



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

---

---

**FACULTAD DE INGENIERÍA**

**SIMULADOR PARA PRUEBAS DE CONTROL DE  
ORIENTACIÓN DE SATELITES.**

**T E S I S**

QUE PRESENTAN  
PARA OBTENER LOS TÍTULOS DE:

INGENIERO ELÉCTRICO - ELECTRÓNICO:

FRANCISCO GABRIEL MÉNDEZ SALAZAR

INGENIERO EN COMPUTACIÓN:

CÉSAR DAVID HUANTE ARANA

DIRECTOR DE TESIS:

**Dr. en I. JORGE PRADO MOLINA**



CD. UNIVERSITARIA MÉXICO D.F.

AGOSTO 2009



Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

---

## DEDICATORIA

A mis padres, amigos, familiares, maestros, compañeros que me han apoyado para lograr esta meta en mi vida.

A mi madre Silvia Salazar Luna que fue y ha sido el pilar fundamental para llegar a ser quien soy por lo cual le agradezco infinitamente su apoyo. Se que aún falta más metas por cumplir y se que seguirás apoyándome de alguna forma u otra. Te quiero mama esta tesis es para ti.

A mi padre †Rufino Donaciano Méndez De la Rosa que fue y seguirá siendo una de mis más grandes inspiraciones para recorrer el largo camino de la vida, me enseñó a no claudicar y a luchar, ahora se que uno vive de esperanzas que lo hacen seguir adelante y muere con otras tantas que lo hacen descansar en paz. Nuevamente gracias Papa. Tomaré las cosas buenas de ti y de las malas aprenderé.

A mi hermana que en una versión anterior no la mencione y con su gran corazón me dijo que no había problema, en estas fechas le he prometido una visita a Paris, para que vea que se le aprecia y mucho, además también he recibido su apoyo incondicional. Te quiero Angélica.

A mis mejores y grandes amigos que en un momento u otro estuvieron conmigo, Vani, Dani, Lucero, Ricardo, Armando, Diego, Francisco Becerra, David.

A todos mis familiares que me apoyaron para seguir adelante con mi carrera en tiempos difíciles, son bastantes, pero a todos los que contribuyeron a cumplir esta meta les aprecio y agradezco en sumo en especial a mi tío Leo y a su familia.

A mis amigos que esta gran Universidad me dio: Lucero, Ricardo, Armando, Rubén, Gaby y a todos los que se me olvide mencionar; también aquellos que he conocido recientemente y que de alguna forma u otra me han ayudado a superarme, ya que compartimos una visión de vida y de libertad financiera Ale, Mariano, Os, Edgar y todos los miembros de Rich Dad México.

Como agradecimiento especial a aquellas personas que me ayudaron en las etapas de mi formación de manera muy especial, a Lalo Cano, a Francisco Becerra en la preparatoria y Lucero en la Universidad. Gracias por pasarme las tareas y ser excelentes amigos.

Y por supuesto a mi compañero de tesis y ahora una gran amigo de vida, David Huante, de el cual me llevo maravillosos momentos, anécdotas y por su supuesto la gratitud de haber trabajado con él.

Francisco

---

## DEDICATORIA

A Dios por otorgarme una vida y guiarme en ella.

A mi padre Rafael Huante que es mi gran inspiración en la vida, por todas las cosas que me ha enseñado, y que gracias a su ayuda pude lograr esta meta. Gracias por ayudarme siempre en todo.

A mi madre Lupita Arana porque siempre me apoyo con su cariño y confianza.

A mis hermanos Francisco y Rafael por apoyarme y entenderme durante todo este tiempo.

A mi amor Rosa Chán que siempre me brindó un apoyo incondicional y que gracias a su confianza, entrega y valor nunca me deje derrotar por los obstáculos. Gracias amor por que sin tu ayuda no lo hubiera logrado siempre vas a ser alguien especial para mí.

A mis más grandes amigos que siempre me ayudaron de una manera desinteresada, Rosa, Omar, Antonio, Daniel, Cristian y Christian.

A mis amigos de la Universidad: Daniel, Fernando, Marcos Peña, David Mannig, Gaby, Máximo, David, Francisco y a todos los que se me olvide mencionar, por su aprecio y apoyo en la carrera.

A todos los compañeros del Instituto de Geografía que de una u otra manera apoyaron a desarrollar este proyecto.

A la DGSCA-DS junto con la Lic. Laura Liyen por haberme aceptado como becario y haberme brindado la mejor de las oportunidades. Así como a mi maestro de la SSW, Mario, por su enseñanza y comprensión.

A mis compañeros de la DGSCA y la SSW, Hugo, Berenice, Sandra, Iván, Claudia, Fernando, Francisco, Gerardo. Y todos aquellos que estuvieron conmigo en ese gran paso de mi vida.

A mi compañero de Tesis Francisco que me dio la oportunidad de trabajar con él y que aparte de la realización exitosa de esta tesis gané un gran amigo. Gracias.

César David

---

## RECONOCIMIENTOS

Agradecemos a la Universidad Nacional Autónoma de México y a la Facultad de Ingeniería por la oportunidad de formar parte de este grandioso recinto que es la máxima casa de estudios, por tantas oportunidades que nos ofrecen para aprender cada día algo nuevo que fortalece nuestra persona.

Al Dr. en I. Jorge Prado Molina, nuestro asesor, por su valioso apoyo en el desarrollo de esta tesis, por su dedicación al proyecto, además de ser una gran persona con la cual compartimos muy buenos momentos y recibimos excelentes consejos.

Al instituto de Geografía, por brindarnos la oportunidad de desarrollar este proyecto, así como a toda la gente que trabaja en el, y que de una u otra manera contribuyó a este gran trabajo, que ya ha alcanzado trascendencia internacional.

A cada uno de nuestros profesores porque gracias a sus enseñanzas ahora somos Ingenieros.

Y por supuesto a nosotros mismos por habernos trazado una meta de vida y haberla cumplido.

¡Felicidades David y Francisco!

# ÍNDICE

DEDICATORIA	i
RECONOCIMIENTOS	ii
ÍNDICE	iv
RESUMEN	vi
<b>1. INTRODUCCIÓN.</b>	<b>1</b>
1.1 Antecedentes.	1
1.2 Sistema de simulación para pruebas de orientación y control de satélites pequeños.	2
1.3 Plataforma de Simulación	3
1.3.1 Características	4
1.3.1.1 Medio sin fricción	5
1.3.2 Balanceo de la plataforma	6
1.3.2.1 Balanceo manual y automático	8
1.4. Computadora de abordo	9
1.4.1. Características de la computadora de abordo	10
1.5. Sistema de monitoreo	10
1.5.1. Interfaz Gráfica	10
1.5.2 Interfaz bidireccional inalámbrica.	12
1.5.3 Protocolo de comunicación.	13
1.5.4 Envío de comandos.	13
1.6 Programación de rutinas en C.	14
<b>2. COMPUTADORA DE ABORDO.</b>	<b>15</b>
2.1. Características y desempeño.	15
2.2. Arquitectura.	18
2.2.1 Puertos	19
2.3. Interfaz bidireccional inalámbrica.	20
2.3.1. Protocolo de comunicación entre radiomódems.	20
2.3.2. Principio de funcionamiento de radiomódems.	21
2.3.3. Tarjeta de desarrollo y su configuración.	23
2.4 Librerías y Modulos adicionales	24
<b>3. SENSORES Y ACTUADORES.</b>	<b>26</b>
3.1. Sensores	27
3.1.1. Sensor de sol	27
3.1.2. Sensor de Tierra	28
3.1.3. Brújula Electrónica	29
3.1.4. Circuito auxiliar de regulación y divisor de voltaje.	29
3.2. Actuadores	31
3.2.1. Bobinas magnéticas	31
3.2.1.1. Geometría	32
3.2.1.2. Cálculo de dipolos magnéticos.	32
3.2.1.3. Ubicación de las Bobinas en la plataforma	34
3.2.2. Ruedas Inerciales	35
3.2.2.1. Sistemas de control de ruedas de reacción	36
3.2.2.2. Salidas de potencia, Puente H	36
3.2.3. Masas deslizantes.	37
3.2.3.1. Diseño y operación de las masas deslizantes.	38
3.2.3.2. Motor a pasos.	39

---

3.2.3.3. Control del motor a pasos	40
3.2.3.3.1 Utilización de un PIC para el control de los motores a pasos.	40
3.3. Ensamble final de los contenedores de circuitos.	42
3.3.1 Cajas "Xenix y Meteoro".	42
3.3.2. Conectores Exteriores	45
<b>4. INTERFAZ GRÁFICA ENTRE LA COMPUTADORA DE ABORDO Y LA ESTACIÓN TERRENA.</b>	<b>48</b>
4.1. Descripción.	48
4.2. Comandos.	51
4.3. Instalación de nuevas rutinas de orientación y control.	52
4.4. Almacenamiento de datos de orientación de la plataforma.	53
4.4.1. Despliegue de datos de orientación	53
4.4.2. Almacenamiento de datos en una Base de datos.	54
<b>5. PROGRAMACIÓN DE RUTINAS EN LENGUAJE DE ALTO NIVEL Y PRUEBAS DE FUNCIONAMIENTO.</b>	<b>56</b>
5.1. Programación de la computadora de abordó.	56
5.1.1. Pruebas de funcionamiento del sistema de monitoreo de la plataforma de simulación.	57
5.2. Balanceo automático de la plataforma de simulación.	58
5.2.1. Balanceo manual.	58
5.2.2. Balanceo Automático.	59
5.2.2.1. Curvas de respuesta.	61
5.3. Pruebas de oscilación simple.	62
5.4. Control de orientación con bobinas magnéticas.	66
5.4.1. Encendido de bobinas para control de orientación.	66
5.4.1.1. Curvas de respuesta	68
5.5. Control de orientación con ruedas inerciales.	69
5.5.1. Curvas de respuesta	71
<b>6. CONCLUSIONES</b>	<b>75</b>
<b>APÉNDICE A</b>	<b>76</b>
<b>APÉNDICE B</b>	<b>79</b>
<b>APÉNDICE C</b>	<b>81</b>
<b>REFERENCIAS</b>	<b>86</b>

## Resumen

Se presenta la siguiente generación de una plataforma utilizada para pruebas de funcionamiento de algoritmos de control de orientación de satélites pequeños. Una PC en una sola tarjeta, que hace las veces de computadora de abordo, se encarga de ejecutar algoritmos de balanceo automático, freno magnético, orientación con bobinas magnéticas, ruedas inerciales, y desaturación de ruedas inerciales, todos ellos implementados en lenguaje de alto nivel.

Una interfaz gráfica, desarrollada de manera específica para esta aplicación, se encarga de enviar comandos y cargar nuevas rutinas de control, a través de un sistema de comunicación bidireccional inalámbrico. Éste también permite el monitoreo de los tres ejes de referencia de la plataforma para su almacenamiento y análisis en post-proceso.

Se describe con detalle la programación desarrollada, la actualización en la circuitería y las pruebas de funcionamiento efectuadas en este nuevo sistema de simulación.



## Capítulo

## 1

## INTRODUCCIÓN

## 1.1 Antecedentes

En el Laboratorio de Percepción Remota Alternativa y Tecnología Avanzada del Instituto de Geografía de la UNAM, se han realizado diversos proyectos para generar la infraestructura necesaria con el fin de llevar a cabo los procesos de detección de orientación y de control de estabilización de satélites pequeños. Se ha diseñado y construido un sistema de simulación que permite probar físicamente el comportamiento de algoritmos, sensores y actuadores. Se han programado algunos algoritmos de detección de orientación y de control de apuntamiento y se ha generado un medio que permite simular físicamente la falta de fricción que existe en el espacio exterior (Prado J. 2007).

Una parte importante de este sistema la constituye la plataforma de simulación, que es donde se genera un medio con fricción despreciable, y es aquí donde se lleva a cabo la integración y las pruebas de funcionamiento de todos los dispositivos y programas desarrollados. Sin embargo, el añadir componentes en un medio sin fricción, aún hasta un simple cable, causa necesariamente un desbalanceo. Por este motivo, las señales que entren o salgan de la plataforma necesariamente deben hacerlo de manera inalámbrica. El desbalanceo no permite llevar a cabo experimentos confiables, ya que los pares externos a la plataforma, que son principalmente de origen gravitacional, causan que los actuadores estén sobredimensionados, lo que implica aumento en masa y energía; recursos muy valiosos en un satélite pequeño.

Para mantener los pares externos al mínimo, un sistema automático lleva a cabo un balanceo preciso de la plataforma, al tiempo que se efectúa un monitoreo de los movimientos en tres ejes para visualizar directamente la orientación y tener una referencia rápida de lo que está ocurriendo con los experimentos. Este sistema de monitoreo utiliza un medio de almacenamiento de la información de la orientación, en archivos que permiten analizar posteriormente el comportamiento de la plataforma.

Anteriormente todo esto fue desarrollado con base en un microcontrolador HC11, utilizando lenguaje ensamblador para programar todas las rutinas. A través del paso de los años, los nuevos avances tecnológicos y su adaptación y asimilación en la industria, ofrecen mejoras significativas en cuanto a velocidad de desarrollo y funcionamiento, con herramientas más versátiles y capacidades aumentadas, lo que hace necesaria una actualización. En nuestro caso, una nueva plataforma de simulación con mejoras en cuanto a la computadora de abordo, el sistema de monitoreo inalámbrico, así como nuevos actuadores y circuitos de potencia; proveerán de un sistema moderno, confiable y con mayores posibilidades de aplicación.

Este trabajo de tesis consistió en el desarrollo una nueva plataforma con base en la anterior (Contreras F., 2004), pero con la implementación de una computadora en una sola tarjeta (microcontrolador Rabbit 4000) de características más adecuadas para nuestro fin; con un sistema de comunicación full-duplex integrado al mismo circuito y con una prominente mejora en los circuitos de potencia.

Cabe señalar que nuestro proyecto es parte intermedia de la evolución de este tipo de plataformas; se han tomado las ideas más recientes para su actualización.

## 1.2. Sistema de simulación para pruebas de orientación y control de satélites pequeños.

Este sistema sirve para probar de manera experimental el desempeño de componentes en desarrollo, como sensores, actuadores y algoritmos que serán incluidos posteriormente en equipos espaciales. Además hace las veces de plataforma estructural para la integración de sensores, actuadores, baterías, sistemas de comunicaciones y controladores, permitiendo así una evaluación de su desempeño global.

Este sistema está integrado por: a) una plataforma circular suspendida sobre un balero de aire esférico que es donde se genera un medio sin fricción, b) tres ruedas inerciales que constituyen el grupo de actuadores para control de orientación de la plataforma, c) tres bobinas magnéticas, localizadas en ejes mutuamente perpendiculares, que de-saturan las ruedas inerciales y además proporcionan un sistema de control de respaldo, d) dos inclinómetros y un magnetómetro para determinar cualquier desviación angular en los tres ejes de la plataforma y e) dos masas deslizantes para el balanceo automático. También cuenta con un sistema de monitoreo inalámbrico que transmite la orientación de los tres ejes, durante las pruebas. Estos datos son desplegados y almacenados, permitiendo así la evaluación de sensores, actuadores y algoritmos en un post-proceso. En la figura 1.1 se muestra el diagrama esquemático de este sistema.

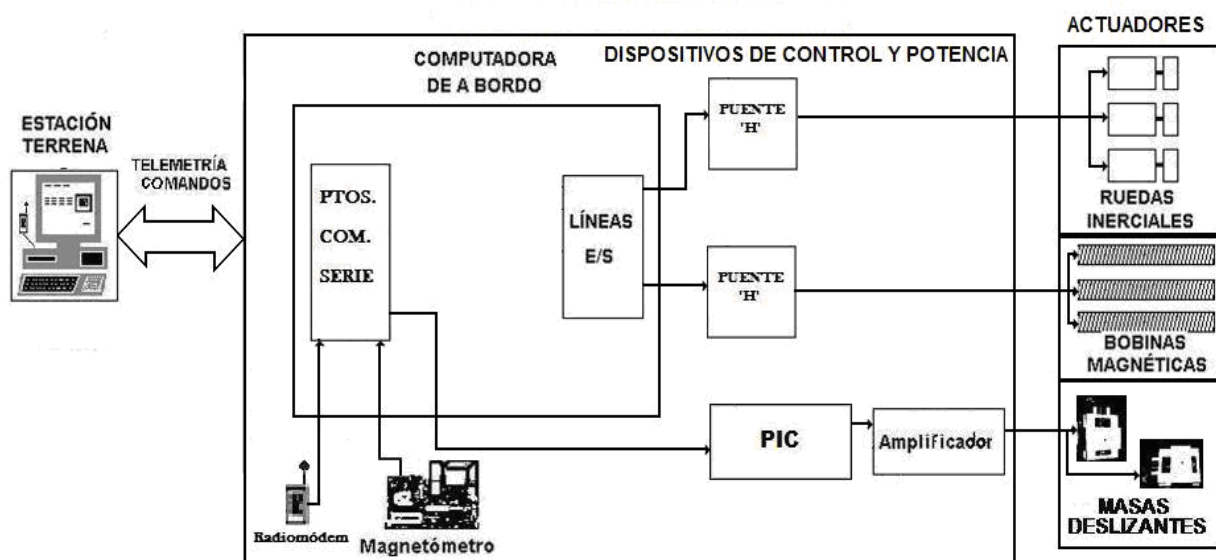
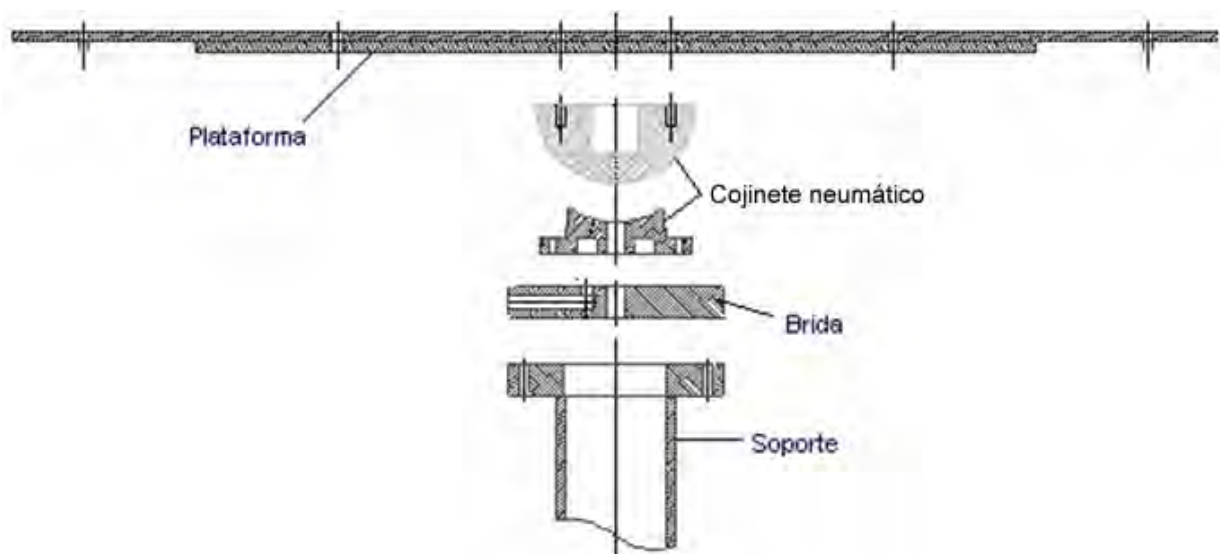


Figura 1.1.- Diagrama de bloques del sistema de simulación para prueba de algoritmos de control de orientación para satélites pequeños.

### 1.3 Plataforma de Simulación

Para poder llevar a cabo el desarrollo y las pruebas de funcionamiento de los sistemas de control de orientación de satélites de una manera realista y objetiva, es necesario contar con un equipo que nos permita simular las condiciones ambientales del espacio exterior como son: vacío, microgravedad, falta de fricción, etc. Como no es posible emular al mismo tiempo todas estas condiciones, entonces es necesario tener diferentes equipos para este propósito. En particular, lo que más nos interesa desde el punto de vista dinámico, es la falta de fricción. Por esta razón debemos contar con un simulador que nos permita tener, movimiento angular en los tres ejes y un medio con fricción prácticamente nula. El diseño básico de este tipo de simuladores, consiste de una plataforma móvil donde se colocan los componentes del sistema, suspendida sobre un soporte que permita el movimiento en los tres ejes a controlar, con fricción despreciable (figura 1.2). Esto nos lleva necesariamente a un cojinete neumático esférico como la única solución para soportar a la plataforma (Prado J. et. al. 2005).

Los cojinetes neumáticos son dispositivos auxiliares muy utilizados para el desarrollo de la programación y la instrumentación de sistemas para determinar la orientación de naves espaciales. Aunque existen otras posibilidades de simular la falta de fricción, éstas presentan inconvenientes como el bloqueo que ocurre con los sistemas basculantes, por ejemplo, además de que no presentan fricción despreciable (Prado J. et. al. 2005).



*Figura 1.2.- Diagrama general de la plataforma de simulación donde se muestran los componentes principales.*

En el Laboratorio de Percepción Remota Alternativa y Tecnología Avanzada del Instituto de Geografía de la UNAM, se ha trabajado por varios años para obtener un diseño básico de este tipo de simuladores, que consiste de una plataforma que además sirve como medio de integración de:

baterías, controladores, sensores, actuadores y todos los componentes del sistema de balanceo automático (Contreras F., 2004).

Debemos tener en cuenta que simulador debe ser totalmente autónomo, es decir, la energía para funcionar debe venir de él mismo, y la transmisión de información sólo es permisible a través de señales electromagnéticas.

### 1.3.1 Características

El simulador consiste de una plataforma circular de 76 cm de diámetro construida con espuma de estireno emparedada por fibras de carbono-epoxy, un balero de aire esférico multifujo, con diámetro de 10 cm, maquinado en bronce SAE 62 con acabado a espejo y con tolerancias muy cerradas, con una capacidad de carga nominal de 76 kg. El sistema completo se muestra en la (figura 1.3).

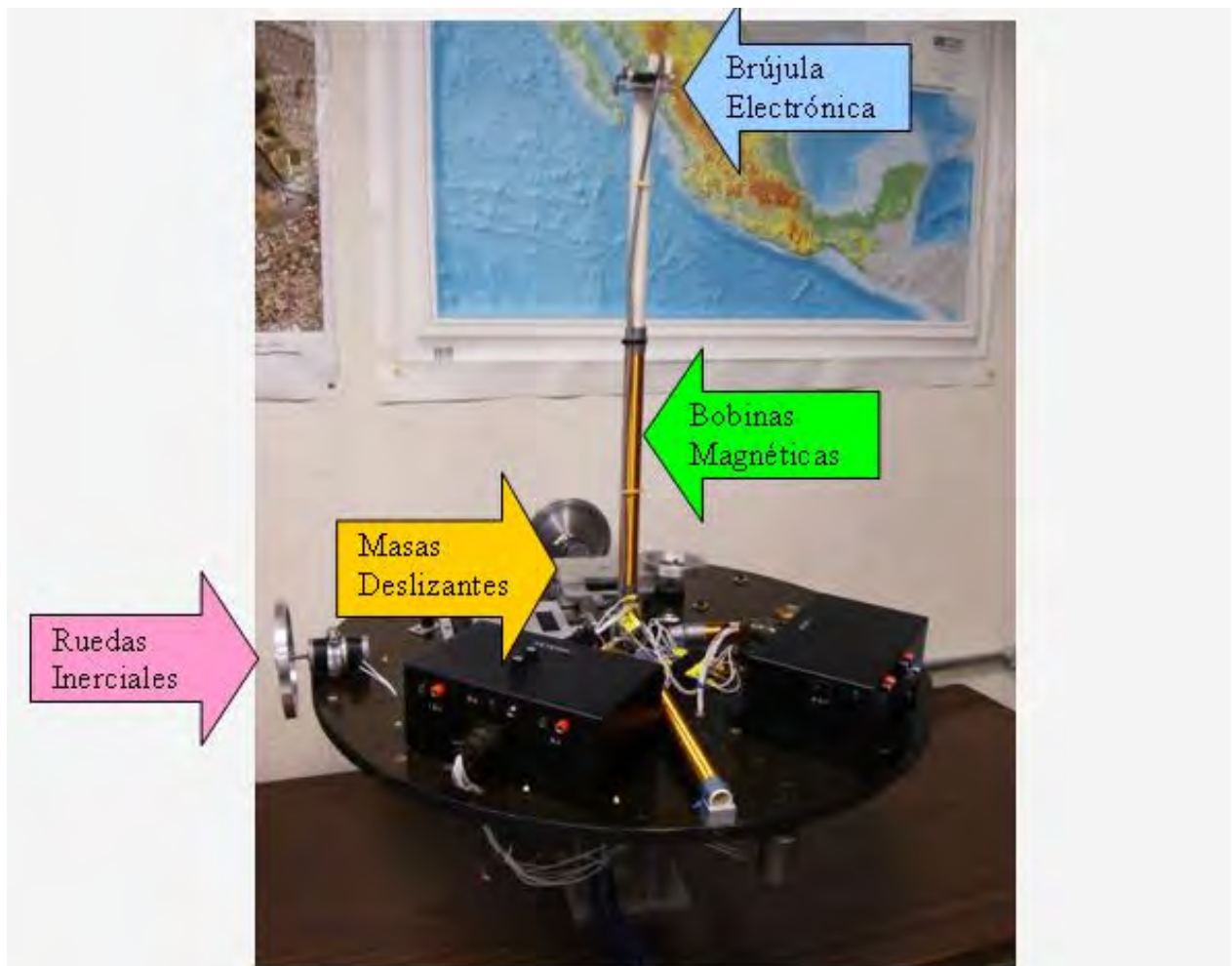


Figura 1.3.- Plataforma de simulación de un medio sin fricción, que sirve como medio de integración de todos los componentes de los subsistemas de control de orientación.

Los componentes que constituyen los subsistemas de control de orientación en la plataforma son: un microprocesador Rabbit LP-3500, un módem inalámbrico, una brújula electrónica equipada con dos inclinómetros (X,Y), tres bobinas magnéticas, tres ruedas inerciales, seis puentes H como interfases de potencia (tres para las bobinas magnéticas y tres para las ruedas inerciales), una tarjeta de control que consta de un circuito integrado MAX 232 y un PIC 18f452, un circuito de potencia para los motores a pasos, reguladores de voltaje para la alimentación de todos los circuitos y baterías: todos ellos tienen una posición conocida y específica para mantener en equilibrio en la plataforma.

Las tarjetas descritas anteriormente se encuentran situadas dentro de dos cajas con un acomodo vertical o en pila, dichas cajas reciben los nombres de Xenix y Meteoro, para diferenciarlas fácilmente entre ellas.

Xenix contiene los siguientes elementos:

- ❖ Computadora industrial en una sola tarjeta Rabbit Lp3500.
- ❖ Radiomódem para comunicación bidireccional inalámbrica.
- ❖ Tarjeta para alimentación y divisores de voltaje para las salidas tipo “sinking” del microcontrolador.
- ❖ 3 puentes H para los motores que impulsan las ruedas inerciales.

Meteoro contiene los siguientes elementos:

- ❖ Tarjeta del PIC.
- ❖ Circuito de Potencia para los motores a pasos que controlan las masas deslizantes.
- ❖ 3 Puentes H para suministrar potencia suficiente a las bobinas magnéticas.

La alimentación del sistema está dividida en tres secciones. La primera está constituida por una batería de 12 [Volts], la cual proporciona energía al Rabbit, el Radio-Módem y la brújula electrónica; la segunda sección la compone una batería de 18 [Volts] que alimenta a los circuitos de potencia para impulsar los motores de las ruedas inerciales y de las bobinas magnéticas, y la tercera etapa, consta de una batería de 8 [Volts] para el PIC y el circuito de potencia de los motores a pasos.

Durante las pruebas se encontró que aislar la alimentación por secciones proporciona al sistema un mejor aprovechamiento de energía y menos variaciones de corriente a los subsistemas, sobre todo a aquellos más sensibles, como el Rabbit y la brújula electrónica.

#### **1.3.1.1. Medio sin fricción.**

Un cojinete neumático esférico, consiste básicamente de una semiesfera y una copa. Es la unión de estos dos componentes lo que produce el colchón de aire que provee del medio sin fricción (figura 1.4). El aire es introducido por la parte baja de la copa y sale por la periferia, es decir, por su parte superior. La semiesfera se sujeta firmemente a la plataforma y se completa el sistema que simula un medio sin fricción.

Debemos subrayar que una característica importante de los cojinetes neumáticos, es el hecho de necesitar pequeñas presiones y gastos de aire para soportar una carga dada, aunque es necesario colocar filtros para impedir el paso del agua y el aceite ya que éstos últimos pueden romper

fácilmente el delgado colchón de aire. Es necesario mantener en condiciones adecuadas el medio sin fricción, es decir, tener una superficie de contacto limpia y bien pulida entre las partes que conforman el balero de aire.

El cojinete forma un colchón de aire de  $0.0254$  mm entre la semiesfera y la copa. Es de tipo multiflujo, con seis perforaciones capilares con diámetro de  $0.55$  mm proporcionalmente espaciadas. Las dimensiones finales de la semiesfera son de  $99.9348 \pm 0.0414$  mm, mientras que la copa tiene un diámetro de  $100.4727 \pm 0.0235$  mm. Este permite el movimiento de  $360^\circ$  en el eje de guiñada y de  $\pm 50^\circ$  en los ejes de rotación y cabeceo. La presión de trabajo del aire suministrado del exterior, depende de la carga; teniendo un valor de  $3.2$  kg/cm<sup>2</sup> con una masa de  $35$  kg.



*Figura. 1.4.- Semiesfera y copa, partes fundamentales de un balero de aire esférico.*

### **1.3.2. Balanceo de la plataforma.**

El simulador permite llevar a cabo una amplia gama de pruebas de detección de orientación y control de estabilización para satélites pequeños. Sin embargo, para poder efectuar pruebas confiables es necesario que este se encuentre balanceado, lo que ayuda a minimizar los efectos causados por los pares gravitacionales.

Un conjunto de masas deslizantes es utilizado para efectuar un balanceo en el simulador. Para mantener el control de la masa total del sistema y sus propiedades inerciales, las masas utilizadas deben ser móviles y totalmente controlables (Juárez A. 2001). Dichas masas pueden cumplir la doble función de servir como lastre y para llevar a cabo el ajuste fino de localización del centro de masa y de los ejes de inercia principales.

Esencialmente, debe llevarse el centro de masa de la plataforma, al centro geométrico del balero de aire esférico, dicho balero se muestra en la (figura 1.5).

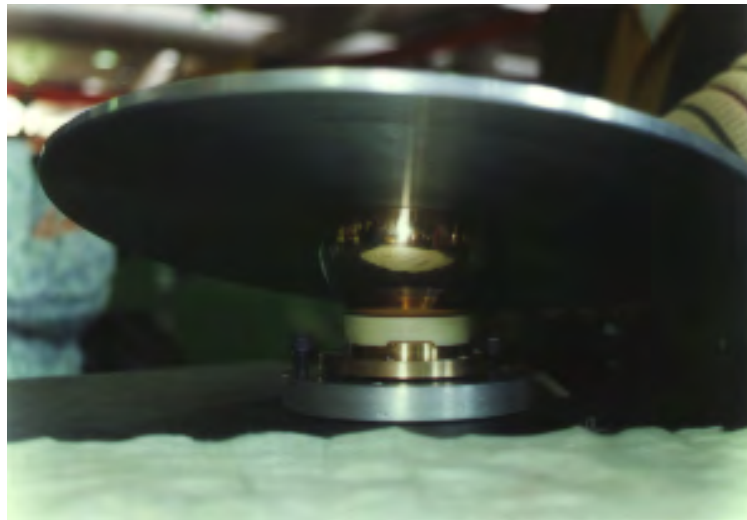


Figura 1.5 Plataforma de simulación y balero de aire esférico donde se produce un medio sin fricción.

El método de balanceo utilizado se lleva a cabo únicamente en dos ejes (Juárez A. 2001). Esto se realiza, por medio de las señales de dos inclinómetros que determinan la orientación de la plataforma con respecto al plano horizontal y de dos masas deslizantes para reposicionar el centro de masa (figura 1.6). Este balanceo es exclusivamente estático y su justificación está basada en el hecho de que los satélites estabilizados por giro, están prácticamente en desuso y no se encuentran contemplados en nuestras aplicaciones.

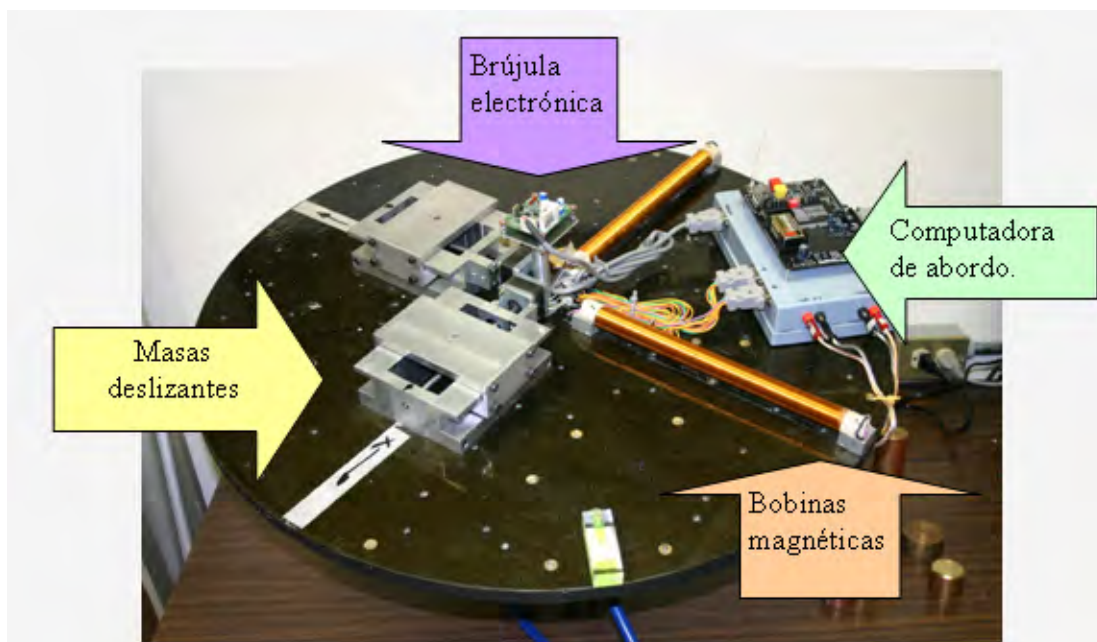


Figura 1.6.- Plataforma de simulación con el sistema de balanceo automático y dos bobinas de control magnético.

