arduino en el conversor analítico digital que esta entre 0 y 1024 que su vez se convierten en voltajes, realizando una regla de tres para convertirlos en una escala de 0 a 5 voltios.

Llevando a un valor promedio en los primeros 500 valores, luego de esto se halla la R/s y esta su vez se encuentra el R/0 que es en sí es la calibración, y una vez hecho este procedimiento se pasa a ser confiable ver (tabla 13).

Tabla 13. Valores confiables aplicado por calibración matemática

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	TIME	MQ8 (A0)	MQ3 (A1)	MQ6 (A2)	MQ2 (A3)	MQ5 (A4)	MQ8 (A0)	MQ3 (A1)	MQ6 (A2)	MQ2 (A3)	MQ5 (A4)	MQ8 (A0)	MQ3 (A1)	MQ6 (A2)	MQ2 (A3)	MQ5 (A4)
2		Analog Values				Voltaje Values				AVG Voltaje Values						
3	4:12:42 PM	211	90,000000	129,00	47	145,000000	1,03027	0,43945	0,62988	0,22949	0,70801	1,0319697	0,44558115	0,6344	0,22859	0,7102
4	4:12:43 PM	210	90,000000	129,00	47	145,000000	1,02539	0,43945	0,62988	0,22949	0,70801					
5	4:12:44 PM	210	90,000000	129,00	47	145,000000	1,02539	0,43945	0,62988	0,22949	0,70801	38,451036	102,213005	68,8143	208,732	60,4023
6	4:12:45 PM	211	90,000000	130,00	47	145,000000	1,03027	0,43945	0,63477	0,22949	0,70801					
7	4:12:46 PM	210	90,000000	129,00	47	145,000000	1,02539	0,43945	0,62988	0,22949	0,70801	0,5493005	1,70355008	6,88143	21,2342	9,29266
8	4:12:47 PM	211	90,000000	129,00	47	145,000000	1,03027	0,43945	0,62988	0,22949	0,70801					
9	4:12:48 PM	210	91,000000	129,00	47	145,000000	1,02539	0,44434	0,62988	0,22949	0,70801	37,850467	106,363636	67,5758	203,333	60,6207
10	4:12:49 PM	211	90,000000	129,00	47	145,000000	1,03027	0,43945	0,62988	0,22949	0,70801					
11	4:12:50 PM	210	91,000000	129,00	47	145,000000	1,02539	0,44434	0,62988	0,22949	0,70801	68,906666	62,43646	9,82002	9,57575	6,5235
12	4:12:51 PM	211	90,000000	130,00	47	145,000000	1,03027	0,43945	0,63477	0,22949	0,70801					
13	4:12:52 PM	210	90,000000	129,00	47	145,000000	1,02539	0,43945	0,62988	0,22949	0,70801	53,1077	2,7059E+65	23,8031	7,47234	0,81392
14	4:12:53 PM	211	91,000000	129,00	47	145,000000	1,03027	0,44434	0,62988	0,22949	0,70801	53,1077	2,7059E+65	23,8031	7,47234	0,81392
15	4:12:54 PM	210	90,000000	130,00	47	145,000000	1,02539	0,43945	0,63477	0,22949	0,70801					
16	4:12:55 PM	211	90,000000	129,00	47	145,000000	1,03027	0,43945	0,62988	0,22949	0,70801					
17	4:12:56 PM	211	90,000000	129,00	47	145,000000	1,03027	0,43945	0,62988	0,22949	0,70801					
18	4:12:57 PM	211	91,000000	130,00	47	145,000000	1,03027	0,44434	0,63477	0,22949	0,70801					
19	4:12:58 PM	211	90,000000	129,00	47	145,000000	1,03027	0,43945	0,62988	0,22949	0,70801	976,97	2E+31	88158	987,99	1163,8
20	4:12:59 PM	211	90,000000	130,00	46	146,000000	1,03027	0,43945	0,63477	0,22461	0,71289	-0,688	19,01	-3,597	-2,162	-3,874
21	4:13:00 PM	210	91,000000	129,00	47	145,000000	1,02539	0,44434	0,62988	0,22949	0,70801	H2	CH4	H2	H2	H2
22	4:13:01 PM	211	90,000000	129,00	47	145,000000	1,03027	0,43945	0,62988	0,22949	0,70801					
23	4:13:02 PM	210	90,000000	129,00	47	145,000000	1,02539	0,43945	0,62988	0,22949	0,70801					
24	4:13:03 PM	211	90,000000	130,00	47	145,000000	1,03027	0,43945	0,63477	0,22949	0,70801					
25	4:13:04 PM	211	90,000000	130,00	47	146,000000	1,03027	0,43945	0,63477	0,22949	0,71289					
26	4:13:05 PM	210	90,000000	129,00	47	145,000000	1,02539	0,43945	0,62988	0,22949	0,70801					
27	4:13:06 PM	210	91,000000	129,00	47	145,000000	1,02539	0,44434	0,62988	0,22949	0,70801					
28	4:13:07 PM	211	90,000000	130,00	46	146,000000	1,03027	0,43945	0,63477	0,22461	0,71289					
29	4:13:08 PM	211	90,000000	130,00	46	145,000000	1,03027	0,43945	0,63477	0,22461	0,70801					

