

**DISEÑO DEL SISTEMA DE CONTROL FISICO PARA EL
POSICIONAMIENTO DE LA ANTENA DE LA ESTACIÓN TERRENA DEL
PROYECTO CUBESAT EN LA UNIVERSIDAD SURCOLOMBIANA**

DIEGO MAURICIO ANDRADE VIDARTE

LEONARDO ERNESTO NÚÑEZ ROMERO

**UNIVERSIDAD SURCOLOMBIANA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA ELECTRÓNICA
NEIVA
2007**

**DISEÑO DEL SISTEMA DE CONTROL FISICO PARA EL
POSICIONAMIENTO DE LA ANTENA DE LA ESTACIÓN TERRENA DEL
PROYECTO CUBESAT EN LA UNIVERSIDAD SURCOLOMBIANA**

DIEGO MAURICIO ANDRADE VIDARTE

LEONARDO ERNESTO NÚÑEZ ROMERO

**Trabajo presentado como requisito para optar al título de
INGENIEROS ELECTRÓNICOS**

**Director,
ING. AGUSTIN SOTO OTALORA
INGENIERO ELECTRÓNICO**

**Asesor,
ING. RAMIRO PERDOMO
INGENIERO ELECTRONICO**

**UNIVERSIDAD SURCOLOMBIANA
FACULTAD DE INGENIERÍA
PROGRAMA DE INGENIERÍA ELECTRÓNICA
NEIVA
2007**

Nota de aceptación:

Firma del presidente del jurado

Firma del jurado

Firma del jurado

Neiva, _____ 2007

A Dios quien nunca me ha desamparado y me ha llenado de fuerza y valor para cada día ser mejor.

A mis padres quienes me acompañaron en más de una travesía, mi mamá quien madrugaba a tenerme el desayuno, siempre a mi lado, mi consejera, mi aliento, y mi papá que no se acostaba hasta no ver que mis cosas estaban saliendo bien, siempre me enseñó a trabajar duro, a ponerle la cara a los problemas y vencer los obstáculos, los adoro.

A mis hermanos, con quienes siempre cuento, siempre me apoyaron, mi hermanita, la niña linda de la casa, mi consentida; mi hermano, mi ejemplo, mi orgullo, siempre a mi lado, siempre dándome ánimo, los quiero mucho.

A la niña hermosa que me ha acompañado por gran parte de este largo trecho. Esa mujer que me da ánimo, cree en mí y es la dueña de mi corazón, el amor de mi vida. A ti Vane te amo y te llevo siempre en mi corazón.

A mi compañero de tesis, aventuras y desventuras, borracheras y de más, con quien compartí gran parte de mi tiempo no solo en aulas sino extra muros, mi gran amigo, excelente compañero.

A todos mis Familiares que siempre me han acompañado, apoyado y acogido en todos los momentos de mi vida.

A mis compañeros de estudio y amigos de parranda, de tomata y gozata, con quienes pase momentos inolvidables, que espero se repitan.

Leonardo Ernesto Núñez Romero

A mi tía Nubia que me ofreció su techo, su apoyo, y compañía, aspectos que hicieron realidad el sueño de realizar mis estudios, gracias a ella puedo decir que estoy cumpliendo una meta.

A mis padres, quienes inculcaron en mí las ganas de conseguir lo que se sueña. Mi madre que con su incondicional apoyo y con esas palabras que siempre saben dar aliento me sacó de esos momentos en que pensaba todo estaba perdido, ¡a ti madre te quiero mucho! A mi padre que siempre me enseñó como hay que hacer las cosas, hacerlas bien, y en él tengo a mi mejor amigo.

A mi vitamina anímica personalizada, Tancho, mi hermano que ha sido en el tiempo que vive en Neiva un excelente amigo, una persona que cuando me ve con el ánimo por el suelo siempre resuelve el problema con un apunte de total espontaneidad.

A mi compañero de tesis, de trabajo y demás, que no solo ha sido una llave de trabajo sino un verdadero amigo que a estado ahí cuando lo necesito. Y en general a esas personas que considero con todo mi corazón mis amigos.

A toda mi familia en general, con quienes he compartido tanto, a esas personas que aprecio no solo por ser parientes sino por que son unos excelentes seres aún con sus problemas.

