CAPÍTULO IV - Desarrollo

1. ***Módulo de alimentación y control de motores***

Se desarrolló un circuito para llevar a cabo la alimentación y regulación de velocidad de los motores de corriente continua del chasis de cuadricóptero Draganflyer V utilizado en el presente trabajo. En primer lugar, se decidió utilizar una batería de Polímero de Litio (Li-Po, del inglés Lithium Polymer) de 3 celdas, ya que es la configuración de alimentación de los motores de corriente continua recomendada por el fabricante en **[Draganfly Innovations 2006].** Por otro lado, en **[Nadales 2009]**, se describen las características de funcionamiento de los motores en cuestión, y la relación entre el voltaje de alimentación y la corriente consumida por los mismos al hacer girar las hélices del cuadricóptero. A continuación se detallan las características de los motores citadas previamente:

* Modelo de motores DC: Mabuchi RC 280SA.
* Voltaje de alimentación recomendado: 4,5 a 12 Voltios.
* Consumo sin carga: 0,14 Amperios.
* Consumo medio: 4 Amperios.
* Peso individual: 47 Gramos.



Ilustración 1: Relación voltaje-corriente de los motores DC del chasis Draganflyer V.

Tomado de **[Nadales 2009]**

En base a las características de los motores anteriormente expuestas, se puede estimar un consumo promedio de 16 Amperios al alimentar los cuatro motores de forma continua, y de hasta 24 amperios considerando un pico de consumo de hasta 2 amperios por motor al realizar el encendido de los mismos, los cuales deben ser provistos por la batería de polímero de litio de forma instantánea. Para una batería de polímero de litio cualquiera, se tiene que:

Para satisfacer la necesidad de consumo de los motores del cuadricóptero se utilizó una batería Li-Po Yuntong de 3 celdas, con una capacidad de 1350 miliAmperios por Hora (mAH), y una tasa de descarga de 25C. Para la batería seleccionada se estimaron los siguientes valores de consumo y duración:

Para regular la velocidad de los motores de corriente continua del cuadricóptero, se utilizó Modulación por Ancho de Pulso (PWM, del inglés Pulse Width Modulation), enviada desde los puertos de la tarjeta Arduino Nano 3.0, para realizar una conmutación rápida entre los estados de encendido y apagado de los motores, pudiendo mantener un control de lazo abierto sobre su velocidad a partir del ciclo de trabajo del PWM emitido.

****

Ilustración 2: Diagrama de bloques del sistema de control de velocidad de los motores DC del cuadricóptero.

Para realizar la conmutación a alta velocidad de cada motor en un solo sentido, se utilizó un MOSFET IRFZ44N y un diodo de rodada libre, para forzar la descarga de la inductancia del motor al cerrar el canal del MOSFET. Estos fueron seleccionados de entre todos los componentes disponibles en el mercado venezolano por su alta velocidad de conmutación, alta tolerancia y estabilidad ante valores altos de corriente y voltaje, y baja resistencia drenador-surtidor en el caso del IRFZ44N. Se realizaron pruebas para caracterizar la conductividad y resistencia interna del MOSFET ante distintos valores de voltaje aplicados en la compuerta del mismo, al encender uno de los motores del cuadricóptero de forma continua (Ciclo de trabajo de 100% con un voltaje Vgs fijo), con una alimentación de 12V -cercana al voltaje máximo de carga de la batería de polímero de litio seleccionada para alimentar el conjunto-. A continuación se presenta el diagrama del circuito de prueba utilizado, y las curvas características identificadas a partir de los datos recogidos:



Ilustración 3: Diagrama de circuito para estimación de Vgs óptimo para conmutar MOSFET con carga de motor DC en el surtidor.

Ilustración 4: Relación **Voltaje de compuerta** – **Corriente drenador-surtidor** del MOSFET IRFZ44N, en presencia de carga del motor.

Ilustración 5: Relación **Voltaje de compuerta** - **Corriente drenador-surtidor** del MOSFET IRFZ44N, en presencia de carga del motor.

A partir del análisis realizado se identificó el rango de valores entre 4,5 y 6 voltios como el rango de valores de tensión de compuerta óptimos para el funcionamiento del MOSFET con los motores del cuadricóptero como carga en el drenador. Considerando el efecto de descarga de la batería a utilizar, se decidió utilizar un voltaje de al menos 6V, siendo éste cota superior del rango de valores óptimos identificado, como tensión de activación de la compuerta del MOSFET IRFZ44N.

