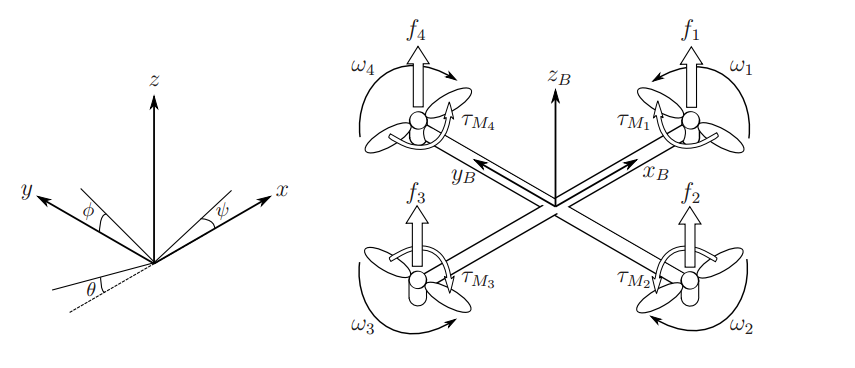
CAPÍTULO II – Marco Referencial

## Cuadricóptero

Un cuadricóptero es un helicóptero con cuatro (4) rotores que estan dirigidos hacia arriba y colocados en forma de cuadrado, equidistantes del centro de masa.  Está basado en tres (3) ejes ortogonales llamados roll (eje “z”), yaw (eje “x”)  y pitch (eje “y”), donde el origen está ubicado en el cruce de los ejes.Los cuadricópteros son controlados por el ajuste de las velocidades de cada rotor. En la Ilustracion 2. 1 se puede observar la estructura básica de un cuadricóptero  con las velocidades angulares (w), torques (t) y fuerzas creadas por los cuatro (4) rotores (f) numeradas del 1 hasta el 4.



Ilustracion 2. Estructura básica de un cuadricóptero

Fuente: <http://sal.aalto.fi/publications/pdf-files/eluu11_public.pdf>

1. **Unidad de Medición Inercial (IMU)**

Una unidad de medida inercial o IMU es un componente electrónico basado en sensores de aceleración y velocidad angular (acelerómetros y giróscopos respectivamente) la cual reporta el movimiento y orientación Ilustracion 2. 2 que sufre dicha unidad. Es el componente principal de sistemas de guía inercial usados en vehículos aéreos, espaciales, marinos y aplicaciones robóticas.



Ilustracion 2. Orientacion porporcionada por una IMU

Fuente: [Bonastre 2010]

**Componentes de una IMU**

Cualquier unidad de medida inercial está compuesta como mínimo por un acelerómetro y un giróscopo para captar una aceleración y una velocidad angular en concreto. Generalmente, es interesante que las IMUs capten la aceleración y la velocidad angular en los tres ejes de coordenadas para conocer el movimiento exacto del componente.

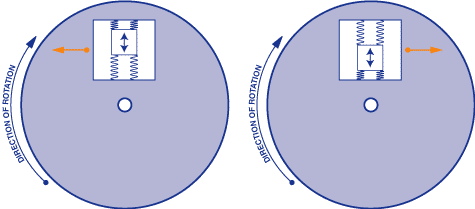
1. **Acelerómetro**

Instrumento capaz de medir aceleración en uno, dos o tres ejes. Existen varios tipos de acelerómetros, dependiendo de su fabricación y funcionamiento. Las IMUs incorporan acelerómetros integrados en silicio, utilizando la tecnología llamada MEMS6, debido a la necesidad de reducir el tamaño total de la unidad. La mayoría de éstos son capacitivos, y calculan la aceleración mediante el voltaje obtenido entre dos placas una de las cuales varía su posición dependiendo del movimiento del acelerómetro. Se caracterizan por ser muy precisos en situaciones estables y tener un gran error en situaciones vibratorias o movimientos muy inestables. [Bonastre 2010]

1. **Giróscopo**

El giroscopio es un tipo de sensor que se utiliza para medir la velocidad angular de un cuerpo en rotación.