Recordemos que los requisitos para el equilibrio estático son simplemente que la suma de todas las fuerzas en el sistema, sean iguales a cero.

$$\Sigma F - ma = 0 \quad (1)$$

Donde:

$\Sigma F$ , es la suma de todas las fuerzas.

$m$ , es la masa y

$a$ , es la aceleración del sistema.

Esto equivale a equilibrar todos los pares existentes en la plataforma, manteniendo su centro de masa en el centro del cojinete neumático. En la figura 1.6 se pueden apreciar las masas deslizantes; localizadas en dos ejes perpendiculares, una brújula electrónica; ubicada en el centro de la plataforma, que hace las veces de magnetómetro y que adicionalmente cuenta con dos inclinómetros que son los que proporcionan las señales de control para lograr el balanceo en dos ejes, el sistema de transmisión inalámbrica; que envía la orientación en tres ejes, proporcionada por la brújula, y por último; el microcontrolador que se encarga de ejecutar el programa de balanceo manejando los motores de pasos, a través de una etapa de potencia (Contreras F. 2004).

### 1.3.2.1. Balanceo manual y automático.

Para poder proceder con el balanceo automático, se necesita forzosamente que el sistema tenga un equilibrio inicial cercano al óptimo, por tanto, se tiene que hacer un acomodo manual de tal manera que los componentes en la plataforma logren que el centro de masa del sistema quede en el centro del cojinete neumático y ligeramente hacia abajo del eje Z para darle estabilidad al sistema. Esto se lleva a cabo partiendo del hecho de que la plataforma de simulación se encuentra inicialmente balanceada cuando no tiene ningún otro componente, debido a que la masa se encuentra distribuida de manera homogénea en toda su superficie. El procedimiento que seguiremos consiste básicamente en mantener balanceada la plataforma; añadiendo un par de componentes cada vez, contrarrestando cada uno el efecto de desbalanceo causado por el otro.

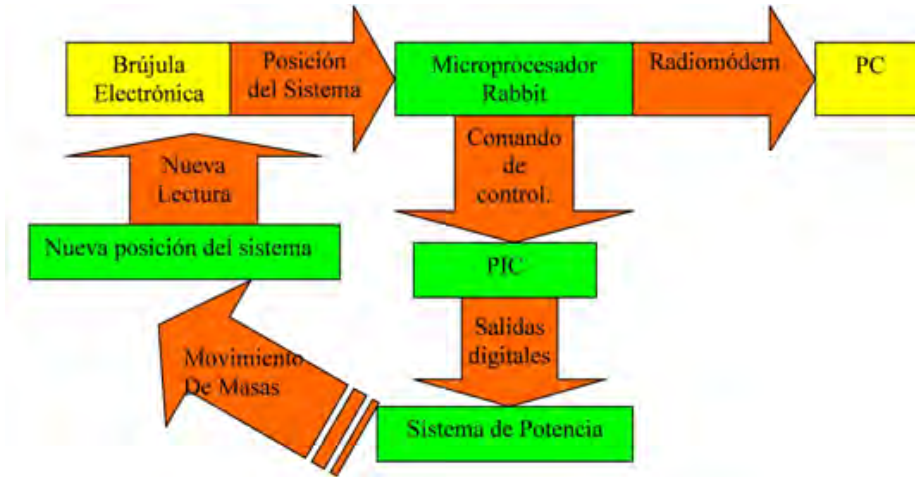
La importancia de esta estrategia consiste en que una vez alcanzado el equilibrio, es posible añadir más componentes, siguiendo el mismo procedimiento y manteniendo sin alteraciones el balance (Juárez A. 2001). El balanceo manual es indispensable para localizar la plataforma a un determinado ángulo de la horizontal, para que entonces entre en funcionamiento el balanceo automático y se lleve a la plataforma a un umbral mínimo de  $\pm 0.5$  grados en los ejes X, Y. Quince grados de desviación son suficientes para evitar que el procedimiento de balanceo sea demasiado tardado.

Una vez que se tiene a la plataforma dentro del rango señalado, se procede a la prueba de balanceo; que en breves pasos, se realiza como se describe a continuación:

La información de orientación es proporcionada por una brújula electrónica y comparada en un microcontrolador, contra el umbral de  $\pm 0.5^\circ$  en los ejes de alabeo y cabeceo; del resultado obtenido de dicha comparación, el microcontrolador manda una secuencia de pulsos, los cuales hacen que los motores de pasos giren en sentido horario o antihorario, lo cual induce el movimiento de las masas deslizantes en una dirección u otra. Cada eje se maneja de manera independiente; es decir, primero se ajusta el eje X y posteriormente el Y, hasta lograr que la



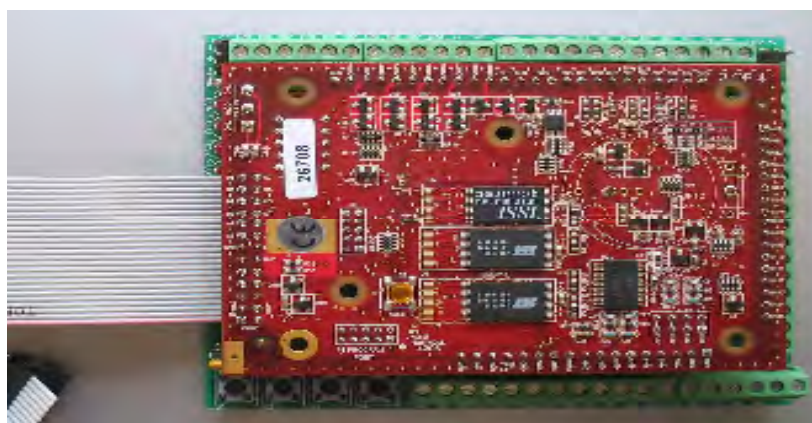
plataforma esté totalmente horizontal. En la [figura 1.7](#) se aprecia la secuencia de balanceo automático.



*Figura 1.7.- Diagrama de bloques del subsistema de balanceo automático.*

#### 1.4. Computadora de abordo.

La computadora de abordo tiene como componente principal un microprocesador Rabbit 3000 con su tarjeta de desarrollo LP Fox 3500 ([figura 1.8](#)). El Rabbit 3000 cuenta con una gama especial de elementos que lo hacen mucho más poderoso que su antecesor, el microcontrolador MCHC11, es además un kit que viene acompañado de dos radiomódems para una comunicación serial full-duplex; siendo de particular importancia resaltar que se puede programar con un compilador de C llamado Dynamic C que cuenta con un amplio contenido de funciones que se encuentran en librerías diseñadas para hacer su uso más rápido y eficiente.



*Figura 1.8.- Computadora de abordo Rabbit 300, con su tarjeta de desarrollo LP Fox 3500 vista en anverso.*

### 1.4.1. Características de la computadora de abordo

La tarjeta LP3500 pertenece, por su estructura, al grupo de computadoras llamadas “single-board” de bajo consumo de energía. Esta cualidad es muy deseable para aplicaciones donde la alimentación proviene de celdas solares o baterías como es nuestro caso. Es una tarjeta de desarrollo que trabaja con un microcontrolador de la marca Rabbit, del cual se presenta un listado que permite conocer sus características principales.

- ❖ Microprocesador Rabbit 3000 que puede operar hasta 7.4 MHZ.
- ❖ 512 kbits de RAM.
- ❖ 512 kbits de Flash.
- ❖ 26 puertos digitales (entradas y salidas).
- ❖ salidas PWM.
- ❖ 6 puertos seriales.
- ❖ Batería de respaldo para el reloj principal.
- ❖ Librería que consta de funciones para hacer multitareas.

Se pueden consultar las características más importantes y especificaciones técnicas en los manuales citados. ([Rabbit Semiconductor 2007](#)).

## 1.5. Sistema de monitoreo

El monitoreo de la plataforma se hace a través de un programa que hace uso de una interfaz gráfica que ha sido desarrollada en Visual Basic y cuyo objetivo es permitir la ejecución de las diferentes rutinas en la plataforma, a través del control de los actuadores, suponiendo una estación terrena alejada del simulador. Este sistema de monitoreo se encarga de generar una base de datos que a su vez permite graficar y analizar el comportamiento de la plataforma durante las pruebas de balanceo automático, oscilación simple, pruebas con bobinas magnéticas y ruedas inerciales, etc. El programa es muy amigable y accesible, además permite entre otras cosas, la configuración del puerto serial de la computadora, establecer los umbrales de precisión en las pruebas de balanceo y ejecutar las diferentes pruebas de orientación y control.

Parte esencial de la interfaz gráfica es su sencillez, para este fin se ha diseñado una página con tablas que despliegan los datos de orientación de la brújula electrónica, que en este caso son las señales provenientes de los inclinómetros y el magnetómetro; detectores principales de la brújula. La idea es mantener una retroalimentación del sistema de manera constante para monitorear en tiempo real la orientación de la plataforma. Cada rutina o programa tiene un determinado umbral, casi en todos los casos determinado por el usuario y se programó de modo que cuando llegue al valor establecido, se despliegue en la pantalla un mensaje que indica que se ha logrado la orientación deseada.

### 1.5.1. Interfaz Gráfica

La interfaz gráfica consta de dos páginas principales; la primera la constituye una pantalla de bienvenida ([figura 1.9](#)) y la segunda es denominada de control ([figura 1.10](#)), ya que permite al usuario mandarle un comando a la plataforma y además también presenta el estatus de cada una de las acciones llevadas a cabo.



Figura 1.9.- Pantalla de bienvenida del sistema de monitoreo.



Figura 1.10.- Pantalla de control del sistema.

### 1.5.2. Interfaz bidireccional inalámbrica.

La selección de los nuevos radiomódems ha sido muy conveniente, ya que presentan características muy adecuadas, al ser instrumentos de relativamente bajo costo para aplicaciones embebidas. También presentan una afinidad con varios elementos que permiten ser reconfigurados a través de un programa bastante accesible el cual maximiza su potencial, puesto que es un dispositivo simple y robusto, con capacidad de trabajar en una red punto a punto o multipunto (figura 1.11).

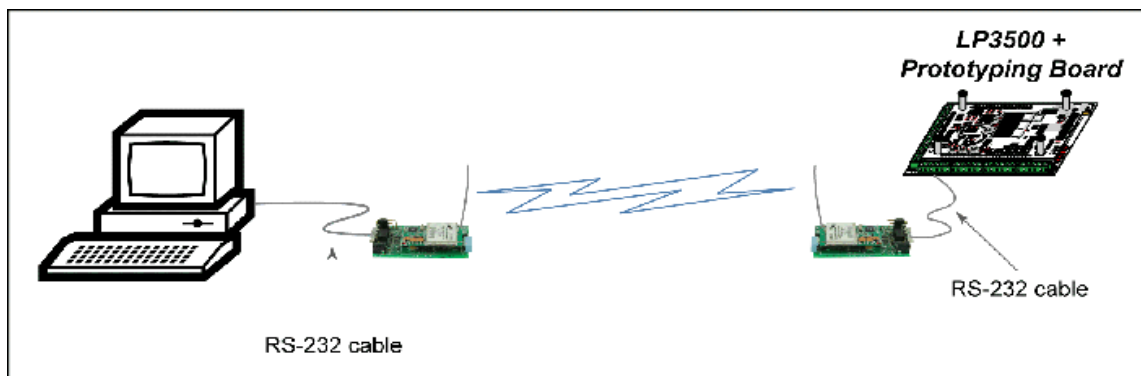


Figura 1.11.- Conexión entre la PC y el microcontrolador Rabbit por medio de los radiomódems.

La comunicación que se puede llevar a cabo con los radiomódems (figura 1.12) tiene varias topologías, nosotros usamos aquella que tiene comunicación punto a punto, con la cual a su vez se puede trabajar de dos formas: una es half-duplex y la otra (la cual usamos) es full-duplex, es decir, bidireccional; a un baudaje de 9600, seleccionado para poder sincronizar los datos de los subsistemas, en un tiempo considerablemente rápido.



Figura 1.12.- Radiomódem Maxstream.

### 1.5.3. Protocolo de comunicación.

La comunicación entre todos los componentes del sistema de simulación, se lleva a cabo siguiendo el protocolo RS-232 que funciona de manera asíncrona. La cantidad de datos y su procesamiento para nuestros fines son apropiados para manejar este protocolo serial; de esta manera sólo tuvimos que configurar los equipos de forma sencilla; puesto que todos los subsistemas que manejamos cuentan con circuitería y programación para el uso de esta interfaz. Cabe señalar que se realizaron varias pruebas para acoplar a una velocidad estable todo el sistema, ya que por ejemplo, se necesitan los datos siempre listos en el microprocesador Rabbit, también el tiempo de procesamiento debe ser constante en todas las tareas a realizar para poder mandar señales de control a los circuitos de potencia y estos a su vez, una señal a los actuadores para que éstos últimos modifiquen la orientación de la plataforma. En algunos casos es necesario introducir retrasos entre los ciclos de control, debido a la naturaleza misma de la plataforma; ya que es necesario esperar a que la oscilación intrínseca de este tipo de aparato se reduzca y se establezca una nueva posición que sea realista y no se vea influenciada por las oscilaciones, por esta razón se debe tener un retardo para tener congruencia con los datos recibidos en la PC y las acciones que fueron ejecutadas por los actuadores.

### 1.5.4. Envío de comandos.

Hay dos partes esenciales que utilizan comandos para poder dar una respuesta al sistema, uno de ellos es el Rabbit que espera comandos de la PC y el otro subsistema es el PIC 18f452 que a su vez espera señales de control del Rabbit.

El envío de comandos entre el Rabbit y la PC se hace a través de la interfaz gráfica, donde se monitorea y controla el desarrollo de las pruebas en el sistema de simulación. Al estar usando un protocolo de comunicación serial, enviamos un comando escribiendo un carácter de control y esperamos a que el receptor, en este caso el Rabbit, mande un eco para confirmar que el carácter ha llegado. De esta forma generamos un conjunto de caracteres de control que son reconocidos en el núcleo de nuestro sistema y que nos permite tener un control total del mismo, siendo este un método confiable para asegurar que los resultados esperados se vean reflejados en nuestra plataforma.

Para el caso del PIC, al ser un circuito secundario, se puede decir que tiene funciones limitadas de control en el sistema, aunque cumple su función al ser controlado de la manera descrita anteriormente.

Para el envío de comandos desde la PC se mandan dos caracteres, uno el que indica que actuador se utilizará y otro para especificar el control que se desea realizar. En las [tablas 1.1 y 1.2](#) mostramos el conjunto de caracteres de control para el microprocesador Rabbit.

**Tabla 1.1.- Se muestra el primer caracter de control para cada actuador**

<i>Caracter</i>	<i>ASCII</i>	<i>Actuador</i>	<i>Descripción</i>
<i>a</i>	97	<i>Motores a pasos</i>	<i>Balanceo</i>
<i>b</i>	98	<i>Ruedas</i>	<i>Modo de reacción e inercia</i>
<i>c</i>	99	<i>Bobinas</i>	<i>Freno Magnético</i>

**Tabla 1. 2.- Segundo caracter de control para el modo de balanceo con motores a pasos.**

<i>Caracter</i>	<i>ASCII</i>	<i>Descripción</i>
0	48	Umbral 3°
1	49	Umbral 2°
2	50	Umbral 1°
3	51	Umbral 0.5°

De estas tablas se obtiene un comando completo. Por ejemplo, para realizar el balanceo por medio de las masas deslizantes a un umbral de tres grados, la PC le manda al Rabbit el comando **a0**. El primer carácter, una **a** le indica que se realizará un balanceo por medio de las masas deslizantes, entonces el Rabbit se va a la rutina del programa para mover los motores a pasos de las masas deslizantes a través del PIC; el **0** corresponde a un umbral de tres grados; lo que significa que el Rabbit estará realizando la rutina de balanceo hasta que los valores obtenidos de la brújula estén en un umbral de 3 grados; es decir el Rabbit recibió la cadena **a0** en su puerto serial E.

En el capítulo tres se describirán con detalle los comandos a usar cuando el microcontrolador Rabbit se comunica con el PIC y en el capítulo cuatro se describen los comandos de control entre la computadora de abordo (microcontrolador Rabbit) y la estación terrena (PC).

### 1.6. Programación de rutinas en C.

Una de las mejoras sustanciales de este sistema, con respecto a su antecesor, es la programación efectuada en un lenguaje de alto nivel, llevado a cabo a través del compilador diseñado para el microprocesador Rabbit : Dynamic C ([Rabbit Semiconductor 2007](#)). El utilizar librerías propias evita el tener que programar rutinas como las que incluyen el uso del punto flotante. Esta es una de las aportaciones de este trabajo de tesis, porque permite una programación más simple de los algoritmos de control. Adicionalmente, el uso de un lenguaje estructurado permite un manejo más comprensible y lógico, por lo que el programa principal está compuesto de varias rutinas; una por cada modo de control, lo que significa que por cada uno de ellos, se tiene un procedimiento de control que es parte de una secuencia maestra o multitareas que permite atender la orden del usuario.

## Capítulo

## 2

## COMPUTADORA DE ABORDO.

**2.1. Características y desempeño.**

En este capítulo se describe, el funcionamiento de la computadora de abordo. Ésta se encuentra conformada por la tarjeta LP3500, a la que nos referimos en este trabajo como microcontrolador o tarjeta Rabbit. Su componente principal es un microcontrolador de la marca Rabbit™ en su tercera generación ([Rabbit Semiconductor 2007](#)). Su velocidad de procesamiento puede alcanzar los 54 Mhz, suficiente para la aplicación que nos ocupa. Es un sistema de bajo consumo que puede trabajar con un mínimo de 3 volts; además, en las salidas de potencia podemos alcanzar el valor de voltaje de alimentación máximo que puede llegar a 12 volts para nuestro caso. Esto es de gran utilidad para aplicaciones de rápida respuesta y que a la vez represente una necesidad de determinada potencia ([figura 2.1](#)). Cabe señalar que en el diseño se configuraron las salidas a 5 volts para manejar los actuadores, para lo cual se utilizaron divisores de voltaje, que envían las señales de control a los circuitos de potencia; siendo estos últimos polarizados a partir de la energía que suministra la plataforma por medio de su sistema de baterías.

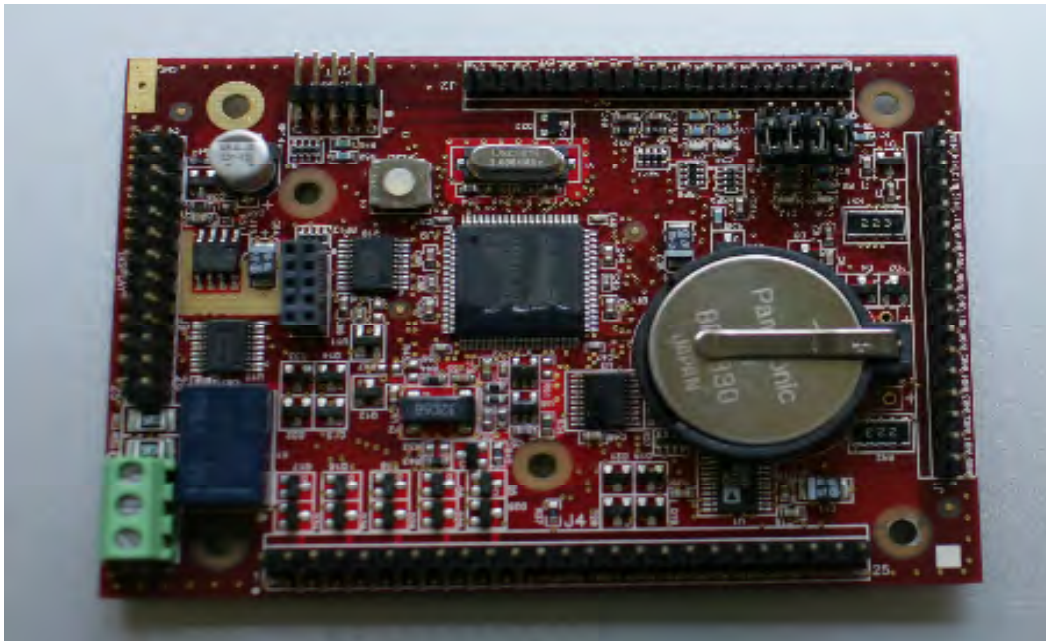


Figura 2.1.- Tarjeta Lp3500 con el procesador Rabbit 3000 en el centro.

Es necesario tener un sistema de simulación para pruebas de algoritmos de control de orientación fiable, que nos garantice un desempeño adecuado en las tareas asignadas, pero sobre todo que se encuentre acorde con los avances tecnológicos que serán implementados en la siguiente generación de satélites pequeños, es por eso que el funcionamiento de la plataforma debe ser óptimo, y por tanto, tener una computadora de abordo con características adecuadas para efectuar las tareas de orientación y control de manera efectiva.

Teniendo como premisa lo antes citado, en esta parte mostramos las propiedades específicas con que cuenta esta computadora.

La tarjeta cuenta con un sistema de protección de picos de voltaje de hasta 60 VDC o 42.2 VAC. Su batería de respaldo ha sido diseñada para soportar condiciones potencialmente adversas como polarización invertida o flujos de corriente excesivos y asegura mayor versatilidad al integrar un equipo de transmisión inalámbrica. Esta computadora en una sola tarjeta cuenta con seis subsistemas que pueden ser manejados independientemente para enfocarse en una tarea específica y los cuales son:

- ❖ Modos de alimentación.
- ❖ Entradas y salidas digitales.
- ❖ Entradas convertidoras A/D.
- ❖ Salidas de potencia para control PWM (Pulse Wide Modulation) modulación por ancho de pulso.
- ❖ Circuito relevador de salida.
- ❖ Memoria.
- ❖ Librerías especializadas para cada uno de estos sistemas y para su control en conjunto, lo cual es de gran utilidad y fue uno de los criterios fundamentales para la elección del microcontrolador.

En la (figura 2.2) se muestra el diagrama a bloques del microcontrolador.

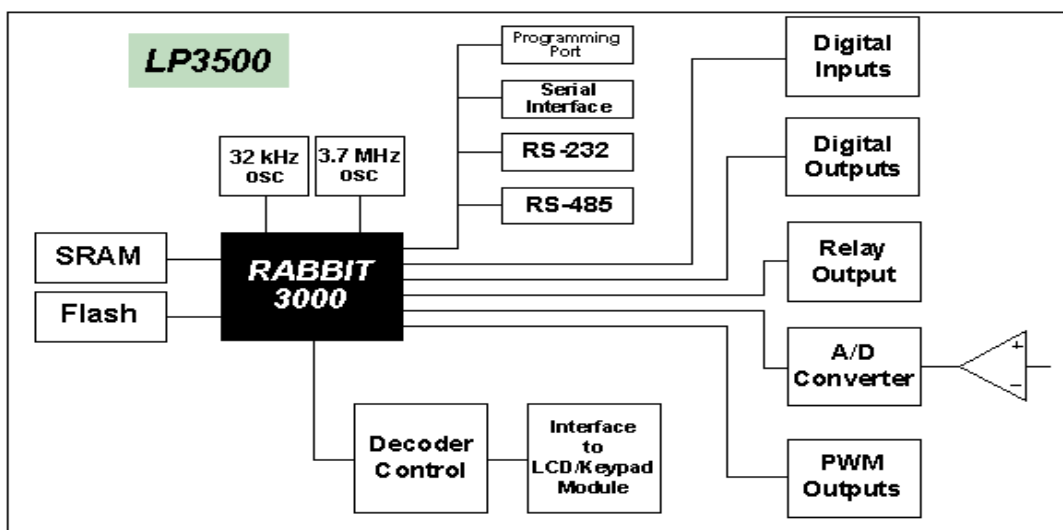


Figura 2.2.- Diagrama de bloques del microcontrolador Rabbit 3000



Una característica importante para nuestra aplicación, es el manejo de energía que hace este microcontrolador, ya que tiene la capacidad de funcionar en diferentes modos de ahorro de energía, por lo que cuenta con una biblioteca con funciones específicas. Por ejemplo, se pueden desconectar algunos puertos del Rabbit que no se utilizan, como la sección de comunicación RS-232 manejada por el puerto PG1 que consume 8 mA. Tomemos en cuenta que el funcionamiento normal del microcontrolador consume alrededor de 20 mA cuando todos sus puertos están encendidos y alrededor de 100  $\mu$ A cuando está en el modo de “ahorro de energía” ([Manual LP3500](#)).

Cuenta con una batería de respaldo para su reloj y la memoria SRAM (Static Random Access Memory), que entra en operación cuando se elimina la fuente de alimentación principal; esta es una batería de litio de 265 mAh, que mantiene en modo activo a los elementos anteriores. Esta fuente puede tener una vida útil de hasta 240 días, trabajando de forma continua, además se puede conectar una batería adicional en la entrada secundaria de la tarjeta LP3500, que podría alimentar al sistema por completo.

Las salidas del Rabbit presentan dos configuraciones diferentes “sourcing” y “sinking”; en nuestro proyecto usamos ambas, solo que para el caso de las salidas tipo sinking tuvimos que acoplarlas de modo que se obtuvieran señales de 5V en alto y 0V en nivel bajo; esto se logró a través de un divisor de voltaje. La [figura 2.3](#) muestra los dos tipos de salida mencionadas, así como los pines a los que está asociada cada una de ellas.

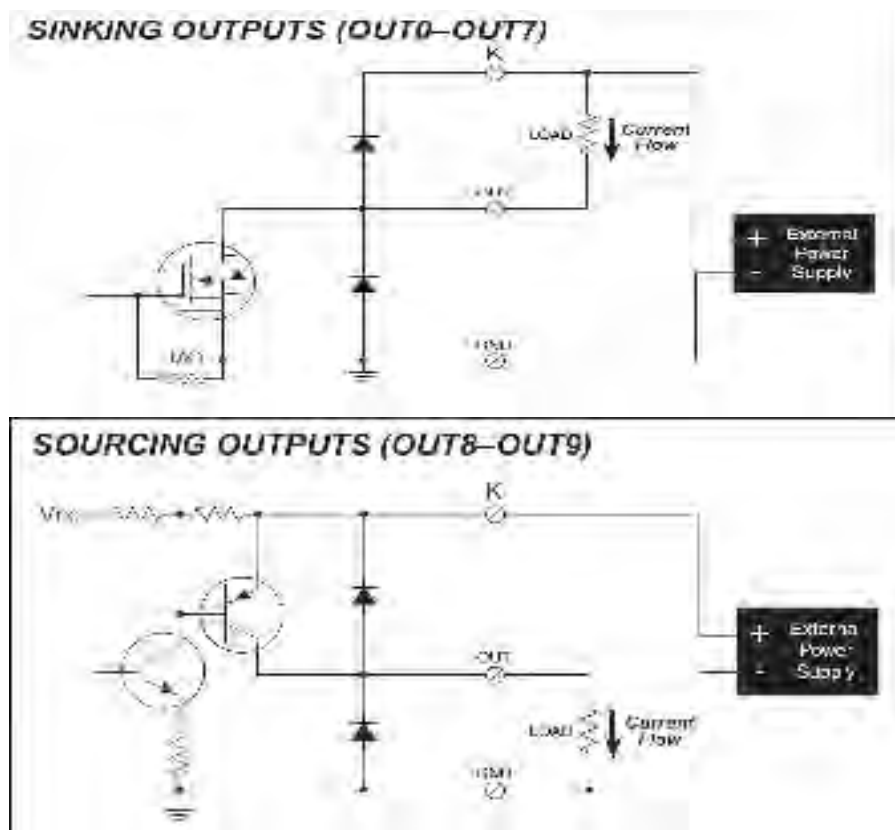


Figura 2.3.- Opciones de salidas que proporciona el microcontrolador.

Las salidas de tipo “sinking” funcionan cuando el microcontrolador cierra la malla a tierra. La carga se conecta antes del interruptor y directamente a la fuente externa; mientras que en la configuración “sourcing” la carga recibe la corriente del pin de salida, mientras que la fuente externa no está conectada directamente a la carga.

## 2.2. Arquitectura.

El núcleo de nuestra tarjeta de desarrollo (LP3500) es el microcontrolador Rabbit 3000, el cual fue diseñado para procesos de control de baja y mediana escala, tiene un bus de datos de 8 bits, de arquitectura Harvard. La tarjeta tiene capacidad para unas 50,000 líneas de programación bajo el compilador C.

El voltaje de alimentación nominal de la tarjeta es de 2.8 volts, proporcionado por un regulador de voltaje, cuyo diagrama se muestra en la figura 2.4. Sin embargo, el micro puede funcionar sin problemas en un intervalo de 1.8 a 3.6 volts. Es importante señalar que el voltaje de polarización de la tarjeta puede llegar hasta 30 VDC, y que las salidas pueden estar acopladas o no a esta alimentación, esto va a depender de una resistencia conectada a una terminal llamada +k, dicha resistencia sirve como puente entre las salidas descritas en la figura 2.3. Esto es, las salidas sourcing presentarán el valor de voltaje al que esté conectado la terminal +k, en nuestro caso 9 V, pero gracias al uso del divisor logramos un promedio de 5 V.

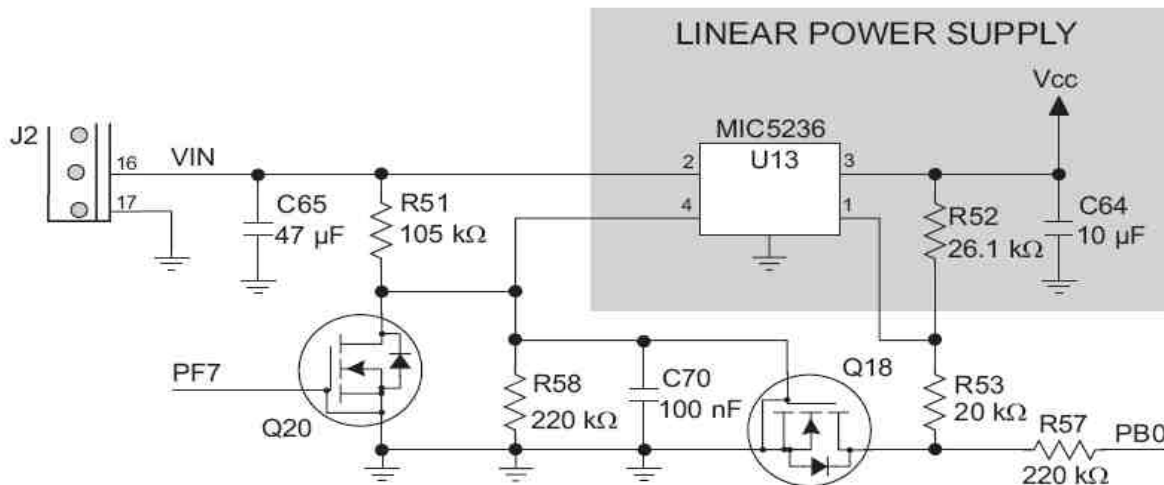


Figura 2.4 Regulador para el microprocesador Rabbit situado en la tarjeta de desarrollo LP3500.

En la figura 2.5 se describe de manera detallada la arquitectura del microprocesador Rabbit 3000.

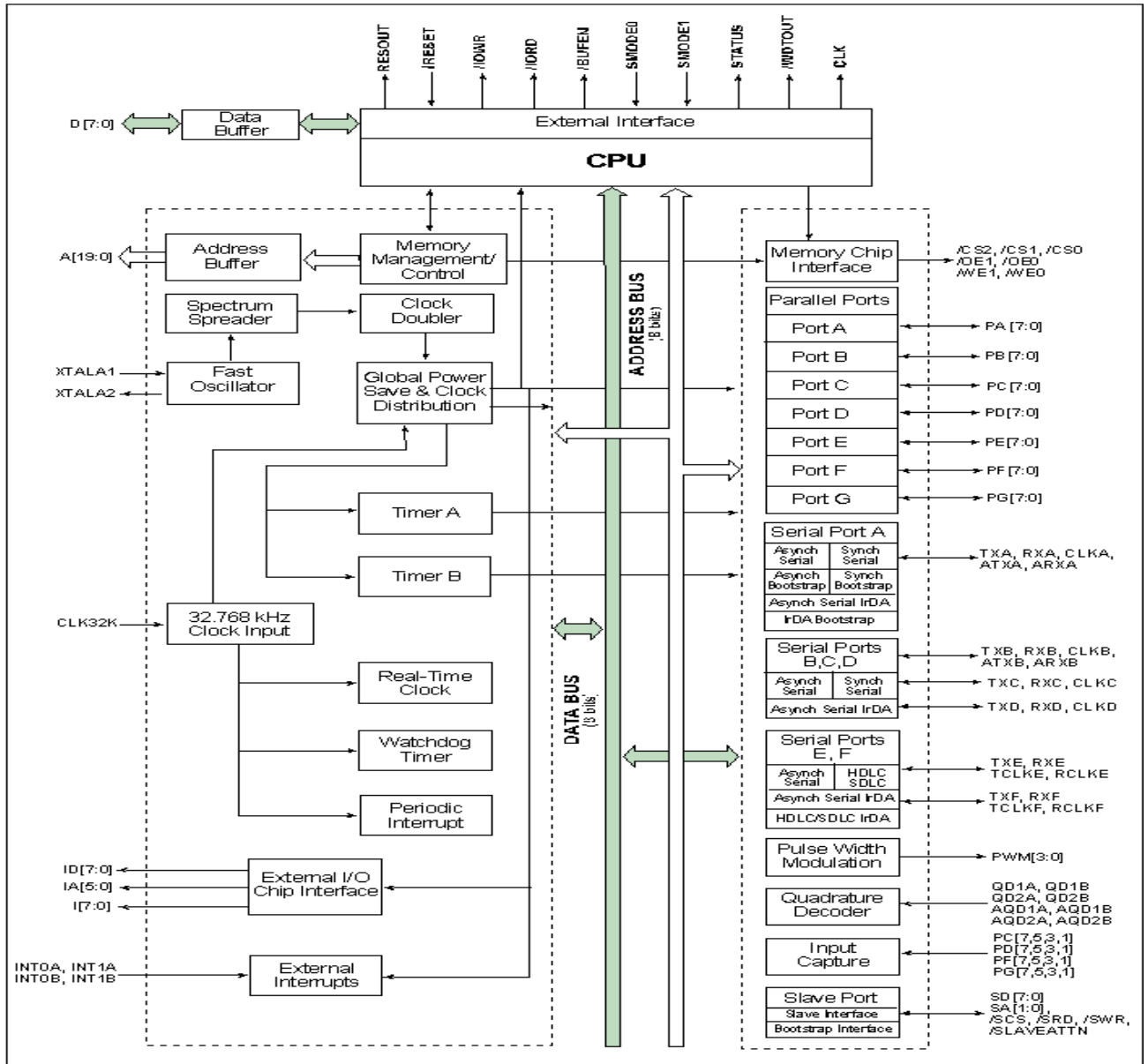


Figura 2.5.- Diagrama que muestra la arquitectura del microprocesador.

