

**METODOLOGÍA DE  
CUADRICÓPTERO CON CONTROL  
Y ESTABILIDAD IMU, USANDO  
APLICACIÓN EN DISPOSITIVO  
MÓVIL ANDROID**

**QUADRICOPTER METHODOLOGY  
WITH CONTROL AND IMU  
STABILITY, USING APPLICATION  
IN MOBILE ANDROID DEVICE**

**METODOLOGIA  
QUADROCOPTERS COM  
CONTROLE E ESTABILIDADE DA  
IMU, USANDO O APLICATIVO NO  
DISPOSITIVO MÓVEL ANDROID**

**Franklin Pineda Torres**

Facultad de Ingeniería, Programa de Ingeniería Electrónica, Grupo de Investigación en Automática y Robótica, Universidad Autónoma de Colombia **franklin.pineda@fuac.edu.co**

**Alexander Cardona López**

Facultad de Ingeniería, Programa de Ingeniería de Sistemas, Grupo de Investigación Computation Intelligence. Institución Universidad Autónoma de Colombia **alexander.cardona@fuac.edu.co**

*Fechas de recepción: 28 de junio 2017*

*Fecha aprobación: 26 de agosto 2017*

## Resumen

La metodología expuesta, se presenta como una agrupación de tecnología, para el desarrollo de un prototipo funcional tipo dron controlado desde una aplicación para dispositivos Android que flexibiliza su manejo y puesta en marcha. La conjugación de la tecnología inalámbrica desde un *Smartphone* hacia el aeromodelo y sus referentes respuestas, ubican al nivel IMU (*Inertia Measurement Unit*) como una opción apropiada para la definición de estabilidad y control en los servomotores de los alerones. El IMU usa el acelerómetro y el giroscopio del dispositivo móvil para enviar las señales de control a un XBEE que se comunica con un microcontrolador Arduino, encargado de ejecutar las órdenes de control PID y a su vez realimentar los lazos respectivamente. App-inventor como software de uso on-line, es versátil en su programación para el diseño de aplicaciones, permite capturar las señales del IMU de forma sencilla y con ellas los puntos de referencia necesarios para el control del cuadricóptero.

**Palabras clave:** App-inventor, arduino, dron, microcontrolador, teléfono inteligente.

## Abstract

The exposed methodology is presented as a technology grouping for the development of a functional prototype type drone, which is controlled from an application for Android devices that makes its management and start-up more flexible. The conjugation of the wireless technology from a Smartphone to the model airplane and its referenced answers, locate the IMU (Inertia Measurement Unit) as an appropriate option for the definition of stability and control in the servomotors of the ailerons. The IMU uses the accelerometer and the gyroscope of the mobile device to send the control signals to a XBEE that communicates with an Arduino microcontroller, in charge of executing the PID control commands and in turn feeding the loops respectively. App-inventor as software for on-line use, it is versatile in its programming for the design of applications, it allows to capture the signals of the IMU in a simple way and with them the points of reference necessary for the control of the quadcopters.

**Keywords:** App-inventor, arduino, drone, microcontroller, smart phone.

## Resumo

A metodologia exposta é apresentada como um grupo de tecnologia para o desenvolvimento de um protótipo funcional tipo controlado por drone de Sde uma aplicação para dispositivos Android que é flexível de gestão e implementação. A conjugação da tecnologia sem fio de um Smartphone ao modelo de avião e suas referências, localizado no nível IMU (Inertia Measurement Unit) como uma opção apropriada para a definição de estabilidade e controle nos servomotores do aileron. O IMU usa o acelerômetro e o giroscópio do dispositivo móvel para enviar os sinais de controle para um XBEE que se comunica com um microcontrolador Arduino, Encarregado de executar os comandos de controle PID e, por sua vez, alimentar os loops, respectivamente. App-inventor como software para uso on-line, É versátil na sua programação para o design de aplicações, permite capturar os sinais da IMU de forma simples e com eles os pontos de referência necessários para o controle dos quadros cópios.

**Palavras-chave:** App-inventor, arduino, drone, microcontrolador, telefone inteligente

## I. INTRODUCCIÓN

El aeromodelismo, catalogado como deporte en Colombia desde 1999, casi 50 años después de serlo en el mundo (Hurtado, 2013), consiste en el diseño y la fabricación de pequeñas aeronaves que vuelan sin tripulación a bordo, esto permite a las naves ser controladas remotamente sin la preocupación de accidentes que pongan en riesgo la vida de las personas, además facilita el acceso a zonas remotas y de constante peligro para el hombre. En el año 2015, la Aeronáutica civil de Colombia reglamentó las directrices para la operación de los

sistemas aeronáuticos piloteados remotamente (RPAS) (Aerocivil, 2017).