Este sensor aprovecha el efecto coriolis el cual aparece cuando un objeto se mueve en un sistema de referencia en rotación, y consiste en que dicho objeto se vea afectado por una aceleración respecto al sistema en rotación. Esta aceleración es perpendicular al eje de giro del sistema y varia según el objeto se acerca o se aleje



Ilustracion 2. 3 efecto coriolis

Fuente: <http://www.analog.com/library/analogdialogue/archives/37-03/Gyro-03.gif>

Las unidades de medida inercial utilizan giróscopos MEMS, es decir, integrados y de tamaño reducido. La salida de dicho sensor es un voltaje, la variación del cual indica en grados por segundo (V/º/s) la velocidad angular sufrida por el sensor. Se caracterizan por tener un error constante y lineal llamado “*bias*” el cual se debe tener en cuenta. [Bonastre 2010]

## Sensor de Ultrasonido

Es un sensor utilizado para medir distancias, el cual emite pulsos ultrasónicos que se reflejan sobre un objeto o una superficie.   Cuando el eco es recibido por el sensor puede calcular la distancia a la que se encuentra utilizando la diferencia de tiempo que tardó un pulso en ir hasta el objeto y regresar. Los materiales pueden ser sólidos o líquidos. Sin embargo han de ser deflectores de sonidos.

## XBee

Es el nombre de la especificación de un conjunto de protocolos de alto nivel de comunicación inalámbrica para su utilización con radios digitales de bajo consumo, basada en el estándar IEEE 802.15.4 de redes inalámbricas de área personal (wireless personal area network, WPAN). Las principales características de ZIGBEE son:

* Velocidades comprendidas entre 20 kB/s y 250 kB/s.
* Alcance de 10 a 75mts.
* Redes cambian los canales en forma dinámica en caso de interferencias.
* Alto ahorro de energía.

Ventajas

* + Ideal para conexiones punto a punto y punto a multipunto.
  + Reduce tiempos de espera en el envío y recepción de paquetes.
  + Detección de Energía (ED).
  + Baja ciclo de trabajo - Proporciona larga duración de la batería.
  + Hasta 65.000 nodos en una red.
  + Son más baratos y de construcción más sencilla.
  + La tasa de transferencia es muy baja.

Desventajas

* Sólo manipula textos pequeños comparados con otras tecnologías.
* Poca cobertura por ser de tipo WPAN.
* No compatible con bluetooth.

La clasificación de dicha tecnología puede ser:

1. De acuerdo a los dispositivos:
   * Coordinador ZIGBEE: se encarga de controlar la red y los caminos que deben seguir los dispositivos para conectarse entre ellos.
   * Router ZIGBEE: interconecta dispositivos separados en la topología de la red.
   * Dispositivo final: se comunica con su nodo padre pero no puede transmitir información destinada a otros dispositivos.
2. De acuerdo a su funcionalidad:
   * Dispositivo de funcionalidad completa: puede funcionar como Coordinador o Router ZIGBEE.
   * Dispositivo de funcionalidad reducida: tiene capacidad y funcionalidad limitadas con el objetivo de conseguir un bajo costo y una gran simplicidad.

## Arduino

Arduino es un proyecto de desarrollo de tarjetas controladoras de hardware libre, de bajo costo y fácil programación, con el fin de acelerar el proceso de prototipado y desarrollo de proyectos, y apoyar la educación en electrónica. Consta de una placa con entradas analógicas y digitales, y salidas digitales, y de un entorno integrado de desarrollo que se apoya sobre el lenguaje de programación Processing. Las primeras tarjetas Arduino utilizaban el microcontrolador ATMEL ATmega328, un chip de bajo costo y amplias capacidades de memoria y manejo de entradas y salidas por medio de sus puertos. Conforme ha ido avanzando el tiempo, se han desarrollado tarjetas Arduino con mejores procesadores, como lo son los modelos Arduino Mega2560 (8 bits), Arduino Tre (32 bits, ARM) y Arduino Galileo (32 bits, x86). [Banzi 2011]



