

# Desarrollo e implementación de un sistema en un drone para evaluar la calidad del aire

José Sebastián Borja Porras<sup>1</sup>, Dr. Luis Carlos Méndez González<sup>2</sup>

**Resumen**— En esta investigación se propone el desarrollo y la integración de un sistema meteorológico en un drone el cual permite detectar las concentraciones de ozono, monóxido de carbono, material particulado, así como temperatura y humedad por medio de distintos sensores de bajo costo. Los datos son transmitidos en tiempo real por radiofrecuencia a 433mhz y posteriormente procesados utilizando la plataforma de Arduino. Al ser un sistema dinámico que puede desplazarse libremente por el aire, nos permite recopilar datos de distintas zonas y compararlos, esta información nos ayuda a conocer la calidad del aire y nos permiten tener un mejor entendimiento sobre las partículas contaminantes que se encuentran en el mismo. Gracias a esta información podemos detectar donde se encuentran las principales fuentes que perjudican el aire que respiramos y así tomar medidas para reducir los niveles de estas partículas dañinas para el ser humano.

**Palabras clave**— Calidad del aire, Drone, Arduino, Sensores

## Introducción

En la actualidad se ha visto un incremento en la contaminación del aire principalmente en zonas urbanas desarrolladas, en zonas donde el uso de vehículos de combustión interna es el principal medio de transporte y en lugares donde no se tiene una correcta conciencia sobre el impacto ambiental. En muchas ocasiones las personas desconocen los niveles de contaminación que hay en el lugar en el que viven ya que no tienen los medios para medir la concentración de sustancias contaminantes o los equipos con los que cuentan en la ciudad donde viven son poco accesibles y están limitados a zonas específicas.

Según reportes ecológicos en 2016 Ciudad Juárez es la tercera ciudad más contaminante de la república mexicana, esto se debe principalmente a los contaminantes vehiculares, restos de animales muertos expuestos en la vía pública y a la basura tirada.

Actualmente la ciudad cuenta con estaciones de monitoreo fijas en ciertos puntos que nos pueden describir la calidad del aire en zonas específicas de la ciudad, sin embargo, estas estaciones son fijas y no abarcan todas las regiones de la ciudad, dejando puntos en los que se desconoce el nivel de contaminación.

Existen diferentes partículas que son las que provocan el deterioro de la calidad del aire, tales como el material particulado PM10, el ozono troposférico y el monóxido de carbono que en altas concentraciones llegan a ser muy tóxicos para el ser humano, varios estudios nos han dado indicios de que la exposición prolongada a estos gases puede traer como consecuencia diferentes efectos en la salud como enfermedades cardiovasculares y respiratorias.

El principal objetivo de este proyecto es desarrollar un sistema que nos permita conocer los niveles de las distintas partículas contaminantes que se encuentran en el aire por medio de sensores acoplados a un drone y transmitirlos en tiempo real hasta una computadora portátil con la ayuda del software de Arduino IDE. Este es un sistema dinámico que no necesita de la instalación de una gran infraestructura y equipo costoso para realizar sus pruebas, además de que gracias a sus características puede abarcar amplias zonas e incluso llegar a las partes difíciles de alcanzar por el ser humano.

## Descripción del Método

### *Selección de componentes*

#### *Microcontrolador*

En este estudio se utilizó el microcontrolador código abierto Arduino debido su bajo costo y sus relativamente poderosas características. Son fáciles de programar, son ampliamente utilizadas, existe vasta documentación, por lo que nos proporciona muchos beneficios ante otras plataformas. Los microcontroladores son pequeños sistemas computacionales para aplicaciones de baja potencia y baja memoria, estos consisten en un circuito integrado en una tablilla con capacidad de lectura, escritura, memoria, entradas y salidas. Hoy en día existe una gran variedad de placas de Arduino disponibles basadas en diferentes microcontroladores, de diferentes tamaños y precios los cuales nos pueden ofrecer soluciones para distintas tareas específicas.