Múltiples aplicaciones en el campo aeronáutico han generado robots que se adaptan a espacios aéreos de la sociedad con diversos tipos y formas especiales. Los UAV (vehículos aéreos no tripulados) se clasifican de acuerdo a sus características técnicas más comunes (Sánchez, 2016) -ver Fig. 1.

El cuadricóptero como plataforma, se ha destacado por tener ciertas características únicas de vuelo, como, por ejemplo, poder flotar y permanecer quieto en el aire, despegue y aterrizaje vertical, alta maniobrabilidad y mecánica simple (Figueiredo, 2012).

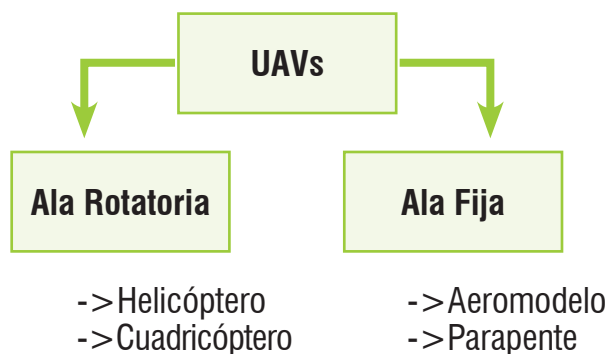


Fig. 1. Tipos de UAVs.

Fuente: Autores.

El cuadricóptero es en esencia un helicóptero con cuatro alerones cada uno con un motor (ver Fig. 2) y en su cuerpo central reposan los circuitos de comunicación, control y suministro de energía. Los motores utilizados en los multi-rotors<sup>1</sup> son de tipo eléctrico

“brushless” (sin escobillas) o de imanes permanentes. Estos evitan fricción y el desgaste de energía consecuente (Sendering, 2011). Cada motor debe poseer un controlador de velocidad ESC (*Electronic Speed Control*) cuya misión es regular la potencia eléctrica suministrada para la respectiva sustentación en el aire. La unidad de control que suele ser un microcontrolador recibe las señales de los sensores para actuar como una unidad estabilizadora del vuelo; los sensores más comunes son el giroscopio; que sirve para medir la

<sup>1</sup> Multirotor denominador para los drones con más de un rotor. Los cuadricópteros poseen cuatro rotores – uno por motor.

orientación (Diaz, 2015), el acelerómetro los cambios de velocidad en el tiempo, el barómetro, la brújula y el GPS. Estos tres últimos utilizados en drones sofisticados con auto-control de plano para la presión atmosférica y el posicionamiento global.

La unidad de medición inercial IMU, es quizá el componente más importante que poseen los drones. Actualmente es un dispositivo electrónico que mide e informa acerca de la velocidad, la orientación y las fuerzas gravitacionales (Hoflinger, 2013) que se ejercen en movimiento dentro del dispositivo, usando para ello, uno o más acelerómetros y giroscopios<sup>2</sup>. La lectura de estas variables permite proporcionar herramientas de control y estabilidad para tener un sistema de dron funcional descrito en su totalidad en los siguientes apartes.



**Fig. 2.** Base de Simulación.

Fuente: Autores.

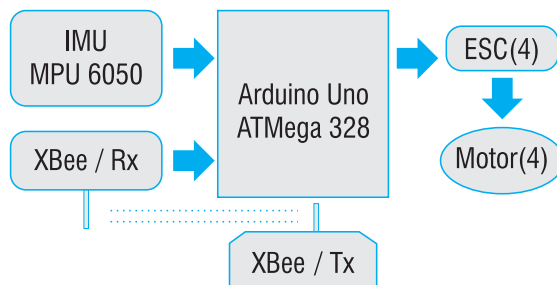
<sup>2</sup> El dispositivo IMU consta generalmente de tres acelerómetros y tres giroscopios; uno por cada eje: adelante-atrás, izquierda-derecha, arriba-abajo.

## II. Metodología General

El proceso metodológico inicia con el planteamiento de una arquitectura donde brevemente se interpreta la disposición de los diferentes elementos electrónicos a utilizar, seguida de la implementación física y lógica respectivamente.

### A. Arquitectura General

El sistema electrónico propuesto consta del IMU, referencia MPU6050, que se comunica con una tarjeta arduino y microcontrolador ATmega328 encargado de las acciones de control y calibración del cuadricóptero, las señales son recibidas utilizando tecnología XBEE desde un dispositivo android con aplicación programada en App-Inventor. El diagrama de bloques se puede apreciar en la Fig. 3.