Por último al ser que todo lo puede, a el ser que busqué cuando solo él podía darme paz, y un respiro de alivio cuando tenia situaciones que me presionaban y que no sabia como resolver, gracias a ti "chuchito", quien ha logrado darme toda la fuerza y la voluntad de seguir adelante sin importar los obstáculos, y claro a toda su comitiva.

Diego Mauricio Andrade Vidarte

AGRADECIMIENTOS

Los autores expresan agradecimientos a:

Dios, por no desampararnos y siempre estar a nuestro lado.

Al Programa de **Ingeniería Electrónica** de la Universidad **SURCOLOMBIANA**, a todos los Profesores con quienes compartimos aula de clase y a aquellos que indirectamente nos dieron un consejo y nos extendieron su mano.

Al Ingeniero **Agustín Soto**, por el apoyo brindado como director del proyecto.

Al Ingeniero **Ramiro Perdomo**, por su contribución como asesor del proyecto.

A todo el grupo **CUBESAT**, que a pesar de no estar conformado como tal, nos brindo la idea y la oportunidad de llevar a cabo este interesante proyecto.

Al señor **Roberto Diacono**, quien nos facilitó la importación del Rotor Yaesu G-5500.

A nuestras familias y a todas aquellas personas que quizá olvidamos mencionar, quienes con su colaboración y ayuda hicieron posible la realización de este proyecto.

CONTENIDO

	Pág.
RESUMEN	11
RESUMEN	11
ABSTRACT	12
INTRODUCCIÓN	13
1. TEORÍA DE CUBESAT	14
1.1 GENERALIDADES	14
<i>1.1.1. Estación Terrena</i>	<i>15</i>
<i>1.1.2. Componentes de la estación terrena</i>	<i>15</i>
2. CONTROL DE ANTENAS PARA UN SATELITE CUBESAT	17
2.1. ASPECTOS GENERALES	17
2.2. MÉTODOS DE SEGUIMIENTO.....	18
3. COMPONENTES PARA LA REALIZACIÓN DEL CONTROL	19
3.1. EL ACTUADOR PARA EL MOVIMIENTO DE LA ANTENA.....	19
3.1.1. Generalidades del rotor Yaesu G-5500.....	21
3.1.2. Ventajas del Rotor yaesu G-5500.....	25
3.2. EL DISPOCITIVO INTERFAZ ICU G-5500.....	28
3.2.1. Generalidades y Características	33
3.2.2. Instalación y Funcionamiento	34
3.3. COMUNICACIÓN USB	41
3.3.1. USB 2.0.....	43
3.3.2. Dispositivos de Interfaz Humana (HID).....	44
3.4. COMPATIBILIDAD Y APLICABILIDAD	46
4. IMPLEMENTACION Y PROYECCION	46
5. COSTOS.....	46
CONCLUSIONES	48
RECOMENDACIONES	49
GLOSARIO.....	50
BIBLIOGRAFÍA	52
ANEXOS	53

LISTA DE CUADROS

Pág.

<i>Cuadro 1 Pines del puerto de control remoto.....</i>	<i>23</i>
<i>Cuadro 2 Especificaciones Yaesu G-5500.....</i>	<i>24</i>
<i>Cuadro 3 Comparación entre motores eléctricos y el rotor Yaesu G-5500</i>	<i>28</i>
<i>Cuadro 4 Correspondencia de pines entre el DB9 y el puerto de control externo de Yaesu G-5500</i>	<i>29</i>
<i>Cuadro 5 Características del microcontrolador PIC18F4550.</i>	<i>29</i>
<i>Cuadro 6 Descripción de la comunicación PC-Hardware.....</i>	<i>37</i>
<i>Cuadro 7 Descripción de funciones de los pines del conector USB.....</i>	<i>43</i>