<sup>1</sup> José Sebastián Borja Porras es Alumno de 12° semestre de la carrera de Ingeniería Mecatrónica de la Universidad Autónoma de Ciudad Juárez, Chihuahua, México. [sebastian.borja94@outlook.com](mailto:sebastian.borja94@outlook.com)

<sup>2</sup> El Dr. Luis Carlos Méndez González es Profesor de tiempo completo de la carrera de Ingeniería Mecatrónica en la Universidad Autónoma de Ciudad Juárez, Chihuahua, México. [luis.mendez@uacj.mx](mailto:luis.mendez@uacj.mx)

Arduino Nano, basado en ATmega328P es la placa que se empleó para el circuito principal, debido a sus reducidas dimensiones (18mm x 45mm) y su peso ligero (7g) es ideal para ser montado en equipos movibles, además cuenta con 30 puertos, 8 analógicos y 22 digitales las cuales son suficientes para realizar las conexiones necesarias con los sensores. La principal función del Nano es leer los datos de los sensores y procesarlos para poder enviarlos a una base en tierra.

Para el circuito de recepción de la señal se utilizó un Arduino Mega 2560 basado en ATmega2560, este microcontrolador está diseñado para proyectos más complejos, con 54 pines de los cuales 16 son entradas análogas y una entrada de alimentación nos proporciona una gran amplitud de posibilidades. A pesar de ser más grande (101.55mm x 53.3) y pesado (37g) que el Arduino Nano, el Mega2560 será utilizado como base terrestre de manera que esto no presenta ningún inconveniente. Este dispositivo será conectado a una computadora mediante el puerto USB y a través del software Arduino IDE y de su monitor serial podremos observar las distintas mediciones de los sensores.

#### *Sensores*

Los sensores montados en el dron serán los encargados de monitorear la calidad del aire mediante la medición de ozono, monóxido de carbono, material particulado, temperatura y humedad. Para poder ser utilizados correctamente y obtener resultados precisos estos sensores deben de cumplir una serie de características

Consumo de energía. La energía consumida es una importante restricción ya que el Arduino tiene un límite de corriente permitida por pin, exceder estos límites nos puede traer como consecuencia que se quemé dicho pin. Además, al estar utilizando la misma batería para alimentar al dron, un consumo alto de corriente puede reducirnos el trabajo que puede realizar el dron, así como limitar su tiempo de vuelo.

Tamaño. Los sensores deben de contar con un tamaño reducido para poder montarse en el dron

Peso. Los motores en conjunto con las hélices del dron tienen un límite de empuje que pueden realizar, al sobrepasar estos límites el dron no podrá mantenerse en vuelo.

Precisión de medición. Las medidas tomadas deberán estar dentro de los parámetros observables

Bajo costo. Al tener un presupuesto limitado se tendrá que tener especial cuidado en este punto

Disponibilidad. Es la posibilidad de adquirir los sensores en Cd. Juárez en un periodo corto.

#### *Selección de sensor de monóxido de carbono*

Sí bien en el mercado existe una gran variedad de sensores para medir las concentraciones de monóxido de carbono, la mayoría son equipos costosos que son utilizados para medir las concentraciones en interiores, otros sensores que pueden ser utilizados para medir concentraciones en ambientes externos como el 3SP\_CO\_1000 Package 110-102 y el AS-MLV-P2 tienen un peso ligero y dimensiones reducidas, pero tienen un precio más elevado y su disponibilidad es menor. El sensor Mq2 el cual ha sido utilizado en distintos proyectos y del cual existe una mayor documentación, cuenta con un peso aproximado de 9 gramos, un consumo cerca de 900mw, un precio de 5.29 dólares y unas dimensiones de 35mm x 28mm x 23, además que es sencillo de programar, lo que lo hace el más adecuado para esta aplicación.

#### *Sensor de ozono*

En este caso se optó por usar el sensor Mics2614 por sus reducidas dimensiones de 5mm x 7mm x 1.1mm, su reducido consumo energético de 100mw y su rango de detección de 10 a 1000ppb, aunque debido su costo de 40 dólares y su largo periodo de entrega de aproximadamente 4 meses, se decidió utilizar el sensor Mq131, el cual es un sensor más utilizado y con una mayor disponibilidad. Este sensor tiene un precio de 17 dólares, un rango de detección de 10 a 2000 ppb, un peso de 9g, un consumo energético de 900mw, dimensiones de 32mm x 22mm x 30mm y necesita de un único periodo de 48h de precalentamiento para su correcto funcionamiento.