Fuente: Elaboracion propia.

En el caso para este trabajo solamente se obtienen el MQ4 y el MQ135, pero en el repositorio <https://github.com/Yercar18/Dronefenix> se deja todas las librerías y las calibraciones de estos sensores y lo cual se constituye como un gran aporte que se hace desde el trabajo de la maestría en ingeniería ambiental, brindando un aporte a la comunidad científica, debido a que esto genera una confiabilidad para el uso de estos sensores para la medición de la calidad del aire.

6.27 PCB

Como se quería que la estación fuera compacta y precisa se llevó a cabo un diseño de circuito impreso que se conectara a través del Arduino con un conjunto de componentes electrónicos ver (figura 50).

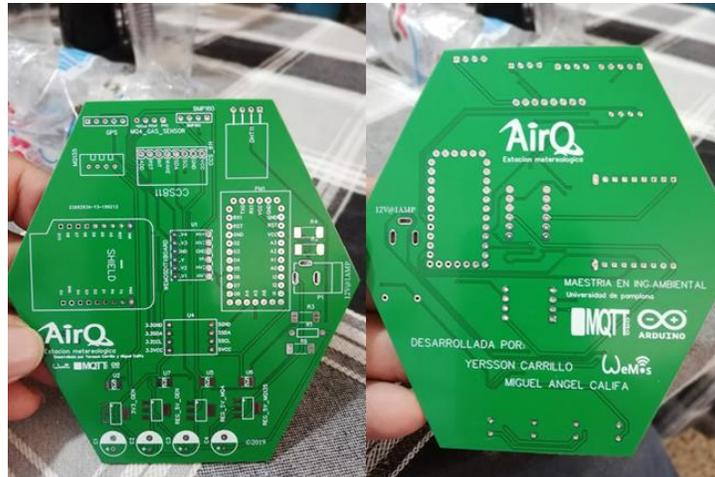


Figura 49. PCB placa de circuito impreso

Fuente: Elaboracion propia.

Instalación de los componentes a la PCB, teniendo la PCB se procede al armado de la estación donde es más fácil, se puede llevar cabo sin ninguna dificultad debido que todo está señalado e indicado donde debe ir se procede al ensamble ver (figura 51 y 52).

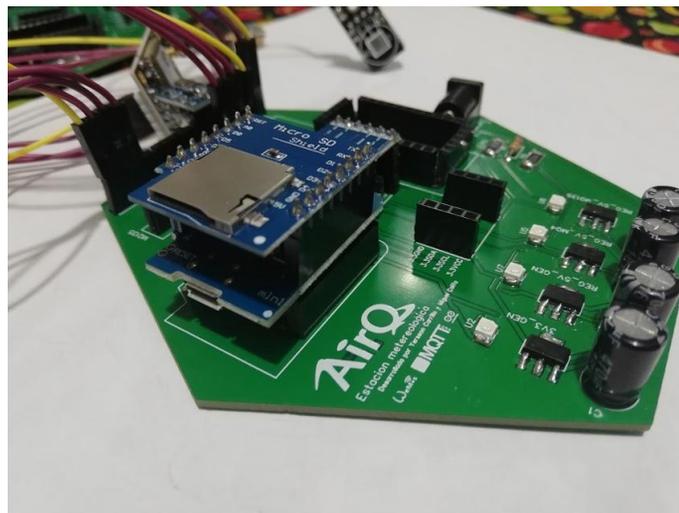


Figura 50. PCB con Arduino mini y wemos

Fuente: Elaboracion propia.