**Fig. 3.** Diagrama de Bloques.

Fuente: Autores

### B. Implementación

Se compone de la agrupación e interacción que proporcionan de los diferentes elementos físicos y lógicos: la hechura del prototipo, el modelado dinámico de los motores, los sistemas de comunicación con sus respectivos circuitos electrónicos de control y por último la programación de la aplicación para el dispositivo móvil.



### 1) Sistema Base de Simulación.

Allí se debe tener en cuenta los cuatro ale-  
rones cada uno con su respectivo motor,  
una base giratoria esférica, movilidad y  
estabilidad de la mesa de simulación (ver  
Fig. 4). La base de pruebas puede tener di-  
versas formas ya que ella sólo sirve en las  
etapas iniciales del prototipo, de hecho, en  
algunos casos solamente es necesario la  
puesta en marcha de dos motores.

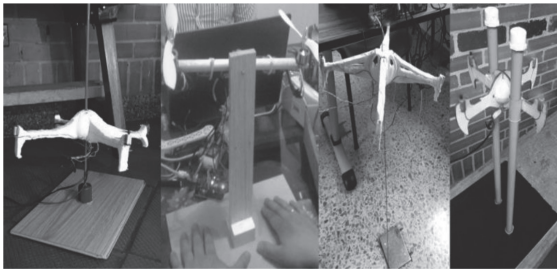


Fig. 4. Bases de Simulación.

Fuente: Autores.

### 2) Modelado de los Motores.

Consiste en la obtención de un modelo ma-  
temático que describe el comportamiento  
dinámico de los motores. El sensor del  
movimiento del motor del Drone se realiza  
con un encoder y un optoacoplador refe-  
rencia H22 ver Fig. 5, este montaje tiene  
una salida de señal en frecuencia, la cual  
debe ser convertida a voltaje utilizando un  
LM2907, este circuito electrónico entrega  
la información a un microcontrolador PIC  
con intervalos de tiempo cada 10  $\mu$ seg.

El modelo resultante para motores dc ge-  
neralmente es de primer o segundo orden  
(Chen, 2000), al utilizar el toolbox de iden-  
tificación en Matlab, que permite deducir  
un modelo dinámico ya sea en función de  
transferencia o en espacio de estado de

tipo continuo o discreto (ident, 2017), va-  
lidamos con un porcentaje de aceptación  
de 89% nos entrega un modelo de primer  
orden, Fig. 6.

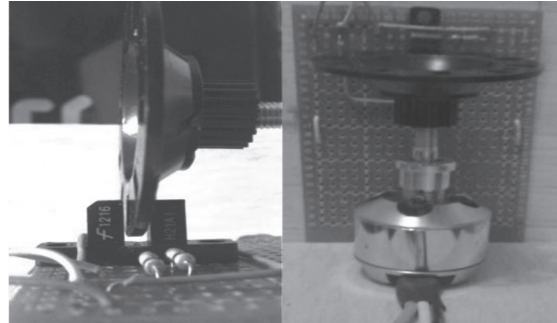


Fig. 5. Vista lateral y superior del montaje  
optoacoplador-encoder.

Fuente: Autores.

Generalmente se debe validar el modelo con  
una confiabilidad superior al 70%.

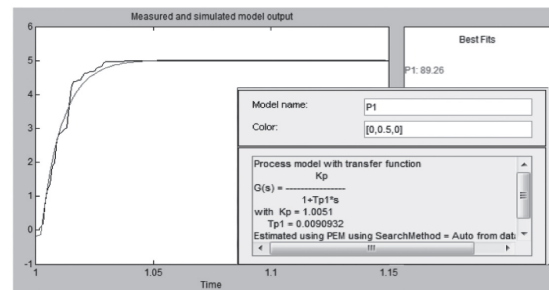


Fig. 6. Validación del Modelo.

Fuente: Autores.

La Fig. 7 muestra los sistemas de refe-  
rencia para describir la dinámica del cua-  
dricóptero, el *frame* {B} como sistema  
coordinado del cuerpo inercial y el *frame*  
{E} como sistema coordinado del espa-  
cio. Cada superficie de control está aso-  
ciado a un propio eje proporcionando una  
fuerza (Vila, 2011); estos tres ejes pueden  
ser deducidos de los ángulos de Euler (*roll*  
 $\varphi$ , *pitch*  $\theta$ , *yaw*  $\phi$ ) (Barrientos,1997) y

definen la forma de movimientos relativos que puede desempeñar el dron durante su operación.