LISTA DE FIGURAS

	Pág.
<i>Figura 1 Fotografía Cubesat, vista externa y vista interna</i>	14
<i>Figura 2 Elementos de una estación terrena y sus conexiones</i>	17
<i>Figura 3 Orbitas terrestres utilizadas para poner Satélites.....</i>	18
<i>Figura 4 Actuadores hidráulico, neumático y eléctrico</i>	20
<i>Figura 5 Rotor Yaesu G-5500 y su unidad de control.</i>	21
<i>Figura 6 Rotores de Azimut y Elevación</i>	22
<i>Figura 7 Unidad de control manual de Yaesu G-5500</i>	22
<i>Figura 8 Puerto de control remoto.....</i>	23
<i>Figura 9 Instalación del rotor junto con su unidad de control.</i>	24
<i>Figura 10 Motor AC.....</i>	25
<i>Figura 11 Servomotor, estructura interna y apariencia externa.....</i>	26
<i>Figura 12 Motor paso a paso.....</i>	27
<i>Figura 13 Circuito de monitoreo de conexión con el Rotor Yaesu G-5500....</i>	30
<i>Figura 14 Circuito de captura de posición.</i>	31
<i>Figura 15 Conexión de disparo y control del Rotor.</i>	31
<i>Figura 16 Control interno para el disparo de los relevos que alimentan los motores en el Yaesu G-5500.....</i>	32
<i>Figura 17 Fuente del ICU G-5500.....</i>	33
<i>Figura 18 Aplicación ICU-G-5500.....</i>	35
<i>Figura 19 Diagrama flujo del programa en Visual Basic.....</i>	38
<i>Figura 20 Diagrama flujo del programa para el microcontrolador.....</i>	40
<i>Figura 21 Conectores USB Tipo A y B.....</i>	43
<i>bFigura 22 Diagrama del reconocimiento de descriptores de un HID.</i>	45
<i>Figura 23 Costos de la implementación del control físico de las antenas.</i>	46

LISTA DE ANEXOS

	Pág.
<i>ANEXO A Diagrama esquemático del Rotor Yaesu G-5500</i>	<i>53</i>
<i>ANEXO B Esquemático completo del dispositivo interfaz ICU G-5500. Circuito principal y circuito frontal.</i>	<i>54</i>
<i>ANEXO C Circuitos impresos de la interfaz ICU-G-5500. Circuitos impresos principal y frontal.</i>	<i>56</i>
<i>ANEXO D Distribución de los elementos en las placas finales del ICU G-5500.</i>	<i>57</i>

RESUMEN

Los sistemas satelitales son hoy por hoy, no solo la base, sino también la causa del rápido avance de las telecomunicaciones, y del acortamiento de las fronteras para la transmisión de diferentes tipos de información.

Un satélite CUBESAT, es un micro satélite de medidas 10x10x10 cm., y de peso menor o igual a 1Kg., que fue creado para fines educativos, experimentales y de investigación. Son satélites no comerciales de órbita baja, que viajan alrededor de la tierra, a una velocidad mayor a la velocidad de rotación de ésta y tiene un promedio de vida en el espacio de 3 años.

La Universidad Distrital, pionera en nuestro país en proyectos de investigación, inició ya hace un par de años, el proyecto CUBESAT, con el cual, se quiere llegar a colocar en órbita un satélite de este tipo, con el fin de realizar experimentos e investigación, y por que no, acercar al país en el campo internacional a las ciencias aerospaciales poniendo en órbita uno de los primeros satélites colombianos. Como parte de estos experimentos e investigaciones, la Universidad Distrital se interesó en nuestra universidad, para la creación de una base terrena en esta ciudad, para tener una comunicación entre las instituciones por medio del CUBESAT.

El presente Proyecto, parte de la necesidad de que en nuestra estación terrena se cuente con un sistema de antenas que mantenga una comunicación estable con el satélite, y lo siga en su trayectoria cuando haya posibilidad de comunicación con éste. Este sistema físico de control es una pequeña parte de todo el engranaje que hay que desarrollar para poder llevar a cabo la estación terrena. Nuestro sistema es un dispositivo que se encarga de la comunicación y control entre el Computador servidor y el rotor de la antena, el cual junto con un software de seguimiento preciso, cumple a cabalidad con los requerimientos de estabilidad, precisión y velocidad de seguimiento requeridos para este tipo de proyectos.

ABSTRACT

Recently, satellite systems are not only the base; they are the essential of the telecommunication improvement and the end of the wall for the transmission of information.

A CUBESAT satellite is a micro satellite of size 10x10x10 cm. and a maximum weight of 1Kg. which was made in order to educational, experimental and researching projects. They are non-commercials and low orbit satellites that goes around the earth, with a velocity higher than the rotational earth speed, and have a space life about of 4 years.