Ilustracion 2. Imagen cara superior Arduino Nano 3.0

Fuente: <http://arduino.cc/es/Main/ArduinoBoardNano>

La placa Arduino Nano 3.0, la misma que se utilizará para el desarrollo del Trabajo Especial de Grado, tiene las siguientes características:

Tabla 2. Características del Arduino Nano 3.0

|  |  |
| --- | --- |
| **Característica** | **Descripción** |
| Microcontrolador | Atmel ATmega328 |
| Frecuencia | 16 MHz |
| Memoria SRAM | 2 KB |
| EEPROM | 1 KB |
| Memoria Flash | 32 KB |
| Entradas analógicas | 8 |
| Entradas/salidas digitales (De los cuales 6 proveen PWM) | 14 |
| Interrupciones | 2 externas, |
| Protocolos de comunicación | USART, , SPI, AREF |
| Tensión de operación | 5V |
| Tensión de entrada (recomendada) | 7-12 V |
| Tensión de entrada (límites) | 6-20 V |
| Corriente máxima por cada pin | 40 mA |
| Dimensiones | 18,5mm x 43,2mm |

Fuente: <http://arduino.cc/es/Main/ArduinoBoardNano>

## Teoría de Control.

La teoría de control es una teoría matemática que rige la manipulación de los parámetros que afectan el comportamiento de un sistema, para producir un comportamiento deseado u óptimo. [Zabczyk 1993].

La teoría de control se ocupa del diseño de algoritmos de regulación de estado, observadores, e identificación de sistemas. Un sistema de control puede definirse como un arreglo de componentes acoplados de tal manera, que el arreglo pueda comandar, dirigir, o regularse a sí mismo o a otro sistema. [Dulhoste 2011]. Un sistema de control está constituido por entradas, salidas y estados [Vidyasagar 2010].

Se dice que un sistema o planta está en lazo abierto cuando las entradas no son afectadas o modificadas por los valores en las salidas de la planta [Rodríguez 2013]. La mayoría de los sistemas de lazo abierto son estables con entradas de referencia limitadas. De lo que normalmente carecen los sistemas de lazo abierto es de velocidad y precisión suficientes para seguir la entrada de referencia aplicada al sistema [Alciatore 2008].

Para un preciso control de un sistema es necesario usar retroalimentación de los sensores (por ejemplo, un codificador o un tacómetro). Al restar una señal de retroalimentación de una señal de entrada deseada (llamada valor de referencia de entrada), se tiene una medición del error en la respuesta. Al cambiar continuamente la señal de comando al sistema con base en la señal de error, se puede mejorar la respuesta del sistema. A esto se le llama control por retroalimentación o de lazo cerrado [Alciatore 2008].



Ilustracion 2. Sistemas de control con y sin realimentación

Fuente: <http://ayciaguillo.blogspot.com/2013/02/1-clase.html>

## Algoritmo Proporcional-Integral-Derivativo (PID)

Los controladores generales pueden tomar muchas formas, pero la mayoría de las aplicaciones industriales usan controladores PID o proporcional-integral-derivativo. La forma matemática de un controlador PID, donde la señal de error se expresa como e(t) es la siguiente:

Ecuacion 2. Formula matematica del controlador PID

Fuente: [Alciatore 2008]