#### *Sensor de temperatura*

Conocer la humedad relativa tanto como la temperatura es indispensable para observar que los sensores estén dentro de los parámetros en que las mediciones se hacen de manera correcta. El DHT11 es un módulo que utiliza un capacitor y un termistor calibrados digitalmente para detectar los niveles de humedad relativa y temperatura. Este sensor cuenta con un microcontrolador de 8 bits de alto rendimiento lo que nos proporciona una salida digital de las variables a medir. Este sensor es de muy bajo costo con un precio de 6 dólares. Al ser capaz de medir dos variables, este módulo nos presenta una ventaja al reducir el número de sensores a utilizar. El DHT11 Es de los mejores en términos de precisión con un error del 5% en humedad relativa de 5% a 95% y un error de  $\pm 2^{\circ}\text{C}$  para temperaturas de  $-20^{\circ}\text{C}$  a  $60^{\circ}\text{C}$ . Aunque el sensor DHT22 presenta una mayor precisión y un mayor rango de lecturas, el DHT11 es adecuado para este propósito además tener dimensiones más reducidas de tan solo 15.5mm x 12mm x 5.5mm, un peso 2.5g y una corriente de suministro de 0.5mA para lecturas y  $60\mu\text{A}$  en reposo.

#### *Sensor de material particulado*

Para este proyecto se utilizó el sensor Shinyei PPD42 que mide las concentraciones de material particulado PM10. Para realizar las mediciones, primero el aire pasa por dentro del sensor, pasa a través de una resistencia que

funciona como elemento de calentamiento, al estar dentro del sensor las partículas atraviesan un punto de detección que se encuentra en medio de un LED infrarrojo y un receptor de luz que recibe la luz dispersada a través de un lente que lo convierte en una señal y esto en un pulso. Este sensor aplica un filtro pasa bandas para remover las partículas más pequeñas o el ruido (El tamaño mínimo de las partículas detectadas es de 1µm), después el sensor cuenta el tiempo en el detectó las partículas independientemente de su tamaño. Esta cantidad de tiempo llamado LPO (low pulse occupancy) se puede considerar como la opacidad del aire circulando a través del sensor. El PPD42 tiene unas dimensiones de 59mm x 45mm x 22mm, un peso de 24g y un consumo de corriente de 90mA además de ser un sensor disponible en distintas partes del mundo con una entrega rápida y un costo de 9 dólares.

*Módulo de transmisión de datos*

Si bien Xbee nos proporciona una amplia variedad de soluciones para los problemas de transmisión de datos inalámbricamente, para este caso se utilizó un módulo de radiofrecuencia (RF) de 433mhz el cual es de muy bajo costo (6.95 dólares) e incluye el transmisor y receptor, el transmisor que es el cual nos enviará los datos procesados por el Arduino, tiene unas dimensiones de 22mm x 19mm x .5mm y un peso de aproximante 5 gramos lo cual lo hace un excelente elemento para ser cargado por el drone. Este dispositivo tiene un poder de transmisión de 10mw y transmite a 433mhz de amplitud modulada con un ratio de transferencia de hasta 4kbps y nos puede transmitir desde unos pocos centímetros hasta 200 metros dependiendo de la antena y el voltaje que se le suministre (3.5v a 12v) siendo más que suficiente para abarcar las distancias a las que se encontrará volando nuestro sistema.

*Selección del drone*

Para la selección del drone se tomó como factor principal el peso que pueda llevar sin que influya de manera significativa en la distancia máxima de vuelo que pueda alcanzar, ni que afecte el control de este, además de que se encuentre a un precio accesible. Se hizo una comparación entre distintos modelos y se encontró que existe una gran cantidad de drones capaces de levantar un peso mayor a 1kg tales como el 3D ROBOTICS X8+ y el DJI S900 READY TO FLY, pero estos tienen un costo aproximado de 2000 y 3200 dólares respectivamente por lo cual no son una opción para un sistema de bajo costo. Los drones que mejor cumplieron con los factores son el DJI Phantom 3 y el Syma 8XC venture. El phantom 3 puede levantar un peso aproximado de 300g, cuenta con un excelente sistema de estabilización automático, puede ser controlado a una distancia de 1 km y tiene un tiempo de vuelo de 25 minutos. A pesar de que el Phantom es un dispositivo conveniente tiene un costo de 500 dólares lo cual sigue siendo un precio elevado por lo que finalmente se seleccionó el Syma 8XC. Este drone tiene un precio de 75 dólares y puede levantar un peso de 200gr que supera el peso total de los componentes a levantar, además que cuenta con un control manual para ajustar la ganancia de los motores los que no ayuda enormemente a la hora de estabilizarlo y con un batería de 7.4v y 2000mAh nos puede proporcionar de 10 a 12 minutos de vuelo.