The Distrital University, pioneer in our country in investigation projects, started since two years ago, the CUBESAT project, which wants to put on orbit a CUBESAT satellite in order to do experiments, research and in a near future put our country near of the international field of the aero-space science. Like part of these experiments and investigations, the Distrital University was interested in our university, for the creation of an earth base in this city, to have a communication between the institutions by means of the CUBESAT.

The present Project, starts from the necessity of in our earth station counting on a system of antennas that maintains a stable communication with the satellite, and it follows it in his trajectory when there is possibility of communication with this. This physical system of control is a small part of all the gear that there is to develop for being able to carry out the earth station. Our system is a device that is in charge of the communication and control between the Computer and the rotor of the antenna, which along with a software of precise pursuit, fulfills with the requirements of stability, precision and required tracking rate for this type of projects.

INTRODUCCIÓN

Las tecnologías satelitales, utilizadas desde el siglo pasado, no han sido ajenas a nuestro país, en donde sin ser creadores o desarrolladores de este tipo de tecnología, a diario la vemos y vivimos, llegando al punto casi de depender de ésta. Telefonía celular, televisión satelital, enlaces en vivo con cualquier parte del mundo, son algunos de los sistemas que dependen de este campo.

Universidades de diferentes regiones de nuestro país, poco a poco se han ido interesando en proyectos para desarrollar diferentes tecnologías de punta, siendo una de estas las comunicaciones satelitales. Es por esta razón que la universidad Distrital Francisco José de Caldas desde hace ya un par de años, decidió llevar a cabo el proyecto CUBESAT, el cual sería desarrollado en convenio con la universidad Surcolombiana, para el intercambio de información entre una y otra.

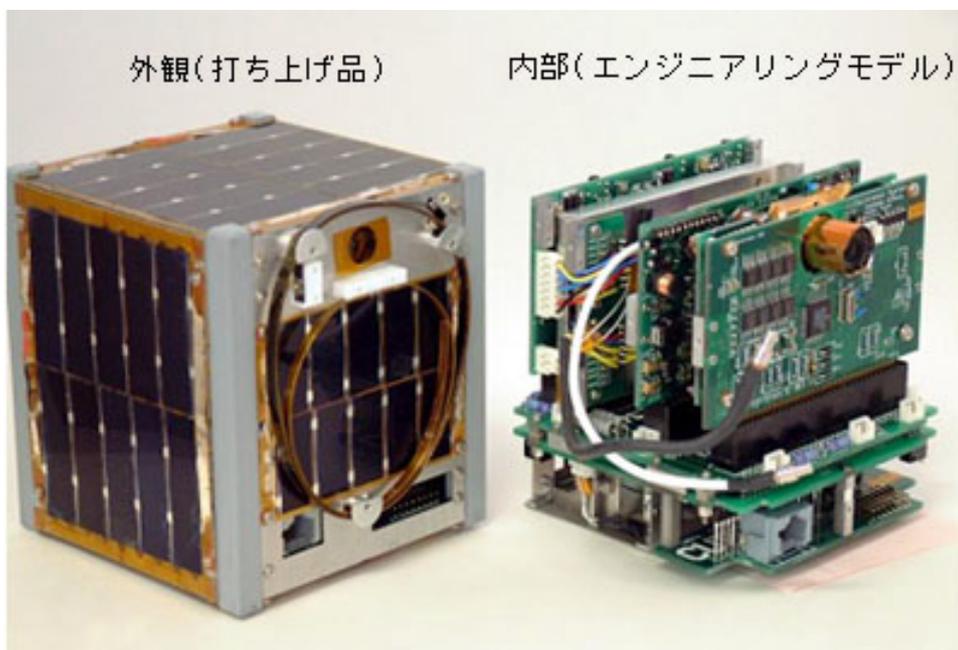
La estación terrena USCO CUBESAT Neiva constaría de un cuarto de control y comunicaciones, dotado de tecnología, en el cual se harán pruebas de enlace con el satélite CUBESAT y a través de este con la universidad Distrital. De aquí nace nuestro proyecto de grado. Para garantizar una comunicación estable con el satélite, se debe contar con un sistema de control para poder seguir (tracking) el satélite durante el tiempo que este tenga línea de vista con nuestra estación. El sistema físico que nosotros desarrollamos, con la ayuda de un software de predicción y seguimiento, cumple a cabalidad con los requisitos mencionados, y garantiza precisión en el seguimiento y la comunicación entre la estación terrena y el satélite.