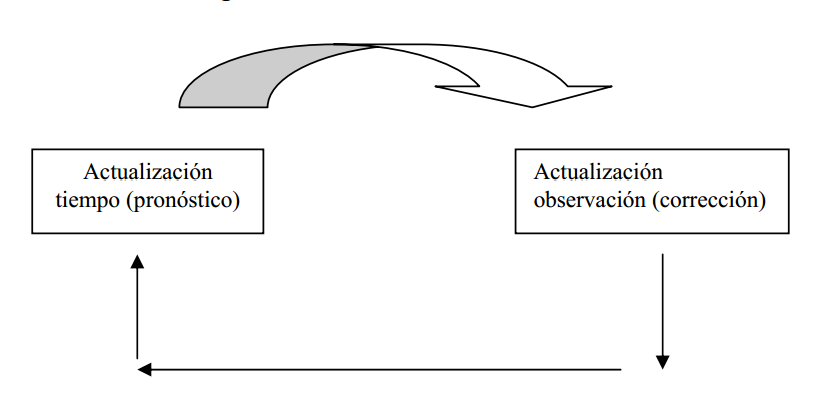
Donde Kp se refiere como la ganancia proporcional, Kd es la ganancia derivativa y Ki es la ganancia integral. El control proporcional es el más intuitivo por que la señal de control es proporcional al error. Mientras más grande sea el error, mayor será la acción correctiva. Una enorme ganancia proporcional crea una respuesta rápida, pero puede conducir a exceso y oscilación, en especial si el sistema tiene poco amortiguamiento. La ganancia derivativa responde a la tasa de cambio de la señal de error. Esto permite al controlador anticipar cambios en la respuesta del sistema, que pueden resultar en menos exceso de oscilación amortiguada. La ganancia integral ayuda a eliminar error de estado estacionario al sumar errores a los largo del tiempo. Mientras más tiempo permanezca el error en un lado de la entrada de referencia deseada, más grande se vuelve la acción correctiva como resultado de la ganancia integral [Alciatore 2008].

## Filtro de Kalman

El filtro de Kalman es un conjunto de ecuaciones matemáticas que proveen una solución recursiva eficiente del método de mínimos cuadrados. Esta solución permite calcular un estimador lineal, insesgado y óptimo del estado de un proceso en cada momento del tiempo con base en la información disponible en el momento t-1, y actualizar, on la información adicional disponible en el momento t, dichas estimaciones. Este filtro es el principal algoritmo para estimar sistemas dinámicos especificados en la forma de estado-espacio (State-space). [Ramirez 2003]

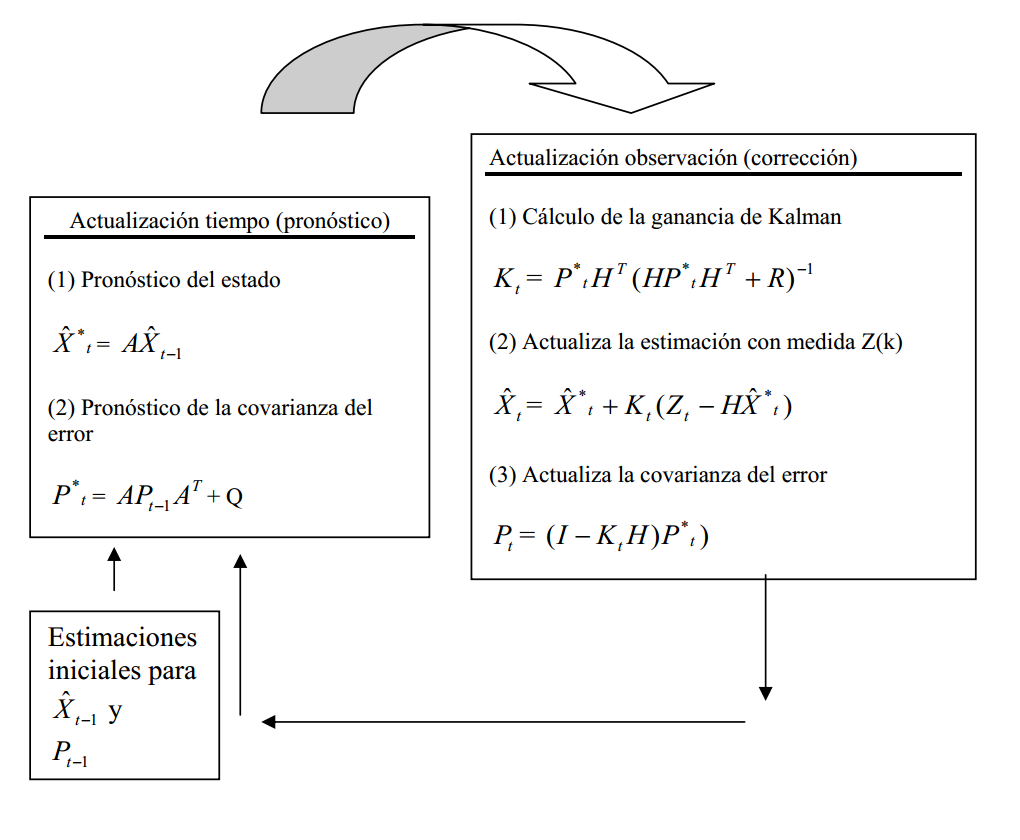
El filtro de Kalman es el principal algoritmo para estimar sistemas dinámicos representados en la forma de estado-espacio En esta representación el sistema es descrito por un conjunto de variables denominadas de estado. El estado contiene toda la información relativa al sistema a un cierto punto en el tiempo. Esta información debe permitir la inferencia del comportamiento pasado del sistema, con el objetivo de predecir su comportamiento futuro. [Ramirez 2003]