Para disminuir los efectos de ruido electromagnético que pudieran ser introducidos por la rápida conmutación de los MOSFETS y los motores

al módulo de lógica, sensores y comunicación, el cual envía las señales de PWM para la regulación de velocidad de los motores, se decidió utilizar opto-acopladores para separar totalmente la etapa . Se seleccionó el modelo 4N26, presente en el mercado de componentes electrónicos venezolano, por su alta velocidad de conmutación, tolerancia a altos valores de voltaje y corriente, bajo precio, y simplicidad de configuración. Para obtener la tensión de salida de 6V para conmutar la compuerta del MOSFET IRFZ44N se diseñó un circuito que hace uso de un divisor de voltaje en el emisor del fototransistor, en configuración colector común, como se presenta a continuación:

****

Ilustración 6: Diagrama de circuito para estimación de carga total de foto-transistor del opto-acoplador 4N26 en configuración colector común.

Para calcular la resistencia total de la carga del colector común del fototransistor se evaluó el tiempo de respuesta del mismo ante impulsos de PWM de 490 Hz, y se seleccionó un valor de 900Ω, a partir del cual se realizó el cálculo de las resistencias R1 y R2 para el circuito de conmutación con divisor de voltaje. El valor calculado para R1 fue de 390Ω, y para R2 fue de 510Ω.



Ilustración 7: Señal de salida del opto-acoplador ante una señal de PWM con un ciclo de trabajo de 98%.



Ilustración 8: Señal de salida del opto-acoplador ante una señal de PWM con un ciclo de trabajo de 23%.



Ilustración 9: Diagrama de circuito de regulación de velocidad diseñado.

Finalmente, se diseñó y construyó una placa de circuito impreso doble capa consistente en un arreglo de cuatro módulos para regulación de velocidad desarrollados en el presente trabajo. A continuación se muestra el diseño y el acabado del circuito luego se ser ensamblado:

**Foto del diagrama de la tarjeta doble capa antes de fabricar.**

**Foto del PCB final.**

1. ***Módulo de lógica, sensores y comunicación***

Se utilizó una tarjeta Arduino Nano 3.0 como unidad de procesamiento principal en el cuadricóptero por poseer las siguientes características:

* 16MHz de velocidad de reloj.
* Puerto serial.
* Posibilidad de establecer comunicación , al ser este un protocolo de uso común para establecer comunicación con Unidades de Medición Inercial.
* 6 puertos con salida de Modulación por Ancho de Pulso (PWM).
* Abundante documentación, y existencia de librerías para acelerar el proceso de desarrollo.
* Dimensiones reducidas.
* Disponible en el mercado venezolano.
* Bajo consumo de corriente.
* 3 temporizadores. Los temporizadores son vitales para realizar una ejecución uniforme de los ciclos de control y lectura de sensores.
* Diseñada para funcionar con niveles lógicos TTL.

Se diseñó una placa de circuito que consta de una serie de puertos pin header para situar la tarjeta Arduino y la unidad de medición intercial, además de conectar el sensor de ultrasonido, las salidas de PWM y el módulo de XBEE mediante cables “jumper”. La misma sirve de interfaz de comunicación entre la tarjeta Arduino, y los demás módulos del cuadricóptero. Además, permite la alimentación de todo el circuito haciendo uso de la etapa de regulación de voltaje embebida en la tarjeta Arduino, la cual puede ser alimentada con voltajes de entre 6 y 20 voltios. A continuación se detalla el consumo promedio de cada componente del circuito, y el consumo total del mismo:

|  |  |
| --- | --- |
| **Componente** | **Consumo promedio** |
| Arduino Nano | 50 mA |
| Pololu minIMU-9 V2 | 10mA |
| XBee Series 1mW Wire Antenna – Series 1 | 50mA |
| Parallax Ping | 30mA |
| 4 Optocouplers 4N26 – Módulo de control de motores | 240 mA |
| **Total** | **380mA** |

Tabla 1: Desglose del consumo promedio por componente, y acumulado, del módulo de lógica, sensores y comunicación.

Para alimentación del módulo de lógica, sensores y comunicación durante la realización de pruebas de vuelo se utilizó una batería de polímero de litio Yuntong de 300mAH, con un coeficiente de descarga de 3C. A continuación se presenta el cálculo de la máxima capacidad de descarga, y duración de la carga de la misma al alimentar el circuito desarrollado:



Ilustración 10: Placa de circuito construída.