Así por ejemplo, para elevar el dron, las velocidades de los motores  $\omega_{1-4}$  se incrementan en igual proporción. La tabla 1 proporciona las demás relaciones que se deben adecuar para el control de cada movimiento, donde el término “mayor que”  $>$  puede ser invertido para generar el movimiento inverso del ángulo en cuestión.

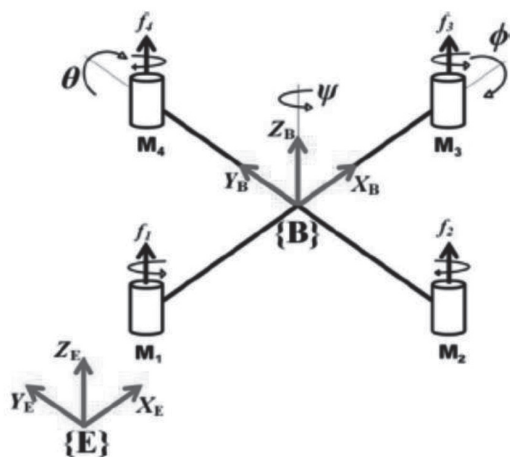


Fig. 7. Sistema de referencia.

Fuente: (Sumantri, 2016)

Tabla 1. Relación de velocidades angulares para el control de los ángulos (roll, pitch, yaw).

Ángulo	Velocidades
Roll	$\omega_4 > \omega_2, \omega_1 = \omega_3$
Pitch	$\omega_1 > \omega_3, \omega_2 = \omega_4$
Yaw	$\omega_2 = \omega_4 > \omega_1 = \omega_3$

Fuente: Autores.

### 3) Estabilidad y Control.

Las funciones de estabilidad y control son de modelo directo, guían y mantienen la velocidad en posiciones requeridas teniendo en cuenta la relación de fuerzas y momentos que afectan al cuadricóptero (Fuentes, 2015). El tipo de estabilidad depende del tipo de ángulo en estudio; en su orden, estabilidad direccional, estabilidad lateral y estabilidad longitudinal, todas ellas enmarcadas en una estabilidad dinámica. La función del sistema de control es generar la mayor estabilidad de vuelo (Guzmán, 2016), cuya implementación se desarrolla sobre la técnica clásica de control PID, cuyo diagrama de bloques lo podemos observar en la Fig. 8.

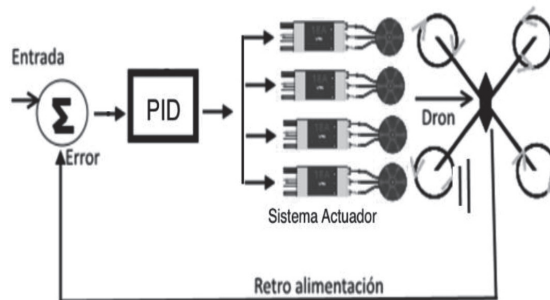


Fig. 8. Diagrama de Bloques del sistema de Control.

Fuente: Autores.

Las constantes encontradas van relacionadas de acuerdo a la estabilidad condicionada de giros, esto significa que estas constantes varían y actúan de acuerdo a las entradas de usuario – auto-regulación de cada ángulo cuyos valores son entregados por la unidad IMU. Actualmente el diseño de los controladores suele ser bastante sencillo con herramientas de software como el caso de *sisotool* de Matlab (siso,2014), donde la sintonización se

realiza con los parámetros de diseño (1) de tiempo de establecimiento menor a 0.5seg y un sobrepaso no mayor a 10%. El programa muestra un controlador diseñado –Fig. 9, donde las constantes de PID para los ángulos roll  $\varphi$  y pitch  $\theta$  son iguales debido a la similitud de los modelos encontrados, intuitivos por la característica de las velocidades angulares –ver tabla 1.

$$\begin{matrix} t_{ss} < 0.5 \text{ seg} \\ ov \leq 10\% \end{matrix} \quad (1)$$

El resultado de los controladores será la base para las correcciones que se hacen de las señales del IMU, implementadas en el Arduino, cuyas simulaciones son coherentes a los parámetros de diseño. El diseño del ángulo yaw, aparece con las

constantes  $K_p = 15.63$ ,  $K_d = 7.18$ ,  $K_i = 3.2488$ .

#### 4) Unidad de Medición Inercial

La importancia de este dispositivo radica en que las medidas obtenidas son usadas en la etapa de control. Al usar 3 giroscopios y 3 acelerómetros, este dispositivo es capaz de medir velocidad, orientación y aceleración de un sistema (Sevilla, 2014). La tarjeta de la Fig. 10 MPU6050, la cual proporciona 6 grados de libertad debe ser calibrada debido a que no siempre se encuentra en posición 100% horizontal, generando errores de medición en las variables, además los acelerómetros añaden un error que se acumula a lo largo del tiempo, por lo cual es necesario trabajar con los valores medios y no con los instantáneos.