Lo que hace al filtro tan interesante es precisamente su habilidad para predecir el estado de un sistema en el pasado, presente y futuro, aún cuando la naturaleza precisa del sistema modelado es desconocida. En la práctica, las variables estado individuales de un sistema dinámico no pueden ser exactamente determinadas por una medición directa. Dado lo anterior, su medición se realiza por medio de procesos estocásticos que involucran algún grado de incertidumbre en la medición. [Ramirez 2003]

****

Ilustracion 2. Ciclo del filtro de kalman

Fuente:[ramirez 2003]

****

Ilustracion 2. Una vision del filtro de Kalman

Fuente:[ramirez 2003]

## Robot Operating System (ROS)

Es una plataforma de software para robótica desarrollada en 2007 por el Laboratorio de Inteligencia Artificial de Stanford (Stanford Artificial intelligence Laboratory). Aunque o es propiamente un sistema operativo, provee librerías y herramientas para ayudar a los desarrolladores de software en la creación de aplicaciones de robots. Está provisto de abstracción de hardware, controladores de diversos dispositivos, visualizadores, pase de mensajes, manejo de paquetes entre otras características. Unos de sus principales rasgos distintivos es el hecho de que es completamente “open source”, es código abierto bajo el estilo de licencia BSD, es libre de usarse, cambiarse y comercializarse. El objetivo principal de ROS es permitir, o facilitar propiamente, a los desarrolladores el diseño construcción y generación de robots cada vez más capaces, consiguiendo aplicaciones de forma sencilla y rápida.[Alvarez 2012]

Fue creado con la finalidad de integrar a gran escala una gran diversidad de sistemas robóticos, que para la actualidad simplifica la escritura de software para robots debido a su constante cambio y crecimiento. [Alvarez 2012]

Objetivos Filosóficos de ROS:

* Red punto a punto: un número de procesos, con posibilidades de diferentes anfitriones conectados en tiempo real por una topología punto a punto.
* Multilenguaje: está diseñado para soportar diferentes lenguajes de programación (C++,Python, Octave y LISP). Para esta característica se emplea un lenguaje neutral y se hace uso de una interfaz de definición de lenguaje, donde se establecen los mensajes que serán empleados para la comunicación entre los módulos.
* Herramientas: para facilitar el manejo de ROS se cuenta con varias herramientas, utilizadas para generar los diversos componentes de ROS.
* Ligero: los algoritmos generados para la operatividad de los robots existentes podrían ser reutilizados fuera del proyecto.
* Software Libre: el código fuente de ROS es público y se encuentra disponible. Distribuido bajo la licencia BSD que permite el desarrollo de proyectos comerciales como de investigación.

Fundamentos:

Las bases operacionales en las que se encuentra implementado ROS son:

* Nodos: son los módulos de software (programas) que componen el sistema.
* Mensajes: son los datos que se utilizan para la comunicación entre los nodos.
* Tópico: es la configuración de la comunicación en los nodos (esta puede ser escuchar o hablar)
* Cliente-Servidor: es el sistema de comunicación definido por una llamada a un servicio, en la que un cliente realiza una petición y un servidor responde a esta.