### 2.2.1 Puertos

Una de las características más importantes para nuestra aplicación es sin lugar a dudas el número de puertos de entrada y de salida con que cuenta nuestro microprocesador, debido a la gran cantidad de sensores y actuadores que deben utilizarse en un momento dado, para cubrir las diferentes opciones de rutinas de orientación y control. Esto lo podemos ver a partir de las

entradas/salidas que lo conforman. El Rabbit 3000 tiene siete puertos paralelos de 8 bits designados por las letras A, B, C, D, E, F y G; cuyos pines comparten otras funciones del Rabbit. También cuenta con seis puertos seriales designados por las letras A, B, C, D, E y F. Éstos pueden operar en forma asíncrona a un baudaje alto. Aunque vienen conectados a la tarjeta de desarrollo de una manera particular, pueden ser configurados y cambiados por el usuario (Rabbit Semiconductor 2007).

### 2.3. Interfaz bidireccional inalámbrica.

El módulo de comunicación inalámbrica lo conforman dos radiomódems de características similares; uno conectado a la computadora de abordo (Tarjeta Lp 3500) y otro conectado a la PC donde se encuentra la interfaz de control que hace las veces de estación terrena. Los dos módulos fueron configurados para transmitir y recibir datos. En la programación del microcontrolador se incluyó una subrutina para mandar un eco y saber que el comando desde la PC ha llegado a la computadora de a bordo, de manera adecuada.

La conexión inalámbrica es indispensable en nuestra plataforma (figura 2.6), ya que en un medio con fricción despreciable, cualquier cable utilizado para enviar o recibir información causaría un desbalanceo inadmisibles, como se mencionó en el capítulo 1.

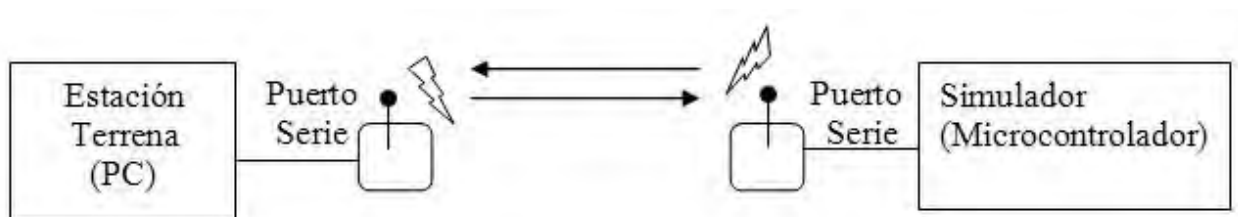


Figura 2.6.- La comunicación que se lleva a cabo entre la estación terrena y la computadora de abordo es de tipo serie, a través de modems inalámbricos.

#### 2.3.1. Protocolo de comunicación entre radiomódems.

La comunicación entre la plataforma de simulación y la estación terrena se hace a través del protocolo serial RS-232, el cual se identifica como una interfaz que designa una norma para el intercambio serie de datos binarios entre un DTE (Equipo terminal de datos) y un DCE (Equipo de Comunicación de datos); teniendo sólo la necesidad de configurar los módems basándonos en este protocolo y hacer las conexiones correspondientes en la PC y en el Rabbit. En la PC se utilizó el puerto **com1** mientras que en el microprocesador se usó el puerto serial E. En las tablas 2.1, 2.2 y 2.3 se presentan las características de comunicación:

Tabla 2.1 Configuración utilizada en los radiomódems

Bauds	9600
Bits de datos	8
Paridad	Ninguna
Bit de paro	1

El protocolo RS-232 utiliza los siguientes elementos para su conexión:

**Tabla 2.2. Asignación estándar de pines para la interfaz RS232.**

Common Ground	G	Tierra
Transmitted Data	TD	Transmisión
Received Data	RD	Recepción
Data Terminal Ready	DTR	Lista Terminal
Data Set Ready	DSR	Envío Listo
Request To Send	RTS	Respuesta
Clear To Send	CTS	Limpia
Carrier Detect	DCD	Acarreo
Ring Indicator	RI	Indicador

Los pines que se utilizan en nuestro sistema de simulación son las conexiones de Transmisión, Recepción y Tierra, los demás quedan disponibles para alguna otra operación del sistema.

**Tabla 2.3. Pines usados en el sistema de comunicación.**

Common Ground	G	Tierra
Transmitted Data	TD	Transmisión
Received Data	RD	Recepción

Es fundamental establecer una sincronización entre los módems ya que de otra manera obtendríamos datos incorrectos a pesar de tener bien configurados los programas de la PC y del Rabbit. Esta sincronización se lleva a cabo a través de el software Maxstream X-CTU (incluido con los radiomódems) y siguiendo la configuración expuesta [en la tabla 2.1.](#)

### **2.3.2. Principio de funcionamiento de los radiomódems.**

El principio de funcionamiento de los radiomódems se basa en un sistema de comunicación transmisor-receptor, mediante ondas electromagnéticas que se propagan por el espacio. Para evitar interferencia y ruido se utiliza una modulación en frecuencia que a la vez sirve para identificar los diferentes canales de transmisión/recepción. La comunicación a su vez puede ser half-duplex o full-duplex; la primera es cuando un radiomódem se configura para que sea exclusivamente transmisor y el otro funcione como receptor, por lo que también se le llama comunicación unidireccional. Por otro lado full-duplex se refiere a una comunicación bidireccional donde los radiomódems son emisor y receptor a la vez. Es importante señalar que mientras uno envía, el otro se dispone a recibir; y no se puede enviar y recibir información al mismo tiempo, sin embargo, debido a que estos sistemas son muy rápidos (hasta 38400 bauds para nuestros radiomódems) y con las condiciones de nuestro sistema, en términos prácticos se envían y reciben datos al mismo tiempo, aunque estrictamente hablando esto no sucede. Entonces, al tener una comunicación full-duplex ambos módems funcionan como transmisor y receptor ([figura 2.7](#)).

La modulación de los radiomódems utiliza el FHSS (Frequency Hopping Spread Spectrum) o Espectro Ensanchado por Salto de Frecuencia, que es básicamente una técnica de modulación en la que la señal se emite sobre una serie de radiofrecuencias aparentemente aleatorias, saltando de frecuencia en frecuencia junto con el transmisor, y que ofrece tres características principales:

1. Las señales en espectro ensanchado son altamente inmunes al ruido y a la interferencia.
2. Son difíciles de interceptar. Una transmisión de este tipo suena como un ruido de corta duración, o como un incremento en el ruido en cualquier receptor, excepto para el que esté usando la misma secuencia utilizada por el transmisor.
3. Las transmisiones en espectro ensanchado pueden compartir una banda de frecuencia con varios tipos de transmisiones convencionales con mínima interferencia.



*Figura 2.7.- Fotografía del radiomódem MaxStream 9 Xcite, usado en nuestro sistema de simulación.*

La configuración de nuestros radiomódems permiten tres tipos diferentes de topologías de red, peer to peer (nodo a nodo), point to point (punto a punto) y point to multipoint (punto a multipunto). Nosotros usamos la topología punto a punto ya que solamente tenemos dos radiomódems en el sistema, uno conectado a la estación terrena y otro en la caja Xenix ligado con la tarjeta Lp3500, en la plataforma.

Los radiomódems usados son del modelo Maxstream 9xcite, entre las características más importantes de utilidad para este proyecto se encuentran:

- \* 4 mW de potencia de salida.
- \* Configuración de 9600 a 38400 bauds.
- \*Puede trabajar con varias topologías de red; nosotros usamos la topología punto a punto.
- \* Ligeros.
- \*Alcance de hasta 30 metros entre tarjetas, con línea de vista.

### 2.3.3. Tarjeta de desarrollo y su configuración.

El microprocesador Rabbit 3000 viene acoplado a una tarjeta de desarrollo para formar el kit Lp3500. Ésta es indispensable para explotar todas las características del microprocesador Rabbit. Cuenta con terminales que permiten la sujeción de cables por medio de un tornillo lo que facilita la conexión entre las tarjetas que componen nuestro sistema. Además presenta salidas para los puertos seriales, salidas de PWM y de potencia. En la [figura 2.8](#) se muestra la tarjeta con sus características principales.

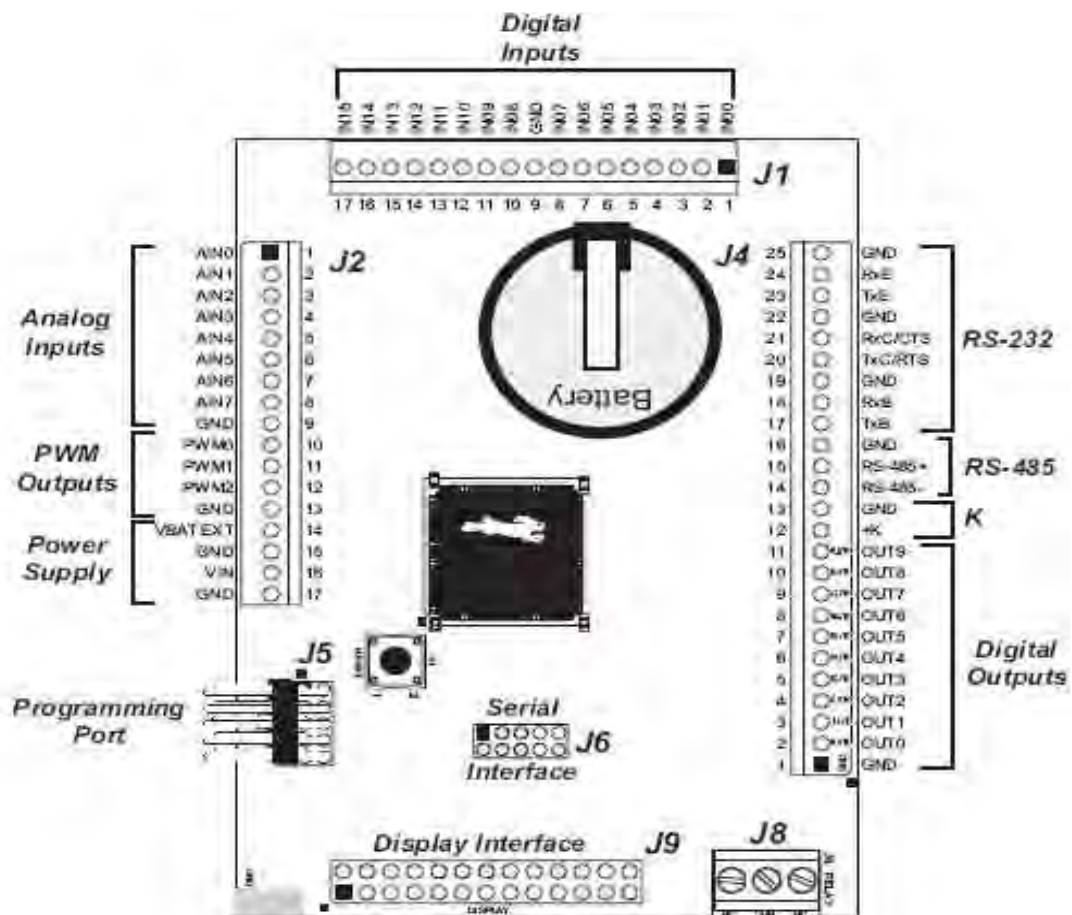


Figura 2.8 Tarjeta Lp 3500 con sus puertos de entrada y salida disponibles.

Los puertos usados se configuran de manera sencilla por programación (Rabbit Semiconductor 2007), algunas salidas cuentan con librerías especializadas, como el PWM y puertos seriales, las cuales son una ayuda para simplificar la programación.

El puerto de programación tiene un cable especial para ser conectado a la PC y quedó dentro de la caja "Xenix". Un punto importante a señalar es que al conectar el cable de programación a una PC, el Rabbit sólo ejecutará la compilación y depuración del programa que se tenga abierto en ese

momento. Es decir, al conectarle este cable el Rabbit dejará de funcionar con el programa previamente cargado y esperará una nueva compilación o depuración desde la PC.

En la figura 2.9 las salidas del Rabbit marcadas como OUT4, OUT5, OUT6, están conectadas a un divisor de voltaje en caso de que la alimentación del sistema fuera mayor y se tuviera necesidad de un voltaje predeterminado, sin embargo, la conexión real para estas salidas es una resistencia para disminuir la corriente hacia los otros circuitos, es decir, se puenteo una resistencia del divisor ya que se usó una alimentación de 9 V para la referencia +K (Veáse sección 2.2). En lo que respecta a las otras salidas, es importante ver el diagrama esquemático de la tarjeta LP3500, ya que en ese lugar se encuentran conectados leds indicadores, por lo que no fue necesario usar divisores, es por esto que en el diagrama general no aparecen para dichas salidas, sin embargo, el circuito del divisor que se presenta más adelante fue diseñado para acoplar todas las salidas sinking. A continuación se presenta el diagrama de la tarjeta Lp3500 y dichas salidas con sus leds.

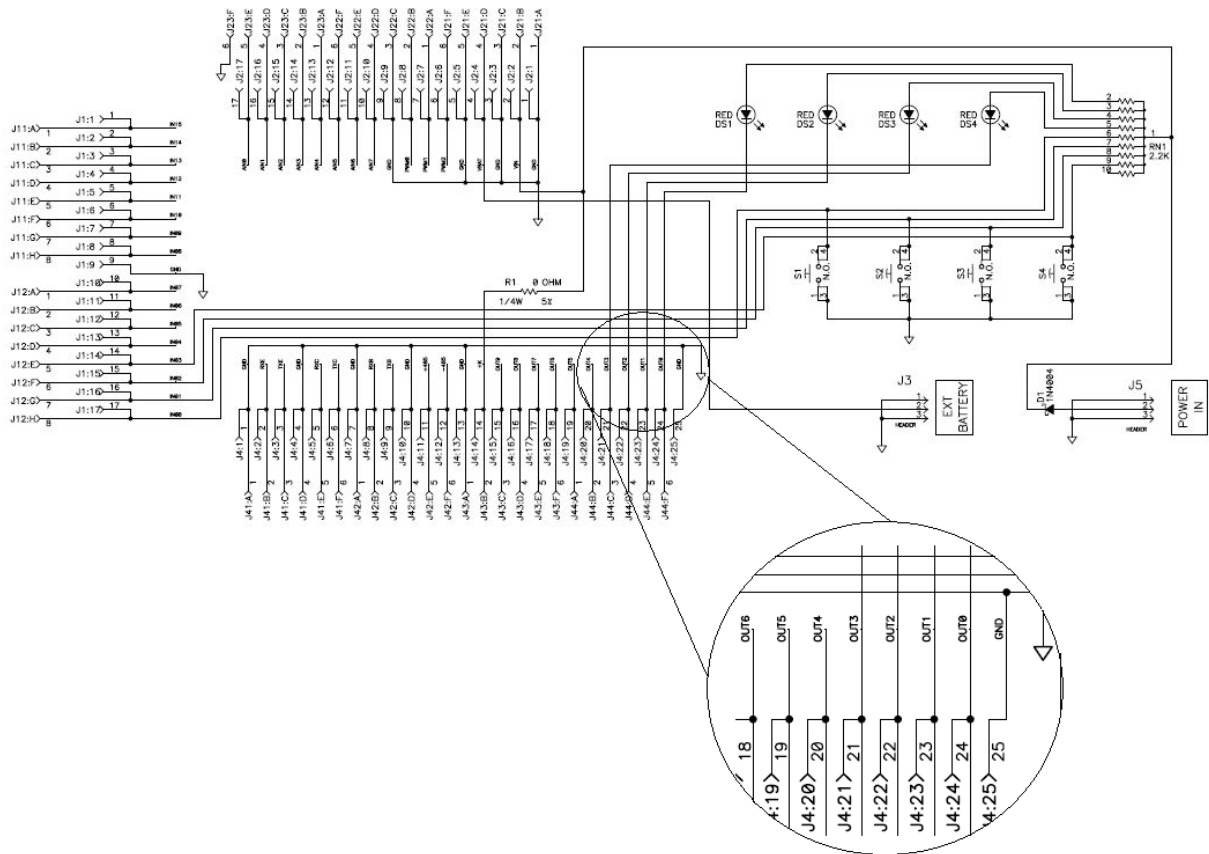


Figura 2.9.- Configuración de la tarjeta LP3500 para las salidas del Rabbit.

### 2.4 LIBRERÍAS Y MÓDULOS ADICIONALES.

Dentro de las ventajas principales de este microcontrolador, se encuentran las librerías; las cuales están enfocadas a una fácil interacción con los subsistemas con que cuenta la tarjeta de desarrollo, y de las cuales resaltan las siguientes características:



- Excepcional velocidad para operaciones de punto flotante, entre las que resaltan las funciones de aritmética y trascendentales.
- Fácil configuración para protocolos RS-232 y RS-485.
- Control de salidas y entradas tanto analógicas como digitales.
- Librería especializada en realizar programas multitareas atendiendo todas sin desperdiciar tiempo de procesamiento, que por las características de nuestro sistema las consideramos ejecutadas en tiempo real.

Gracias a estas librerías tenemos un desarrollo en la programación que nos condujo a construir subprogramas y verificar su comportamiento para una vez probados, acoplarlos a un programa general.

Entre las principales librerías usadas están la de PWM la cual permite la configuración de su ciclo de trabajo y su período. Las librerías de RS-232 que a partir de instrucciones macros se puede configurar los puertos a las velocidades estándar y que gracias a una ventana de salida del compilador Dynamic C, llamada stdio se puede visualizar los datos de recepción y transmisión serial.

Las librerías de configuración para salidas y entradas digitales nos permitió la accesibilidad al manejo de cualquier puerto de la tarjeta, sin tener que desactivar o activar otras funciones que estuviesen ligadas al mismo puerto.

Capítulo

3

SENSORES Y ACTUADORES.

En nuestro sistema de simulación para pruebas de algoritmos de control de orientación para satélites pequeños, utilizamos sensores y actuadores; los primeros sirven para determinar la orientación de la plataforma y los últimos, para cambiarla. Sensores de Sol, Tierra y un magnetómetro, constituyen el juego básico de dispositivos de orientación. Los actuadores están conformados por masas deslizantes, para efectuar un balanceo inicial del sistema bajo prueba; de bobinas magnéticas y ruedas inerciales, que permiten establecer una nueva orientación. El principio de funcionamiento de cada una de estos elementos, se describe en este capítulo, junto con sus circuitos de potencia. En la figura 3.1 es posible observar la conexión que existe entre ellos y la computadora de a bordo.

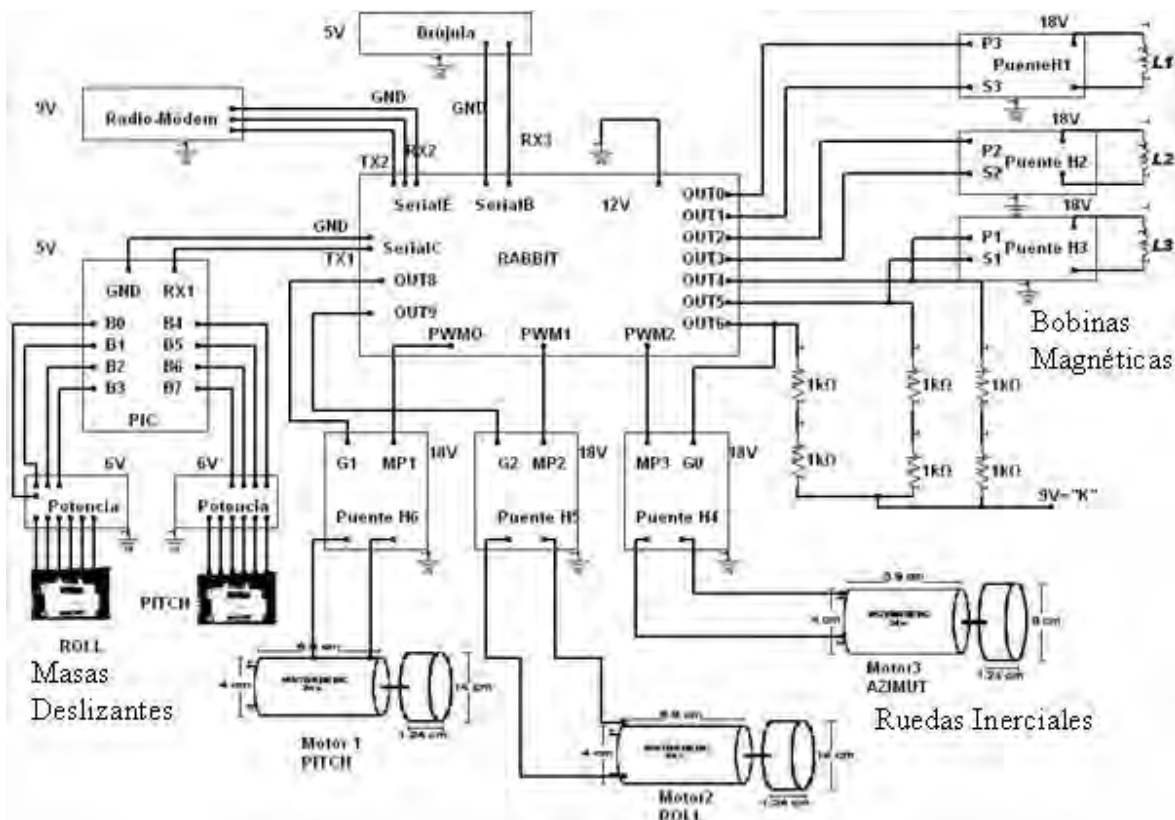


Figura 3.1.- Diagrama general donde se aprecian los sensores y actuadores y su conexión a la computadora de a bordo.

### 3.1 Sensores

El juego de sensores de orientación del sistema de simulación está constituido por un magnetómetro en tres ejes y los sensores de Sol y Tierra, pero en la práctica, es el magnetómetro el más utilizado por la ventaja que representa el brindar la orientación en tres ejes, a través de la brújula electrónica (Z) y un juego de inclinómetros (X,Y) integrados en el mismo dispositivo.

#### 3.1.1 Sensor de sol

Este es un dispositivo que provee información en un eje, usado para detectar la orientación de un satélite pequeño en órbita baja. Su diseño está basado en la utilización de un arreglo lineal de fotodiodos de 256 elementos. La salida analógica es acondicionada y enviada a la computadora de a bordo, para determinar la desviación, con respecto a los ejes de referencia que puede ser de rotación o de cabeceo, y que están fijos al cuerpo de la nave (figura 3.2).

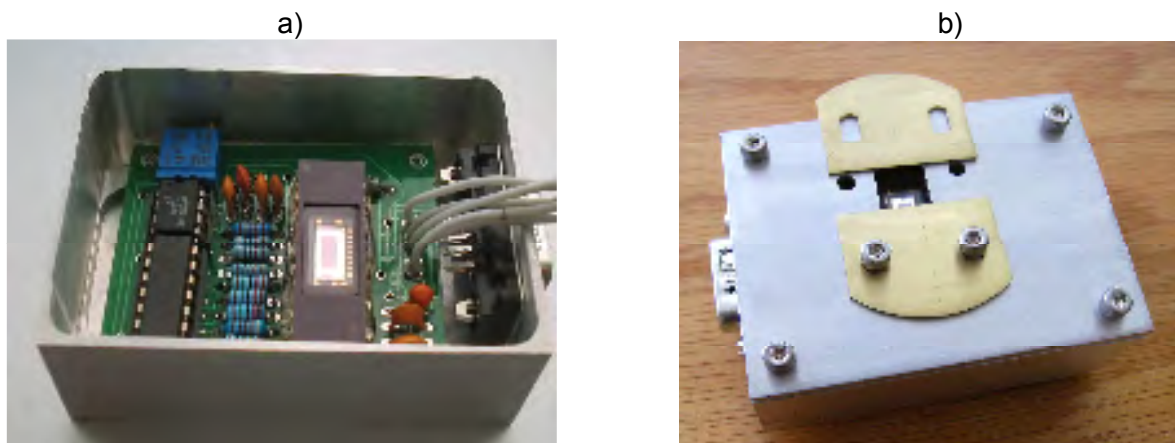


Figura 3.2.- Sensor de sol basado en un arreglo lineal de fotodiodos. a) Vista interior, b) Carcasa.

La precisión obtenida es de  $\pm 0.18^\circ$  y la cobertura o campo de vista es de  $\pm 22.5^\circ$ . Una lámpara de halógeno conectada a una fuente de voltaje regulada, hace las veces del Sol. Con ella se ha efectuado una calibración inicial utilizando una potencia de iluminación similar a la encontrada en órbita terrestre ( $1250 \text{ w/m}^2$ ), para determinar los niveles de ganancia adecuados en los amplificadores. Se ha colocado un filtro óptico que permite atenuar la intensidad de la fuente de iluminación para evitar problemas de saturación en los fotodiodos y la electrónica de acondicionamiento.

El arreglo lineal de fotodiodos utilizado (S3924-256Q) es fabricado por Hamamatsu™, cuenta con 256 elementos, requiere de una potencia 10 mW para operar y tiene una respuesta proporcional a la cantidad de elementos iluminados (Nápoles E. 2009). Este detector genera su señal de salida a través de la integración, en forma de campo eléctrico, en la capacitancia formada en las uniones de cada uno de los fotodiodos. También cuenta con un circuito propio de autobarrido (fabricado con transistores MOS), que lee la carga de manera secuencial con la ayuda de un conmutador direccional conectado a cada uno de los fotodiodos; todo integrado en una sola pastilla de silicón.

### 3.1.2 Sensor de Tierra

Hay básicamente dos configuraciones de los sensores de Tierra: los de barrido y los estáticos. Los primeros utilizan un sistema de rotación, en donde, estando perfectamente alineados los sensores, determinan los límites de la Tierra mediante pulsos eléctricos de igual duración, definidos como alto o bajo según la detección de la Tierra o el espacio frío (figura 3.3).

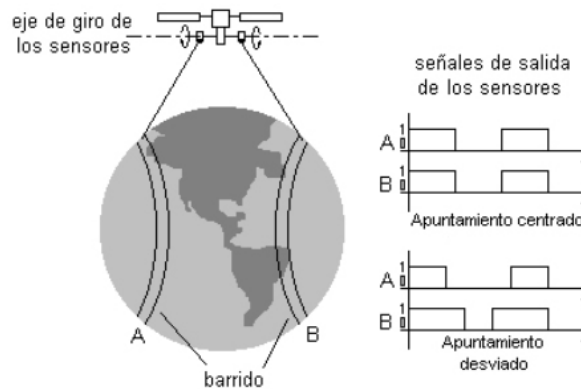


Figura 3.3. Sensores de Tierra rotatorios. La desviación en el apuntamiento del satélite provoca una variación en el ancho de los pulsos obtenidos.

La tendencia de los últimos años ha sido la implementación de sensores de Tierra estáticos, que son más confiables y duraderos. Su funcionamiento se rige por la proyección de la capa de CO<sub>2</sub> de la Atmósfera Terrestre, sobre detectores que trabajan en la banda IR (14 a 16 μm) a través de un sistema óptico de enfoque. En este proyecto, se utiliza un conjunto de cuatro sensores de Tierra estáticos para determinar la orientación en dos ejes independientes de la plataforma (figura 3.4). Los detectores empleados son de selenuro de plomo, con una respuesta espectral entre 1 y 5 μm. Sus señales analógicas son enviadas a un circuito de acondicionamiento y posteriormente a un filtro Sallen-Key para reducción de ruido y finalmente a la computadora de abordo para determinar la orientación. Su precisión es del orden de ± 0.5°.

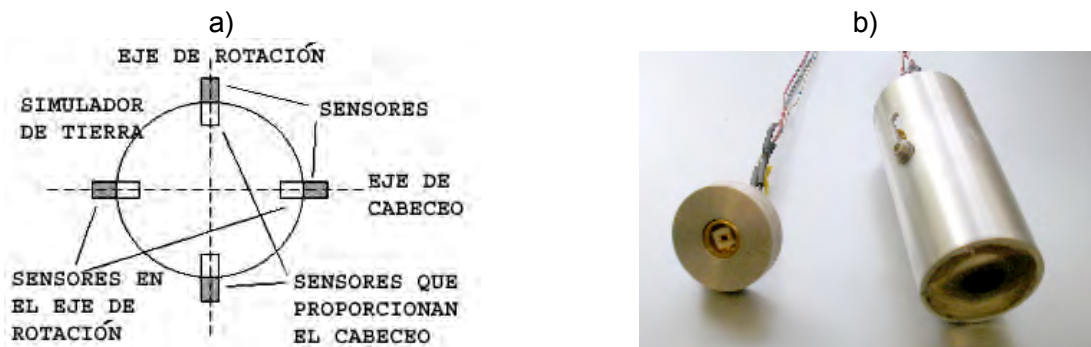


Figura 3.4 a) Arreglo de sensores de Tierra para determinar la orientación en dos ejes independientes y b) fotografía de uno de estos elementos.

### 3.1.3 Brújula Electrónica

Una brújula electrónica llamada EZ-COMPASS-3 ([Advanced Orientation Systems, 2004](#)), con dos inclinómetros incluidos (figura 3.5), es el sensor empleado para obtener los tres componentes vectoriales del campo magnético Terrestre. Este dispositivo entrega una señal de azimut de manera continua con una precisión  $\leq 0.5^\circ$ , tres componentes de campo magnético, la inclinación en los ejes de rotación y cabeceo con una precisión de  $0.25^\circ$  y datos sobre la temperatura, hacia un microcontrolador interno el cual genera una cadena de caracteres con la información de todos sus sensores. Esta cadena llega a un puerto serial lista para ser transmitida y se conecta al puerto B serial del Rabbit. Los inclinómetros también proporcionan las señales de control para llevar a cabo el balanceo automático de la plataforma en el plano horizontal, como se explicará más adelante.

Los ejes X,Y de nuestra plataforma se localizan sobre el plano de la misma, mientras que el eje Z es perpendicular a dicho plano, con dirección positiva hacia arriba. La brújula proporciona de manera continua la inclinación de los ejes de alabeo (X) y cabeceo (Y), y la desviación del eje de guiñada (Z).

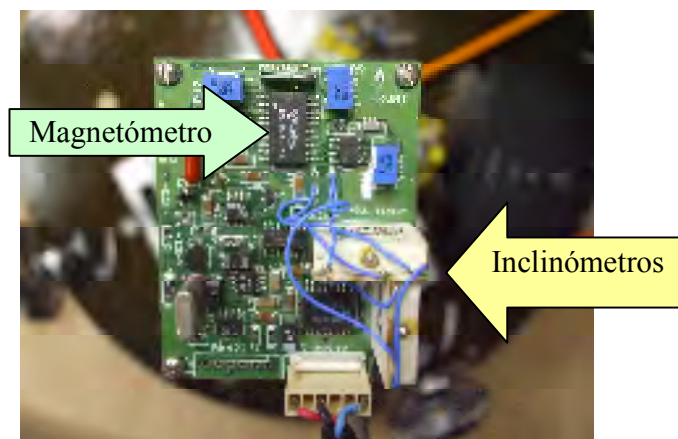


Figura 3.5. Brújula electrónica Ez-compass-3 que incluye un magnetómetro en tres ejes y dos inclinómetros.

### 3.1.4 Circuito auxiliar de regulación y divisor de voltaje.

Para polarizar al microcontrolador Lp3500, la brújula electrónica EZCompass-3 y al radiomódem, así como para lograr el acoplamiento de las salidas del microcontrolador, hacia los circuitos de potencia, se desarrolló una tarjeta especial que pudiera cubrir estas necesidades. En la [figura 3.6](#) se muestra su diagrama electrónico y en [la 3.7](#) se aprecian físicamente los elementos que componen la tarjeta impresa. Como se puede apreciar en el diagrama de [la figura 3.6](#), el divisor de voltaje es utilizado para acoplar las salidas de tipo “sinking”, que son a su vez entradas de los circuitos de potencia. Ya que esta configuración trabaja cerrando el circuito a tierra, la alimentación se toma de una fuente externa y las salidas del Rabbit se consideran tierras físicas. También es importante señalar que la polarización de las tarjetas mencionadas se hizo en escalera; es decir la entrada general llega al regulador de 12 V usado para el Rabbit, esa misma salida se usa para la entrada del regulador de 9 V que usa el radiomódem, y finalmente alimenta a la fuente regulada de 5 V que es usada para la brújula electrónica.