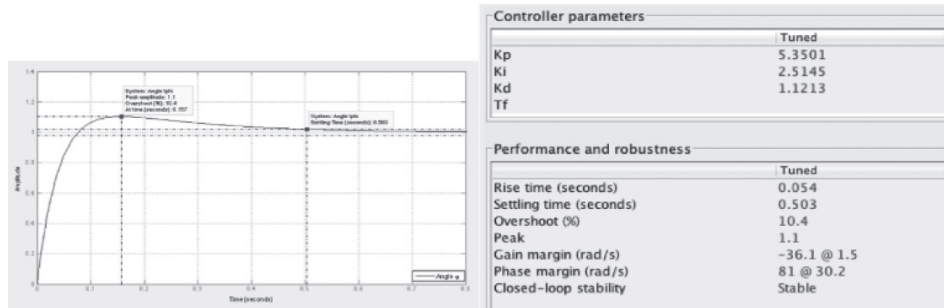
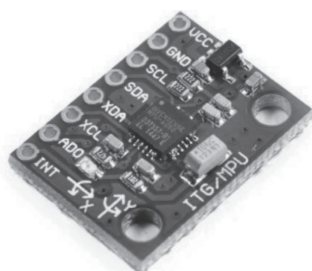


Fig. 9. Autotuning con las constantes de PID para los ángulos roll y pitch, dentro del SISO Design Toolbox.

Fuente: Autores.



Voltaje: 3V-5V  
 Rango Acelerómetro:  
 2g/4g/8g/16g  
 Rango Giroscopio:  
 250,500,1000,2000 °/seg.  
 Sensibilidad: 131 LSB/dps  
 Interfaz I<sup>2</sup>C

Fig. 10. Dispositivo IMU y características técnicas.

Fuente: (Prometec, 2017).

Para el cálculo de ángulos a partir del acelerómetro se usa (2) y a partir del giroscopio se usa (3)

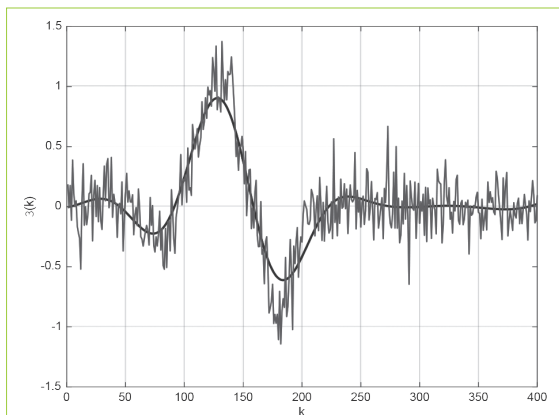
$$\theta_{acel} = \tan^{-1} \left( \frac{x}{\sqrt{y^2+z^2}} \right) \quad (2)$$

$$\theta_{gyro} = \frac{y}{sensibilidad} \quad (3)$$

Aunque existen varios métodos de calibración (Dorveaux, 2009), una de las más sencillas consta de un filtro de complemento; combina el ángulo calculado por el giroscopio con un filtro pasa-altos y el ángulo entregado por el acelerómetro con un filtro pasa-bajos en (4)<sup>3</sup>. De igual forma se realiza los cálculos para los ángulos *roll* y *yaw* –ver (robora, 2017). La salida del filtro para el acelerómetro la podemos apreciar en la Fig. 11.

$$\theta(k) = 0.98(\theta(k-1) + 0.01 * \theta_{gyro}) + 0.02 * \theta_{acel} \quad (4)$$

Las instancias referidas en la programación del Arduino se refieren, primero a la definición del protocolo de comunicación I<sup>2</sup>C, que se puede manejar con la librería <Wire.h>, segundo la definición del MPU a usar y su respectiva inicialización y tercero a los ángulos que se requieren obtener para la disposición del filtro y su respectivo control PID.



**Fig. 11.** Valores entregados por el acelerómetro (señal roja) y salida del filtro (en azul).

Fuente: Autores.