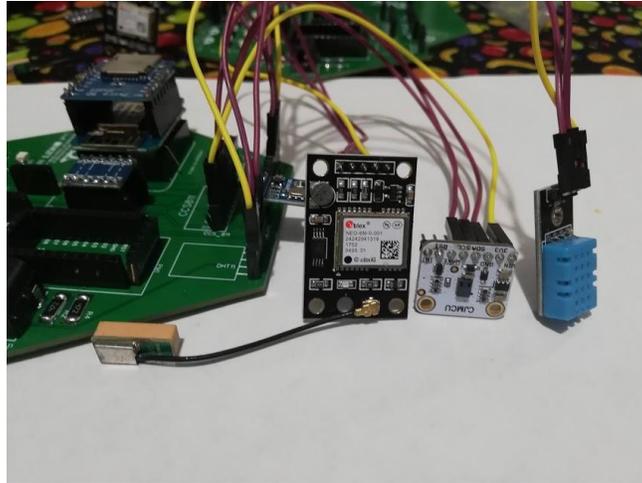


Figura 51. PCB con componentes de temperatura, GPS y voc

Fuente: Elaboracion propia.

6.28 Estación Final

La estación ensamblada y calibrada ver (figura 53) comenzó a registrar variable, esta estación se caracteriza por la toma de datos precisos de temperatura, humedad, presión atmosférica, alcohol, amonio, metanol, dióxido de carbono, compuestos orgánicos volátiles; a su vez tiene su geo posicionamiento, precisando fecha, hora y lugar en la cual se registran los datos.



Figura 52. Estación terminada

Fuente: Elaboracion propia.

ID Base de datos	Temperatura	Humedad	Presion Atmosferica	Alcohol	CH4	NH4	CO2	VOC	Latitud	Longitud	echa Estacio	Fecha sincronizacion
2847239	30	63	978	4,4487247	1,7491703	0,7381469	398	301	7,8930538	-72,509992	/2020 -20-48	10/03/2020 1:48
2847238	30	63	978	2,405373	2,0016783	2,6787202	405	3064	7,8930538	-72,509992	/2020 -20-48	10/03/2020 1:48
2847237	30	63	978	2,3142772	1,9066711	2,3676564	398	3015	7,8930538	-72,509992	/2020 -20-48	10/03/2020 1:48
2847236	30	63	978	0,1971146	2,4883749	0,0870683	401	3035	7,8930538	-72,509992	/2020 -20-48	10/03/2020 1:48
2847235	30	63	978	3,2836947	0,5898793	0,0387461	401	3035	7,8930538	-72,509992	/2020 -20-48	10/03/2020 1:48
2847234	30	63	978	2,3734877	1,0042249	0,7488466	401	3035	7,8930538	-72,509992	/2020 -20-48	10/03/2020 1:48
2847233	30	63	978	1,2192645	1,4112536	1,8615918	395	2993	7,8930538	-72,509992	/2020 -20-48	10/03/2020 1:48
2847232	30	63	978	3,4569976	1,7862174	2,7084915	395	2993	7,8930538	-72,509992	/2020 -20-48	10/03/2020 1:48
2847231	30	63	978	3,6396753	2,7935516	1,4163425	398	3015	7,8930538	-72,509992	/2020 -20-48	10/03/2020 1:48
2847230	30	63	978	2,835096	1,2562609	1,1781315	395	2993	7,8930538	-72,509992	/2020 -20-48	10/03/2020 1:48
2847229	30	63	978	3,5364587	1,0749816	2,0092826	391	2967	7,8930538	-72,509992	/2020 -20-48	10/03/2020 1:48
2847228	30	63	978	1,3691133	1,0794923	2,9330576	395	2993	7,8930538	-72,509992	/2020 -20-48	10/03/2020 1:48
2847227	30	63	978	4,0446855	0,3348836	0,3939844	395	2993	7,8930538	-72,509992	/2020 -20-4	10/03/2020 1:48
2847226	30	63	978	1,6087856	0,6444037	0,3262044	391	2967	7,8930538	-72,509992	/2020 -20-4	10/03/2020 1:48
2847225	30	63	978	4,4963518	0,5104423	0,4587782	391	2967	7,8930538	-72,509992	/2020 -20-4	10/03/2020 1:48
2847224	30	63	978	1,2709406	2,4364872	0,894743	395	2993	7,8930538	-72,509992	/2020 -20-4	10/03/2020 1:48
2847223	30	63	978	0,2737559	1,1370387	2,1924328	391	2967	7,8930538	-72,509992	/2020 -20-4	10/03/2020 1:48
2847222	30	63	978	2,5850806	1,1779432	1,2365712	391	2967	7,8930538	-72,509992	/2020 -20-4	10/03/2020 1:48
2847221	30	63	978	4,150441	0,5354188	0,7300148	391	2967	7,8930538	-72,509992	/2020 -20-4	10/03/2020 1:48
2847220	30	63	978	3,4063629	2,0290443	1,0137684	388	2948	7,8930538	-72,509992	/2020 -20-4	10/03/2020 1:48
2847219	30	63	978	3,3028483	0,8672399	1,3910728	388	2948	7,8930538	-72,509992	/2020 -20-4	10/03/2020 1:48
2847218	30	63	978	2,2560738	2,5950032	2,9140832	388	2948	7,8930538	-72,509992	/2020 -20-4	10/03/2020 1:48