1. ***Estimación de orientación del cuadricóptero***
   1. **Convenciones respecto a los ángulos**

Para simplificar la implementación de los sistemas de control del cuadricóptero, se decidió representar la posición y velocidad angular del mismo en cada eje mediante ángulos de Euler, bajo la siguiente convención:

* Ángulo de Yaw: Ángulo de giro respecto al eje Z del acelerómetro y giroscopio.
* Ángulo de Pitch: Ángulo de giro respecto al eje Y del acelerómetro y giroscopio.
* Ángulo de Roll: Ángulo de giro respecto al eje X del acelerómetro y giroscopio.
* Velocidad de Yaw: Velocidad de giro respecto al eje Z del giroscopio.
* Velocidad de Pitch: Velocidad de giro respecto al eje Y del giroscopio.
* Velocidad de Roll: Velocidad de giro respecto al eje X del giroscopio.
  1. **Descripción de la Unidad de Medición Inercial (IMU)**

Se utilizó una tarjeta Pololu MinIMU-9 v2, la cual está constituida porun giroscopio de 3 ejes L3GD20, y por un acelerómetro de 3 ejes y un compás de 3 ejes LSM303DLHC. La misma provee de una interfaz que permite acceder a las mediciones, por eje, de cada uno de sus sensores, e incluye un regulador de voltaje y un convertidor de nivel de voltaje que permite su operación con una entrada de voltaje de 2,5V a 5,5V.

Para la obtención de datos de la tarjeta Pololu MinIMU-9 v2 se utilizaron las librerías diseñadas por el fabricante para su manejo desde tarjetas Arduino, y se utilizó la configuración por defecto de las mismas en cuanto a tasa de salida de datos, ya que para la obtención de datos del acelerómetro era suficientemente rápida, y para la obtención de datos del giroscopio, es cercana a la velocidad de ejecución del ciclo de control más rápido del cuadricóptero. En principio, para mejorar la estimación incremental de ángulo y el sistema de control de velocidad angular, se aumentó la tasa de salida de datos del giroscopio a 380Hz, pero, ante la pérdida de sensibilidad total del sensor, se decidió utilizar la configuración por defecto del mismo, con una tasa de salida de 95Hz.

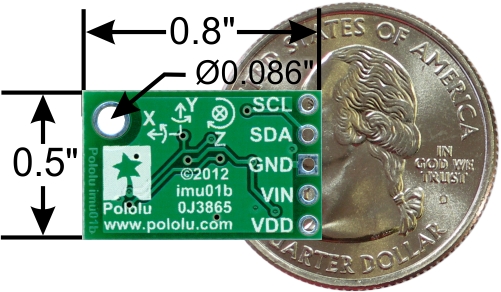


Ilustración 11: Pololu minIMU-9 V2.

Tomado de http://www.pololu.com/product/1268

* 1. **Cálculo de velocidad angular**

El cálculo de velocidad angular se realizó a partir de las mediciones realizadas con el giroscopio. Según la hoja de datos del sensor L3GD20, el mismo puede configurarse para obtener una sensibilidad de 8,75, 17,5 y 70 milésimas de grado por segundo por cada dígito de medición obtenido (mdps/digit – en inglés *millidegrees per second per digit*), y un rango de medición de 250, 500 y 2000 grados por segundo (dps – en inglés *degree per second*). Se decidió configurar el rango de medición a 250 grados por segundo y la senbilidad del sensor a 8,75 mdps/digit, por considerarse rango y senbilidad suficientes para medir las velocidades del cuadricóptero realizando movimientos simples en vuelo. En base a esto último se calculó la ganancia del giroscopio, para convertir todas las mediciones obtenidas mediante el mismo, en milésimas de grado por segundo, a grados por segundo, como se ilustra a continuación:

siendo:

* 1. **Estimación de ángulos de Pitch y Roll a partir del acelerómetro**

Se realizó un estimado del ángulo de inclinación del cuadricóptero a partir de las mediciones del acelerómero, el cual provee una descomposición de la fuerza de aceleración del cuadricóptero en tres (3) ejes perpendiculares (x, y, z). El acelerómetro puede detectar constantemente la fuerza de gravedad, en magnitud, dirección, y fuerza, en cada uno de sus tres (3) ejes, y en base a ello puede establecerse un marco de referencia absoluto a partir del cual calcular los ángulos de Pitch y Roll del sensor, y en consecuencia, del cuadricóptero. La estimación de ángulos se realizó siguiendo el procedimiento expuesto en **[STMicroElectronics 2010]**, el cual es presentado a continuación:



Ilustración 12 - Tomada de **[STMicroElectronics]**

siendo:

Las estimaciones de los ángulos de Pitch y Roll calculadas a partir de los datos del acelerómetro, a pesar de ser precisas y permitir mantener un marco de referencia absoluto, en base a la fuerza de gravedad de la tierra, presentan un alto porcentaje de ruido, ya que el acelerómetro es altamente sensible a perturbaciones provocadas por fuerzas externas que incidan sobre el mismo.