<sup>3</sup> La deducción completa y procedimientos matemáticos se puede encontrar en (Ferrer,2015)

Parte del código donde se captura y se almacena los tres ángulos se presenta en la Fig. 12. La variable *ypr* es un arreglo tipo *float* de tres posiciones donde el primer valor es el ángulo de guiñada –*yaw* expresado en radianes.

```
#ifndef OUTPUT_READABLE_YAWPITCHROLL
  mpu.dmpGetQuaternion(&q, fifoBuffer);
  mpu.dmpGetGravity(&gravity, &q);
  mpu.dmpGetYawPitchRoll(ypr, &q, &gravity);
  Serial.print("ypr\t");
  Serial.print(ypr[0] * 180/M_PI);
  Serial.print("\t");
  Serial.print(ypr[1] * 180/M_PI);
  Serial.print("\t");
  Serial.println(ypr[2] * 180/M_PI);
#endif
```

**Fig. 12.** Código del Arduino donde se captura los ángulos *yaw*, *pitch*, *roll*.

Fuente: Autores.

## 5) Arduino

El microcontrolador Arduino, es el elemento fundamental que permite interactuar con los demás dispositivos de control y comunicación. La Fig. 13 nos da una idea de los demás elementos con los que debe contar electrónicamente, como son: el MPU, los XBEE<sup>4</sup>, encargados de recibir las órdenes asignadas por software en la aplicación Android, y finalmente el Arduino debe generar las órdenes de control a los motores con el ESC (*Electronic Speed Controller*). Todo el sistema se encuentra alimentado por una batería recargable tipo Li-Po abreviatura de Litio y Polímero, estas son ligeras y de gran capacidad de almacenamiento de energía (iCharger, 2017).

<sup>4</sup> Configuración preferible con software X-CTU. Interacción ver (Pineda,2012)



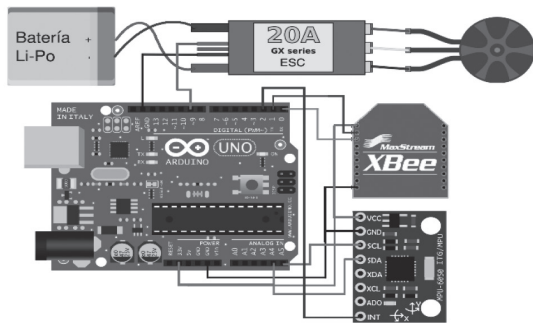


Fig. 13. Conexión de los dispositivos electrónicos.

Fuente: Autores.

### 6) App de Control

Recientemente ha aparecido gran variedad de software para el desarrollo de aplicaciones, algunas de ellas, de código abierto descargables desde internet como *processing* (Linares, 2017), *android-studio* (android, 2017) e incluso de manejo directo desde internet como *App-inventor* (inventor, 2017). Este último posee un entorno de programación visual e intuitivo por medio de bloques e incorpora funciones para el manejo tanto del giroscopio como del acelerómetro –ver Fig. 14. Creado por el MIT, App inventor también permite importar extensiones para el manejo de otros sensores y herramientas propias del *smartphone*, para la comunicación wi-fi con los Xbee se recomienda descargar una extensión e importarla dentro del IDE (extensión, 2017).

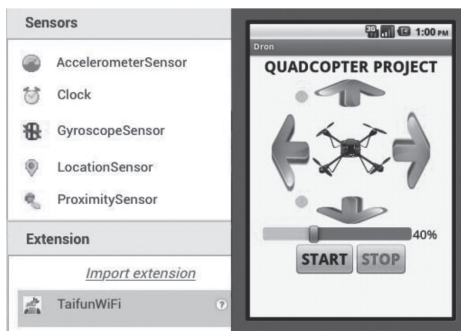


Fig. 14. Funciones del IMU, wi-fi e interfaz de usuario.

Fuente: Autores.

La aplicación de control posee la configuración de la red wi-fi y una interfaz de usuario, vista en la Fig. 14. La interfaz posee cinco botones para direccionamiento del cuadricóptero: el central de la imagen sube o baja, el superior e inferior para direccionarlo hacia delante/atrás y los laterales para llevarlo izquierda/derecha. Posee a su vez una barra que indica el porcentaje de onda PWM que se le suministra a los motores que se encuentren activos. Una vez que se da inicio con el botón *start*, se verifica la conexión del wi-fi para dar inicio al envío de tramas hacia el dron. La trama se compone esencialmente de el valor de PWM y los ángulos (*yaw*, *pitch*, *roll*), desarrollados a partir de las aceleraciones y velocidades angulares. En la Fig. 15 se muestra las funciones principales en forma de diagrama de flujo del programa.