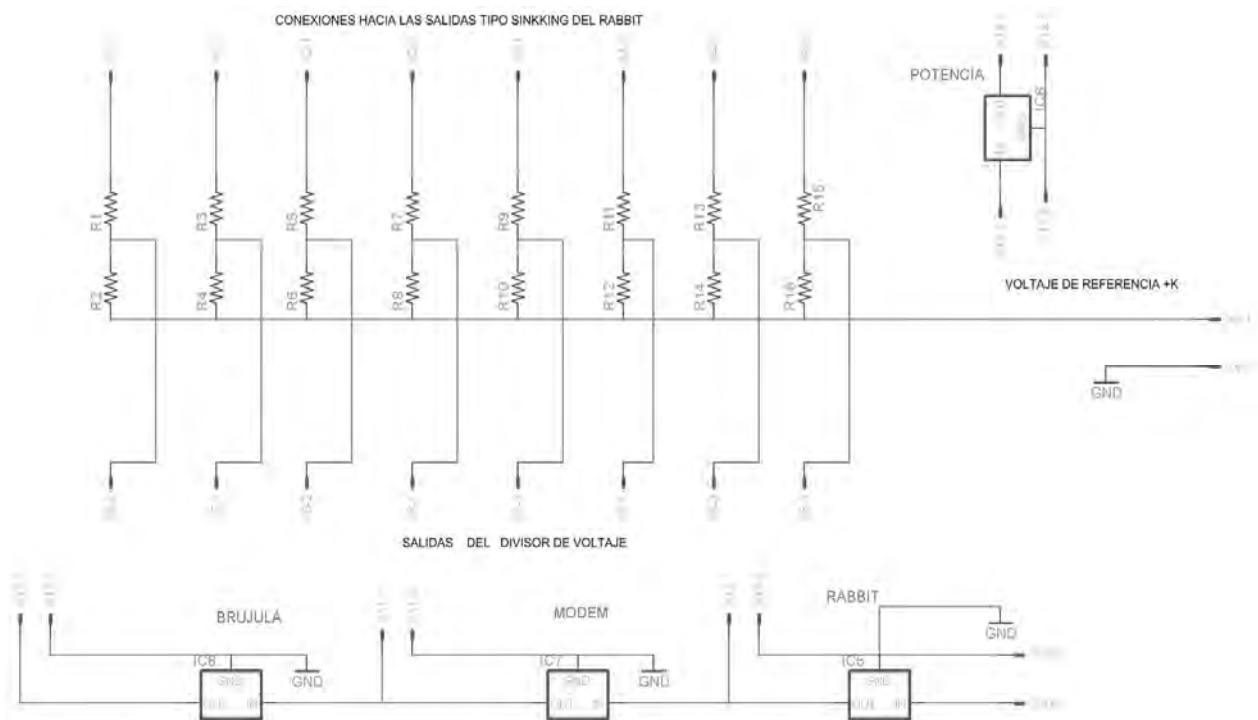


Figura 3.6.- Diagrama eléctrico de la tarjeta de reguladores y divisores de voltaje.

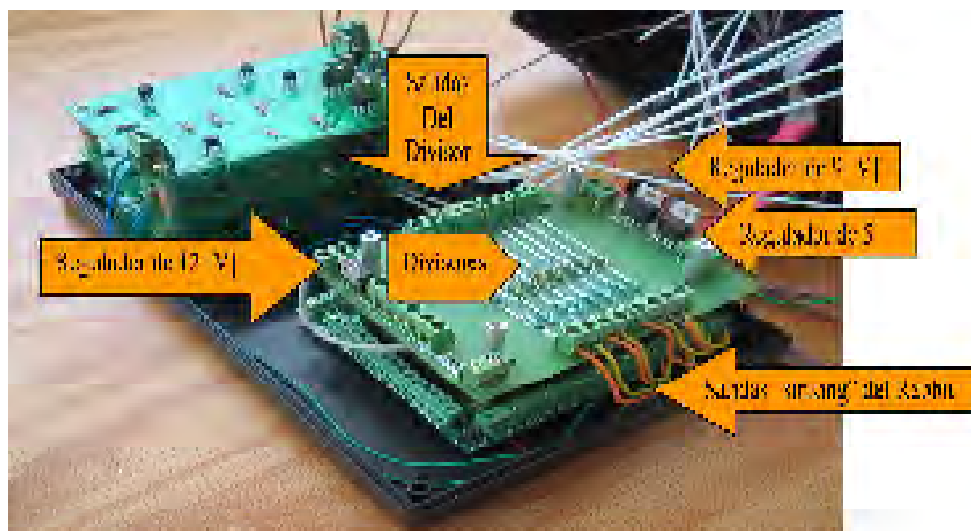


Figura 3.7. Reguladores para alimentar el microcontrolador, la brújula y el radiomódem. La tarjeta también contiene divisores de voltaje para las salidas tipo sinking del Rabbit.

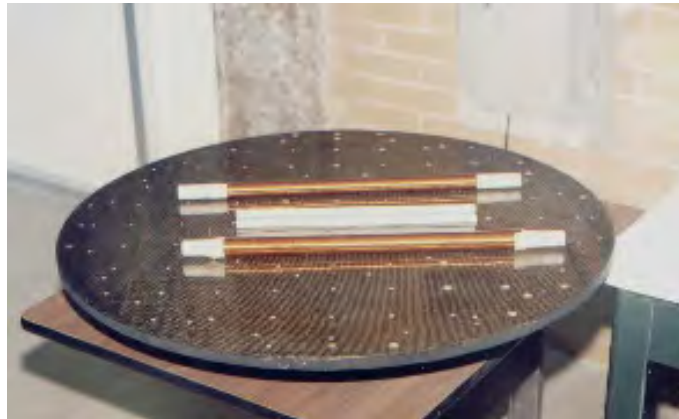
### 3.2. Actuadores

Los actuadores de este sistema son un conjunto de tres bobinas magnéticas y tres ruedas inerciales, éstas últimas impulsadas por motores de corriente directa. Mientras que la orientación de la plataforma es re-direccionada por las ruedas inerciales a través del intercambio de momentum, entre la propia plataforma y el cojinete neumático del simulador; las bobinas magnéticas reciben un par externo al interactuar con el campo magnético terrestre. Esta condición las hace indispensables en el espacio, ya que permiten imprimir un par neto al satélite, cosa que no ocurre con las ruedas inerciales que solo intercambian momentum entre ellas y el cuerpo del satélite.

Otros actuadores en la plataforma son las masas deslizantes que se utilizan para balancear la plataforma de manera automática. Este proceso se lleva a cabo antes de efectuar los experimentos de orientación y control.

#### 3.2.1. Bobinas magnéticas

El principio de funcionamiento de las bobinas magnéticas (figura 3.8) es que al circular corriente en su embobinado se generan momentos dipolares que interactúan con el campo magnético de la Tierra para producir esfuerzos de rotación en la nave.



*Figura 3.8. Imagen de las bobinas magnéticas donde se muestra su tamaño en relación con la plataforma.*

Otra de las grandes ventajas que representan las bobinas magnéticas es en el momento en el que el satélite se desprende del cohete lanzador. En caso de que la nave se encuentre tambaleándose, ésta puede ser estabilizada por medio de la ley de control “menos la derivada de B”:

El momento dipolar  $M_x$  que produce cada bobina magnética está determinado de acuerdo con la siguiente ecuación:

$$M_x = -\frac{dB_x}{dt} \quad (2)$$

donde  $\frac{dB_x}{dt}$  es la derivada con respecto al tiempo de la componente del campo geomagnético a lo largo de la bobina. Como resultado de esta ley, las bobinas actúan como freno magnético y se alinean con las componentes de campo de la Tierra. Se han desarrollado varias leyes de control para manejar el momentum con la finalidad de llevar a cabo el control de orientación. En todos los casos, el sistema de control de orientación determina un vector de error de momentum  $\vec{H}$  y entonces el vector de momento dipolar deseado  $\vec{M}$ , es calculado a través del producto (cruz) entre los vectores  $\vec{H} \times \vec{B}$ . El par resultante  $\vec{M} \times \vec{B}$  tiende entonces a reducir el vector de error  $\vec{H}$ .

Los sistemas de control con bobinas magnéticas, pueden ser utilizados de manera efectiva para llevar a cabo maniobras de apuntamiento y de control de estabilización, en todas las órbitas; incluyendo las geosíncronas (36,000 km). Los sistemas de control magnético son relativamente sencillos, no requieren de partes móviles, de sensores sofisticados, ni de consumibles abordo de la nave. Esto hace que los pares magnéticos sean atractivos para aplicaciones espaciales, sin embargo, estos proporcionan pares poco significativos, lo que restringe la cantidad y la rapidez de las maniobras, ya que su operación depende del valor de las componentes vectoriales del campo magnético en el lugar de la órbita donde se realiza la maniobra y de las dimensiones de las bobinas y de la energía disponible a bordo (Juárez, G. 1999).

### 3.2.1.1 Geometría

Hay dos tipos de bobinas que generalmente son apropiadas para estas aplicaciones, una de ellas es de forma tubular o de cilindro alargado y la otra es siguiendo el contorno perimetral del satélite. Las primeras ofrecen ventajas en cuanto a sus dimensiones reducidas; ya que con un material adecuado en el núcleo, es posible obtener momentos dipolares relativamente grandes. Son fáciles de colocar aunque presentan histéresis y no linealidades. Las bobinas que siguen el contorno del satélite son otra opción viable, ya que aún con las reducidas dimensiones de éste, se puede contar con mayor área en el núcleo y por tanto, con un mayor dipolo, en este trabajo por fines prácticos se decidió usar las de forma tubular.

### 3.2.1.2 Cálculo de dipolos magnéticos.

El dipolo magnético que proporcionan las bobinas está dado por la siguiente expresión:

$$\mathbf{d} = \mu^* (NI) \mathbf{A} \cdot \mathbf{n} \quad (3)$$

donde:

$\mu$ , es la permeabilidad del núcleo.

$N$ , es el número de vueltas.

$I$ , es la corriente a través de la bobina.

$A$ , es el área transversal de la bobina.

$n$ , es un vector unitario normal al plano del enrollamiento.



Lo que obtenemos es un dipolo que tenderá a alinearse con el campo magnético Terrestre produciendo un par externo al satélite. En la figura 3.9 se pueden ver el tipo de líneas de campo generadas en una bobina en forma tubular

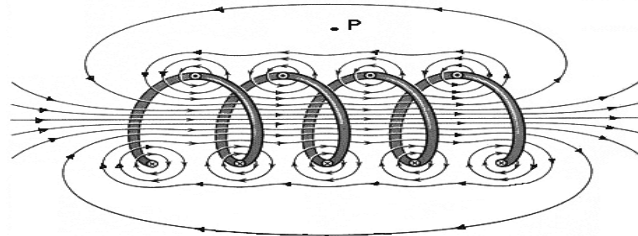


Figura 3.9.- Líneas de campo magnético generadas por una bobina en forma tubular.

Con el fin de aumentar el par disponible generado por las bobinas magnéticas, para este trabajo de tesis se diseñaron y fabricaron nuevas bobinas que utilizan un núcleo de ferrita, cuya permeabilidad es de  $\mu=1000$ . Asimismo se aumentó sustancialmente el número de vueltas, lo que redundó en un par mayor, con relación a las bobinas anteriores que tenían núcleo de aire. Los datos se observan en la tabla 3.1, de esta manera se pasó de  $0.257 \text{ A}\cdot\text{m}^2$  que se tenían en la plataforma anterior, a  $225.55 \text{ A}\cdot\text{m}^2$  lo que implicó un cambio drástico en el par disponible. Esto se hizo para poder visualizar los movimientos de la plataforma, ya que un par demasiado pequeño como el que se tenía, no lo permite.

El diámetro del núcleo es de 1.4 cm, el calibre del cable es 28 AWG y el voltaje aplicado de 18 volts (ver figura 3.10).

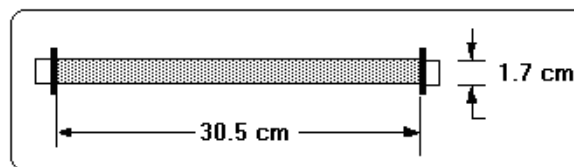


Figura 3.10. Dibujo de una bobina con sus dimensiones actuales. El diámetro del núcleo de ferrita es de 1.4 cm, esto es ligeramente menor que el diámetro del tubo que sirve de sustento al arrollamiento de cobre.

Tabla 3.1. Cálculo del dipolo generado en las bobinas utilizando núcleo de ferrita ( $\mu=1000$ )

$\phi$ bobina	Cable AWG #	vueltas	R en $\Omega$	V en volts	I en amp.	Potencia en w	Masa del núcleo gr	Masa de la bobina gr	Masa total gr	Dipolo en $\text{A}\cdot\text{m}^2$
1.4 cm	28	2600	36.3	18	0.4958	8.923	227.6	180.0	407.6	198.43
1.4 cm	28	6000	73.7	18	0.2442	4.394	227.6	432.4	660.0	225.55

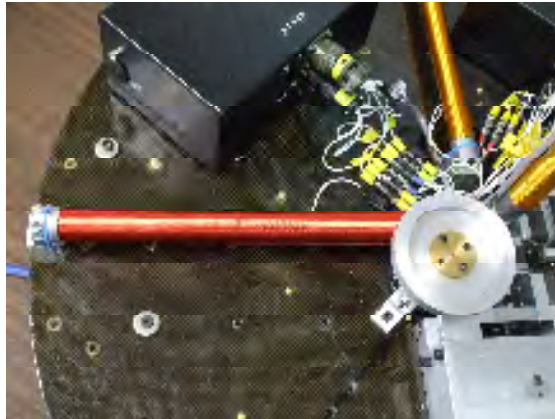


Figura 3.11.- Fotografía de una bobina con núcleo de ferrita y un arrollamiento de 2,600 vueltas, localizada en el eje de rotación ( $\bar{X}$ ) de la plataforma.

### 3.2.1.3. Ubicación de las Bobinas en la plataforma

Al utilizar bobinas de forma tubular la mejor forma de colocarlas es de manera axial o paralela a cada eje de control; con este acomodo tienen una posición perpendicular entre si, y es posible manipular la plataforma en cada uno de sus tres grados de libertad. La disposición final de las bobinas en la plataforma se muestra en la figura 3.12.

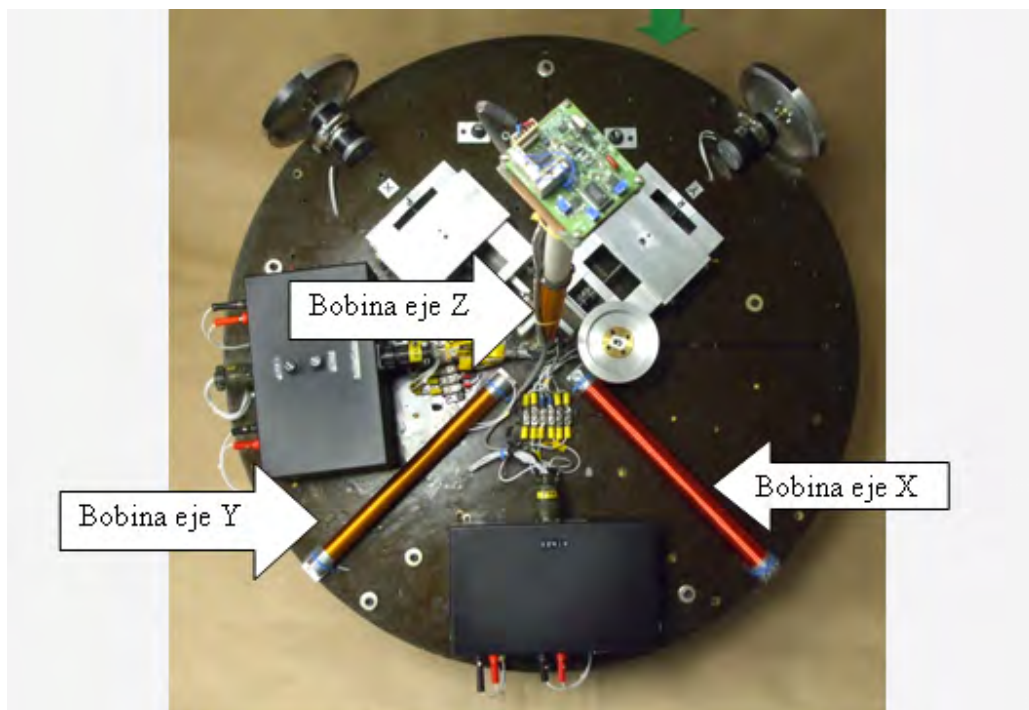


Figura 3.12.- Imagen donde se muestra la colocación de las bobinas en la plataforma de simulación de un medio sin fricción.

### 3.2.2. Ruedas Inerciales

La utilización de ruedas inerciales como actuadores en un satélite pequeño, posibilita un apuntamiento muy fino y permite disponer de pares correctivos importantes. También son útiles cuando es necesario llevar a cabo maniobras de re-orientación sobre algún lugar específico. Son dispositivos indispensables cuando es necesario contar con una capacidad de apuntamiento del orden de una décima de grado o menos, como es el caso de los satélites de percepción remota (Prado J. y Miranda V.M. 1998). Durante la operación normal de las ruedas inerciales, éstas van incrementando la cantidad de momentum angular, por lo que es necesario un sistema que disipe esta energía acumulada. Un conjunto de bobinas magnéticas puede efectuar el trabajo de desaturación, además de que en sí mismo, constituye un sistema de control de orientación de respaldo (Prado J. et. al. 1998). Un motor eléctrico ligado a la estructura de la nave, hace rotar a una rueda que gira libremente, cuando un par perturbador actúa en la nave a lo largo de uno de sus ejes, la rueda reacciona, absorbiendo el par y manteniendo la orientación.

Existen dos modos básicos de funcionamiento de las ruedas inerciales: en modo de reacción y con momentum diferente de cero. En el primer caso tenemos una rueda con velocidad inicial cero y que reacciona girando en un sentido o en otro para compensar una desviación en su eje. En el segundo caso, tenemos una rueda girando a una determinada velocidad; misma que se incrementa o disminuye atendiendo a una perturbación externa.

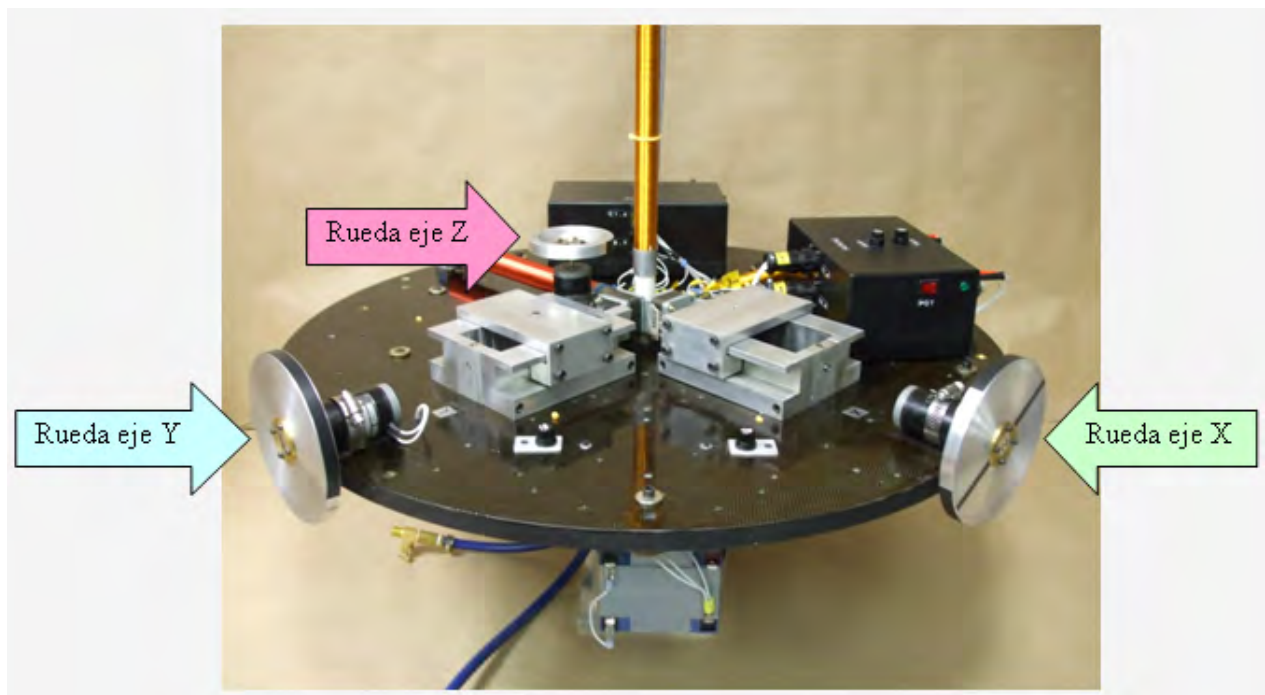


Figura 3.13 Imagen donde se aprecia la posición de las ruedas inerciales en la plataforma.

En este proyecto desarrollamos los dos modos de funcionamiento. Las ruedas fueron re-diseñadas aumentando su tamaño y su forma para que proporcionen un mayor par de control y se han dispuesto en los tres ejes de la plataforma. Las ruedas son discos delgados con diámetro de 14cm,

espesor de 2 mm, y con un anillo exterior con 1.2 cm de ancho, esto para los ejes X,Y y para el eje Z, la rueda tiene un diámetro de 8.8 cm con un espesor de 3.1 mm que también cuenta con un anillo exterior que tiene 1.22 cm de ancho. El disco en el eje Z es de menores dimensiones, debido a que no está expuesto a pares restauradores y la fuerza necesaria para re-orientar la plataforma es menor que en los otros ejes. En la figura 3.13, se aprecia la forma y la diferencia de tamaño de las ruedas inerciales.

### 3.2.2.1. Sistemas de control de ruedas de reacción

El control del giro de las ruedas se hace a través del PWM (pulse width modulation), o modulación por ancho de pulso, los sensores de orientación envían señales de retroalimentación hacia los motores de cada rueda de reacción, para corregir su posición. Si la plataforma se encuentra en una posición muy alejada de la orientación deseada, el ciclo de trabajo es grande y conforme se va acercando a dicha orientación, el ciclo de trabajo disminuye; en otras palabras, mientras se esté lejos de la posición deseada el motor trabajará mayor tiempo y cuando se acerque trabajará un tiempo cada vez menor; esto nos da un mejor control y además ahorra energía ya que los ciclos de trabajo permiten a las baterías recombinar sus elementos y no descargarse tan rápidamente.

### 3.2.2.2. Salidas de potencia, Puente H

Para acoplar las salidas lógicas de la computadora de abordaje con los motores de las ruedas inerciales (y también las bobinas magnéticas), usamos como interfaz un circuito de potencia conocido como puente "H". (ver figura 3.14). Recibe este nombre ya que el circuito eléctrico tiene el perfil de la letra H y su función es tener dos formas de polarización acopladas a las salidas de potencia a partir de pulsos lógicos, de esta manera puede cambiar el sentido de la corriente y conmutarlo; además de proteger al microcontrolador evitando corrientes inversas hacia el. Este tipo de circuitos son necesarios ya que el microcontrolador aunque maneja salidas para soportar una corriente de hasta 200 miliamperes; no tiene protección contra flujos mayores de corriente y podría dañarse.

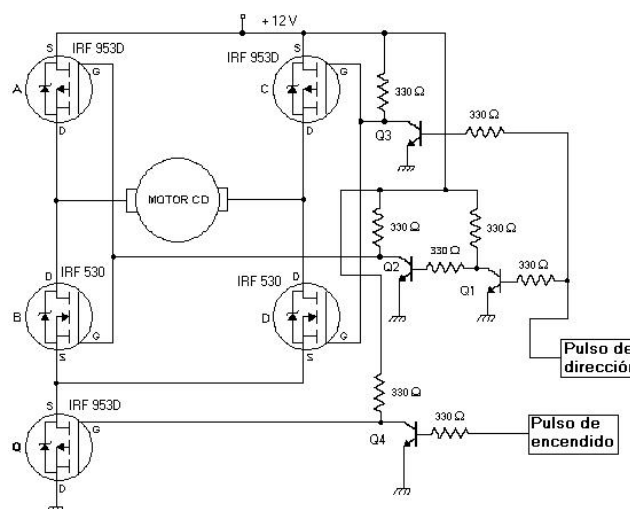


Figura 3.14.- Diagrama electrónico del puente "H", implementado con mosfets.

Un puente H es básicamente un arreglo de cuatro interruptores distribuidos como se muestra en la figura 3.14. Estos interruptores (A, B, C y D) pueden ser transistores bipolares, mosfets, jfets, relevadores o cualquier combinación de elementos. El puente H sirve para que un motor de corriente directa funcione en ambos sentidos, sin tener que manejar voltajes negativos. Este es el caso de las ruedas de reacción que necesitan girar de esta manera. Si se cierran solamente los contactos A y D, la corriente circulará en un sentido a través del motor, y si se cierran solamente los contactos B y C, la corriente circulará en sentido contrario. Resulta evidente que los contactos A y B ó C y D no deben cerrarse al mismo tiempo para evitar un corto circuito.

Los transistores Q1, Q2, Q3 y Q4 son transistores NPN pequeños (2N3904) que se usan como convertidores de nivel. Estos transistores están conectados como inversores y en este caso convierten los niveles lógicos TTL a +12 y 0.0 V para controlar los FETs. En la figura 3.15 se muestra la tarjeta impresa que alberga a todos estos componentes.

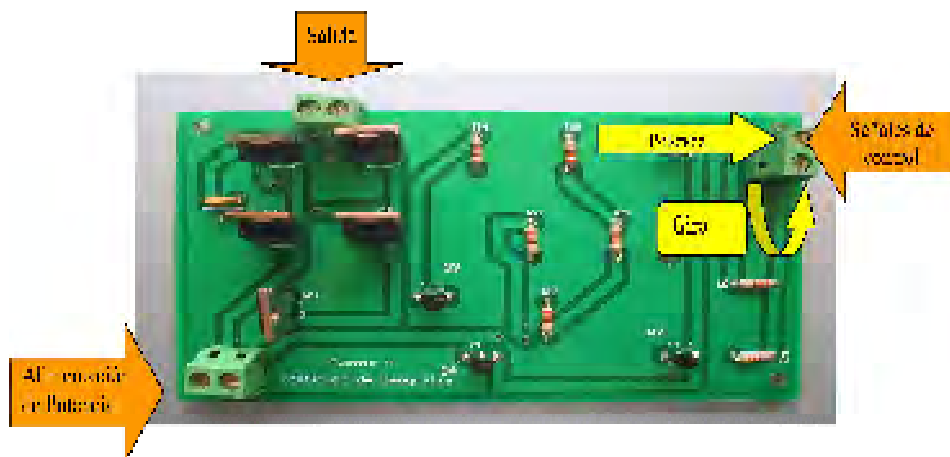


Figura 3.15.- Tarjeta impresa del puente "H".

### 3.2.3. Masas deslizantes.

El simulador permite llevar a cabo una amplia gama de pruebas de detección de orientación y control de estabilización para satélites pequeños. Sin embargo, para poder efectuar pruebas confiables es necesario que este se encuentre balanceado, lo que ayuda a minimizar los efectos causados por los pares gravitacionales. Un conjunto de masas deslizantes es utilizado para efectuar un balanceo en el simulador. Para mantener el control de la masa total del sistema y sus propiedades inerciales, las masas utilizadas deben ser móviles y totalmente controlables (Prado J. (2007)). Dichas masas pueden cumplir la doble función de servir como lastre y para llevar a cabo el ajuste fino de localización del centro de masa de la plataforma de simulación.

La superficie de nuestro simulador conforma un plano de referencia XY, donde en cada uno de estos ejes hay un actuador, el cual está conformado por una masa sobre una platina deslizante que se mueve sobre un tornillo de cuerda milimétrica, el que a su vez está sujeto a un motor de pasos. Cuando éste recibe una serie de pulsos eléctricos, hace que la masa se deslice en un movimiento fino. (figura 3.16).

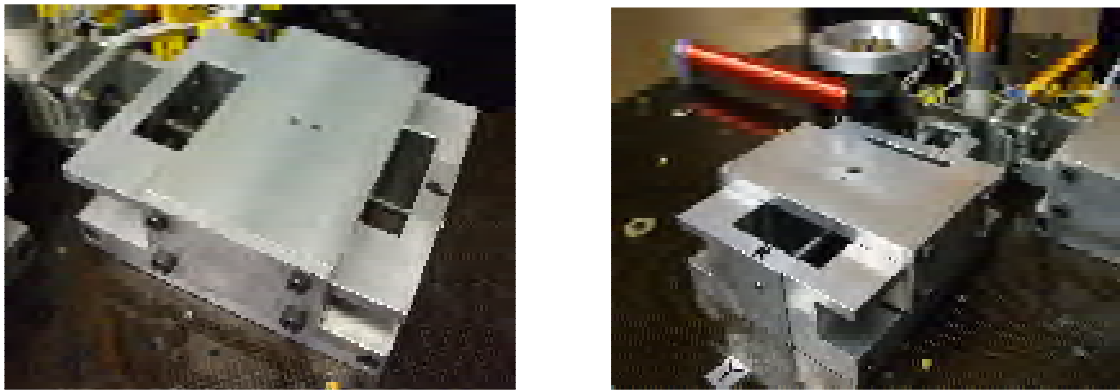


Figura 3.16. Actuator o masa deslizante que sirve para re-localizar el centro de masa de la plataforma.

Al usar las masas deslizantes para balancear nuestra plataforma estamos aprovechando las siguientes características:

- ❖ Desplazan una masa siguiendo un movimiento rectilíneo.
- ❖ Para su construcción se emplearon en la medida de lo posible materiales no magnéticos.
- ❖ La longitud de desplazamiento es de hasta cinco centímetros.
- ❖ Tienen la capacidad de mover una masa de 700 a 1000 gramos.
- ❖ Su centro de masa está localizado prácticamente en su centro geométrico.

### 3.2.3.1. Diseño y operación de las masas deslizantes.

Los sistemas de masas deslizantes son los componentes más importantes en el balanceo de la plataforma, ya que de ellos depende que se logre un par residual de pequeñas proporciones (Juárez A. 2001). La platina tiene un peso de 0.615 kg y cuatro pulsos de control (un paso) mueven su masa a lo largo del tornillo de cuerda milimétrica M7X1, una distancia total de 0.005 mm. El máximo desplazamiento en la platina es de 5 cm; esto permite hacer ajustes grandes, aún con un balanceo manual inadecuado. En la figura 3.17 se muestran los planos de un conjunto completo de una mesa deslizante con su platina, donde son mostrados los centros de masa.

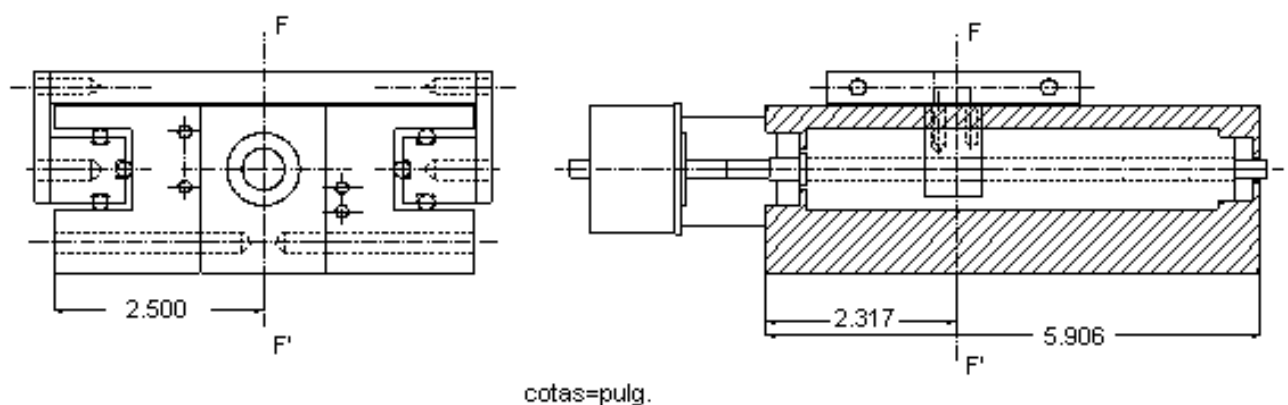
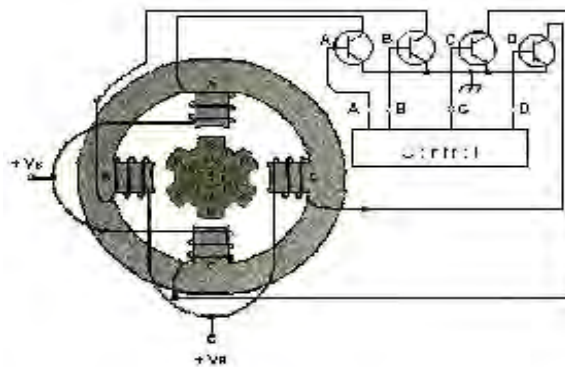


Figura 3.17.- Plano mecánico de una mesa deslizante.

### 3.2.3.2. Motor a pasos.

Una de las ventajas principales al tener las masas deslizantes acopladas a motores de pasos, es el hecho de que podemos mover la platina a posiciones exactas y en desplazamientos muy pequeños. Esto permite ajustar el centro de masa de manera muy precisa, logrando un par residual final del orden de 20 gr-cm lo que para una plataforma de 35 kg significa tener desplazado el centro de masa una distancia de 0.000057141 mm, con respecto al centro geométrico del balero de aire.

Básicamente el control de un motor a pasos consiste en una conmutación electrónica que es una secuencia lógica de energización en sus bobinas. Definiendo de una manera sencilla al motor de pasos podemos decir que es un dispositivo que convierte pulsos eléctricos en movimientos discretos de rotación mecánica (por pasos). Al aplicar a sus bobinas una serie adecuada de impulsos eléctricos, el rotor gira sobre su eje un ángulo fijo. Este ángulo recorrido, que depende de las características del motor, se le llama paso, del que podemos controlar la cantidad, la velocidad y el sentido de giro.



*Figura 3.18.- Diagrama que muestra las conexiones internas de un motor de pasos y su configuración más básica de control.*

Cuando el circuito de control enciende un transistor (A, B, C o D en particular), hay un flujo de corriente de alimentación de CD ( $+V_s$ ) que pasa por ese devanado a través del transistor a tierra. Cuando un solo devanado es energizado, este se encuentra enrollado de tal manera que su polo se vuelve norte magnético. Su flujo emerge de la cara del polo, pasa a través del rotor, entonces completa su trayectoria entrando en la cara del polo directamente opuesta a él.

Existen dos tipos de motores a pasos: los bipolares y los unipolares; éstos últimos son los usados en el sistema aquí descrito, los cuales se componen de dos bobinas cada una separada en dos y reciben el nombre de unipolares porque la corriente que circula en sus embobinados siempre circula en el mismo sentido; además, otra característica muy particular es que presentan seis cables, dos para cada una de sus bobinas y dos para la alimentación de cada una de ellas. Fig(3.18).