Figura 53. Datos obtenidos por la estación visualizados en Excel

Fuente: Elaboracion propia.

En la toma de datos se observan las gráficas constantes sin ningún salto.

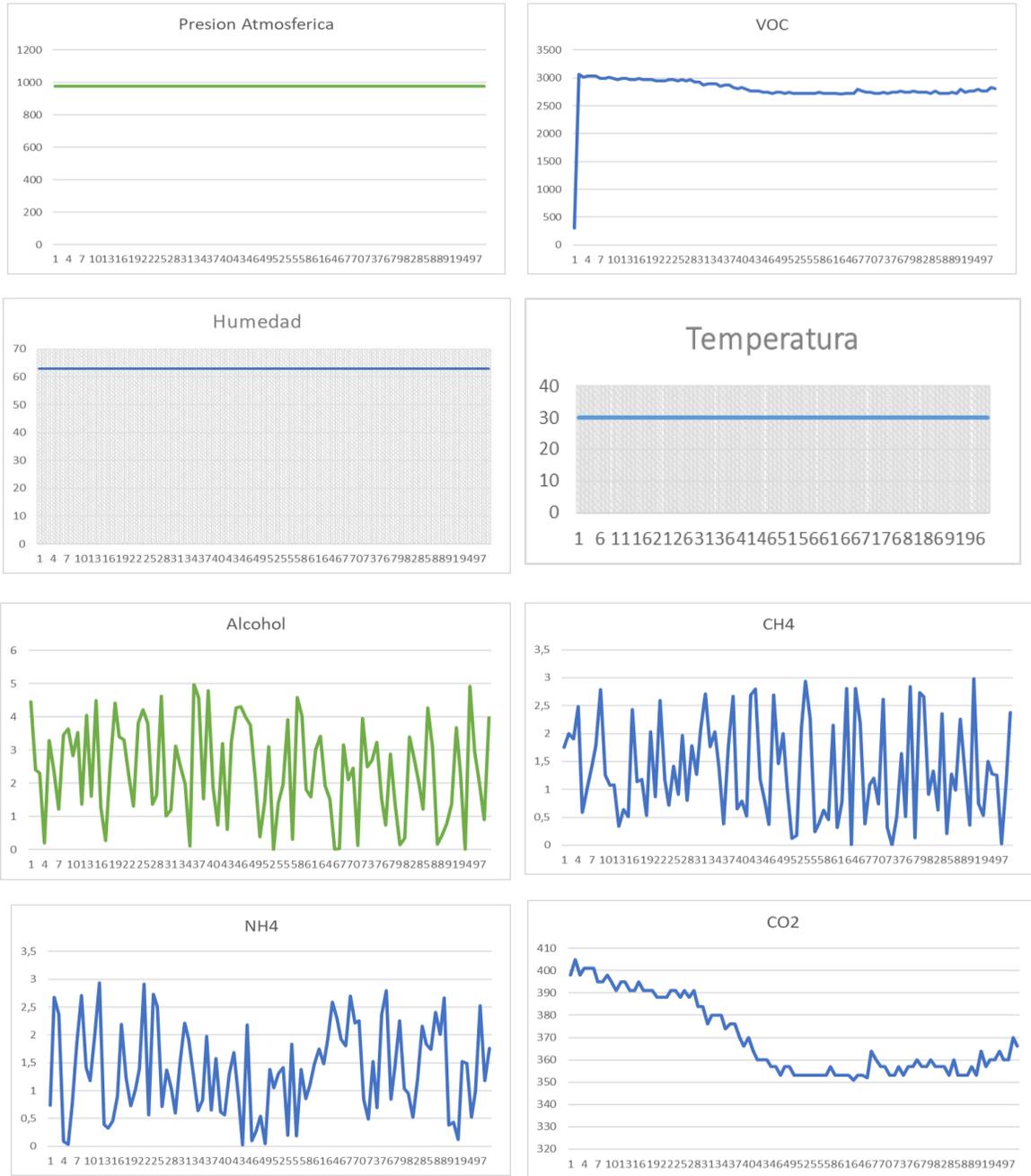


Figura 54. Datos tomados por la estación

Fuente: Elaboracion propia.

Esta estación meteorológica recoge los datos suficientes para monitorear la calidad del aire ver (figura 54 y 55) que son de interés para controlar el aumento de gases contaminantes en las pequeñas ciudades o donde sea requeridas. Pueden estar instaladas en cualquier terreno y parte del mundo, además de no tener un gran costo de elaboración, donde todo está a la mano de quien la quiera construir, debido a que las librerías son de adquisición abierta que permiten gran utilidad.

6.29 Plataforma

Después de realizar un análisis literario sobre las típicas estructuras o tendencias a la hora de elaborar plataformas, (campusmvp, 2020) se seleccionó database first, definido la necesidad como el requerimiento de tener una plataforma centralizada donde se reciba, administre y visualice la información transmitida por los cientos de estaciones meteorológicas que se encuentren instaladas en el territorio nacional.

Basados en ello se ha definido un modelo de base de datos que contiene una tabla de usuarios, una tabla de cosas (Cosas por el concepto de internet de las cosas) y una tabla datos de cosas (con 11 posibles columnas de datos por cada registro), las cuales se relacionan entre sí para garantizar la integridad y coherencia de los datos.

Finalmente, y con el fin de reducir periódicamente el tamaño de la base de datos se ha agregado tres tablas adicionales, datos de cosas semanales, mensuales y anuales. Con ello conseguimos que cada semana, mes y año se elimine la data progresivamente, se calcule un promedio y se almacene únicamente el valor promedio, máximo, mínimo y la cantidad de datos procesados, consiguiendo con ello mantener un registro histórico que está compactado.

Posteriormente, y sobre esta estructura sólida se implementa un servidor nodeJs para generar el API, el cual recibe las peticiones del servidor MQTT broker y las peticiones del frontend, y un frontend hecho con tecnología VueJs para mantener la eficiencia y seguridad informática ante posibles ataques cibernéticos.

Toda esto es implementado en un VPS proveido por el servidor cloud digital ocean, con 20 GB de disco duro SSD, 1 TB de tasa de transferencia limite mensual, 1 GB de memoria RAM y con capacidad de ampliarse al compa que crece el número de usuarios.