## III. RESULTADOS

El dispositivo en conjunto es capaz de sobrevolar diferentes áreas, con el objetivo de adquirir imágenes de dichos lugares, por medio de una cámara y su respectivo almacenamiento en tarjeta de memoria micro-SD. Una de las principales limitaciones a la hora del ensamble y configuración, consistió en la adaptación de los componentes mecánicos y electrónicos. En la Fig. 16, se observa el montaje final, con la cámara en la parte inferior. El sistema de baterías, éstas presentan un tiempo estimado de trabajo comprendido entre 18 a 25 minutos. Adicionalmente, no se puede permitir que estas baterías bajen su voltaje a menos de 11 voltios DC, ya que tienden a descomponerse. Tener en cuenta que al sistema no se le incorporó un sistema de protección de los circuitos electrónicos y se debe manipular en condiciones climáticas aptas para los sobrevuelos.

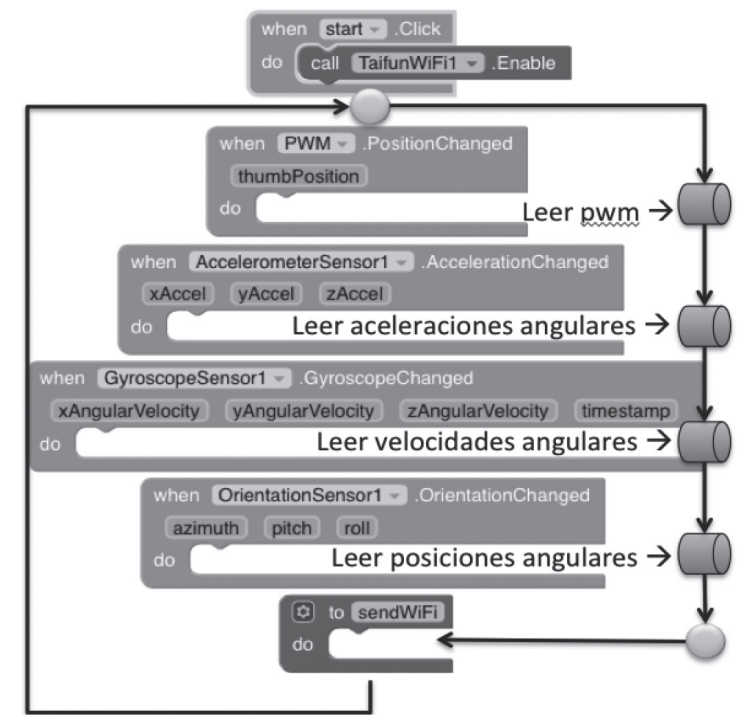


Fig. 15. Diagrama de Flujo con las funciones principales utilizadas en App-Inventor.

Fuente: Autores.

Dentro de los alcances, se tiene la posibilidad de instalar la aplicación móvil en diversos tipos de dispositivos electrónicos que usen sistema Android entre los más comunes los celulares y tablets. El código del software en App-Inventor, aunque es de fácil programación; puesto que se maneja por medio de bloques, la interacción debe tener pocos componentes de usuario para hacerla de rápida interacción con el sistema de comunicación.

Aunque el costo neto total se encuentra alrededor de los 750 mil pesos, en el comercio con características parecidas de tiempo de aplicabilidad y distancia de comunicación, se encuentran con valores superiores a millón doscientos, como ejemplo el ArDrone 2.0 (*power edition*).

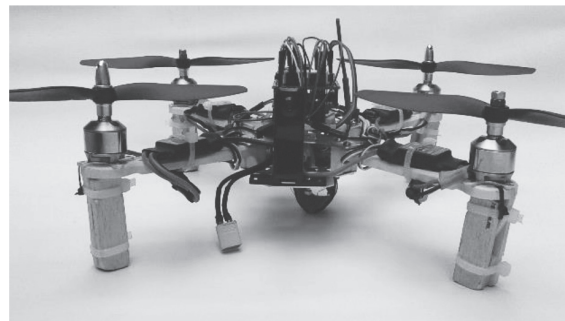


Fig. 16. Montaje Final de Elementos, vista lateral.

Fuente: Autores.

## IV. CONCLUSIONES

En este artículo se ha presentado una metodología para la construcción y diseño del de un cuadricóptero funcional desde una aplicación Android; guiada desde el punto de vista

de utilizar un IMU en el cuadricóptero y el otro propio del *smartPhone*, para el control y la estabilidad en vuelo.

### Trabajo Futuro

Tender a que el proceso de estabilidad y control sea más rápido, y en lo posible, sin necesidad de usar software especializado, quizá con librerías propias dentro del mismo Arduino. De igual forma, proveer una interfaz con vista de la cámara dentro de la App. Finalmente, ayudar a que el costo neto disminuya con elementos de importe más económicos como lo son los XBEE, los ESC y los motores y la respectiva protección de los mismos. De hecho, el mismo Arduino provee peso y elementos innecesarios (sistema de fuente externa, sistema de comunicación y puertos no utilizados) que quizá con otro tipo de microcontrolador sea más económico y de menor peso que disminuiría también el gasto energético.