**3.2.3.3. Control del motor a pasos**

Se tiene una tabla que nos guía para poder mover los vástagos de los motores, en ella, se observa una secuencia lógica de pasos donde cuatro pasos de cuatro bits son una secuencia, la cual se repite hasta completar un giro completo que corresponde a 200 pasos (Contreras F. 2004).

Tabla 3.2.- Secuencia lógica para el control de los motores de pasos.

Pasos	Bobina A	Bobina B	Bobina C	Bobina D	Imagen
1	1	1	0	0	
2	0	1	1	0	
3	0	0	1	1	
4	1	0	0	1	

**3.2.3.3.1. Utilización de un PIC para el control de los motores a pasos.**

Para efectuar el movimiento controlado de los motores a pasos de las masas deslizantes, se utilizó un PIC (Peripheral Interface Controller) o Controlador de Interfaz Periférico, este es un dispositivo de la familia de microcontroladores tipo RISC, y que fue programado usando un Compilador de C. El modelo utilizado es el 18F452 y permite manejar adecuadamente las salidas que corresponden a cada uno de los motores de las masas. La necesidad de usar este microcontrolador fue para tener una disposición de salidas auxiliares, puesto que la tarjeta LP3500 tiene un número limitado de puertos digitales, los cuales son usados para el control de otros actuadores. El PIC funciona haciéndole llegar un caracter en su puerto serial, desde el microcontrolador Rabbit, que a su vez inicia una secuencia o pasos para hacer girar el motor, de esta manera el PIC sólo tiene programada una rutina para recibir determinados caracteres y ejecutar una secuencia por cada caracter recibido. La tabla 3.3 ilustra la relación que existe entre dicho caracter y lo que se ejecuta. La convención de signos está determinada en sentido positivo para el movimiento que realizan las masas hacia el centro de la plataforma y en sentido negativo para el movimiento hacia fuera de la misma.



Tabla 3.3. Relación de caracteres recibidos y ejecución de acciones en el PIC.

CARACTER RECIBIDO	FUNCIÓN A REALIZAR
a	Mueve el eje Y (roll) en sentido -
b	Mueve el eje Y (roll) en sentido +
c	Mueve el eje X (pitch) en sentido +
d	Mueve el eje X (pitch) en sentido -

En la figura 3.19 se muestra el diagrama electrónico del PIC y en la figura 3.20 la tarjeta impresa donde se aprecian sus componentes.

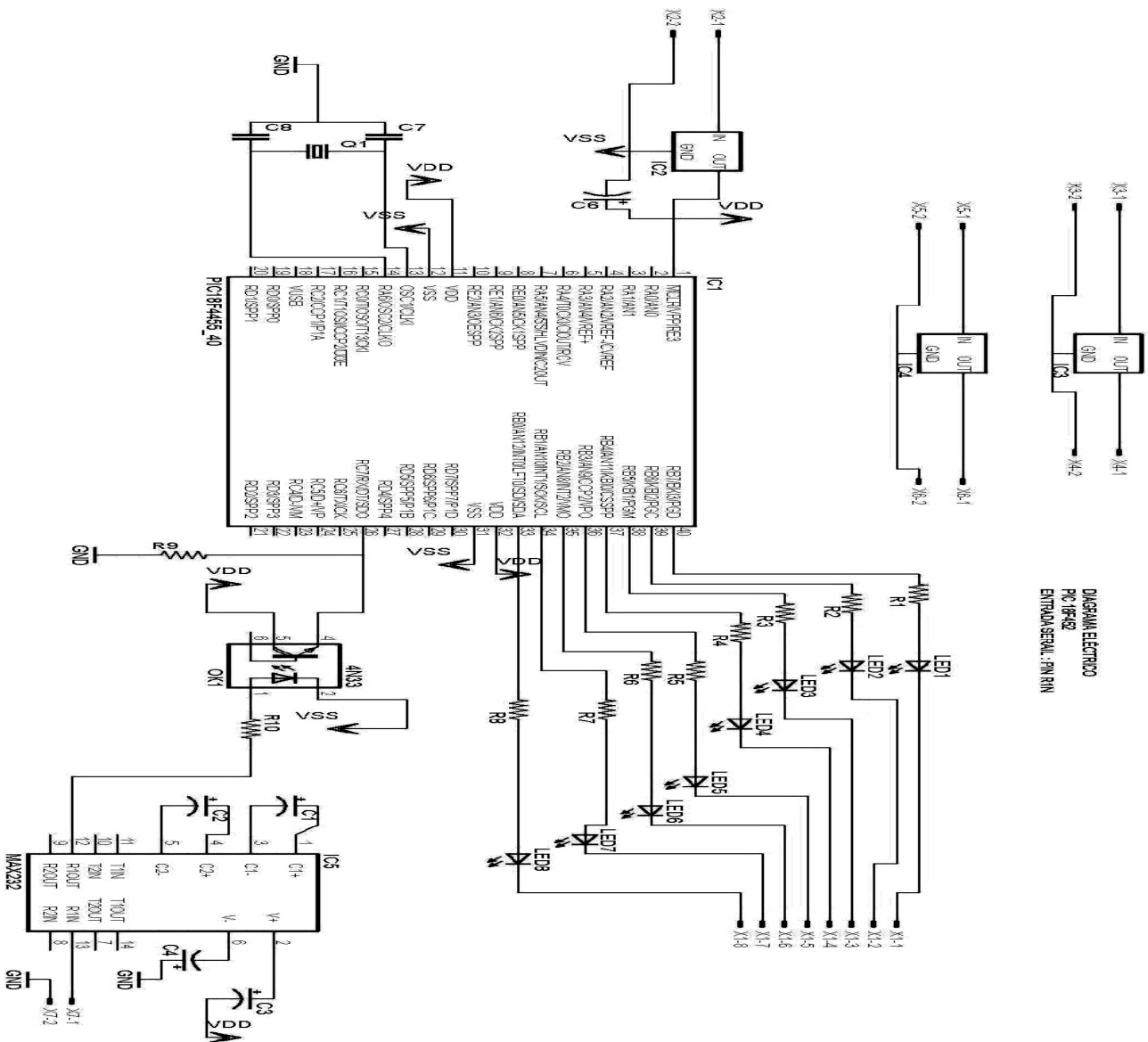


Figura 3.19.- Diagrama electrónico del PIC y su conexión con el microcontrolador y la interfaz de potencia para manejar los motores a pasos de las masas deslizantes.

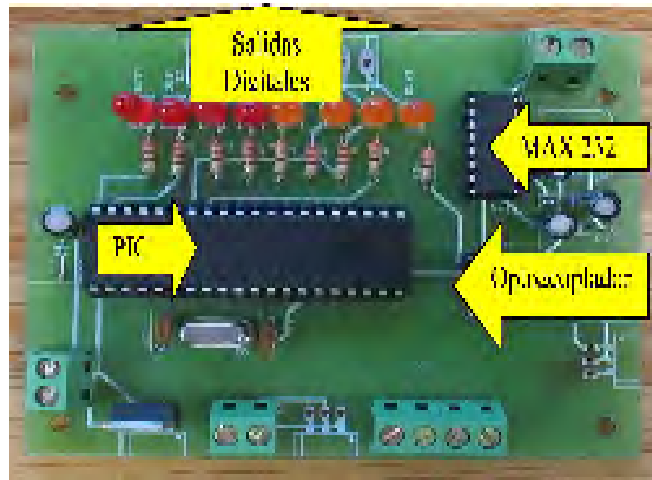


Figura 3.20.- Tarjeta desarrollada para el control del motor a pasos por medio de un PIC.

Una vez teniendo las salidas digitales del PIC, debemos acoplar un sistema de potencia que nos permita transferir dichas salidas a las bobinas del motor. El circuito utilizado se muestra en la figura 3.21.

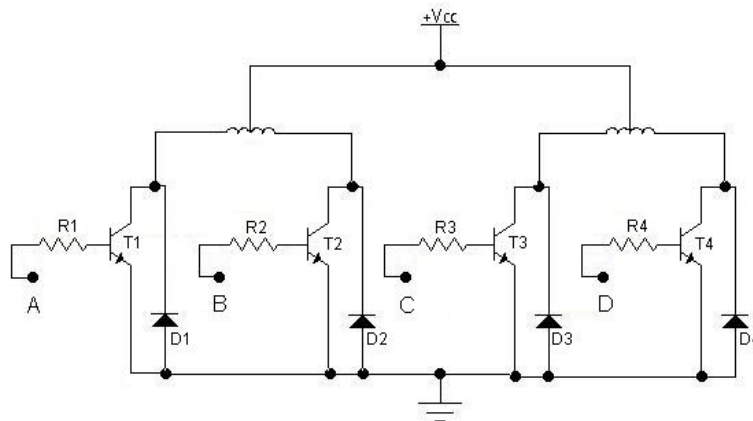


Figura 3.21.- Diagrama del circuito de potencia para manejar los motores a pasos.

### 3.3. Ensamble final de los contenedores de circuitos.

#### 3.3.1 Cajas “Xenix y Meteoro”.

A continuación se muestra el ensamble final de las tarjetas de electrónica, en las cajas correspondientes. Esto tiene la finalidad de hacer una descripción topológica del acomodo de cada una de ellas y clarificar su interconexión. Empezaremos mostrando el interior del contenedor denominado Xenix, (ver figura 3.22) en esta imagen se aprecian las tarjetas de los puentes H4, H5 y H6; que manejan a las ruedas inerciales en los ejes Z, X y Y, respectivamente. A la derecha de estas, se encuentran, en orden ascendente, la tarjeta de desarrollo Lp3500, el Rabbit, la tarjeta de

reguladores y divisores de voltaje y hasta arriba, el radiomódem, que aquí no aparece, pero si en la figura 3.23.

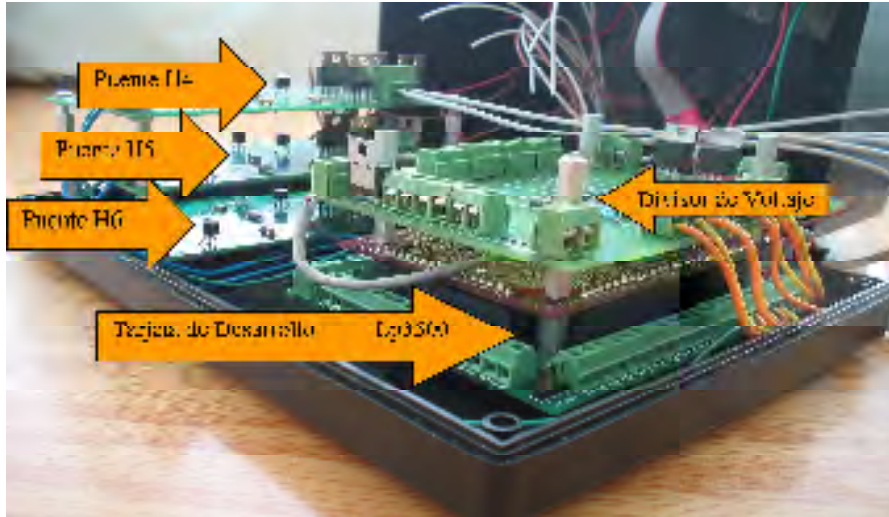


Figura 3.22.- Vista al interior de la caja Xenix en proceso de ensamblado.

En la figura 3.23 se muestra la caja Xenix totalmente armada, en la que se puede apreciar en su parte superior izquierda, los bornes de alimentación para los puentes H (18 V) mientras que a la derecha se encuentran los bornes de polarización para el radiomódem, la brújula y el Rabbit (12 V). El cable de programación se comunica con la PC a través de un conector DB9 para permitir cargar o depurar los programas dentro del microcontrolador.

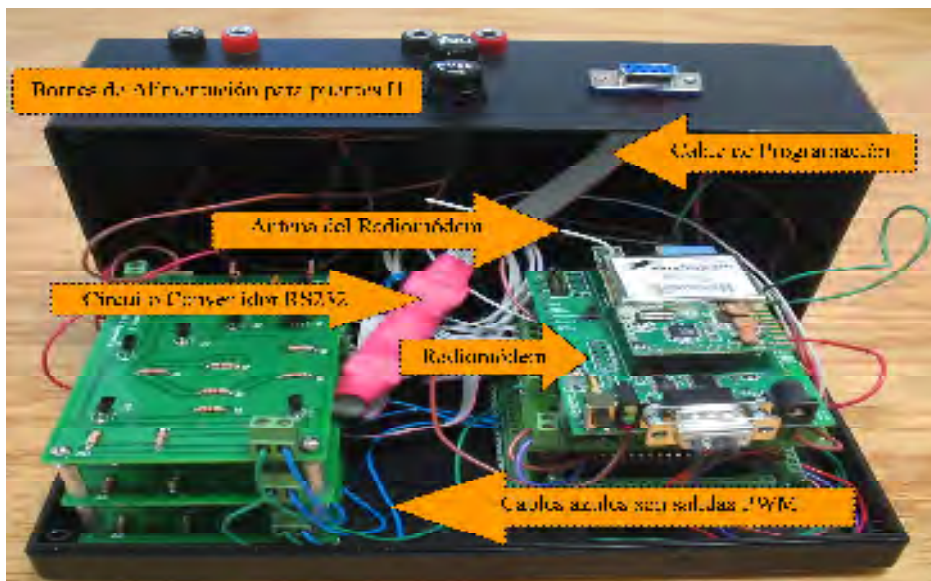


Figura 3.23.- Caja Xenix con vista a sus principales elementos y totalmente armada.

Para terminar con la descripción de la caja Xenix, mostraremos la [siguiente tabla](#) donde se señalan los colores de los cables utilizados para la interconexión entre tarjetas y componentes. El calibre del cable es del no. 20, con excepción de las salidas de potencia hacia las ruedas inerciales donde se usó calibre 16, con forro especial de PVC.

Tabla 3.4.- Designación de colores de cables utilizados en la interconexión de elementos en Xenix.

<b>COLOR</b>	<b>DESCRIPCIÓN</b>
Azul	Salidas PWM
Verde	Señales para sentidos de motores
Rojo	Voltajes positivos
Café	Tierra digital y analógica.
Rojo en Radiomódem	RTS(7)/ TxE(LP3500)
Morado	CTS(8)/RxE(LP3500)
Verde al costado de la caja	Alimentación del led 5 [V]
Blanco AWG16	Señales de Potencia
Blanco AWG22	Señales de TRX y GND para la brújula
Gris	Alimentación principal del Rabbit
Anaranjado y Amarillo	Salidas de potencia del Rabbit

En la figura 3.24.- se muestra el contenedor denominado “Meteoro”, en una de sus etapas de ensamblado, donde se encuentra la tarjeta del PIC y las tarjetas de los puentes H1, H2 y H3 que manejan las bobinas magnéticas en los ejes X,Y y Z. Esta caja es complemento del sistema de control; el acomodo y su diseño son muy parecidos a los de Xenix.

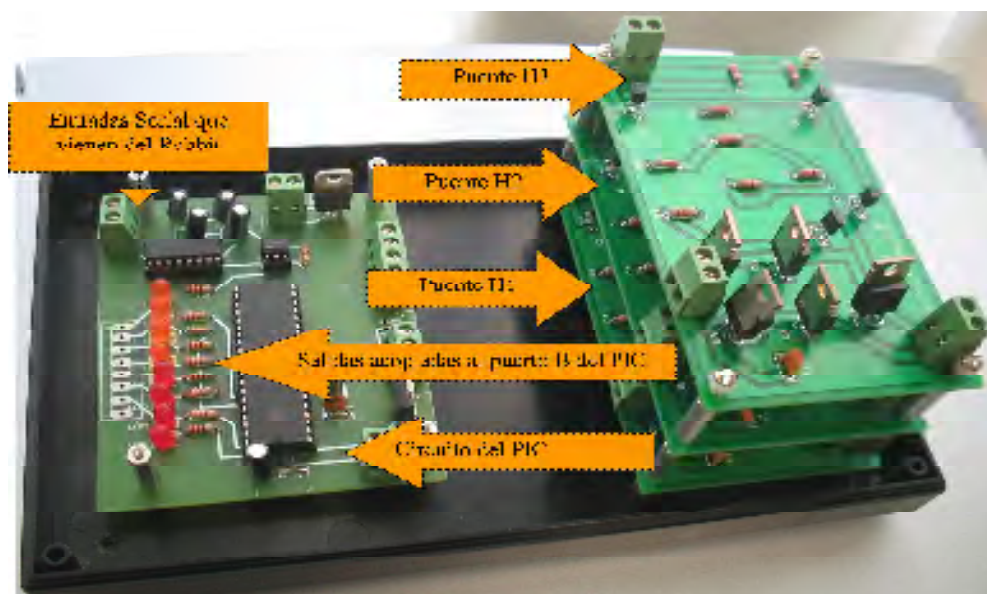


Figura 3.24.- Vista interior de la caja Meteoro en una de sus etapas de ensamblado.

En la tabla 3.5 se presenta el código de colores de cables utilizado en este contenedor. El calibre del cable es el 20, con excepción de las salidas de potencia hacia las bobinas magnéticas donde se usó calibre 16, con forro especial de PVC. En la figura 3.25 se muestra la caja ya terminada, donde se aprecia en el conjunto de tarjetas de la parte izquierda los puentes H y a la derecha, los circuitos de potencia para manejar los motores a pasos de las masas deslizantes, que a su vez se encuentran sobre la tarjeta del PIC.

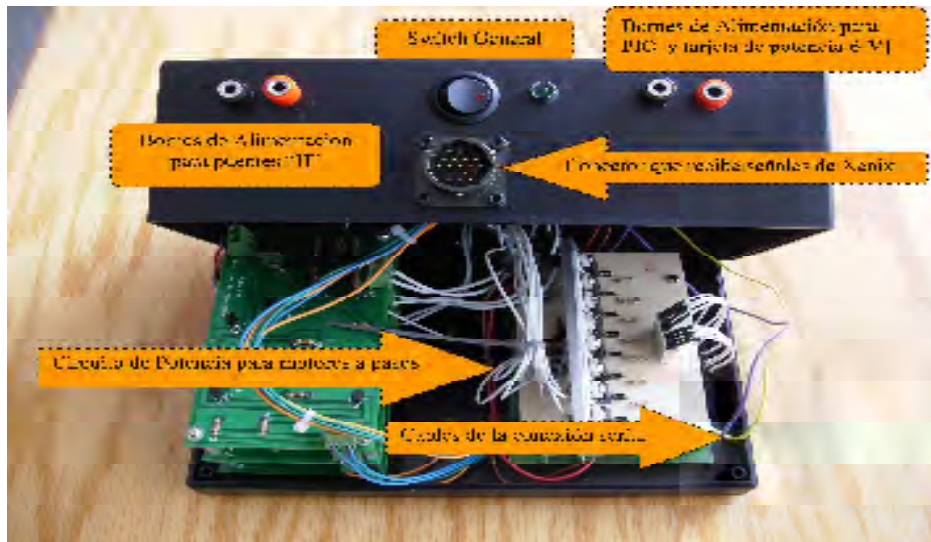


Figura 3.25.- Contenedor Meteoro. A la izquierda el conjunto de tarjetas de los puentes H que manejan las bobinas magnéticas y a la derecha la tarjeta de potencia para los motores de pasos.

Tabla 3.5.- Designación de colores de cables utilizados en la interconexión de elementos en el contenedor denominado Meteoro.

COLOR	FUNCIÓN
Amarillo	Recepción Serial
Café y Morado	Tierra analógica y digital.
Azul	Señales de Potencia y Giro en H3
Verde	Señales de Potencia y Giro en H2
Anaranjado	Señales de Potencia y Giro en H1
Rojo	Voltajes Positivos
Blanco AWG22	Señales digitales (salidas del PIC)
Blanco AWG16	Señales de Potencia

**3.3.2. Conectores Exteriores.**

Los conectores que se usaron son de dos tipos, de la serie 206061-1 de Amphenol, que cumple con estándares militares; de nueve y diecinueve pines. En esta sección se muestra la asignación de pines para tener una identificación positiva de la conexión entre los diferentes componentes de la plataforma de simulación. En total se usaron cinco conectores, dos en Xenix y tres en Meteoro.

Se muestran los conectores que se encuentran sobre las cajas, los de nueve pines son hembras y los de 19 alfileres son machos, ambos para chasis.

Para la colocación y conexión de las masas deslizantes y los motores de DC, es que se nombraron de acuerdo al eje que se ve afectado por su movimiento; así por ejemplo, el motor dos "Roll" se encuentra en el eje X alineado con la masa deslizante del Pitch, debido a que su eje de acción es el eje imaginario que se considera Roll. (Ver la etiqueta de señalización sobre la plataforma). A su vez el motor uno "Pitch" se encuentra alineado con la masa deslizante Roll en el eje Y.

**Conectores colocados en el contenedor Xenix.**

Tabla 3.6 Asignación de pines en el conector C1.X (macho chasis. Amphenol MS3112E14-19P)

M	Tierra física para la brújula.
A	Recepción de datos de la brújula electrónica
B	Alimentación de 5 [V] para la brújula.
L	Tierra física para el PIC.
N	Transmisión hacia el PIC.
P, C, K, U, V, R, D, E	Sin conectar.
J	Señal de sentido de polarización para la Bobina3.
T	Señal de sentido de polarización para la Bobina2.
S	Señal de Encendido para la Bobina2.
H	Señal de sentido de polarización para la Bobina1.
G	Señal de Encendido para la Bobina1.
F	Señal de Encendido para la Bobina3.



Figura 3.26. Distribución de pines para los conectores C1.X y C3.M

Tabla 3.7. Asignación de pines para el conector C2.X (hembra chasis. Amphenol 206705-1)

1,2,3	Sin conectar
4	Potencia para motor 2 "Motor del Roll" eje X
5	Potencia para motor 3 "Motor del Azimut"
6	Potencia para motor 1 "Motor del Pitch" eje Y
7	Potencia para motor 2 "Motor del Roll" eje X
8	Potencia para motor 3 "Motor del Azimut"
9	Potencia para motor 1 "Motor del Pitch" eje Y

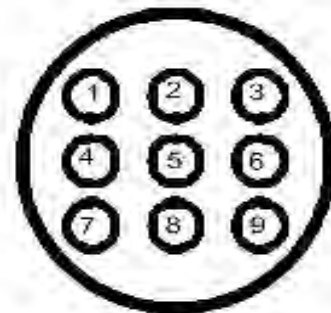


Figura 3.27. Distribución de pines en los conectores C2.X, C1.M y C2.M

**Conectores colocados en el contenedor Meteoro.**

Tabla 3.8. Asignación de pines en el conector C1.M (hembra chasis. Amphenol 206705-1.)

1	Potencia para Bobina 2 en el eje de cabeceo ( <i>Pitch</i> )
2	Potencia para Bobina 3 en el eje del Azimut
3	Potencia para Bobina 3 en el eje del Azimut
4	Terminal A de la bobina del motor a pasos para la masa deslizante del eje X ( <i>Roll</i> )
5	Terminal B de la bobina del motor a pasos para la masa deslizante del eje X ( <i>Roll</i> )
6	Terminal de alimentación de 6 V para la bobina común A-B del motor a pasos para la masa deslizante del eje de rotación X ( <i>Roll</i> )
7	Terminal C de la bobina del motor a pasos para la masa deslizante del eje X ( <i>Roll</i> )
8	Terminal D de la bobina del motor a pasos para la masa deslizante del eje X ( <i>Roll</i> )
9	Terminal de alimentación de 6 V para la bobina común C-D del motor a pasos para la masa deslizante del eje X ( <i>Roll</i> )

Tabla 3.9. Asignación de pines en el conector C2.M (hembra chasis. Amphenol 206705-1)

1	Potencia para Bobina 2 en el eje de cabeceo ( <i>Pitch</i> )
2	Potencia para Bobina 1 en el eje de rotación ( <i>Roll</i> )
3	Potencia para Bobina 1 en el eje de rotación ( <i>Roll</i> )
4	Terminal A de la bobina del motor a pasos para la masa deslizante del eje Y ( <i>Pitch</i> )
5	Terminal B de la bobina del motor a pasos para la masa deslizante del eje Y ( <i>Pitch</i> )
6	Terminal de alimentación de 6 V para la bobina común A-B del motor a pasos para la masa deslizante del eje de cabeceo ( <i>Pitch</i> )
7	Terminal C de la bobina del motor a pasos para la masa deslizante del eje Y Pitch
8	Terminal D de la bobina del motor a pasos para la masa deslizante del eje Y Pitch
9	Terminal de alimentación de 6 V para la bobina común C-D del motor a pasos para la masa deslizante del Pitch

Tabla 3.10 Asignación de pines en el conector C3.M (macho chasis. Amphenol MS3112E14-19P)

M	Transmisión hacia el PIC.
A	Tierra física para el PIC.
B, L, N, P, C, K, U, V, R, D, F	Sin conectar
J	Señal de sentido de polarización para la Bobina3
T	Señal de Encendido para la Bobina 3
S	Señal de sentido de polarización para la Bobina 2
E	Señal de Encendido para la Bobina 2
H	Señal de Encendido para la Bobina 2
G	Señal de Encendido para la Bobina 1

## Capítulo

## 4

**INTERFAZ GRÁFICA ENTRE LA COMPUTADORA DE ABORDO Y LA ESTACIÓN TERRENA****4.1. Descripción.**

La interfaz gráfica es un medio por el cual se comunica la computadora de abordo y la Estación Terrena, que en este caso será la PC donde se seleccionan las diferentes rutinas de balanceo y control de orientación que puede realizar el sistema.

Una interfaz gráfica es aquella que presenta un entorno más amigable para entender el funcionamiento y operación de un sistema a fin de que la reacción del usuario sea más rápida y con menos errores posibles. Para esta aplicación se desarrolló una interfaz escrita en Visual Basic 6.0. En ella se presentan básicamente dos pantallas; una de inicio o bienvenida, y otra, de control que cuenta con tres secciones. La primera pantalla da la opción de entrar al programa, configurar los puertos de comunicación o salir, esto a través de tres botones como se muestra en la figura 4.1, mientras que la segunda, permite establecer algunos parámetros de los ensayos.



Figura 4.1.- Pantalla de inicio de la interfaz gráfica entre la plataforma de simulación y la estación terrena.



Esta interfaz depende, para su operación, de un sistema de comunicación bidireccional inalámbrico, cuyo funcionamiento ha sido explicado en el capítulo anterior y que sirve para que la computadora de abordo pueda recibir las peticiones de la Estación Terrena. En la figura 4.2 se puede apreciar como es, en términos muy generales, la comunicación entre las diferentes Interfaces.

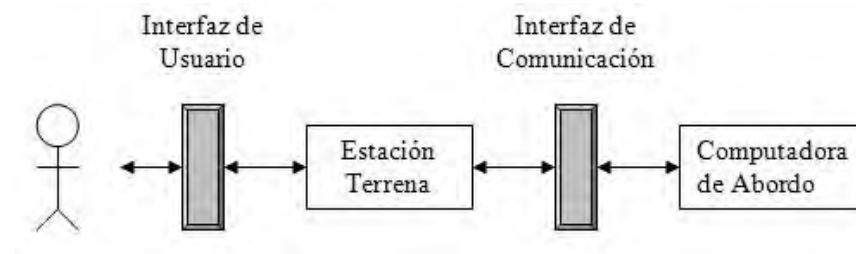


Figura 4.2.- Representación gráfica de la comunicación que existe entre el usuario, la estación terrena y la computadora de abordo.

Una vez que se ha accedido al programa, se debe seleccionar el tipo de experimento a realizar, entre los tres conjuntos principales: balanceo de la plataforma, control de orientación con ruedas inerciales y control con bobinas magnéticas (ver figura 4.3). También se tiene la opción de cancelar una acción para poder empezar otra nueva, o salir del programa.

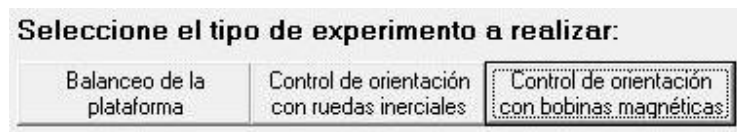


Figura 4.3.- Botones para seleccionar un experimento en la Interfaz Gráfica.

Al momento de seleccionar una opción de ensayo o prueba, el sistema desplegará entonces una pantalla diferente, con los elementos mostrados en la figura 4.4 para que el usuario pueda establecer las condiciones bajo las cuales desea que trabaje el sistema, tales como: el umbral con el que se quiere balancear la plataforma, el eje en particular que se quiere controlar, o como en el caso de las bobinas magnéticas, se puede seleccionar su sentido de polarización.

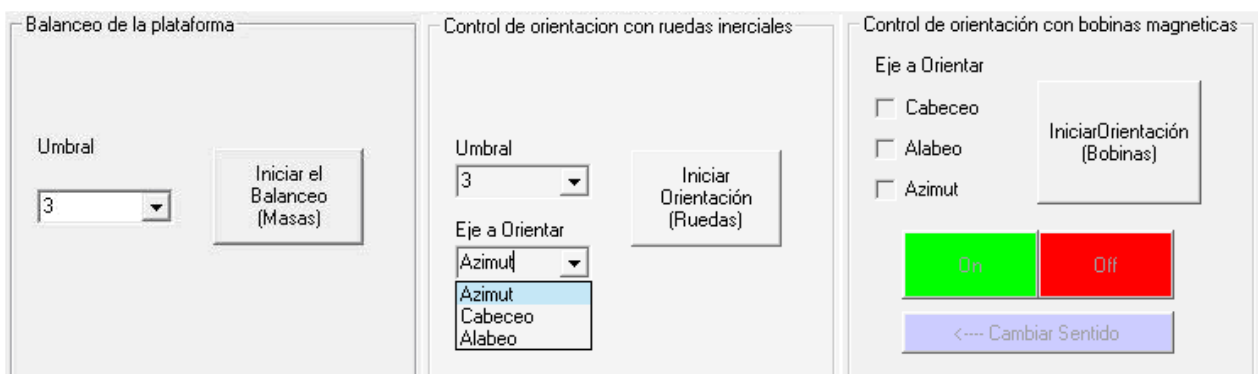


Figura 4.4. Elementos de control para cada conjunto de experimentos: bobinas magnéticas, ruedas inerciales y balanceo automático con las masas deslizantes.

Para iniciar un experimento, de balanceo o control, es necesario seleccionar según sea el caso, el umbral o el eje a controlar, después de esto se oprime el botón **Iniciar** y el sistema empezará a realizar el control y a desplegar la información de la orientación de la plataforma en la tabla de datos. Al oprimir **Iniciar**, causa que el programa envíe un comando, que es un conjunto de caracteres, a la computadora de abordo con el que interpreta y determina que tipo de acción o control realizar y con que opciones.

La interfaz de usuario cuando se lleva a cabo la experimentación y el almacenamiento de datos provenientes de la brújula electrónica, se muestra en la figura 4.5.

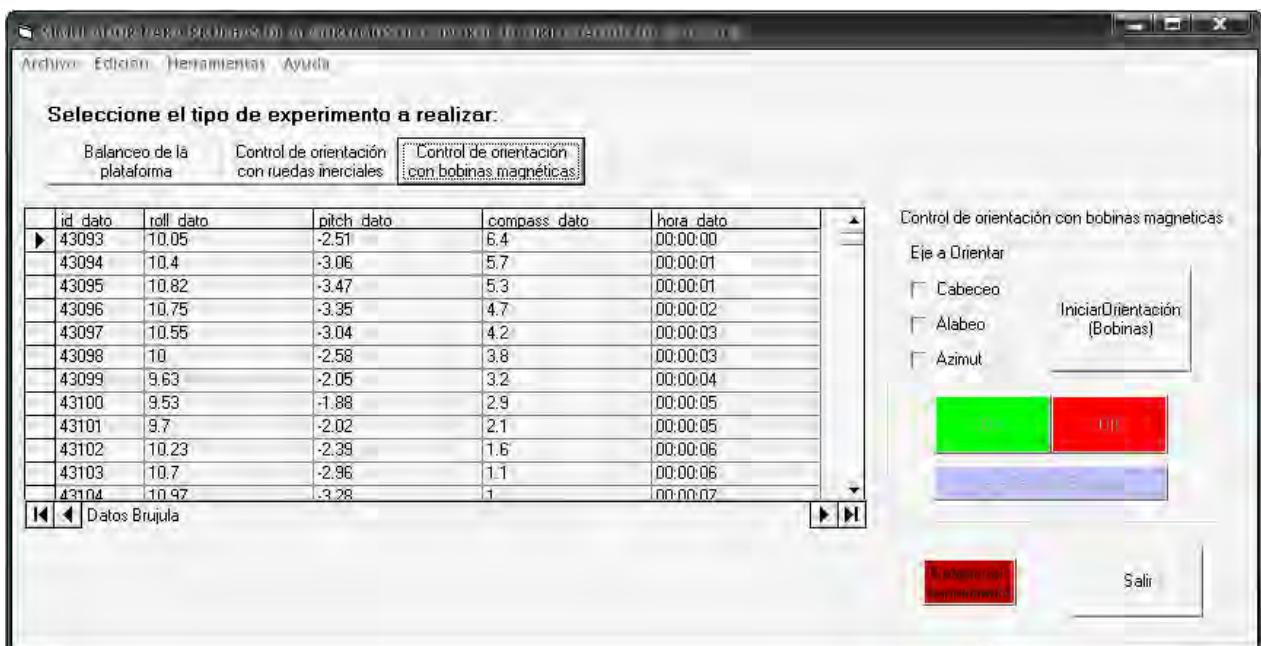


Figura 4.5.- Pantalla de control y almacenamiento de datos.