*Peso y Precio*

El cuadro 1 muestra el peso de los componentes, al tener un total de 66.5 gramos el drone puede maniobrar sin problema. El cuadro 2 muestra el costo total de este proyecto, lo que nos muestra que es bastante económico.

Arduino Nano	7
Sensor Mq2	9
Sensor Mq131	9
Sensor DHT11	2.5
Sensor Shinyei PPD42	24
Moudolo RF 433MHz	5
Otros	10
Total	66.5

Cuadro 1. Peso de los componentes en gramos

Arduino Nano	\$22
Arduino Mega	\$38.50
Sensor Mq2	\$5.29
Sensor Mq131	\$17
Sensor DHT11	\$6
Sensor Shinyei PPD42	\$9
Moudolo RF 433MHz	\$7
Drone	\$73
Otros	\$17
Total	\$195

Cuadro 2. Costo total de los componentes en dólares

## Implementación

### *Circuito de transmisión*

En la figura 1 se muestra un esquema de las conexiones para los sensores y el transmisor de radiofrecuencia. Este circuito tuvo que permanecer 24 horas encendido ya que los sensores MQ tienen que pasar una única vez por un periodo de precalentamiento para que sus lecturas sean correctas.

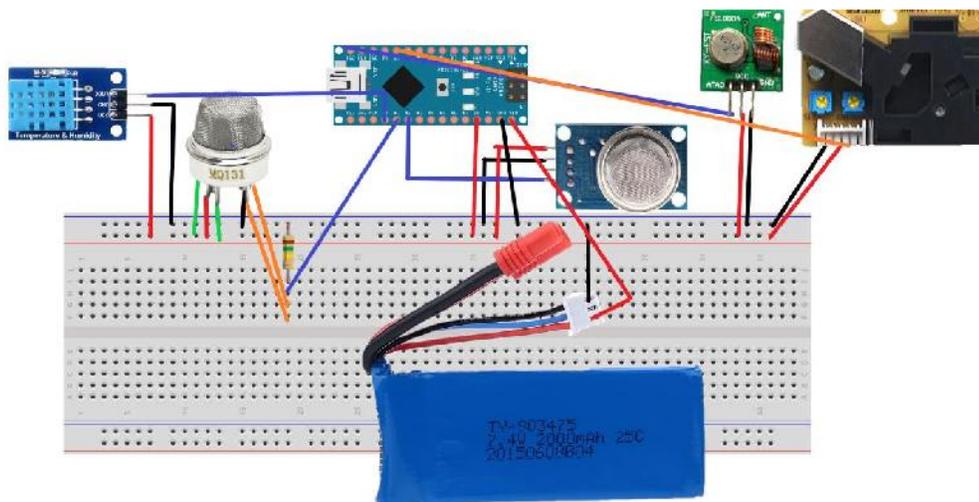


Figura 1. Implementación de los sensores y el transmisor con el Arduino nano

Tras haber realizado la programación de todos los sensores, se realizaron pruebas en las que se exponía el sensor de CO a el fuego de un encendedor y al sensor de material particulado a cúmulos de polvo para ver de qué manera reaccionaban. La figura 2 muestra los resultados que se obtuvieron en el monitor serial de Arduino IDE.

```
concentration = 4380.10 pcs/0.01cf -  
Current humidity = 31.00% temperature = 27.00C  
LPG:2ppm CO:11ppm SMOKE:8ppm  
O3:47ppb  
concentration = 401.69 pcs/0.01cf -  
Current humidity = 31.00% temperature = 27.00C  
LPG:2ppm CO:12ppm SMOKE:8ppm  
O3:47ppb  
concentration = 0.62 pcs/0.01cf -  
Current humidity = 31.00% temperature = 27.00C  
LPG:2ppm CO:14ppm SMOKE:9ppm  
O3:46ppb  
concentration = 0.62 pcs/0.01cf -  
Current humidity = 31.00% temperature = 27.00C  
LPG:2ppm CO:12ppm SMOKE:8ppm  
O3:47ppb  
concentration = 5158.34 pcs/0.01cf -  
Current humidity = 31.00% temperature = 27.00C  
LPG:2ppm CO:12ppm SMOKE:8ppm  
O3:47ppb  
concentration = 50.56 pcs/0.01cf -  
Current humidity = 31.00% temperature = 27.00C  
LPG:2ppm CO:12ppm SMOKE:8ppm  
O3:47ppb
```