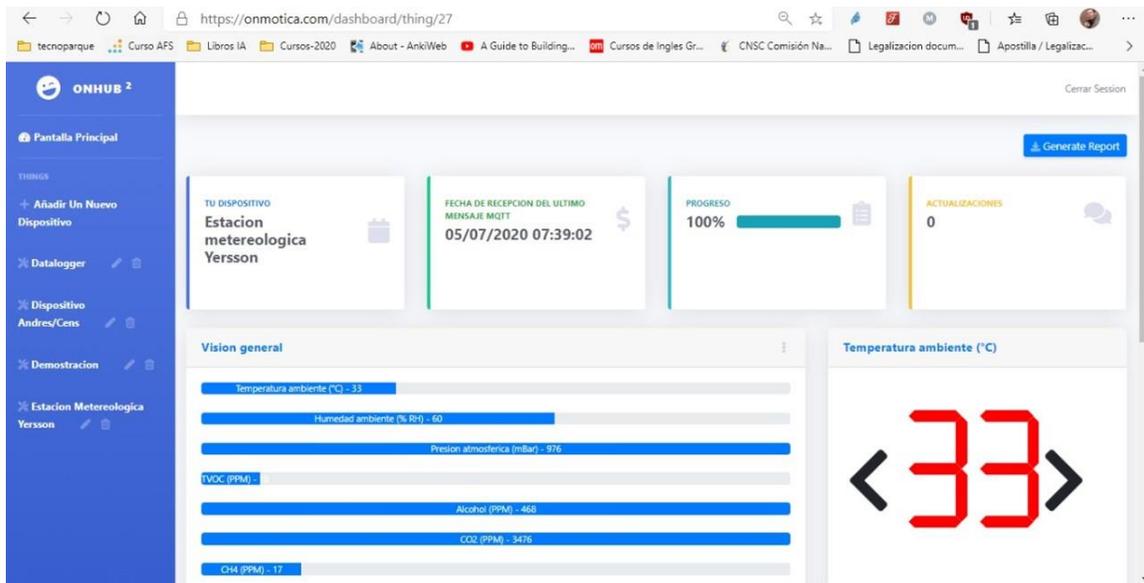


Figura 55. Entorno de la plataforma

Fuente: Elaboracion propia.

La visualización consta de 3 elementos principales ver (figura 56), la barra lateral izquierda donde estan los botones con los cuales se puede seleccionar un dispositivo, o crear uno nuevo, la barra de navegación donde se encuentra la opción de cierre de sesión, el panel de notificaciones donde encuentra información relevante del dispositivo, el gráfico de barras donde se ve la última

lectura registrada por el sistema y el panel led derecho donde encuentra la ultima lectura reflejada en dos display de 7 segmentos.

Por último, un botón denominado generar reporte donde al hacer click, se despliega un formulario en el cual selecciona el rango desde y hasta el cual desea imprimir el reporte.

7. Conclusiones

Este trabajo es de trascendencia porque se creó un dispositivo de tamaño pequeño con grandes características y componentes de gran precisión para la toma de datos meteorológicos y de gases nocivos presentes en el aire para proporcionar datos de contaminantes en lugares determinados.

Esta estación permite llevar una medición de la contaminación atmosférica en lugares donde no haya un control de la calidad del aire, debido a que puede ayudar en la toma de decisiones frente a los agentes contaminantes presentes en el aire del lugar inspeccionado ya que los contaminantes presentes en el aire son una clara muestra de daños en la salud humana.

Se estableció un modelo matemático de calibración y estandarización para las mediciones de la calidad del aire a través de los sensores MQ, en el cual se empleó una relación matemática con base en la información proporcionada por los fabricantes de los sensores a través de sus hojas de datos, utilizando el modelo de regresión matemática lineal que permitió crear las librerías de calibración para los sensores de la serie MQ, las cuales sirven para realizar estudios en el medio ambiente, contaminación del aire y análisis de procesos industriales.