**Demostración del ruido del acelerómetro mediante una gráfica**

**Gráfica del espectro de frecuencias del acelerómetro**

* 1. **Estimación de posición angular a partir del giroscopio**

Se realizó un estimado del ángulo de inclinación del cuadricóptero, mediante integración numérica de las velocidades de rotación de Yaw, Pitch y Roll, como se describe a continuación:

Durante un lapso corto de tiempo, este estimado del ángulo de inclinación en cada eje puede mantenerse preciso, pero, tiende a presentar deriva constante y a alejarse de los valores reales a medir, por no realizarse la misma respecto a un marco de referencia absoluto.

**Gráfica que demuestre la derivada en el tiempo del estimado de ángulo mediante integración numérica de datos del giroscopio**

* 1. **Combinación de las estimaciones de posición angular del acelerómetro y giroscopio**

Al combinar la precisión del acelerómetro para medir inclinación respecto al marco de referencia absoluto del planeta Tierra, con la sensibilidad y estabilidad de la estimación de ángulo realizada a partir de los datos del giroscopio para medir los movimientos de rotación alrededor de cada eje del sensor, puede obtenerse una estimación de ángulo precisa, estable, y de alta sensibilidad.

Como es descrito en **[Burgard 2005]**,si es el estado estimado de un proceso, y una observación acerca del estado del mismo, puede aplicarse la Regla de Bayes para estimar la probabilidad de que el proceso se encuentre en el estado X a partir de Z:

En base a **[Sturm 2013]** y **[Burgard 2005]**, por la Ley de Probabilidad Total, el término puede ser tratado como una constante de normalización , tal que:

Por lo cual:

Para el caso particular del presente trabajo en el que se realizó estimación de posición angular mediante la combinación de datos de los sensores acelerómetro y giroscopio, por lo cual se cuenta con observaciones y , por lo cual el modelo se reduce a:

Se asume además que:

+ )

+ )

Siendo el error de estimación a partir de las observaciones de ambos sensores. Luego:

Finalmente, se decidió utilizar una aproximación del ángulo bajo la suposición de que el error de estimación supone una cantidad despreciable?:

+

El modelo derivado a partir de teoría de probabilidades para la combinación de las estimaciones de ángulo del giroscopio y acelerómetro, coincide con el de un filtro digital de uso altamente extendido conocido como *Filtro Complementario* **[Gaydou 2007], [Colton 2007]**, el cual se fundamenta en la combinación de un filtro pasa bajos y un filtro pasa altos, ambos de primer orden, para la composición de los espectros de frecuencias de dos señales lineales invariantes en el tiempo en una tercera señal de salida. El filtro complementario puede ser descrito mediante las siguientes ecuaciones en el dominio de la frecuencia **[Gaydou 2007]**:



Ilustración 13: Tomada de **[Gaydou 2007]**

En donde:

Función de transferencia del acelerómetro.

Función de transferencia del giroscopio.

Filtro pasa altos de primer orden.

Frecuencia de corte del filtro pasa altos de primer orden.

La ganancia del filtro pasa altos puede ser descrita a partir de la siguiente ecuación:

donde:

Constante de tiempo del filtro.

Diferencial de tiempo.

Así mismo, tanto el modelo probabilístico como el modelo basado en el análisis de señales del proceso de estimación de ángulo pueden considerarse como análogos, ¿por qué? por lo cual mantienen la siguiente relación: Mejorar

El cálculo de la ganancia del filtro pasa altos, y con ello, de y , se realizó mediante prueba y error, teniendo como criterio de selección la estabilidad, precisión y exactitud de la estimación de ángulo realizada, y el tiempo de respuesta de la misma. Se estableció un valor de k=0,03 para la ganancia, el cual fue utilizado para las pruebas de vuelo del cuadricóptero.

**Gráfica que demuestre precisión y estabilidad del ángulo calculado con el filtro complementario**

**Gráfica que demuestre estabilidad del estimado de ángulo ante ruido, por ejemplo, con el cuadricóptero inmovilizado, pero con las hélices girando a alta velocidad.**

1. **Estimación de altura del cuadricóptero**
   1. **Descripción del sensor de altura utilizado**

Para la estimación de altura del cuadricóptero se utilizó un sensor Parallax Ping, el cual opera con niveles lógicos TTL, provee un rango de alcance de entre 2cm y 3m, y un peso de 9 gramos. Adicionalmente, el sensor porta un LED en su cara frontal, que indica si el proceso de estimación de altura se está realizando de forma efectiva.