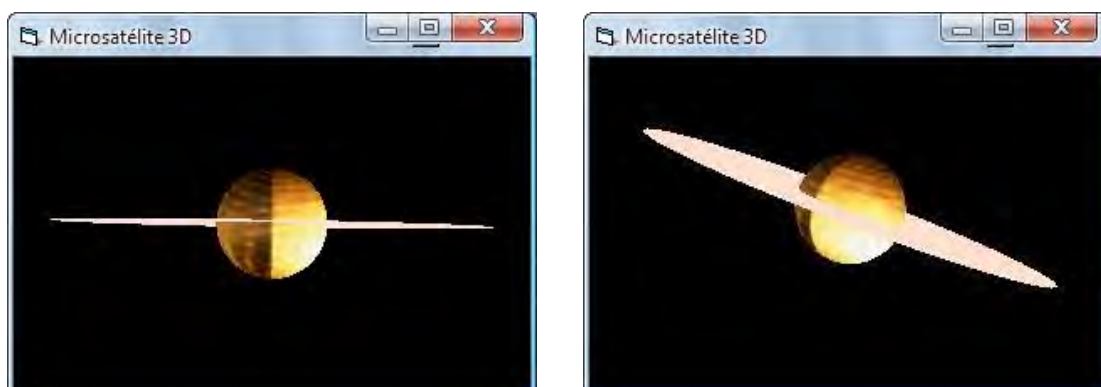


Figura 4.6.- Pantalla donde se aprecia el comportamiento en tiempo real de la plataforma de simulación de acuerdo a los datos que se están capturando.

La interfaz gráfica también cuenta con una pantalla donde se puede observar el movimiento de la plataforma, con el fin de que se pueda apreciar el comportamiento que está teniendo, para verificar la funcionalidad de los controles y que al usuario puede determinar de una manera simple y rápida si el comportamiento va de acuerdo con lo programado en el algoritmo bajo prueba. Esta gráfica en tiempo real facilita la observación y hace más amigable a dicha interfaz. Un ejemplo de la pantalla de Microsatélite 3D se muestra en la figura 4.6.

#### 4.2. Comandos.

Tabla 4.1 Comandos enviados desde la estación terrena hacia la computadora de abordo.

<i>Caracter</i>	<i>ASCII</i>	<i>Actuador</i>	<i>Descripción</i>
<b>Comandos para la inicialización de un control.</b>			
<i>a</i>	97	<i>Masas Deslizantes</i>	<i>Balanceo</i>
<i>b</i>	98	<i>Ruedas Inerciales</i>	<i>Modo de reacción e inercia</i>
<i>c</i>	99	<i>Bobinas Magnéticas</i>	<i>Freno Magnético</i>
<b>Comandos para la selección del Umbral de: Masas Deslizantes y Ruedas Inerciales</b>			
<i>0</i>	48	<i>Masas y Ruedas</i>	<i>Umbral 3°</i>
<i>1</i>	49	<i>Masas y Ruedas</i>	<i>Umbral 2°</i>
<i>2</i>	50	<i>Masas y Ruedas</i>	<i>Umbral 1°</i>
<i>3</i>	51	<i>Masas y Ruedas</i>	<i>Umbral 0.5°</i>
<b>Comandos para la selección del eje a controlar de: Bobinas Magnéticas y Ruedas Inerciales</b>			
<i>0</i>	48	<i>Bobinas y Ruedas</i>	<i>Azimut</i>
<i>1</i>	49	<i>Bobinas y Ruedas</i>	<i>Cabeceo</i>
<i>2</i>	50	<i>Bobinas y Ruedas</i>	<i>Alabeo</i>
<b>Comandos para el cambio de sentido de la corriente y control de encendido en las Bobinas Magnéticas</b>			
<i>C</i>	67	<i>Bobinas</i>	<i>Cambio de sentido Bobinas Magnéticas</i>
<i>N</i>	78	<i>Bobinas</i>	<i>Encendido de las Bobinas Magnéticas</i>
<i>F</i>	70	<i>Bobinas</i>	<i>Apagado de las Bobinas Magnéticas</i>
<i>P</i>	80	<i>Bobinas</i>	<i>Detener Control de Bobinas Magnéticas</i>

Los comandos a utilizar son esencialmente caracteres enviados por el programa que se denomina: Micro-Satélite, hacia la computadora de abordo; que los interpreta como una función a realizar. Al mismo tiempo manda los datos de la orientación de la plataforma de simulación, hacia la Estación Terrena para que se pueda monitorear el experimento de balanceo o estabilización que se está realizando. Además de enviar los datos que proporciona la brújula electrónica, también transmite alertas para informar que la plataforma ha terminado un proceso de control.

Los caracteres que se envían y reciben se apegan al estándar ASCII (American Standard Code for Information Interchange) ya que tanto el controlador de la computadora de abordo como la PC de la estación terrena, manejan rutinariamente este código. El estándar ASCII representa a todos los caracteres por un número en binario, el cual es enviado por los dispositivos en forma de voltaje alto y bajo, o por medio de una codificación en frecuencia. Por ejemplo, si se quiere enviar una letra 'a' por ASCII, dicha letra está representada por el número binario 0110 0001.

En la tabla 4.1 se presentan los comandos utilizados en este proyecto, mismos que son enviados desde la estación terrena hacia la computadora de abordo, con las diferentes opciones de control.

El programa de control de la computadora de abordo tiene una secuencia específica para poder recibir comandos, ya que al entrar en un control específico espera un carácter de un conjunto cerrado de opciones. El diagrama de la figura 4.7 muestra los diferentes flujos de control que puede haber entre la interfaz de la estación terrena y la computadora de abordo.

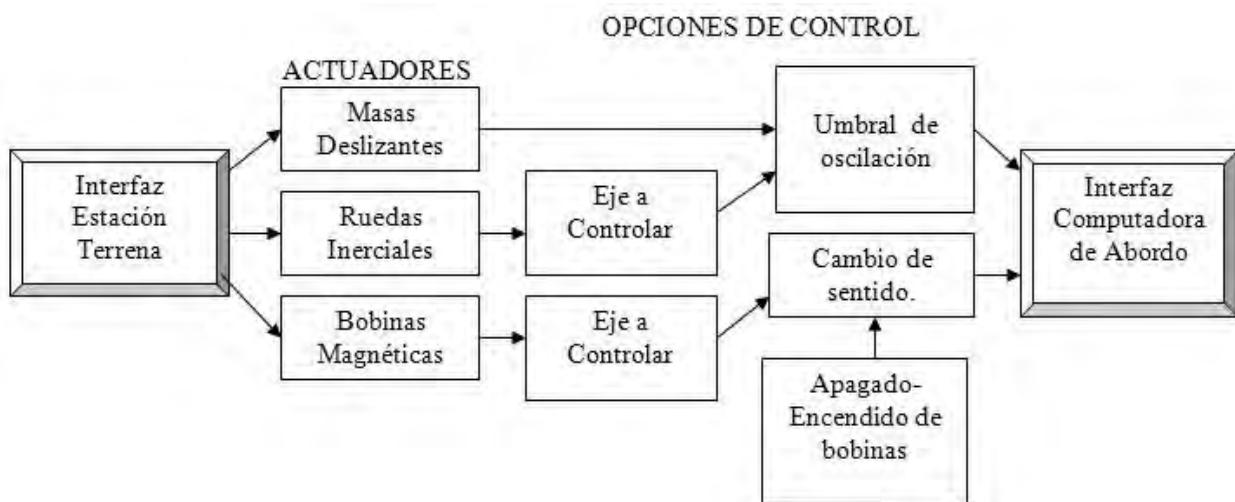


Figura 4.7.- Flujo de control entre la interfaz de usuario y la computadora de abordo.

### 4.3. Instalación de nuevas rutinas de orientación y control.

La instalación de nuevas rutinas de control es esencial para darle continuidad al desarrollo del sistema y mantener actualizadas sus capacidades de control, siendo algo realmente muy sencillo, sólo se necesitan modificar los dos programas, el del microcontrolador Rabbit y la interfaz gráfica en Visual Basic.

En el programa de Visual Basic se pueden agregar más botones para implementar las nuevas opciones, los cuales pueden ser copiados con las mismas sentencias que tienen los botones de control actuales, con la diferencia de solamente cambiar el carácter a enviar por medio del puerto serial. En el programa del Rabbit se tendrá que agregar un nuevo caso en el “select case” general, y escribir la rutina correspondiente.

Crear nuevas rutinas de desarrollo para el sistema en general no tiene muchas complicaciones, ya que fue diseñado para que se puedan agregar en forma independiente a las ya utilizadas, tanto en el caso del programa del microcontrolador como en la interfaz de usuario.

#### 4.4. Almacenamiento de datos de orientación de la plataforma.

El almacenamiento de la orientación de la plataforma es una necesidad imperiosa para el sistema de simulación, ya que es la única manera de poder tener un mejor control en la información y poder graficar y evaluar el comportamiento que va teniendo una prueba a lo largo del tiempo. Sobre todo por la gran cantidad de experimentos que es necesario llevar a cabo para validar algunos resultados. Este sistema de monitoreo nos va a servir para poder ver si la acción que realiza la plataforma es acorde con el algoritmo programado y si es de utilidad en la orientación y control del micro-satélite.

Se decidió que la mejor forma de presentar la información de la orientación era auxiliándose de un programa gráfico, para poder observar el movimiento del simulador en tiempo real en sus tres grados de orientación (alabeo, cabeceo y guiñada). Pero más importante, es almacenar en una base de datos la información de la orientación.

##### 4.4.1. Despliegue de datos de orientación.

La información crucial en este sistema de monitores es la orientación: alabeo (X), cabeceo (Y) y guiñada (Z). Estas variables, junto con el número de evento y el reloj interno de la computadora, son las que el programa de graficación se encarga de monitorear, durante las pruebas de orientación. La denominación de los ejes de referencia es análoga a la utilizada en aviación y se describen de la siguiente manera:

El eje de alabeo o rotación, se encuentra localizado de manera axial al vector velocidad, es decir, se encuentra sobre la trayectoria orbital, esto se ilustra en la [Figura 4.8](#).

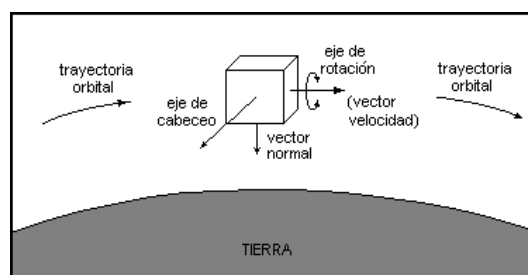
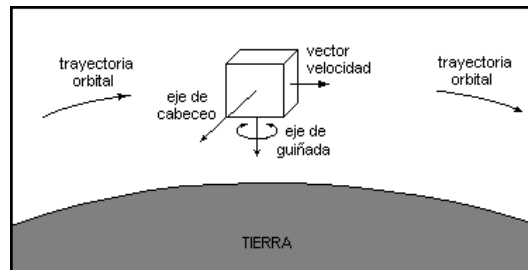


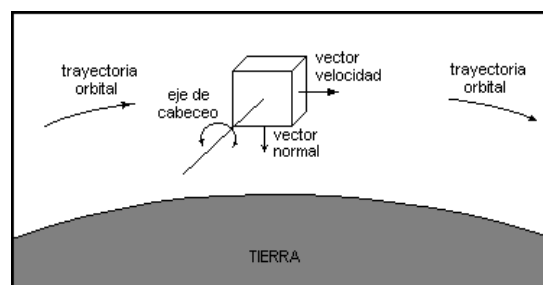
Figura 4.8. Definición del eje de rotación.

El eje de guiñada o azimut se define como el vector que une el centro de masa del satélite, con el centro de masa de la tierra (Figura 4.9).



*Figura 4.9 Eje de guiñada.*

El eje de cabeceo es perpendicular a los ejes de rotación y de guiñada, de tal manera que se forma un sistema de ejes ortogonales (Figura 4.10).



*Figura 4.10 Eje de cabeceo.*

#### **4.4.2. Almacenamiento en una Base de datos.**

La información de la brújula electrónica se almacena mediante el programa Micro-Satélite directamente en una base de datos de Microsoft Access, con el fin de que dicha información se pueda manejar más fácilmente al exportarlos o graficarlos por medio de otro programa o aplicación.

El archivo que contiene la base de datos se encuentra en la carpeta de ejecución del programa Micro-Satélite. En la tabla 4.2 se puede apreciar como se van guardando los registros de la brújula electrónica, así mismo en la tabla 4.3 se describen los atributos de dicha base de datos.

**Tabla 4.2. Ejemplo del tipo de información que despliega la interfaz de usuario en la estación terrena.**

id_dato	roll_dato	pitch_dato	compass_dato	hora_dato
38209	-0.25	-1.69	181.5	28/01/2009 14:21:32
38210	-0.30	-1.71	182.0	28/01/2009 14:21:33
38211	-0.32	-1.59	181.6	28/01/2009 14:21:33
38212	-0.32	-1.86	181.6	28/01/2009 14:21:34
38213	-0.30	-1.76	182.0	28/01/2009 14:21:35
38214	-0.25	-1.78	181.8	28/01/2009 14:21:35
38215	-0.35	-1.62	181.6	28/01/2009 14:21:36
38216	-0.30	-1.76	182.0	28/01/2009 14:21:37
38217	-0.32	-1.71	181.6	28/01/2009 14:21:37
38218	-0.37	-1.76	181.8	28/01/2009 14:21:38
38219	-0.27	-1.64	181.5	28/01/2009 14:21:39

Datos Brújula

**Tabla 4.3 Descripción de las columnas en la tabla de la base de datos del sistema.**

Dato	Descripción
Id_dato	Identificador único para cada dato.
roll_dato	Valor del eje de alabeo del satélite.
pitch_dato	Valor del eje de cabeceo del satélite.
compass_dato	Valor del eje de azimut del satélite.
hora_dato	Hora y fecha en la que se obtuvo el dato del satélite.

## Capítulo

**5****PROGRAMACIÓN DE RUTINAS EN LENGUAJE DE ALTO NIVEL Y PRUEBAS DE FUNCIONAMIENTO.**

En este capítulo se hace una descripción de los algoritmos desarrollados en lenguaje de alto nivel para el sistema de simulación con que cuenta el Laboratorio de Percepción Remota Alternativa y Tecnología Avanzada del Instituto de Geografía de la UNAM. Esta programación fue desarrollada para actualizar, junto con la circuitería, los sistemas de balanceo automático, y de orientación y control. Se presentan las pruebas de funcionamiento efectuadas con todos estos subsistemas.

**5.1. Programación de la computadora de abordó.**

En esta sección se hace una descripción de como fue programada la computadora en una sola tarjeta Rabbit 3500, para llevar a cabo una serie de experimentos como el balanceo automático de la plataforma utilizando diferentes umbrales, así como las pruebas de control de orientación, utilizando bobinas magnéticas y ruedas inerciales.

El procedimiento de programación esta basado en una lógica con diferentes casos independientes entre ellos. Esto se realiza con una sentencia de control llamada "select case" la cual nos da la posibilidad de seleccionar una opción, dependiendo del valor de una variable; que es asignada por un caracter enviado por la estación terrena.

Se consideraron tres tipos de experimentos en el microcontrolador Rabbit cada uno de ellos como un caso en el "select case" principal, los cuales son: balanceo mediante masas deslizantes, control de orientación con bobinas magnéticas y control con ruedas inerciales.

En la figura 5.1 se muestra el diagrama de flujo del programa, las rutinas se ejecutan a través de la interfaz gráfica desarrollada para este proyecto, descrita en el capítulo 4.



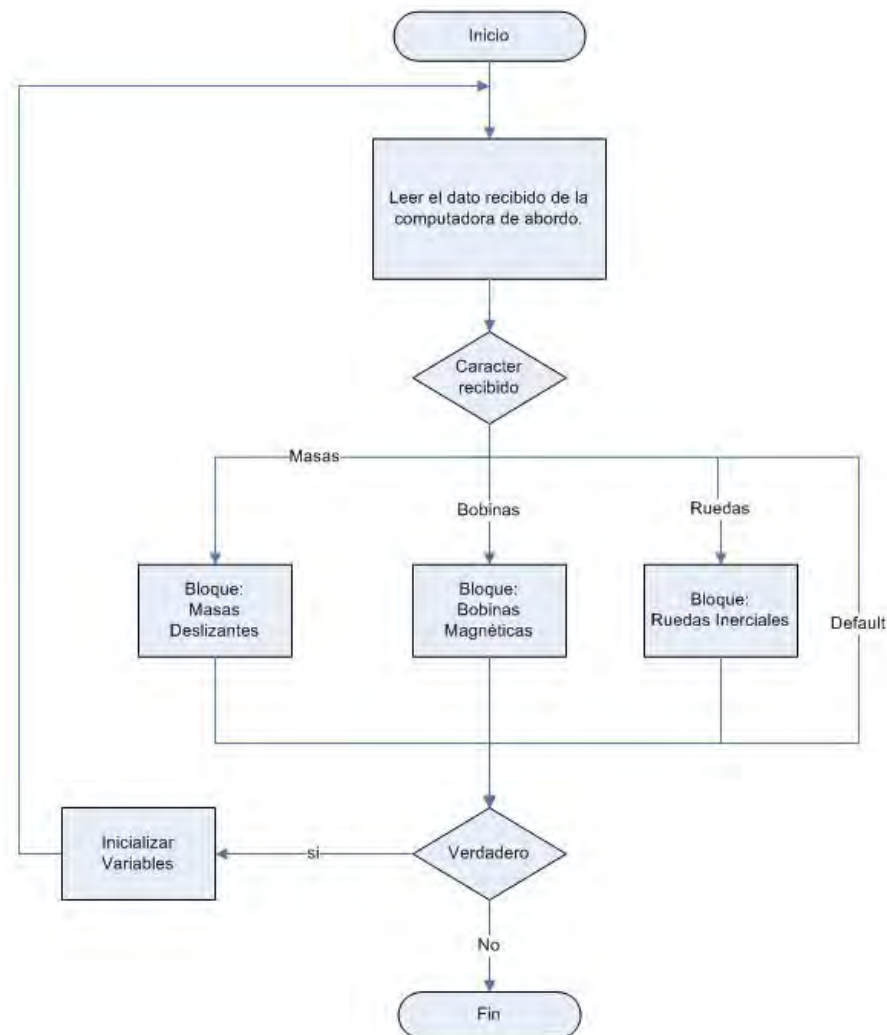


Figura 5.1.- Diagrama de flujo donde se muestra el concepto general de programación desarrollado para esta aplicación.

### 5.1.1. Pruebas de funcionamiento del sistema de monitoreo de la plataforma de simulación.

El sistema de monitoreo fue probado extensivamente durante la gran cantidad de experimentos realizados en la plataforma; para el balanceo automático es necesario recabar los datos de orientación en los ejes X,Y para poder corroborar que la plataforma se encuentre balanceada adecuadamente; para las pruebas de control de orientación, es indispensable llevar a cabo el monitoreo continuo de la orientación del simulador, debido a la gran importancia que tiene el poder caracterizar y evaluar de manera objetiva, el comportamiento y efectividad de los algoritmos de control implementados, de hecho, la necesidad de supervisión, graficación y almacenamiento de datos de orientación fue la necesidad del desarrollo del sistema de monitoreo.

En la versión anterior de este sistema de monitoreo (Contreras F. 2004) hubo fallas con el sistema inalámbrico, debido a interferencias en el lugar donde se efectuaron los experimentos, por lo que se utilizó el modo de prueba para corregir la transmisión-recepción, utilizando para ello las tarjetas de desarrollo. Con el sistema de transmisión implantado para este trabajo de tesis fue eliminado por completo el problema de pérdida de información de orientación de la plataforma. El sistema de

monitoreo cumple con la función de mostrar en formato alfanumérico, la información de la hora y los ángulos de los tres ejes de orientación. Los datos de: número de evento, fecha, hora, eje X, eje Y y eje Z son transmitidos en forma inalámbrica y almacenados en un archivo, para su posterior graficación y análisis (figura 5.2).

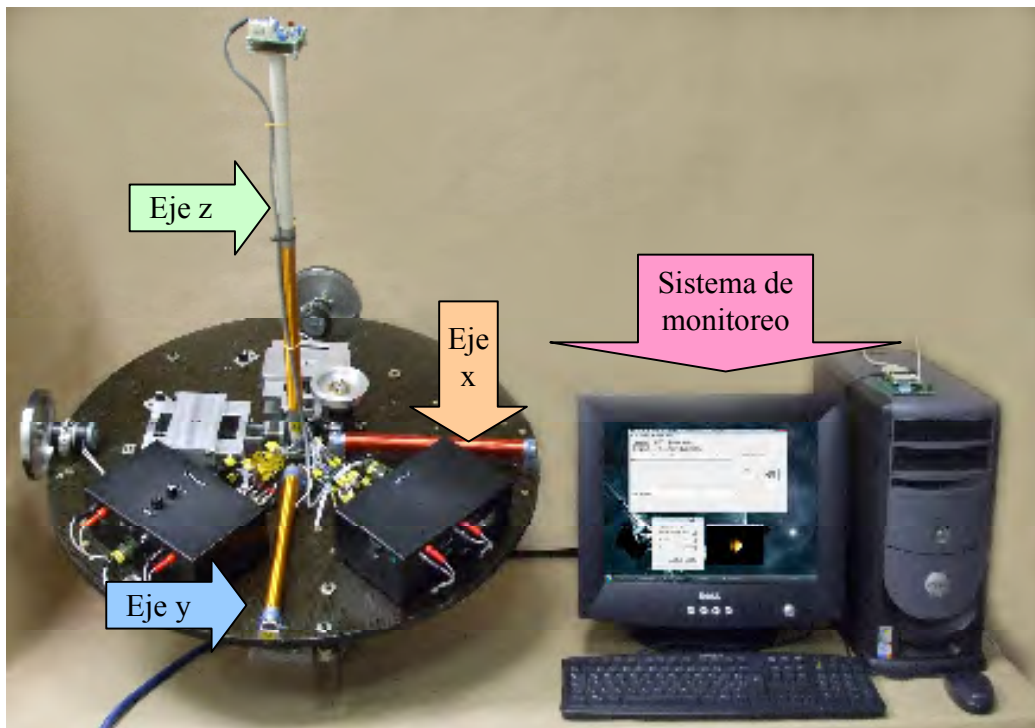


Figura 5.2.- Ejecución de algoritmos en el simulador a través de la interfaz gráfica.

## 5.2. Balanceo automático de la plataforma de simulación.

El simulador presentado en este trabajo permite llevar a cabo una amplia gama de pruebas de detección de orientación y control de estabilización para satélites. Pero, para poder efectuar pruebas confiables es necesario que este se encuentre balanceado, lo que ayudará a cancelar en cierta medida los efectos causados por los pares gravitacionales.

### 5.2.1. Balanceo manual.

El balanceo manual de la plataforma, es un procedimiento que debe seguirse independientemente del método de balanceo automático, que pueda llevarse a cabo posteriormente. La plataforma de simulación se encuentra inicialmente balanceada, es decir, su masa esta distribuida de manera homogénea en toda su superficie. El procedimiento seguido consiste básicamente en mantener balanceada la plataforma, añadiendo un par de componentes cada vez, contrarrestando cada uno el efecto de desbalanceo causado por el otro. La importancia de esta estrategia radica en que una vez alcanzado el equilibrio, es posible añadir más componentes, manteniendo sin alteraciones el balanceo (Juárez A. 2001).

### 5.2.2. Balanceo Automático.

El proceso de balanceo de una plataforma sin fricción, que no va acompañado de un método automático, resulta complicado, tardado y no siempre permite alcanzar resultados satisfactorios, por lo que se utiliza un conjunto de masas deslizantes que bajo el control de un sistema realimentado, permite efectuar un balanceo adecuado. El objetivo primordial, es el tener la capacidad de llevar el centro de masa de la plataforma, hacia al centro de rotación del cojinete neumático esférico (Prado J, 2007).

El balanceo automático se efectúa teniendo como premisa, el hecho de que la plataforma estando en posición horizontal, se encuentra balanceada, entonces, por medio de las señales de dos inclinómetros que determinan la orientación de la plataforma con respecto a la horizontal local se establece si hay alguna desviación y enseguida se utilizan dos masas deslizantes, una por cada eje, para reposicionar el centro de masa.

Se diseñó una rutina para que la computadora de a bordo reciba los datos de orientación de la plataforma, desde la brújula electrónica, colocada en el centro de rotación de la misma, a través de un puerto de comunicación serie. Hay que recordar que la finalidad es preservar la plataforma en el plano horizontal para que se mantenga balanceada. Los datos de alabeo y cabeceo (X,Y) son enviados hacia el microcontrolador Rabbit para ser comparados contra los diferentes umbrales de  $\pm 3^\circ$ ,  $\pm 2^\circ$ ,  $\pm 1^\circ$  y  $\pm 0.5^\circ$ , si la orientación de cada uno de los ejes no se encuentra dentro de este intervalo, entonces el microcontrolador activa a los motores de pasos, los cuales mueven a las masas deslizantes en un sentido o en otro, sucesivamente, hasta estabilizar la plataforma dentro del intervalo deseado.

La descripción del funcionamiento del programa es la siguiente: a) para efectuar el balanceo automático a través del movimiento de las masas deslizantes, se toma el valor de la brújula electrónica y con base en este, ya sea el cabeceo o el alabeo, se multiplica su valor por una constante, (la cual fue obtenida por métodos experimentales, de acuerdo con el movimiento de los motores a pasos y las masas deslizantes), b) al resultado de la multiplicación manda al PIC un número determinado de caracteres para que este mueva la masa ya sea del eje de alabeo o cabeceo, c) si un eje ya esta balanceado no se le envía ningún comando al PIC e inmediatamente vuelve a comprobar el otro eje para terminar de balancearlos, d) cuando los dos hayan terminado de balancearse se envía un mensaje a la computadora de abordaje diciendo que el proceso automático está terminado, y e) a su vez el microcontrolador Rabbit está enviando todo el tiempo los datos de orientación de la plataforma generados por la brújula electrónica, de forma inalámbrica, a la estación terrena para que en esta se puedan visualizar y almacenar.

En la figura 5.3 se muestra el diagrama de flujo de la rutina desarrollada para balancear automáticamente la plataforma de simulación, mediante la utilización de un lazo de control realimentado. El código correspondiente a la programación del microcontrolador Rabbit para el caso de las masas deslizantes se encuentra en el apéndice A.

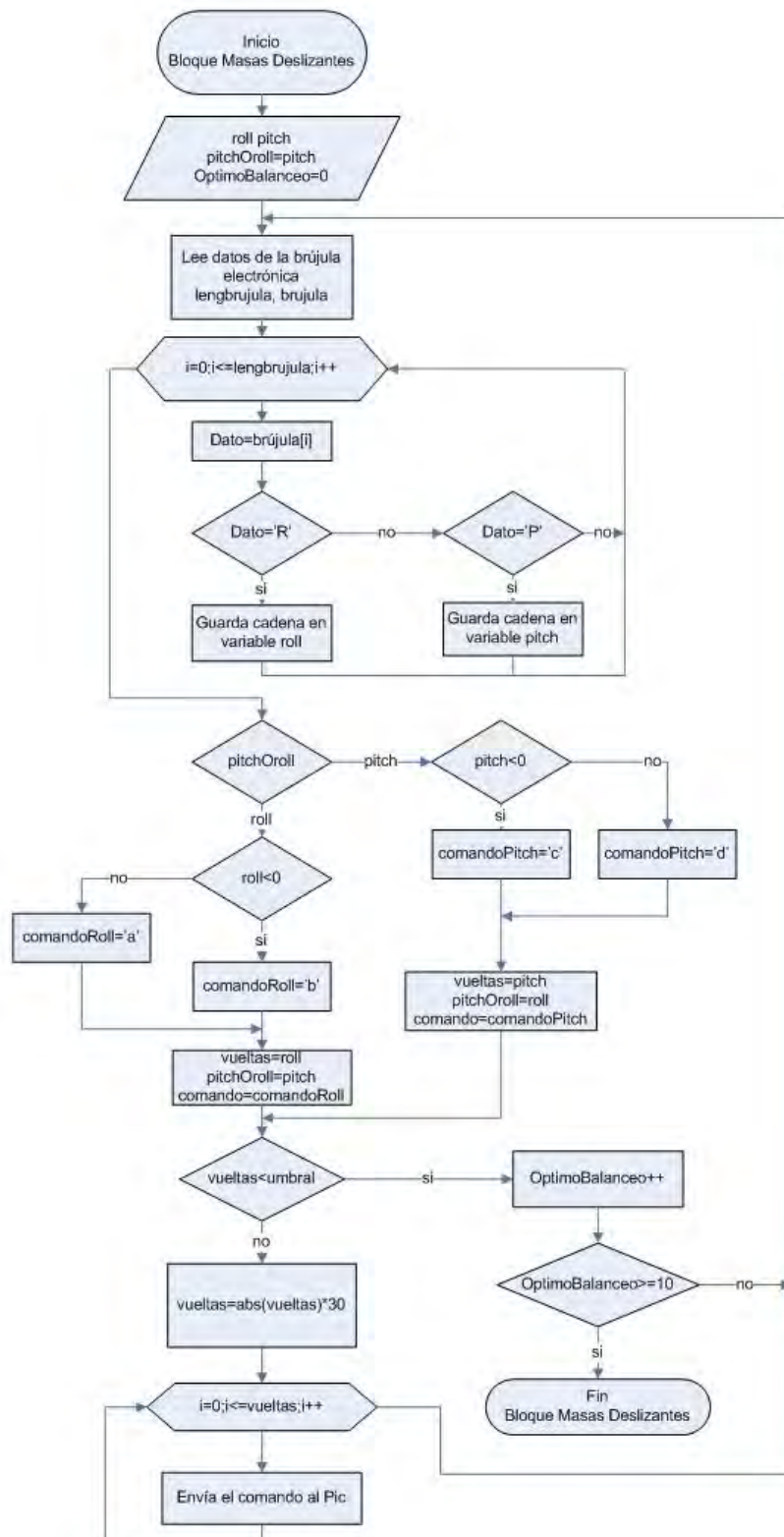


Figura 5.3.- Diagrama de flujo de la subrutina para realizar el balanceo automático del simulador físico de un medio sin fricción.

### 5.2.2.1- Curvas de Respuesta.

En las gráficas de las figuras 5.4 y 5.5, se observa el comportamiento típico de dos pruebas de funcionamiento, donde se demuestra la efectividad del sistema de control en la ejecución del balanceo automático. Es muy importante el introducir un retraso entre cada ciclo de control, debido a la naturaleza oscilatoria de la plataforma durante este proceso, para que éste pueda llevarse a cabo de manera adecuada. En la figura 5.4 se observa que con una desviación inicial en el eje de rotación de +15 grados, el sistema de balanceo automático, lleva a la plataforma al umbral de 0.5 grados en aproximadamente 17 minutos. Esto es debido a un retraso de 7 segundos entre cada ciclo de control por eje. En la figura 5.5 se aprecia como se lleva a cabo el balanceo automático en el eje de cabeceo, en este caso se partió de una desviación inicial de +13° grados.

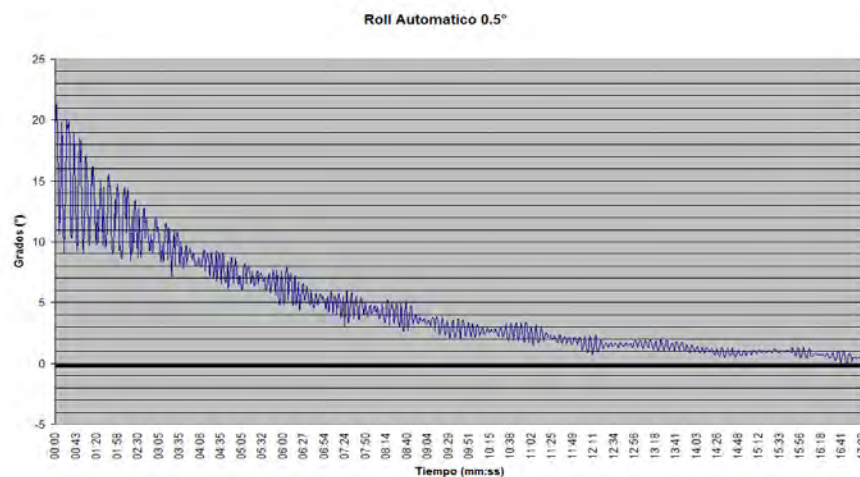


Figura 5.4.- Prueba de balanceo automático en el eje de rotación con un umbral de  $\pm 0.5^\circ$ .

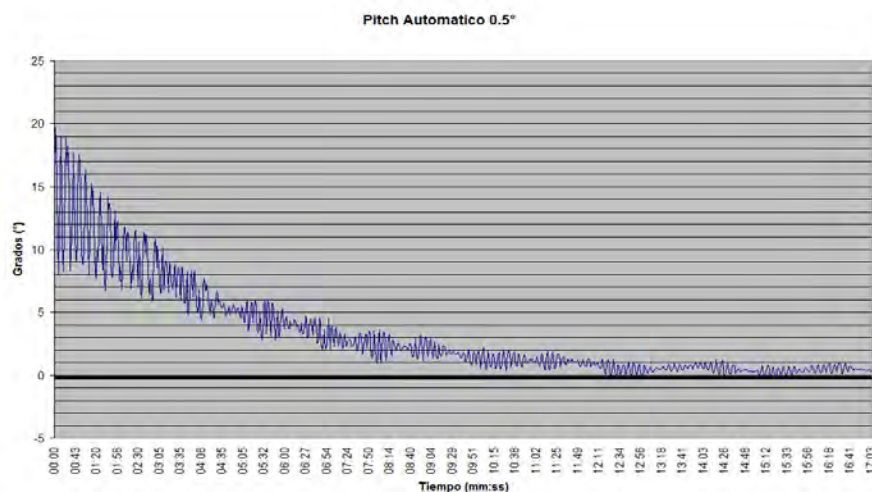


Figura 5.5.- Balanceo en el eje de cabeceo, a partir de una desviación inicial de +13°.

En resumen, comenzando con una desviación de +15 grados en el eje de rotación y de +13 en el eje de cabeceo, la desviación de la plataforma fue llevada hasta el límite de error, en este caso particular de  $\pm 0.5^\circ$ . Las gráficas fueron construidas a partir de 1,211 muestras, lo que significa aproximadamente 17 minutos de iteraciones, hasta que el sistema alcanzó un estado estable.

### 5.3. Pruebas de oscilación simple.

Una vez que se ha llevado a cabo el balanceo, se ha procedido a efectuar diferentes pruebas de oscilación simple para determinar el periodo de oscilación de la plataforma con los umbrales de  $\pm 3^\circ$ ,  $\pm 2^\circ$ ,  $\pm 1^\circ$  y  $\pm 0.5^\circ$ .