Figura 2. Resultados mostrados en el monitor serial de Arduino IDE

### *Transmisión de datos*

Para la transmisión de datos se realizó el diagrama que muestra la figura 3, donde simplemente es colocado el modulo de radiofrecuencia directamente a 5v, tierra y el puerto 8 del Arduino. La programación de este sensor es

realmente sencilla, pero se tienen que programar las dos partes tanto transmisor como el receptor. Las pruebas de transmisión fueron realizadas con los dos Arduino conectados en diferentes computadoras portátiles hasta una distancia de 50 metros y se pudo observar que los valores mostrados en ambos monitores seriales eran los mismos.

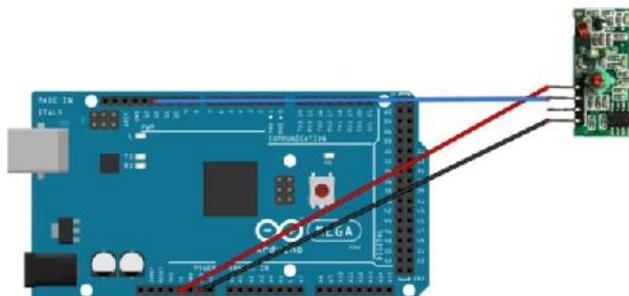


Figura 3. Conexión del receptor de radiofrecuencia en Arduino Mega

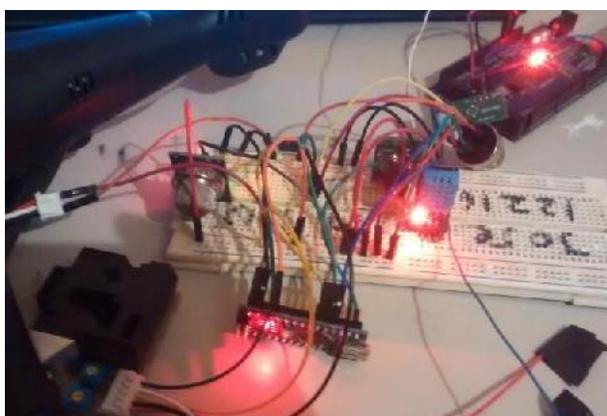


Figura 4. Conexión física de ambos circuitos

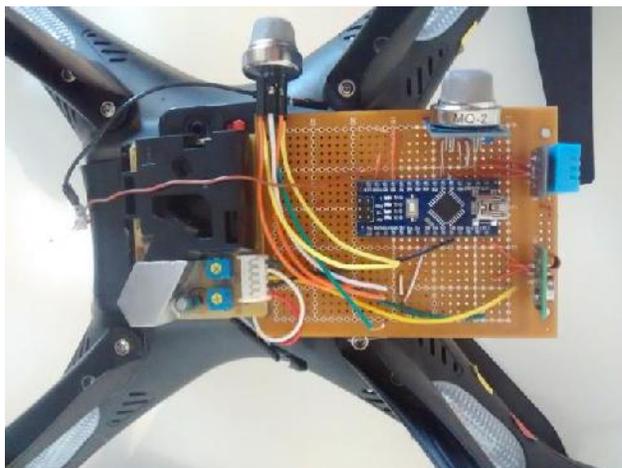


Figura 5. Implementación de circuito final en el drone

### Resultados

Al probar el drone con el circuito implementado, el drone tendía a moverse hacia adelante, esto debido a la carga extra que se le aplicó en la parte delantera del mismo. Esto se solucionó manualmente con el controlador de la ganancia de los motores que se encuentra en el control remoto del drone lo que redujo considerablemente el movimiento erróneo realizado por el drone.