* 1. **Obtención de datos y cálculo de altura. ¿Filtrado de datos? Validación de posición angular al calcular la altura en vuelo?**

1. ***Sistemas de control***
   1. ***Sistema de control de posición y velocidad angular***
   2. ***Sistema de control de altura***
2. ***Comunicación inalámbrica***
   1. ***Descripción de las características y configuración de los módulos XBEE utilizados***
   2. ***Descripción del protocolo de comunicación desarrollado***
3. ***Software de telemetría y comandos***
   1. ***Arquitectura del software***

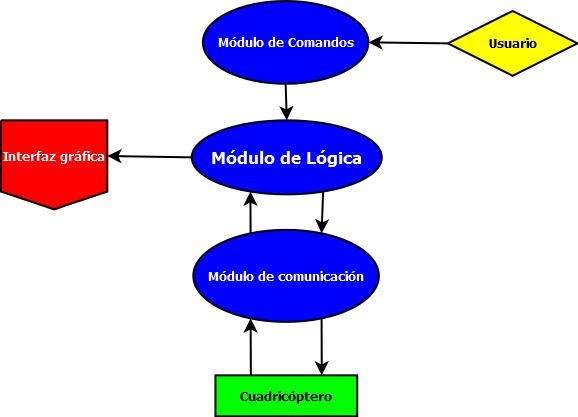


Ilustración 14: Arquitectura de la aplicación desarrollada.

* 1. ***Secuencia de ejecución***
  2. ***Interfaz gráfica***
  3. ***Descripción del módulo de lógica***
  4. ***Descripción del módulo de comandos***

Contiene a la clase HiloComando, que permite que detectar continuamente movimientos en el joystick sin afectar la ejecución de las rutinas de comunicación y graficación de datos.

En la clase VentanaPrincipal del módulo de lógica es instanciado e inicializado un objeto de la clase HiloComando, el cual detecta en su constructor la presencia de un Joystick Logitech Gamepad II, y, a partir de un parámetro de su constructor asigna valor a un atributo de tipo VentanaPrincipal, para ejecutar llamadas a la clase de lógica emulando una interfaz RPC (En inglés Remote Process Call – Llamada remota a procesos).

Al ejecutarse el método Start del objeto de tipo HiloComando contenido en la clase de la ventana principal, se inicia la ejecución de un ciclo que detecta eventos provocados por accionamiento del Joystick, calcula comandos a partir de ellos, y los retransmite mediante llamadas a procesos de la clase VentanaPrincipal para que sean enviados al cuadricóptero por el hilo de comunicación.

* 1. ***Descripción del módulo de comunicación serial***

Contiene a la clase HiloSerial, la cual permite a la aplicación mantener un hilo de ejecución como gestor de la comunicación con el cuadricóptero, sin interrumpir la ejecución de los hilos de graficación y detección de comandos del Joystick.

En la clase VentanaPrincipal del módulo de lógica es instanciado e inicializado un objeto de la clase HiloSerial, el cual, en su constructor se encarga de inicializar y almacenar como atributo un manejador del puerto serial, inicializa variables para el manejo de datos recibidos a través del puerto serial, y declara una variable de control para detener la ejecución del hilo en tiempo real.

Al ejecutarse el método Start del objeto de tipo HiloSerial contenido en la clase de la ventana principal, se inicia la ejecución de un ciclo que:

* Verifica la presencia de datos en el puerto serial, y que en caso de detectar la presencia de los mismos, los recupera del búfer del puerto serial, los decodifica, y valida la presencia de errores en los mismos. En caso de ser recibidos datos válidos, los inserta en una variable para el manejo de datos recibidos hasta que alcanzan una cantidad predefinida, y son enviados a la ventana principal para ser graficados.
* Envía al cuadricóptero comandos que retransmitidos por el objeto VentanaPrincipal, al recibir esta la llamada del hilo de detección de eventos en el joystick.

1. ***Plataforma de pruebas***
   1. ***Montaje para la ejecución de pruebas en un solo eje del cuadricóptero***
   2. ***Montaje para la ejecución de pruebas en vuelo restringido?***
   3. ***Montaje para la ejecución de pruebas en vuelo libre?***