#### Pruebas de oscilación con umbral de $\pm 3$ grados

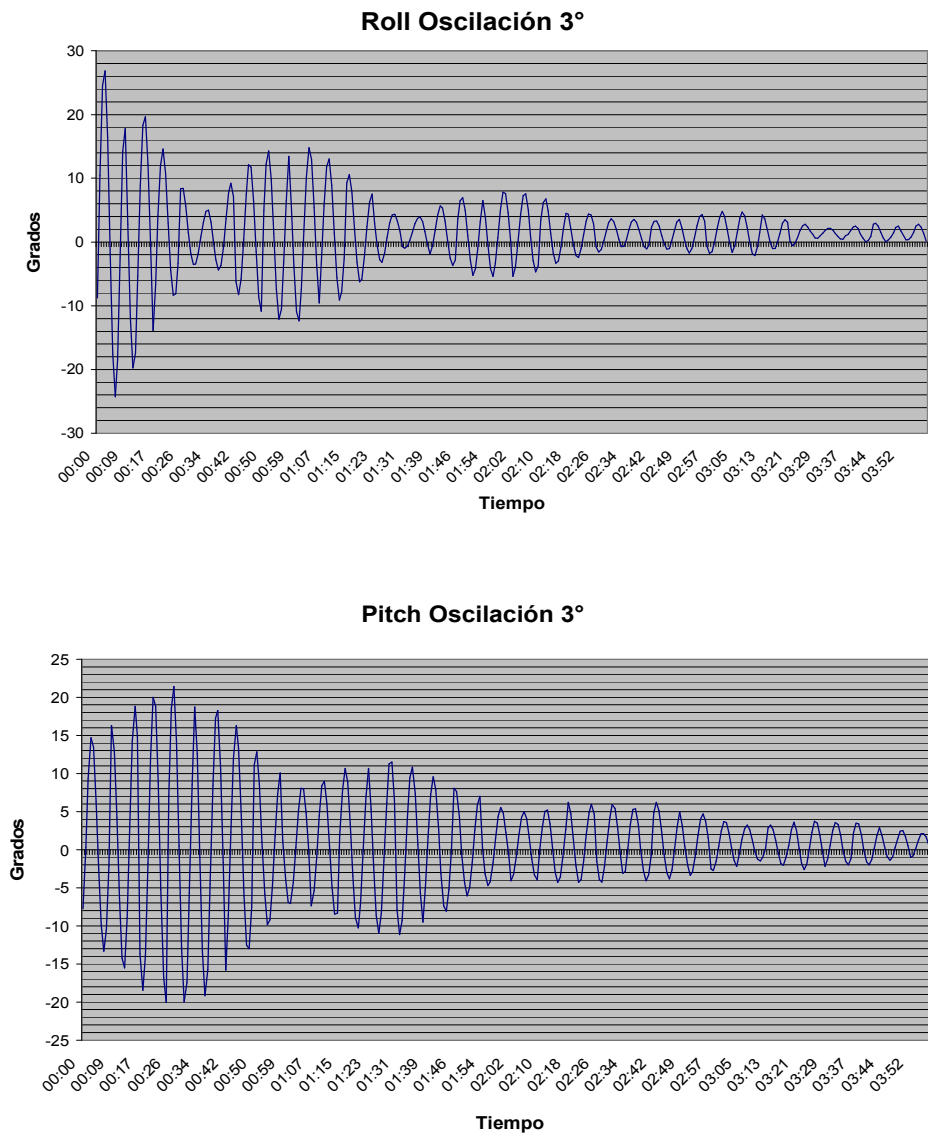
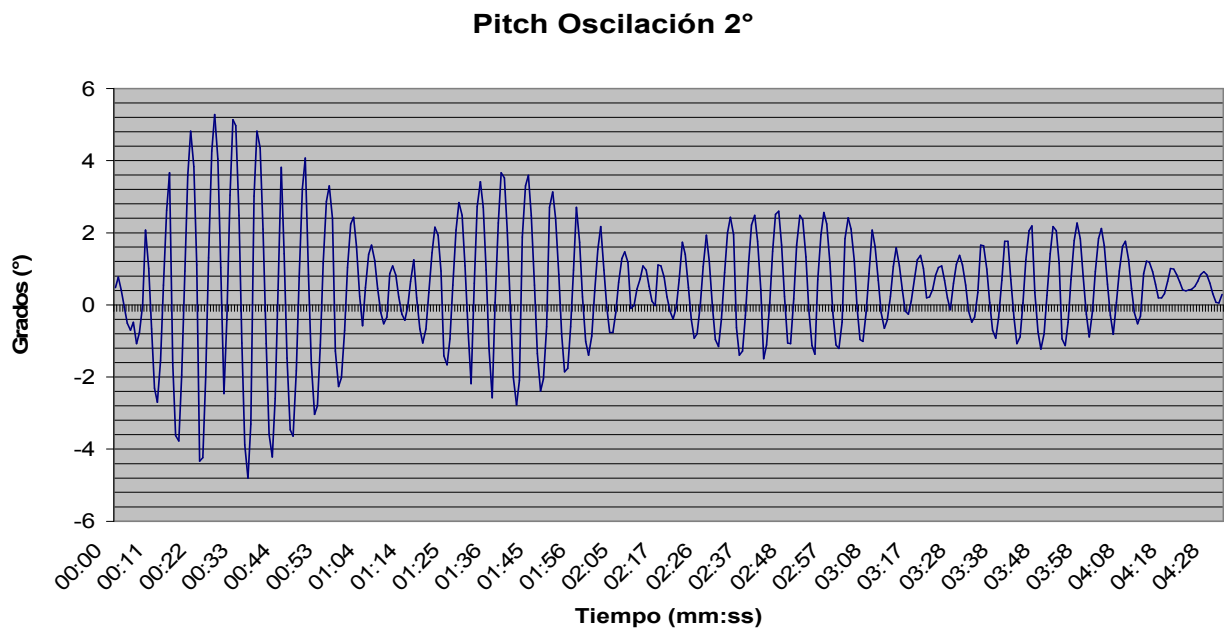
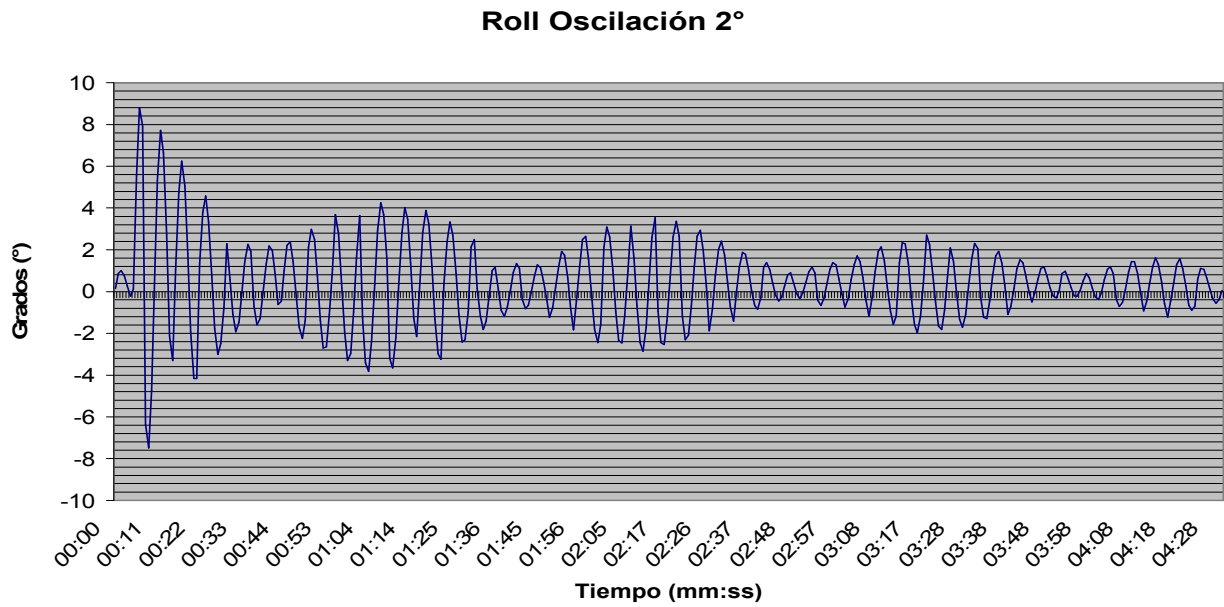


Figura 5.6.- Curvas de oscilación obtenidas a partir de la aplicación de una desviación intencional de 15 grados en el eje de cabeceo.

**Pruebas de oscilación con umbral de  $\pm 2$  grados**



*Figura 5.7.- Curvas de oscilación obtenidas a partir de la aplicación de una desviación intencional de 9 grados en el eje de rotación.*

**Pruebas de oscilación con umbral de  $\pm 1$  grado**

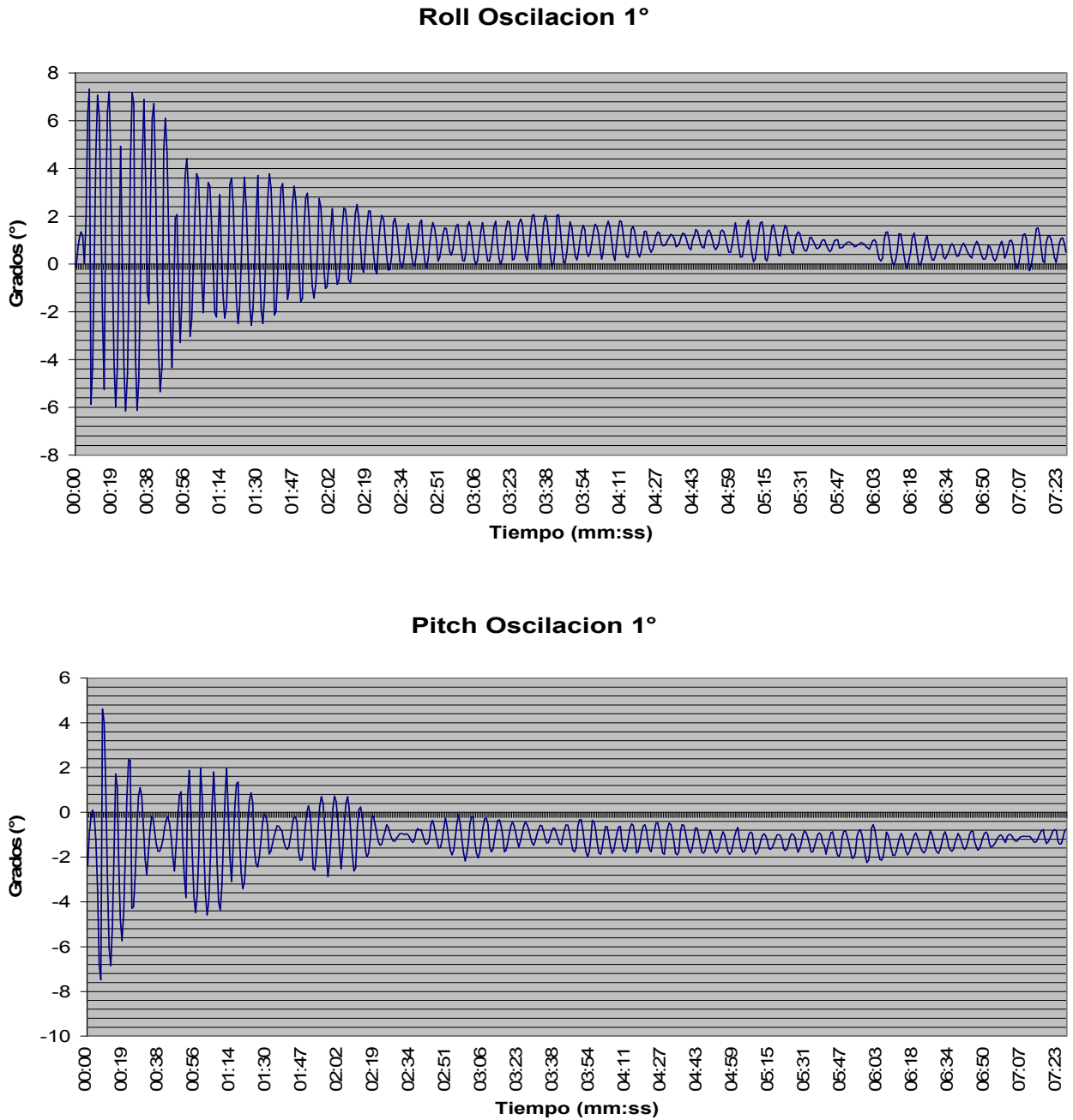


Figura 5.8.- Curvas de oscilación obtenidas a partir de la aplicación de una desviación intencional de -9 grados en el eje de cabeceo.



**Pruebas de oscilación simple con umbral de 0.5 grados**

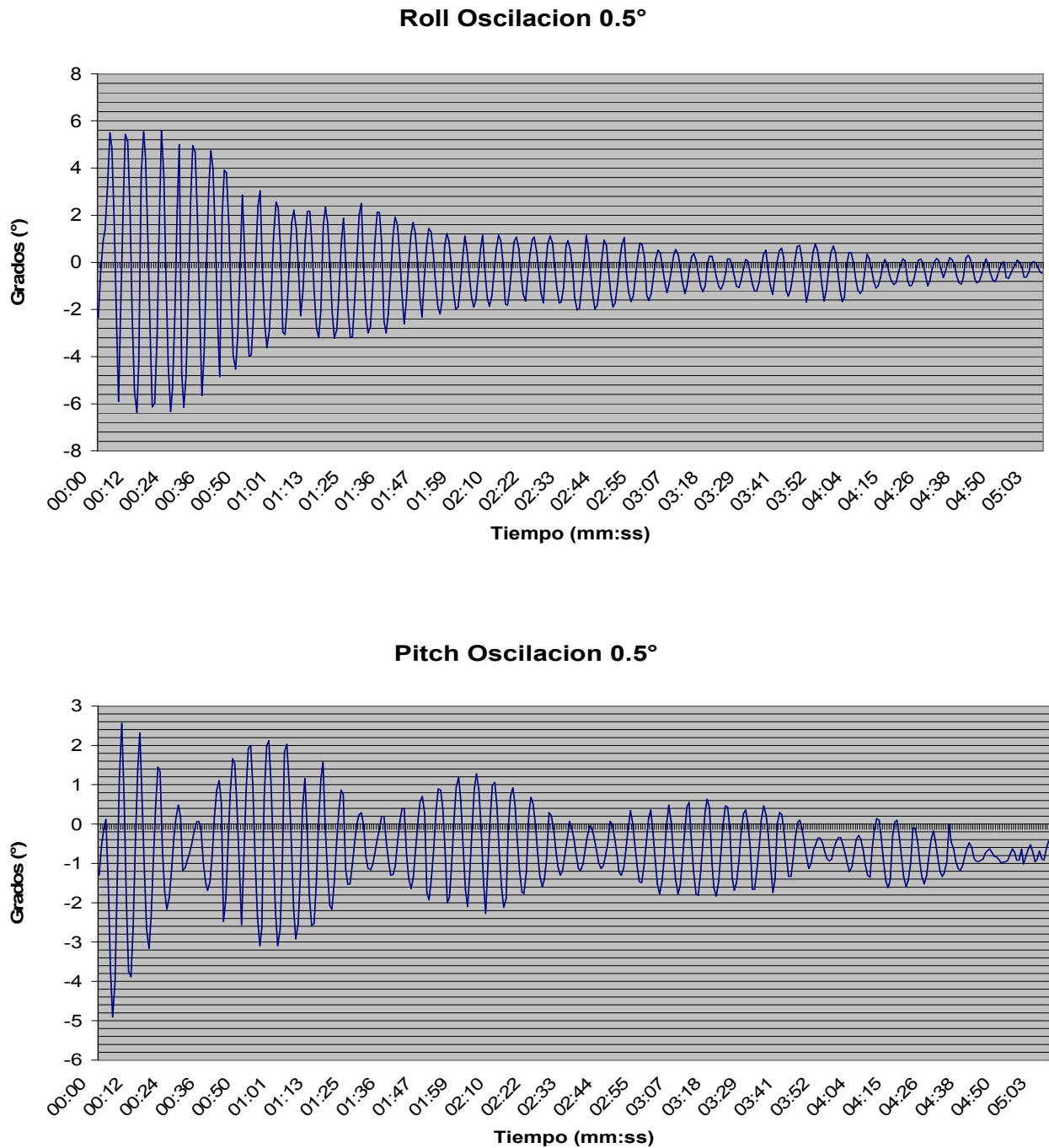


Figura 5.9.- Curvas de oscilación obtenidas a partir de la aplicación de una desviación intencional de -6 grados en el eje de cabeceo.

La experimentación con la oscilación simple, permite generar una gráfica como la que se muestra en la figura 5.10, donde se comprueba que el periodo de oscilación de la plataforma aumenta conforme el par residual disminuye. Además nos indica la tendencia y el límite al que debe llegar la plataforma para encontrarse en el óptimo de balanceo.

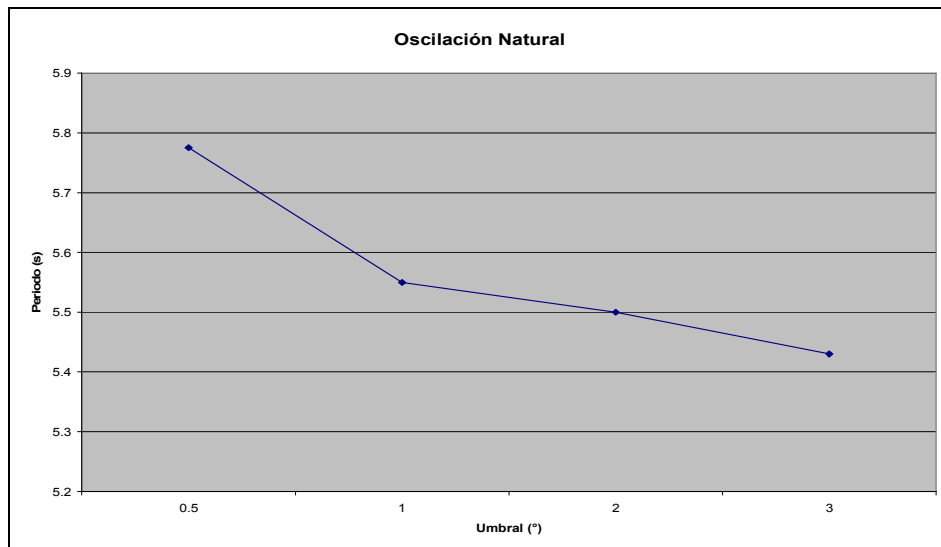


Figura 5.10.- Diferentes umbrales manejados durante el balanceo automático vs. el periodo de oscilación de la plataforma.

#### 5.4. Control de orientación con bobinas magnéticas

Una vez balanceada la plataforma se procede a hacer pruebas de control de orientación con bobinas magnéticas y ruedas inerciales. La utilización de bobinas magnéticas en órbita Terrestre, tiene la doble función de generar pares externos al satélite para modificar su orientación, y como sistema de desaturación para las ruedas inerciales.

##### 5.4.1. Encendido de bobinas para control de orientación.

Este programa fue desarrollado para activar las bobinas magnéticas y generar un dipolo que interactúa con las componentes del campo magnético terrestre, para producir un par externo que modifica la orientación de la plataforma. Al generarse el dipolo en la bobina, ésta trata de alinearse con la componente del campo magnético, de manera similar a lo que ocurre con la aguja imantada de una brújula.

Dadas las restricciones de disponibilidad de un campo magnético cambiante, como el que existe a lo largo de una trayectoria orbital, cualquiera que ésta sea, el único tipo de prueba que pudimos efectuar fue el encender una bobina y cambiar su polaridad para que se alineara con el campo magnético circundante en el Laboratorio.

En este aspecto estuvo muy limitada la experimentación debido a que no se cuenta con un simulador de campo magnético terrestre, sin embargo, los resultados de esta prueba mostraron que la plataforma sí reacciona con el control, y por consiguiente se logró ver un cambio en la orientación del eje del azimut al interactuar con el campo magnético terrestre. La figura 5.11 muestra el diagrama de flujo del procedimiento desarrollado para encender y cambiar la polaridad de las bobinas magnéticas, instalado en el microcontrolador Rabbit.

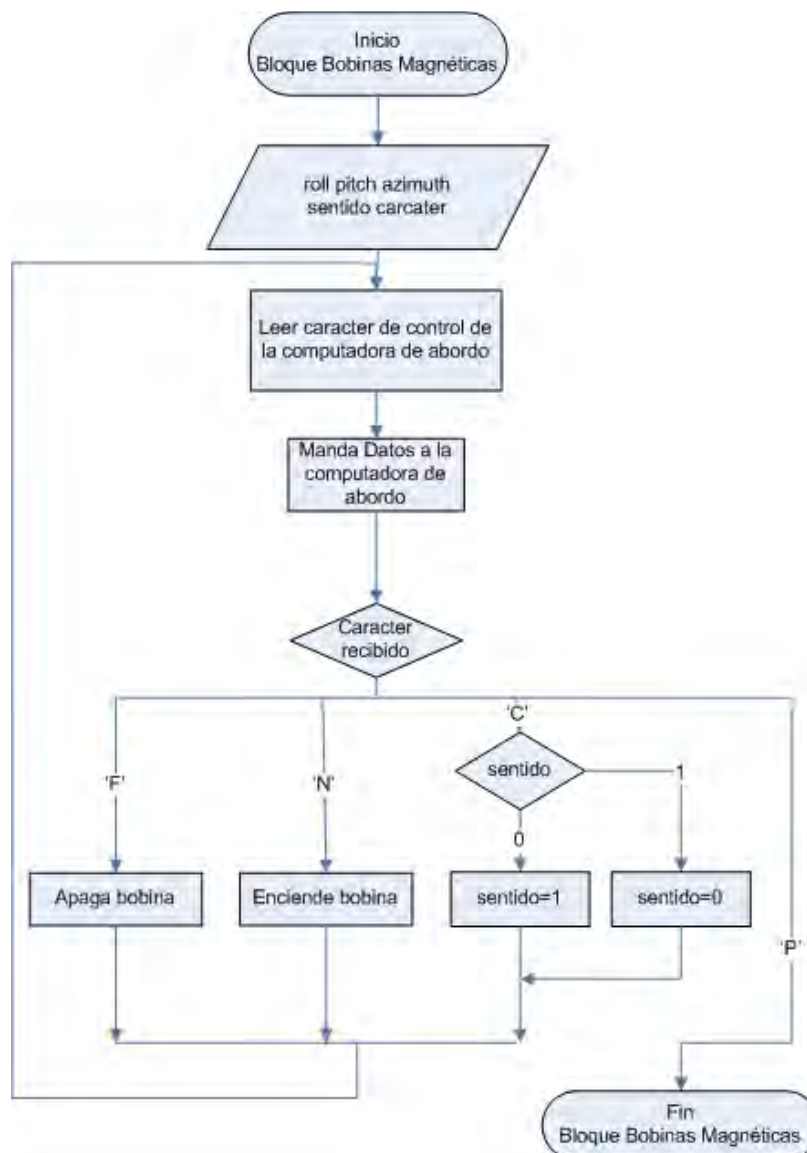


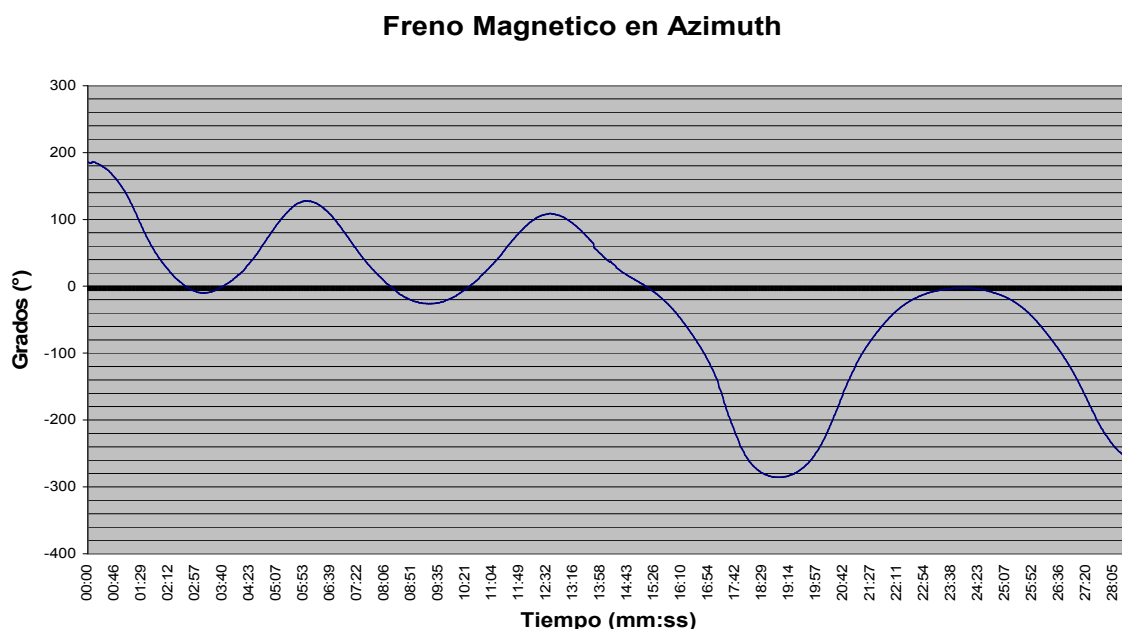
Figura 5.11.- Diagrama de flujo del programa de encendido, apagado y cambio de polaridad de una bobina magnética.

Las bobinas magnéticas se encenderán, cuando las ruedas inerciales se encuentren en un estado de saturación, su acción permanecerá hasta que las ruedas regresen a su operación nominal. En esencia, la misma rutina de control de encendido y cambio de polaridad sirve para reorientar al satélite cuando se aplica una fuerza de frenado en alguna rueda inercial del satélite.

El código correspondiente a la programación del microcontrolador Rabbit para el caso de bobinas magnéticas se encuentra en el apéndice B. Esta rutina se usa para el encendido y cambio de polaridad de las bobinas magnéticas; para maniobras de control y de reorientación ante la aplicación de un freno o cuando el momentum acumulado en las ruedas inerciales lleva a éstas a una zona cercana a la saturación.

#### 5.4.1.1. Curvas de respuesta

En la [figura 5.12](#) se muestra una maniobra de orientación sobre el eje de guiñada o azimut, ésta se inició con la plataforma en una dirección de  $185.7^\circ$  en su azimut. Al aplicar la corriente a la bobina la plataforma comenzó a girar hasta detenerse en  $-10.2^\circ$  alineándose con el norte del dipolo magnético terrestre; después de alcanzar este valor la plataforma se mantuvo cambiando su posición. Después se invirtió la polaridad, lo que significó un cambio en el sentido de giro de la plataforma; el sistema reaccionó alcanzando los  $-180^\circ$  que es donde encontraría la alineación con el sur del dipolo magnético terrestre, sin embargo la inercia que presentó la plataforma hizo que ésta continuara girando hasta alcanzar los  $-285.6^\circ$ , presentando un nuevo patrón de oscilación. Esto quiere decir que la plataforma tiende a alinearse con el dipolo magnético terrestre. Pero debido a que estos pares son relativamente poco influyentes en nuestro sistema, esta se mantenía oscilando. Esta prueba permitió apreciar que la plataforma tiende a alinearse con el campo magnético terrestre.



*Figura 5.12.- Prueba para demostrar la factibilidad de estabilizar la plataforma alrededor del eje Z. Aproximadamente a los 15 minutos se hizo el cambio de polaridad.*

### 5.5. Control de orientación con ruedas inerciales.

El control por medio de las ruedas inerciales se realiza de dos formas diferentes, una de ellas se aplica para los ejes de cabeceo y alabeo, y la otra para el eje de azimut. En el primer caso, correspondiente a los ejes X,Y, la brújula envía 30 datos a la computadora de abordo y con el último de ellos, el microcontrolador decide cual será el eje a controlar, y dependiendo del valor de la desviación y del signo de la misma, se manda a encender la rueda correspondiente a un sentido u otro. En el momento en que la brújula electrónica indica que el eje ha alcanzado el valor de cero, se termina el proceso de orientación y se manda un mensaje a la estación terrena, donde se indica que el proceso ya terminó. En la [figura 5.13 a\)](#) se muestra la colocación de las ruedas en la plataforma y en la [figura 5.13 b\)](#) un detalle de las mismas. En el caso del eje de azimut se realiza casi la misma operación salvo que en este modo, dependiendo del valor de desviación, se mueve más tiempo o menos tiempo la rueda de dicho eje, y después cuando llega la plataforma dentro del umbral señalado, el control deja de trabajar hasta que exista una nueva desviación.



*Figura 5.13.- a) Juego de ruedas inerciales en la plataforma y b) detalle de la rueda Inercial en el eje de Alabeo.*

En la [figura 5.14](#) se muestra el diagrama de flujo del procedimiento utilizado para lograr un correcto apuntamiento de la plataforma, incrementando o disminuyendo el tiempo que gira la rueda correspondiente al eje que presenta la desviación. En el apéndice C, se presenta el listado del programa de control de orientación con ruedas inerciales.

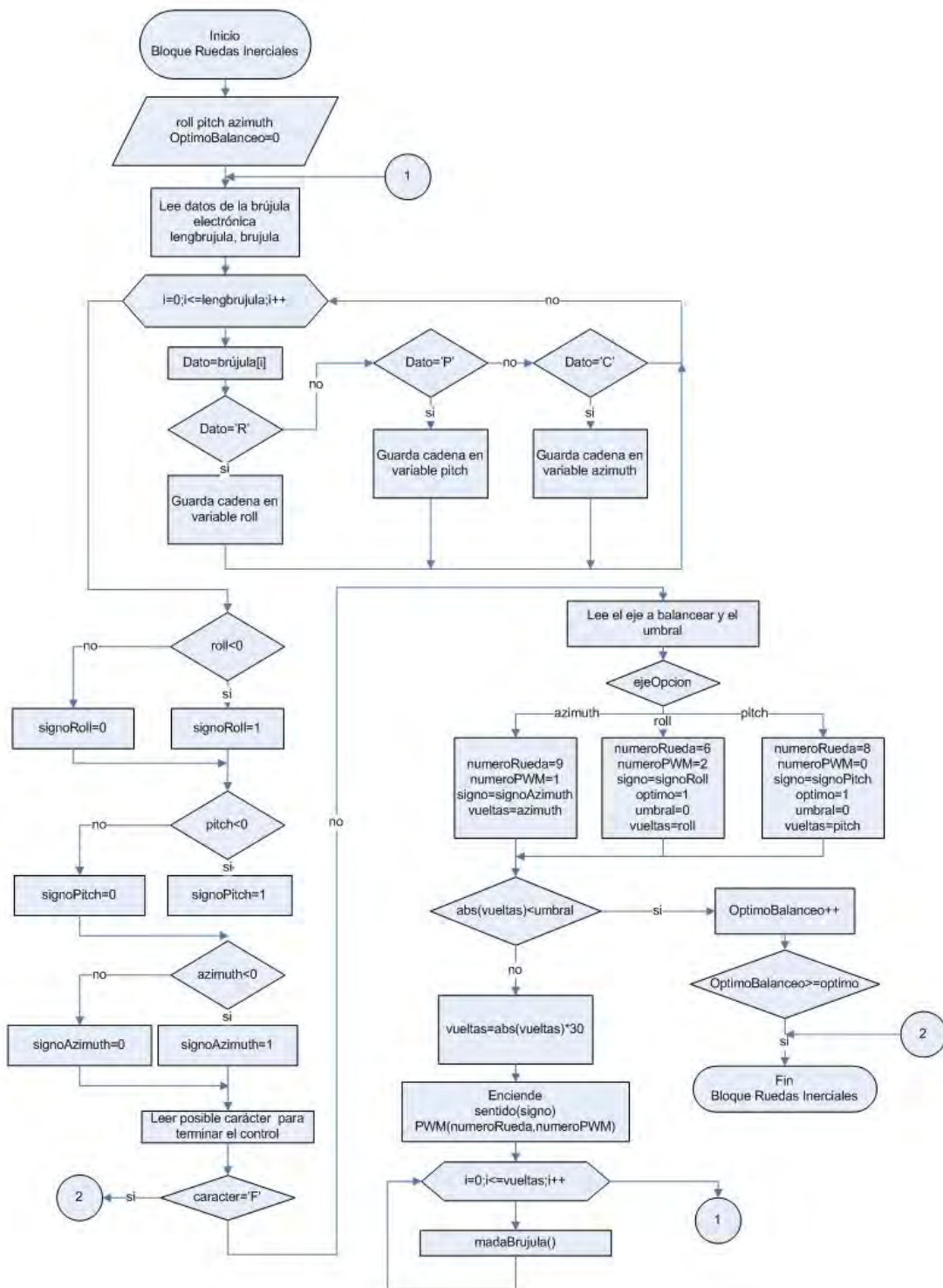
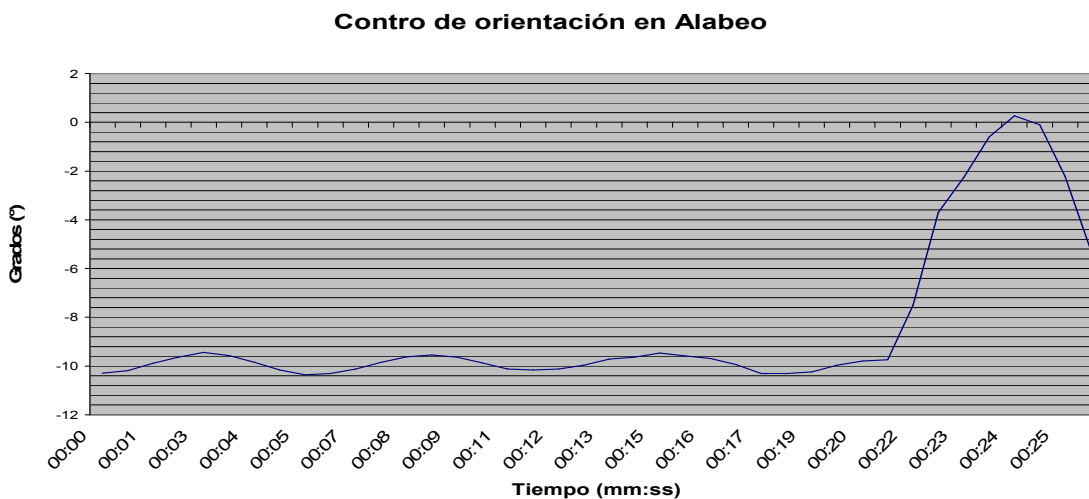
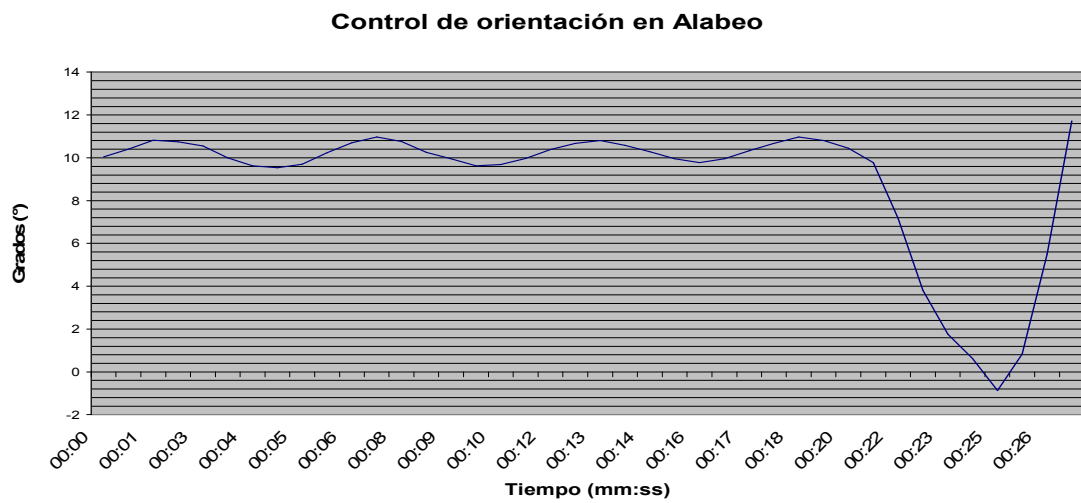


Figura 5.14.- Diagrama de flujo del algoritmo para controlar la orientación de un eje, por medio de una rueda inercial.