Se desarrolló un sistema de adquisición y plataforma de visualización de datos meteorológicos para la medición de la calidad del aire a través de Arduino, donde se comprobó la toma específica de los agentes contaminantes presentes en el aire, y a su vez a través de la plataforma se puede realizar un seguimiento en tiempo real.

8. Recomendaciones

Se recomienda hacer un estudio de la calidad del aire a partir de la toma de datos usando la estación meteorológica en la cual se puedan identificar puntos neurálgicos y conocer el panorama la contaminación atmosférica en la ciudad de Pamplona y San José de Cúcuta.

Generar un sistema de alerta temprana en torno a la contaminación del aire usando las estaciones meteorológicas propuestas en este trabajo y llevarlo como propuesta de Doctorado.

Bibliografía

- Adafruit. (s,f). Adafruit CCS811 Air Quality Sensor. Recuperado de: <https://cdn-learn.adafruit.com/downloads/pdf/adafruit-ccs811-air-quality-sensor.pdf>
- Aprendinedo, A. (2018). *aprendiendoarduino.wordpress.com*. Recuperado de: aprendiendoarduino.wordpress.com:
<https://aprendiendoarduino.wordpress.com/2018/04/14/sensores-arduino-3/>
- Arduino Community. (2020). Logo. Recuperado de: <https://www.arduino.cc/en/Trademark/CommunityLogo>
- ARDUINO. (2019). Arduino pro mini. Recuperado de: <https://store.arduino.cc/usa/arduino-pro-mini>
- ARDUINO. (2019). Que es arduino. Recuperado de: <http://arduino.cl/que-es-arduino/>
- Biblioteca del Congreso Nacional de Chile. (15 de Noviembre de 2018). *la-contaminacion-del-aire-en-latinoamerica*. Recuperado de: www.bcn.cl/observatorio/americas/noticias/
- Built With Wimba Create. (s,f). Estructura básica de un programa. Recuperado de: http://dfists.ua.es/~jpomares/arduino/page_04.htm
- Califa, M., Contreras, G. & Carrillo, Y. (2019). *miguel5612/MQ Sensors Lib: Arduino Preview V1.03* Recuperado de: <https://zenodo.org/record/3384301#.XZ9Fmma221s>
- Carrillo-Amado, Y. R., Califa-Urquiza, M. A., & Ramón-Valencia, J. A. (2020). Calibración y estandarización de mediciones de calidad del aire usando sensores MQ. *Respuestas*, 25(1).
<https://doi.org/10.22463/0122820X.2408>