## V. REFERENCIAS

AeroCivil. (2017) Normatividad. [online] <http://www.aerocivil.gov.co/normatividad>

Android (2017), Software Android Studio. [online]: <https://developer.android.com>

Barrientos A. (1997), Fundamentos de Robótica. Mc Graw Hill. Universidad Politécnica de Madrid.

Sevilla L., (2014), Modelado y Control de un Cuadricóptero. Proyecto fin de Carrera Universidad Pontificia de Comillas.

Chen Tsong. (2000) Linear System Theory and Design. Third Edition. Oxford University Press.

Diaz, T.J. (2015) "Lights, drone... action," in *Spectrum, IEEE*, vol.52, no.7, pp.36-41.

Dorveaux E. (2009), Iterative calibration method for inertial and magnetic sensors. 48th IEEE Conference on Decision and Control. Shanghai, China.

Extension. (2017). Wi-fi Manager Extension. [online]: <https://puravidaapps.com/wifi.php>

Ferrer D. (2015), Adquisición de Datos IMU, en un sistema embebido. Proyecto de Grado. Universidad Politécnica de Valencia.

Figueiredo H., Saotome O. (2012) Simulation Platform for quadricopter: using Matlab and X-plane. SBR-LARS ISBN 978146734650-4. IEEE Brazil.

Fuentes J. (2015) Arquitectura cognitiva híbrida para la navegación autónoma de UAVs mediante mapas topológicos visuales", [online] [http://oa.upm.es/22579/1/JUANPABLO\\_FUENTES\\_BREA.pdf](http://oa.upm.es/22579/1/JUANPABLO_FUENTES_BREA.pdf)

Guzmán A. (2016) Cálculo de variables de control PID para Drones Cuadcopter. Reaxion Ciencia y Tecnología Universitaria.

Hoflinger, F., Muller, J., Rui Zhang., Reindl, L.M., Burgard, W. (2013) "A Wireless Micro Inertial Measurement Unit (IMU)," in *Instrumentation and Measurement, IEEE Transactions on*, vol.62, no.9, pp.2583-2595, Sept. 2013 doi: 10.1109/TIM.2013.2255977

Hurtado Rocío. (2013) Aeromodelismo, el arte de volar con los pies en la tierra. [online]

<http://www.eltiempo.com/archivo/documento/CMS-12735957>

iCharger (2017). Introduction to LiPo Batteries. [recurso Online]: <http://www.icharger.co.nz/buying/resources-faq/>

- Ident (2017), System Identification Toolbox, Trial Software [online]:  
<https://www.mathworks.com/products/sysid.html>
- Inventor. (2017), Software App-Inventor. [online]: <http://appinventor.mit.edu/explore/>
- Linares J. (2017) Introducción a Processing. Departamento de Sistemas y Computación, Escuela Politécnica Superior d'Alcoi. [online]  
[http://users.dsic.upv.es/~jlinares/grafics/processing\\_spa\\_1.pdf](http://users.dsic.upv.es/~jlinares/grafics/processing_spa_1.pdf)
- Pineda F. (2012) Control de dispositivos de seguridad a través de internet, utilizando el protocolo IEEE 802.15.4. *Revista Clepsidra* No.15. ISSN: 1900-1355
- Robora. (2017), realidad Aumentada Aplicada a Competiciones de Robótica. [online]<https://robora.wordpress.com>
- Sánchez B., Tapia J., Rosa P. (2016) Drones Social Aspects and Social Applications, *Visión Electrónica Universidad Distrital* Vol.10 No.2.
- Sensinger, J.W., Clark, S.D., Schorsch, J.F. (2011) "Exterior vs. interior rotors in robotic brushless motors," in Robotics and Automation (ICRA), 2011 *IEEE International Conference*. pp.2764-2770, 9-13 May 2011 doi: 10.1109/ICRA.2011.5979940
- Sevilla L., (2014), Modelado y Control de un Cuadrícóptero. Proyecto fin de Carrera Universidad Pontificia de Comillas.
- Siso (2014), Control System Design Toolbox, Manual de Usuario. Mathworks.
- Sumantri B. (2016) Least square based sliding mode control for a quad-rotor helicopter and energy saving by chattering reduction. Vol 66-67. *Mechanical Systems and Signal Processing*.
- Vila O. (2011) Modelización de aeronaves no tripuladas con Simulink. Proyecto de Grado Escuela Universitaria de Ingeniería. España. [online] <http://upcommons.upc.edu/bitstream/handle/2099.1/12919/memoria.pdf>