1. TEORÍA DE CUBESAT

1.1 GENERALIDADES

El programa CUBESAT (CUBE: cubo; SAT: satélite) fue una idea original del profesor Robert Twig de la Universidad de STANFORD en la década de los 90s. En base a este programa, todo proyecto de tipo CUBESAT debe cumplir con ciertas especificaciones y normas. El tamaño del microsatélite debe ser de 10x10x10 cm. y su peso no mayor a 1 Kg. Son microsatélites dotados de alta tecnología enfocados para proyectos estudiantiles e investigativos.

Figura 1 Fotografía Cubesat, vista externa y vista interna



Tomado de:

<http://www.aerospace.t.u-tokyo.ac.jp/headline/images/topics/cubesat.jpg>

En base a esto, La Universidad Distrital de Colombia, lleva a cabo el proyecto 'Cubesat UD', que cuenta con el respaldo de dos grupos de investigación de dicha universidad los cuales son el Capítulo de Comunicaciones del IEEE_UD y el grupo de telecomunicaciones de la universidad Distrital GITUD.

Con el fin de ampliar el proyecto, vinculando a éste otra universidad del país, la Universidad Distrital, decidió, vincular a la Universidad Surcolombiana como parte del mismo, para llevar a cabo en esta última, la construcción de una base terrena, la cual permitirá la realización de diferentes pruebas de comunicaciones entre Universidad (Usco o Distrital), satélite Cubesat y Universidad (Usco o Distrital).

El CubeSat UD será puesto en órbita, por medio de un vehículo espacial Ruso, junto a otros proyectos CubeSat. El CubeSat estará ubicado dentro de un dispositivo creado para su lanzamiento, el cual contiene otros 2 CubeSat. La órbita en la que será puesto el CubeSat UD, es LEO (Low Earth Orbit), a una altitud aproximada de 800 Km.

1.1.1. Estación Terrena

El proyecto Cubesat Usco, se basa en la creación de una estación terrena en nuestra ciudad para la transmisión, recepción y manipulación de la información enviada hacia y desde el satélite. El Proyecto se divide en 2 grandes grupos, uno encargado del control y seguimiento del satélite y el otro encargado de las comunicaciones y manipulación de datos enviados y recibidos.

El grupo de control y seguimiento, al cual nosotros pertenecemos, está dividido en dos subgrupos, el primero es el encargado, del hardware de control (parte física), todo lo relacionado con el acople entre el computador (software) y el servomecanismo y el acople entre este y las antenas. El segundo grupo se encarga del desarrollo de un software inteligente para el seguimiento del satélite y la comunicación de este con el hardware. En este caso, nuestro proyecto es llevar a cabo el hardware de control.

1.1.2. Componentes de la estación terrena

Una estación terrena para una comunicación satelital requiere de diferentes dispositivos y de tecnología para poder llevar a cabo todas las funciones que a esta le competen. Los principales componentes son:

Antenas:

Las antenas son las encargadas de la captura y envío de las señales RF necesarias para realizar la comunicación. Las antenas deben cumplir con diferentes parámetros de ganancia y respuesta en frecuencia, tanto para el enlace de subida como para el de bajada. Por esta razón se utilizarán dos antenas.

Actuadores:

Para el seguimiento del satélite, se necesita de al menos 2 actuadores para lograr mover las antenas, uno que llevara a cabo el movimiento del azimut y otro se encargara de la elevación. En nuestro caso escogimos el rotor Yaesu G-5500, rotor que cuenta con rotores de combinación de azimut/elevación.

Control del Actuador:

Para sincronizar el rotor con el PC y las antenas, hemos desarrollado un dispositivo que se encarga de este control, creando una interfaz USB entre el PC y el rotor de las antenas. Es un dispositivo versátil, ligero y preciso, que cumple con los requisitos de velocidad y precisión a la hora de realizar el control.