Control Automatico Educacion. (s,f). Salidas digitales. Recuperado de:

<https://controlautomaticoeducacion.com/arduino/salidas-digitales/>

DHT11 Humidity & Temperature Sensor. (s,f). DHT11 Technical Data Sheet Translated Version

1143054. Recuperado de: <https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf>

BMP180 Data sheet. (s,f). BMP180 Digital pressure sensor. Recuperado de: [https://cdn-](https://cdn-shop.adafruit.com/datasheets/BST-BMP180-DS000-09.pdf)

[shop.adafruit.com/datasheets/BST-BMP180-DS000-09.pdf](https://cdn-shop.adafruit.com/datasheets/BST-BMP180-DS000-09.pdf)

Naylamp Mechatronics SAC . (2016). Tutorial Arduino y memoria SD y micro SD. Recuperado

de: https://naylampmechatronics.com/blog/38_Tutorial-Arduino-y-memoria-SD-y-micro-SD-.html

Electro Componentes. (s,f). condensadores electroliticos condensadores electroliticos. Recuperado

de: <https://www.electrocomponentes.es/condensadores-electroliticos/condensadores-electroliticos-100uf-16v-20-5x11mm-124-.html>

Codigo Electronica. (s,f). Resistencia eléctrica. Recuperado de:

<http://codigoelectronica.com/blog/resistencia-electrica>

Sparkfun. (s,f). Technical data mq-6 gas sensor. Recuperado de:

<https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-6.pdf>

Ferdoush, S. & Li, X. (2014). Wireless Sensor Network System Design using Raspberry Pi and

Arduino for Environmental Monitoring Applications. *Procedia Computer Sciencie*, 34(10-110).

Final Test S.A. (2019). ¿Que es un Datalogger?. Recuperado de:

<https://www.finaltest.com.mx/product-p/art-4.htm>

Herrador, R. E. (2009). Guía de Usuario de Arduino. *I.T.I. Sistemas, I(1)*.

IDEAM. (2014). Participación ciudadana. Recuperado de:

<http://www.ideam.gov.co/web/atencion-y-participacion-ciudadana/glosario>

IDEAM. (2015). Calidad del aire. <http://www.ideam.gov.co/web/contaminacion-y-calidad-ambiental/calidad-del-aire>

Ingeniería Mecafenix. (2017). Arduino ¿que es, como funciona? y sus partes. Recuperado de:

<https://www.ingmecafenix.com/electronica/arduino/>

Instituto de Hidrología, Meteorología y Estudios Ambientales. (2015). Calidad del Aire.

Recuperado de: <http://www.ideam.gov.co>

Instituto de Hidrología, Meteorología y Estudios Ambientales. (2008). Protocolo para el monitoreo y seguimiento de la calidad del aire. Bogotá: IDEAM.

Instituto de Hidrología, Meteorología y Estudios Ambientales. (2017). *Informe del Estado de la Calidad del Aire en Colombia*. Bogota : IDEAM.

Instituto de Investigaciones Agropecuarias. (2006). *Redes de estaciones meteorologicas automaticas*. Chile: Trama Impresores S.A.

Loza, E.. (2013). *La dimensión social de la universidad del siglo xxi creación del programa de aprendiazaje-servicio en la universidad técnica de ambato*. madrid: universidad complutense de madrid . Universidad Técnica de Ambato. Ambato, Ecuador.

Ministerio de Ambiente Sostenible. (2019). Recuperado el 26 de Enero de 2019, de <http://www.minambiente.gov.co>: <http://www.minambiente.gov.co/index.php/asuntos-ambientales-sectorial-y-urbana/gestion-del-aire/contaminacion-atmosferica>

Ministerio de Ambiente, Vivienda y Desarrollo Territorial. (2008). Protocolo para el monitoreo y seguimiento de la calidad del aire. en mavdt, & mavdt (ed.), *protocolo para el monitoreo y seguimiento de la calidad del aire* (pág. 285). Bogota : MAVDT .