Tras haber ajustado las velocidades de los motores el drone pudo permanecer estable a una distancia aproximada de 60 metros por un tiempo de 8 a 10 minutos, el tiempo total de vuelo que el drone puede permanecer, se redujo en un par de minutos esto debido a que la batería en el drone también estaba suministrando corriente a el circuito principal con los sensores y el Arduino.

La transmisión de los datos en tiempo real fue efectiva, tardando un tiempo de aproximadamente 2 segundos para recibir una señal. Algunos de los resultados obtenidos por el drone se encuentran en el cuadro 3. Estos datos nos indican que existen niveles elevados de ozono y de partículas PM10 donde  $0.01CF=283ml = Pie^2$ . Los datos de CO solo mostraran cuando las concentraciones sobrepasen las 1000 ppm que son muy elevadas en el aire ambiental.

```
Humedad = 30.00% Temperatura = 29.00°C
Material particulado = 0.62 pcs / 0.01cf
Monoxido = 0.00 ppm
Gas lp = 0.00ppm
Ozono =48.00 ppb

Humedad = 30.00% Temperatura = 29.00°C
Material particulado = 0.62 pcs / 0.01cf
Monoxido = 0.00 ppm
Gas lp = 0.00ppm
Ozono =45.00 ppb

Humedad = 30.00% Temperatura = 29.00°C
Material particulado = 0.62 pcs / 0.01cf
Monoxido = 0.00 ppm
Gas lp = 0.00ppm
Ozono =46.00 ppb

Humedad = 30.00% Temperatura = 29.00°C
Material particulado = 0.62 pcs / 0.01cf
Monoxido = 0.00 ppm
Gas lp = 0.00ppm
Ozono =47.00 ppb
```

Cuadro 3. Datos obtenidos de los sensores mostrados en el monitor serial de Arduino IDE

### Conclusiones

Este proyecto consiguió seleccionar los componentes adecuados para desarrollar un sistema de bajo costo que nos permitió conocer las concentraciones de distintas partículas que pueden llegar a ser nocivas y traer distintos problemas para el ser humano. Sin embargo, los sensores necesitan una calibración adecuada para obtener datos más precisos. Para un trabajo futuro se propone incrementar el presupuesto con la finalidad de seleccionar sensores más robustos, así como utilizar un modulo Xbee que nos proporcionará más alcance de transmisión y un drone que sea capaz de volar a una altura mayor.

### Referencias

- Jiang B. y Huacón C. F., "Cloud-based smart device for environment monitoring," *2017 IEEE Conference on Technologies for Sustainability (SusTech)*, Phoenix, AZ, 2017, pp. 1-6. doi: 10.1109/SusTech.2017.8333472
- Maraj A., Berzati S., Efendi I., Shala A., Dermaku J. y Melekoglu E., "Sensing platform development for air quality measurements and analysis," *2017 South Eastern European Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNM)*, Kastoria, 2017, pp. 1-5
- Morones, M. "Juárez, la tercera ciudad del país con peor calidad del aire" *El Diario*. Enero 2017. Juárez. Consultado por internet el 2 de abril de 2017 Dirección de internet: [http://diario.mx/Local/2017-01-07\\_8144d24a/juarez-la-tercera-ciudad-del-pais-con-peor-calidad-del-aire/](http://diario.mx/Local/2017-01-07_8144d24a/juarez-la-tercera-ciudad-del-pais-con-peor-calidad-del-aire/)
- Olaya J., Reina J., Torres L., Paz M. y Pereira. L. "Modelación no paramétrica de la contaminación promedio octohoraria del aire debida al Monóxido de Carbono y al Ozono Troposférico," *Ingeniería y Competitividad*, Vol 16, No. 1, p. 147 - 157 (2014)
- Reyna-Carranza M. A., Mérida-Palacio J. V. y Soria-Rodríguez C., "Acute effects of the PM10 and O3 in the healthy school children pulmonary function from Mexicali, BC, Mexico," *2010 Pan American Health Care Exchanges*, Lima, 2010, pp. 2-2.
- Wang Y. y Chi Z., "System of Wireless Temperature and Humidity Monitoring Based on Arduino Uno Platform", *2016 Sixth International Conference on Instrumentation & Measurement, Computer, Communication and Control (IMCCC)*, Harbin, 2016, pp. 770-773.