**5.5.1. Curvas de respuesta**

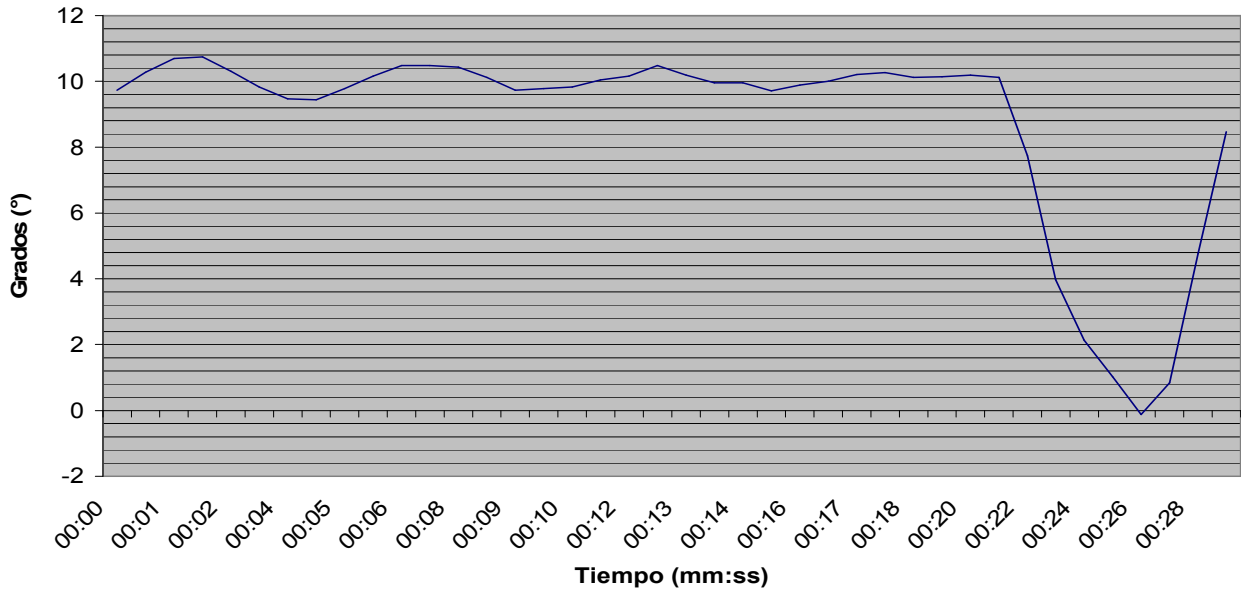
Cuando se llega a la orientación deseada en el caso de los ejes X,Y el sistema automático termina sus ciclos de control, sin embargo, los pares gravitacionales regresan a la plataforma a su posición inicial causando una oscilación. Esto se puede apreciar claramente en las figuras 5.15 y 5.16, donde se presentan dos curvas de respuesta en los ejes de rotación y cabeceo, respectivamente, de una maniobra de reorientación hacia la posición horizontal. El apuntamiento estable se alcanza en unos cuantos segundos, lo que demuestra que las ruedas inerciales pueden llevar a cabo acciones de control y apuntamiento de una manera muy rápida y efectiva.

En el eje Z esta condición de oscilación no ocurre y el sistema de control permanece funcionando en un procedimiento cíclico, alcanzando el apuntamiento establecido.



*Figura 5.15.- Curvas de respuesta en el eje de alabeo ante una desviación de +10 y -10 grados respectivamente.*

**Control de orientación en Cabeceo 3°**



**Control de orientación en Cabeceo 3°**

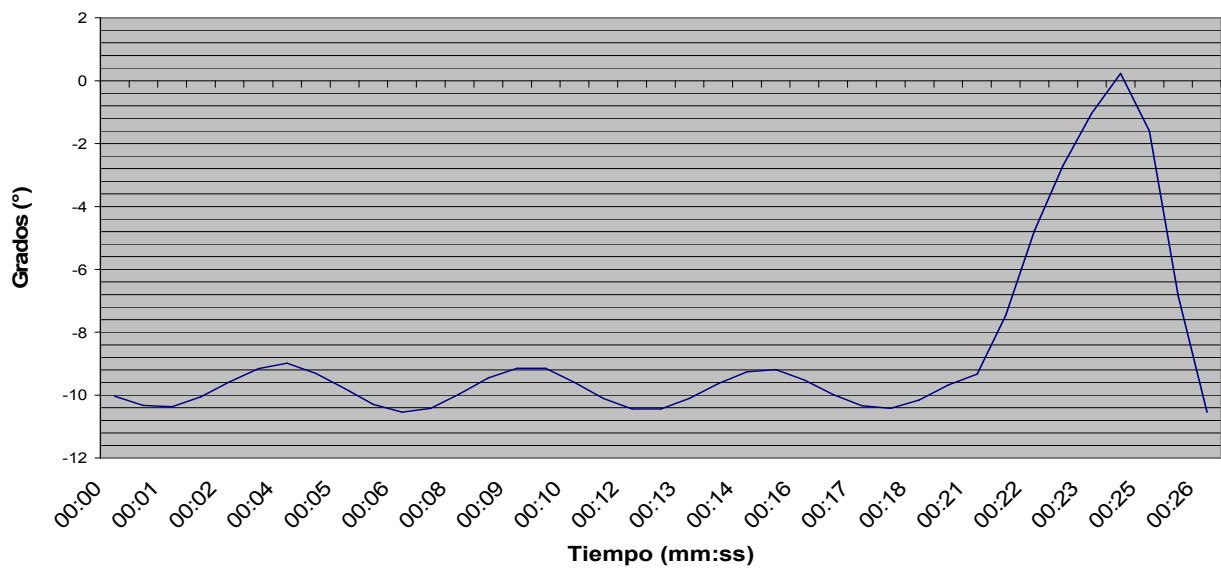


Figura 5.16.- Curvas de respuesta en el eje de cabeceo ante una desviación positiva y negativa respectivamente.



Las respuestas en los ejes de rotación y cabeceo son muy similares, como se hace evidente en las gráficas anteriores. Una vez alcanzada la orientación deseada, la plataforma sufre un par gravitacional restaurador que inicia una oscilación, por eso su movimiento hacia la posición inicial de desviación.

En la figura 5.17 se observa la respuesta de la plataforma de simulación en una prueba en el eje Z, que tiene una desviación inicial. Estas son curvas típicas del comportamiento de la plataforma cuando ésta es orientada hacia una dirección específica, en este caso una desviación de 0°. La velocidad de respuesta promedio de la plataforma es de 3° por cada segundo. Se puede corregir aún más rápido dicha orientación, sin embargo, esto causaría oscilaciones no deseadas en la plataforma, por tal motivo no se aumento la potencia ni la velocidad de la rueda al máximo.

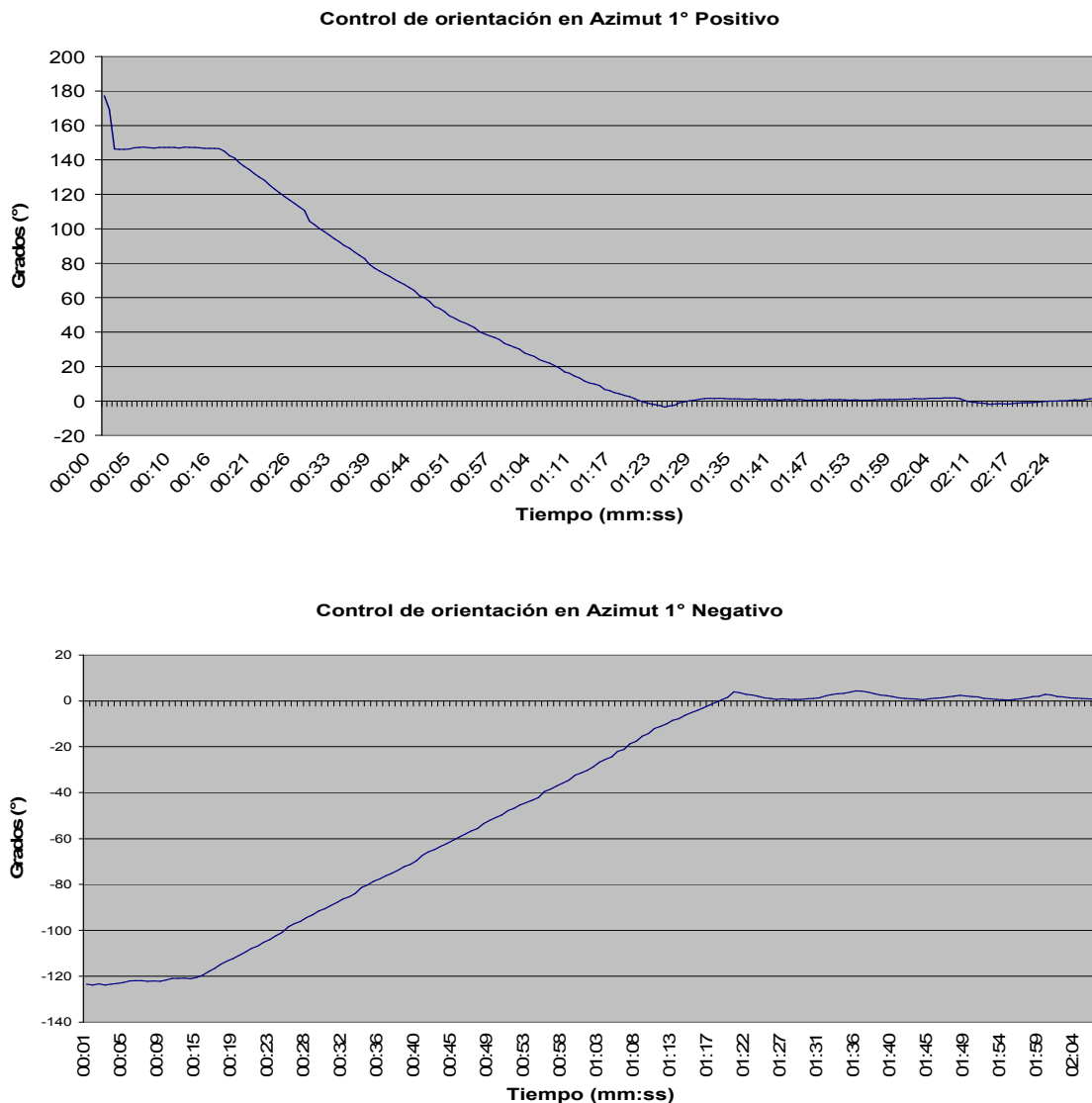


Figura 5.17.- Respuesta de la plataforma con el algoritmo de orientación, en el eje Z, que la dirige hacia una posición de cero grados ante una desviación de +143° y -120° respectivamente.

Finalmente, [en la figura 5.18](#) se muestra una fotografía donde se aprecia la localización de la rueda inercial en el eje Z.

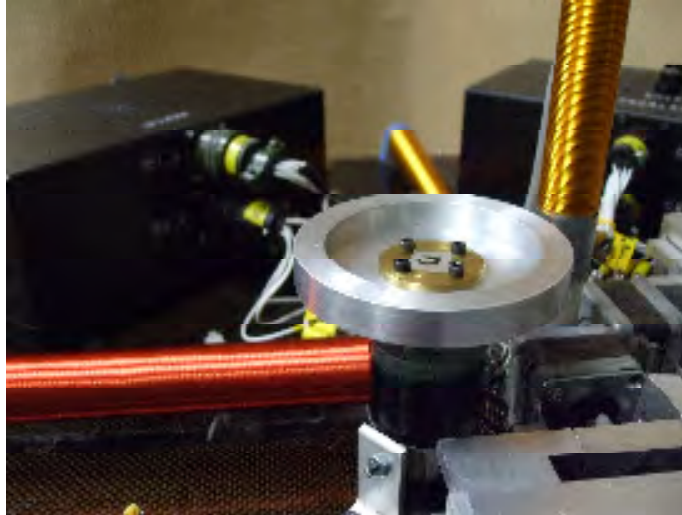


Figura 5.18.- Localización de la rueda sobre el eje Z, en la plataforma de simulación.

## CONCLUSIONES

Del trabajo realizado, podemos desprender las siguientes conclusiones, de manera puntual:

- Se desarrolló una nueva versión de un sistema de simulación para pruebas de algoritmos de orientación y control, para satélites pequeños (menores a 50 kg).
- El uso de la tarjeta LP3500 con su microcontrolador Rabbit 3000 es muy adecuado para la plataforma, debido a que sus especificaciones resultaron suficientes para los requerimientos del sistema, además de que es compacta y consume poca energía.
- Se desarrollaron las interfaces de potencia para manejar los motores que utilizan las masas deslizantes y las ruedas inerciales, logrando un adecuado funcionamiento de ambos sistemas.
- Se llevaron a cabo pruebas de funcionamiento con bobinas magnéticas y ruedas inerciales en una plataforma de simulación de un medio sin fricción, para su implementación en satélites pequeños.
- Se diseñaron e implementaron nuevas bobinas con núcleo de ferrita para obtener un mejor dipolo magnético de la plataforma, con un consumo bajo de energía.
- Las ruedas inerciales pueden proveer pares importantes para orientar un satélite pequeño, aunque tienen la desventaja de consumir mucha energía.
- Se mejora la confiabilidad del sistema comparado con sus antecesores.
- La comunicación entre una estación terrena y la plataforma es bidireccional.
- La plataforma de simulación es totalmente autónoma.
- Las interfaces de potencia son robustas y confiables.
- Se programó en un lenguaje de alto nivel, el cual fue un cambio radical para esta plataforma.
- Se programó una nueva Interfaz 100% amigable y de fácil configuración.
- Diseño y estética apreciables en todos los elementos del sistema.
- El trabajo en equipo siempre da mejores resultados.

## A

**Código de programación para llevar a cabo el balanceo automático de la plataforma de simulación utilizando dos masas deslizantes.**

```

case 97:          /*CASO DE MASAS DESLIZANTES*/
{
  while ((umbralOpcion = serEgetc()) == -1);
  switch(umbralOpcion) //Cambia la variable umbral dependiendo del dato recibido desde la PC.
  {
    case 48:
    {
      umbral=3;
    }
    break;
    case 49:
    {
      umbral=2;
    }
    break;
    case 50:
    {
      umbral=1;
    }
    break;
    case 51:
    {
      umbral=0.5;
    }
    break;
    default:
    {
      umbral=0;
      optimoBalanceo=0;
      comandoPC=0;
      umbral=0;
      comandoPC=0;
      serEwrFlush(); //limpia el buffer de transmisión
      serErdFlush(); // limpia el buffer de entrada o recepción
      goto inicio;
    }
    break;
  }
}
if(umbral!=0 && comandoPC!=0)
{
  while(1)
  {
    serBwrFlush(); //limpia el buffer de transmisión
    serBrdFlush(); // limpia el buffer de recepción
    mandaBrujula();
    getBrujula(); //actualiza la variable lenbrujuja y brujula
    for(i=0;i<=lenBrujula;i++) //Obtiene R roll(alabeo) y P pitch(cabeceo)
    {
      if(brujula[i]=='R')
      {
        posRoll=i;
      }
      if(brujula[i]=='P')
      {
        posPitch=i;
      }
      if(brujula[i]=='T')
      {
        posTemperatura=i;
      }
    }
  }
}

```

```

    }
}
j=0;

lenRoll=posPitch-posRoll-1;
for(i=posRoll+1;i<posPitch;i++) //Obtiene el valor de alabeo
{
    rollValor[j]=brujula[i];
    j++;
}
for(;j<6;j++)
{
    rollValor[j]='0';
}
roll = strtod(rollValor, &endptr);
j=0;

lenPitch=posTemperatura-posPitch-1;
for(i=posPitch+1;i<posTemperatura;i++) //Obtiene el valor de cabeceo
{
    pitchValor[j]=brujula[i];
    j++;
}
for(;j<6;j++)
{
    pitchValor[j]='0';
}
pitch = strtod(pitchValor, &endptr);

if(rollValor[0]=='-') //Compara si el valor del alabeo es negativo
{
    vueltasRoll=rollValor[1]-48; //Cambia el valor int. de ASCII al valor real
    comandoPicRoll='b';
    if(lenRoll==6)
    {
        vueltasRoll=vueltasRoll*10;
        vueltasRoll=vueltasRoll+rollValor[2]-48;
    }
}
else //En caso de ser positivo el valor de alabeo
{
    vueltasRoll=rollValor[0]-48;
    comandoPicRoll='a';
    if(lenRoll==5)
    {
        vueltasRoll=vueltasRoll*10;
        vueltasRoll=vueltasRoll+rollValor[1]-48;
    }
}
if(pitchValor[0]=='-') //Compara si el valor del cabeceo es negativo
{
    vueltasPitch=pitchValor[1]-48;
    comandoPicPitch='c';
    if(lenPitch==6)
    {
        vueltasPitch=vueltasPitch*10;
        vueltasPitch=vueltasPitch+pitchValor[2]-48;
    }
}
else //En caso de ser positivo el valor de alabeo
{
    vueltasPitch=pitchValor[0]-48;
    comandoPicPitch='d';
    if(lenPitch==5)
    {
        vueltasPitch=vueltasPitch*10;
        vueltasPitch=vueltasPitch+pitchValor[1]-48;
    }
}
if(pitchOroll==0) //Decide el turno del eje de cabeceo
{
    vueltas=vueltasPitch;
    comandoPic=comandoPicPitch;
    valor=pitch;
    pitchOroll=1;
}
else //Decide el turno del eje de alabeo

```

```

{
    vueltas=vueltasRoll;
    comandoPic=comandoPicRoll;
    valor=roll;
    pitchOroll=0;
}
if(abs(valor)<umbral)    //Comprueba si el valor del cabeceo o alabeo están dentro del umbral
{
    vueltas=0;
    optimoBalanceo++;    //aumenta el contador para verificar si esta balanceado
}
else                    //de lo contrario inicializa el contador
{
    optimoBalanceo=0;
}
if(optimoBalanceo<1)    //Si esta fuera del umbral realiza manda los comandos al PIC
{
    for(i=0;i<vueltas*30;i++)
    {
        serCputc(comandoPic);
        waitfor(DelayMs(10));
    }
}
if(optimoBalanceo>=10) //si lleva 10 resultados óptimos dentro del umbral sale del control
{
    for(i=0;i<=50;i++)
    {
        serEputc(36);
        waitfor(DelayMs(5));
        serEputc(76);
        waitfor(DelayMs(5));
        serEputc(13);
        waitfor(DelayMs(30));
    }
    goto inicio;
}
for(j=0;j<=7;j++)      //Manda datos de la brújula a la computadora de abordo
{
    serBwrFlush();      //limpia el buffer de transmisión
    serBrdFlush();      // limpia el buffer de entrada o recepción
    mandaBrujula();     //Manda los datos de la Brújula a la PC
}
}
}
break;

```

## B

**Código de programación para encender y cambiar de polaridad las bobinas magnéticas.**

```

case 99:          /*CASO DE BOBINAS MAGNÉTICAS*/
{
waitfor(DelayMs(100));
while ((ejeOpcion = serEgetc()) == -1);
switch(ejeOpcion) //Cambia la variable umbral dependiendo del dato recibido desde la PC.
{
case 48:
{
bobinaPot=1;
bobinaSen=0;
}
break;
case 49:
{
bobinaPot=2;
bobinaSen=3;
}
break;
case 50:
{
bobinaPot=4;
bobinaSen=5;
}
break;
default:
{
umbral=0;
optimoBalanceo=0;
comandoPC=0;
umbral=0;
comandoPC=0;
serEwrFlush(); // limpia el buffer de transmisión puerto E
serErdFlush(); // limpia el buffer de recepción puerto E
goto inicio;
}
break;
}
}
sentidoBobinas=0;
while(1)
{
serBwrFlush(); // limpia el buffer de transmisión puerto B
serBrdFlush(); // limpia el buffer de entrada o recepción puerto B
mandaBrujula(); // actualiza la variable lenbrujula y brujula
controlPCTemp=serEgetc(); // toma el caracter enviado por la computadora de abord

if(controlPCTemp=='F' || controlPC=="F") //caso 'F' Apaga la bobina
{
controlPC="F";
digOut(bobinaPot,0); // Salida SINKING de potencia 1=ON,0=OFF
digOut(bobinaSen,sentidoBobinas); // Salida SINKING de sentido
}
if(controlPCTemp=='N' || controlPC=="N") //caso 'N' Enciende la bobina
{
controlPC="N";
digOut(bobinaPot,1); // Salida SINKING de potencia 1=ON, 0=OFF
digOut(bobinaSen,sentidoBobinas); // Salida SINKING de Sentido
}
if(controlPCTemp=='C') // caso 'C' Cambia sentido de polarización

```

```
{
  if(sentidoBobinas==0)
  {
    sentidoBobinas=1;
  }
  else
  {
    sentidoBobinas=0;
  }
}
if(controlPCTemp=='P')          // caso 'P' Termina el control
{
  controlPCTemp=0;
  controlPC=0;
  digOut(bobinaSen,0);
  optimoBalanceo=0;
  comandoPC=0;
  umbral=0;
  comandoPC=0;
  serEwrFlush();               // limpia el buffer de transmision puerto E
  serErdFlush();               // limpia el buffer derecepción puerto E
  goto inicio;
}
}
break;
```



## C

**Código de programación para re-orientar la plataforma de simulación con las ruedas inerciales.**

```

case 98:                /*CASO DE RUEDAS INERCIALES*/
{
    pitchOrolloazimuth=0;
    pwmOut(0, 0);
    pwmOut(1, 0);
    pwmOut(2, 0);
    waitfor(DelayMs(30));
    while ((umbralOpcion = serEgetc()) == -1);
    switch(umbralOpcion) //Cambia la variable umbral dependiendo del dato recibido.
    {
        case 48:
        {
            umbral=3;
        }
        break;
        case 49:
        {
            umbral=2;
        }
        break;
        case 50:
        {
            umbral=1;
        }
        break;
        case 51:
        {
            umbral=0.5;
        }
        break;
        default:
        {
            umbral=0;
            optimoBalanceo=0;
            comandoPC=0;
            umbral=0;
            comandoPC=0;
            serEwrFlush(); // limpia el buffer de transmision del puerto E
            serErdFlush(); // limpia el buffer de recepción del puerto E
            pitchOrolloazimuth=0;
            goto inicio;
        }
        break;
    }
}
if(umbral!=0 && comandoPC!=0)
{
    waitfor(DelayMs(30));
    while ((ejeOpcion = serEgetc()) == -1);
    ejeOpcion=ejeOpcion-48;
    if(ejeOpcion!=0 && ejeOpcion!=1 && ejeOpcion!=2)
    {
        goto inicio;
    }
    for(i=0;i<=30;i++)
    {
        mandaBrujula();
    }
}

```

```

while(1)
{
    pwmOut(0, 0);
    pwmOut(1, 0);
    pwmOut(2, 0);
    serBwrFlush();           //limpia el buffer de transmision del puerto B
    serBrdFlush();          // limpia el buffer de recepci3n del puerto B
    getBrujula();           //actualiza la variable lenbrujula y brujula
    for(i=0;i<=lenBrujula;i++) //Obtiene R roll (alabeo), P pitch (cabeceo) y C azimuth (azimut)
    {
        if(brujula[i]=='R')
        {
            posRoll=i;
        }
        if(brujula[i]=='P')
        {
            posPitch=i;
        }
        if(brujula[i]=='T')
        {
            posTemperatura=i;
        }
        if(brujula[i]=='C')
        {
            posAzimuth=i;
        }
        if(brujula[i]=='*')
        {
            posFinal=i;
        }
    }
    j=0;
    lenRoll=posPitch-posRoll-1;
    for(i=posRoll+1;i<posPitch;i++) //Separa en la variable rollValor el valor de alabeo
    {
        rollValor[j]=brujula[i];
        j++;
    }
    roll = strtod(rollValor, &endptr);
    j=0;
    lenPitch=posTemperatura-posPitch-1;
    for(i=posPitch+1;i<posTemperatura;i++) //Separa en la variable pitchValor el valor de
cabeceo
    {
        pitchValor[j]=brujula[i];
        j++;
    }
    pitch = strtod(pitchValor, &endptr);
    j=0;
    lenAzimuth=posFinal-posAzimuth-1;
    for(i=posAzimuth+1;i<posFinal;i++) //Separa en la variable azimuthValor el valor de
azimut
    {
        azimuthValor[j]=brujula[i];
        j++;
    }
    azimuth = strtod(azimuthValor, &endptr);
    //Obteniendo el valor de ROLL
    if(rollValor[0]=='-') //Compara si el valor del alabeo es negativo
    {
        vueltasRoll=rollValor[1]-48; //Cambia el valor int de ascii al valor de alabeo
real
        signoRuedaRoll=1;
        if(lenRoll==6)
        {
            vueltasRoll=vueltasRoll*10;
            vueltasRoll=vueltasRoll+rollValor[2]-48;
        }
    }
    else
    {
        vueltasRoll=rollValor[0]-48;
        signoRuedaRoll=0;
        if(lenRoll==5)
        {
            vueltasRoll=vueltasRoll*10;
            vueltasRoll=vueltasRoll+rollValor[1]-48;
        }
    }
}

```

```

    }
}
//Obteniendo el valor del Pitch
if(pitchValor[0]=='-') //Compara si el valor del cabeceo es negarivo
{
    vueltasPitch=pitchValor[1]-48; //Cambia el valor int de ascii al valor de cabeceo
real
    signoRuedaPitch=1;
    if(lenPitch==6)
    {
        vueltasPitch=vueltasPitch*10;
        vueltasPitch=vueltasPitch+pitchValor[2]-48;
    }
}
else
{
    vueltasPitch=pitchValor[0]-48;
    signoRuedaPitch=0;
    if(lenPitch==5)
    {
        vueltasPitch=vueltasPitch*10;
        vueltasPitch=vueltasPitch+pitchValor[1]-48;
    }
}
//Obteniendo el valor del Azimuth
if(azimuth>180) //Compara si el valor del azimuth es mayor a 180°
{
    vueltasAzimuth=(360-azimuth); //Cambia el valor int de ascii al valor de azimuth
real
    signoRuedaAzimuth=1;
}
else
{
    vueltasAzimuth=azimuth; //Cambia el valor int de ascii al valor de azimuth real
    signoRuedaAzimuth=0;
}
switch(ejeOpcion) //Segun el eje a controlar se asignan los puertos de PWM y
Sentido
{
    case 0: //Balanceo con Azimut
    {
        pwmOut(0, 0);
        pwmOut(1, 0);
        vueltas=vueltasAzimuth;
        numeroRueda=6;
        numeroPWM=2;
        signoRueda=signoRuedaAzimuth;
        optimoRuedas=100;
        valor=vueltasAzimuth;
        if(vueltas<20)
        {
            waitfor(DelayMs(30));
        }
        if(vueltas>15)
        {
            vueltas=vueltas/8;
        }
    }
    break;
    case 1: //Balanceo con Cabeceo
    {
        pwmOut(2, 0);
        pwmOut(1, 0);
        vueltas=vueltasPitch;
        numeroRueda=8;
        numeroPWM=0;
        signoRueda=signoRuedaPitch;
        valor=pitch;
        optimoRuedas=1;
        if(vueltas>9)
        {
            vueltas=vueltas;
        }
    }
    break;
    case 2: //Balanceo con Alabeo
    {

```

```

    pwmOut(0, 0);
    pwmOut(2, 0);
    vueltas=vueltasRoll;
    numeroRueda=9;
    numeroPWM=1;
    signoRueda=signoRuedaRoll;
    valor=roll;
    optimoRuedas=1;
    if(vueltas>9)
    {
        vueltas=vueltas;
    }
}
break;
default:
{
    pitchOrrollOazimuth=0;
}
break;
}
if(abs(valor)<=umbral) //comprueba el valor obtenido con el del umbral
{
    digOut(numeroRueda, signoRueda);
    pwmOut(numeroPWM, 0);
    vueltas=0;
    optimoBalanceo++;
}
else
{
    optimoBalanceo=0;
}
if(optimoBalanceo<1)
{
    digOut(numeroRueda, signoRueda);
    pwmOut(numeroPWM, 0.80); //Enciende PWM
    for(i=0;i<vueltas;i++)
    {
        serBwrFlush();
        serBrdFlush();
        mandaBrujula();
        controlPC = serEgetc(); //Lee si la estacion terrena manda a terminar el
control
    }
    if(controlPC=='F')
    {
        for(i=0;i<=50;i++)
        {
            serEputc(36);
            waitFor(DelayMs(5));
            serEputc(76);
            waitFor(DelayMs(5));
            serEputc(13);
            waitFor(DelayMs(30));
        }
        digOut(numeroRueda, signoRueda);
        pwmOut(numeroPWM, 0);
        goto inicio; //Regresa al inicio del programa
    }
}
digOut(numeroRueda, signoRueda);
pwmOut(numeroPWM, 0); //Apaga el PWM
}
else
{
    for(i=0;i<5;i++)
    {
        serBwrFlush();
        serBrdFlush();
        mandaBrujula(); //manda datos a la estación terrena
        controlPC = serEgetc();
    }
    digOut(numeroRueda, signoRueda);
    pwmOut(numeroPWM, 0); //Apaga el PWM
}
if(optimoBalanceo>=optimoRuedas) //Si llega a un estado optimo va al inicio del programa
{
    for(i=0;i<=50;i++)
    {

```

```
        serEputc(36);
        waitFor(DelayMs(5));
        serEputc(76);
        waitFor(DelayMs(5));
        serEputc(13);
        waitFor(DelayMs(30));
    }
    goto inicio;
}
controlPC = serEgetc(); //Lee si la estacion terrena manda a terminar el control
if(controlPC=='F')
{
    for(i=0;i<=50;i++)
    {
        serEputc(36);
        waitFor(DelayMs(5));
        serEputc(76);
        waitFor(DelayMs(5));
        serEputc(13);
        waitFor(DelayMs(30));
    }
    goto inicio;
}
}
}
}
break;
```

## Referencias

ADVANCED ORIENTATION SYSTEMS (2004) "Ez-Compass-3 Application Manual". Linden, NJ. USA. pp 1-12.

Contreras F. (2004). "Pruebas de control de estabilización para un satélite pequeño empleando bobinas magnéticas y ruedas inerciales". Tesis de Licenciatura. Ingeniería Electrónica. Facultad de Ingeniería, UNAM. pp. 1- 86

Juárez A. (2001) "Balanceo automático de un simulador para control de orientación de satélites, utilizando masas deslizantes". Tesis de Licenciatura. Ingeniería Mecánica. Facultad de Ingeniería, UNAM. pp. 1- 79.

Juárez G. (1999). "Utilización de bobinas magnéticas para control de orientación de satélites pequeños". Tesis de Licenciatura. Ingeniería Electrónica. Tesis de Licenciatura. Facultad de Ingeniería, UNAM. pp. 1-68.

Nápoles E. (2009). "Sensor de Sol Basado en Arreglos Lineales de Fotodiodos para Detección de Orientación de un Satélite" ". Tesis de Licenciatura, Ingeniería electrónica. Facultad de Ingeniería, UNAM. pp1-66

Prado J., Bisiacchi G., Juárez G. (1998) "Utilización de bobinas magnéticas para control de orientación de satélites pequeños". SOMI XIII Congreso Nacional de Instrumentación. Ensenada, BCN. México. Octubre 5-9. Memorias del Congreso. pp 408-412.

Prado J., Miranda V.M. (1998) "Sistema de adquisición de datos para determinar la orientación de un satélite en tres ejes". SOMI XIII Congreso Nacional de Instrumentación. Ensenada, BCN. México. Octubre 5-9. Memorias del Congreso. pp 413-417.

Prado J. (2007) "Sistema de Simulación para pruebas de algoritmos de orientación y control de satélites pequeños" Tesis de Doctorado. Programa de maestría y doctorado en Ingeniería, Facultad de Ingeniería, UNAM. México. pp1-171.

Rabbit Semiconductor (2007) "Wireless Control Application kit". Rabbit and MaxStream Modules. (5 CD's.)

Wertz J.R. (1990) "Spacecraft Attitude Determination and Control". Kluwer Academic Publishers. pp 1-860