Ministerio de Ambiente, Vivienda y Desarrollo Territorial. (2010). Por la cual se modifica la Resolución 601 del 4 de abril de 2006”. Recuperado de: <http://www.ideam.gov.co/>: <http://www.ideam.gov.co/documents/51310/527391/Resoluci%C3%B3n+610+de+2010+-+Calidad+del+Aire.pdf/aba3635e-af6e-4990-a863-29ace89b413a>

Ochoa, A., Denisse, L. & Delgado, T. (2018). Iternativa Open Source en la implementación de un sistema IoT para la medición de la calidad del aire. . *Revista Cubana de Ciencias Informáticas* , 189-204.

Olimex. (s,f). Technical data MQ-135 Gas Sensor. Recuperado de: <https://www.olimex.com/Products/Components/Sensors/Gas/SNS-MQ135/resources/SNS-MQ135.pdf>

Organización Mundial de la Salud (2014). 7 millones de muertes cada año debidas a la contaminación atmosférica. Recuperado de: <https://www.who.int/mediacentre/news/releases/2014/air-pollution/es/>

Organización Mundial de la Salud. . (2006). *Guías de calidad del aire de la OMS relativas al material particulado, el ozono, el dióxido de nitrógeno y el dióxido de azufre*. Ginebra: OMS.

PCE. (2018). Estación meteorológica . Recuperado de: www.pce-iberica.es: <https://www.pce-iberica.es/medidor-detalles-tecnicos/que-estacion-meteorologica.htm>

Pérez, J. (2017). La calidad del aire en colombia: un problema de salud pública, un problema de todos. *BIOSALUD* , 5-6.

Ramírez , L., & Ávila, M. (Abril de 2017). Evaluación de precipitación y evaporación en estaciones climatológicas convencionales y automatizadas. *Novum Scientiarum*, 2(10-24).

Romero , M., Diego, F. & Alvarez, M. (2006). La contaminación del aire: su repercusión como problema de salud . *Rev Cubana Hig Epidemiol* , II(44).

Sánchez, E. (marzo de 2012). Diseño de un sistema de control domótico basado en la plataforma Arduino. *Etsinf*, 1(1).

Sistema de Informacion Ambiental de Colombia. (16 de Noviembre de 2018). Monitoreo del aire. Recuperado de: <http://www.siac.gov.co/>

Sistema de Informacion Ambiental de Colombia. (2018). Recuperado el 18 de Enero de 2019, de <http://www.siac.gov.co>: <http://www.siac.gov.co/monitoreoaire>

Sparkfun. (s,f). Technical data MQ-4 Gas Sensor. Recuperado de: <https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-4.pdf>

Ublox. (2011). *NEO-6 u-blox 6 GPS Modules*. EU: ARM .

U-blox. (2011). NEO-6. Recuperado de: https://www.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_%28GPS.G6-HW-09005%29.pdf

Valls, J. M. (Agosto de 2017). Efectos de la calidad del aire sobre la salud. *Actualizaciones*(511-514).

World Health Organization. (2000). Air Quality Guidelines for Europe. Copenhagen: World Health Organization.

SparkFun Electronics. (s,f). Arduino Pro Mini 328 - 5V/16MHz. Recuperado de:

<https://www.sparkfun.com/products/11113>

Bastelgarage. (2020). GPS Modul NEO-6M kompatibel. Recuperado de:

<https://www.bastelgarage.ch/gps-modul-neo-6m-kompatibel>.

Llamas, L. (2008). Adaptar 3.3V a 5V (y viceversa) en arduino con level shifter. Recuperado de:

<https://www.luisllamas.es/arduino-level-shifter/>

Agspecinfo. (s,f). Dev-11113: arduino pro mini 328 5 v a 16 MHZ. Recuperado de:

<http://www.agspecinfo.com/pdfs/D/DEV11113.PDF>

Prometec. (s,f). Wemos D1 ESP8266 WIFI. Recuperado de: [https://www.prometec.net/wemos-](https://www.prometec.net/wemos-d1-esp8266-wifi/)

[d1-esp8266-wifi/](https://www.prometec.net/wemos-d1-esp8266-wifi/)

Neitzel, F., & Klonowski, J. (2011). Mobile 3d mapping with a low-cost uav system .

International Archives of the Photogrammetry, Remote Sensing and Spatial Information

Sciences, 1-6.