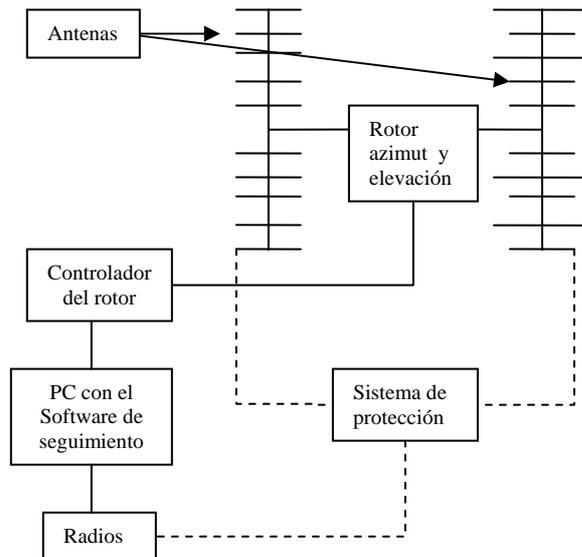
Software de seguimiento:

Este software se encarga por medio del desarrollo de diferentes ecuaciones matemáticas de enviar la información de la precisa ubicación del satélite en el espacio en tiempo real. Este se comunica por medio del controlador al rotor el cual a su vez lleva a las antenas a la ubicación deseada.

Radios:

Son los encargados de la modulación y demodulación de las señales enviadas y recibidas respectivamente. Son los que convierten las señales RF en señales que puedan ser entendidas por la computadora, y por supuesto, se encargan de adaptar la información enviada por el PC para poder ser irradiada en el espacio.

Figura 2 Elementos de una estación terrena y sus conexiones



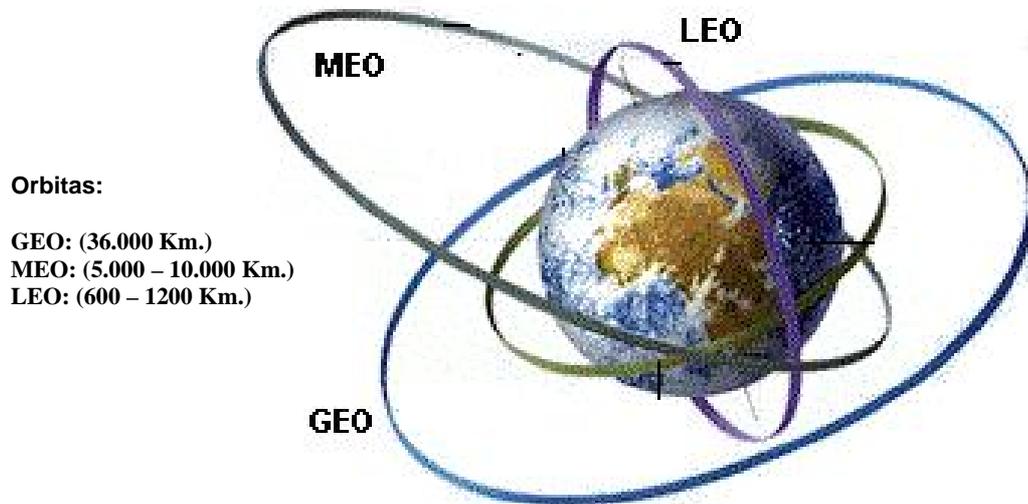
En general estos son los elementos necesarios para llevar a cabo una estación terrena. El correcto acople y la perfecta sincronización entre estos, hará que el funcionamiento de la estación sea exitoso.

2. CONTROL DE ANTENAS PARA UN SATELITE CUBESAT

2.1. ASPECTOS GENERALES

La mayoría de satélites con son utilizados para la realización de comunicaciones de tipo comercial y militar son de tipo geoestacionarios, es decir, que se encuentran en la orbita GEO (aprox. 36.000 Km.) y que giran alrededor de la tierra a una velocidad igual a la velocidad de rotación de ésta. Por tanto son satélites que 'no se mueven', ya que al ir a la par con la tierra, siempre estarán visibles en un mismo punto en el espacio. Un control para una antena de este tipo, sería algo innecesario, ya que la probabilidad de perder de vista al satélite es muy reducida, y se debe a factores como nubosidad, vientos y aspectos climáticos naturales que pueden hacer cambiar las condiciones del enlace.

Figura 3 Orbitas terrestres utilizadas para poner Satélites.



Tomado de:

http://www.saftbatteries.com/100-MS_Space/images/30-10-10.gif

En nuestro caso, el seguimiento no es tan sencillo. Un satélite cubesat, es un satélite muchísimo más pequeño que un satélite comercial, que irradia mucha menor potencia y que por este motivo debe ser puesto en una órbita baja LEO (aprox. 1.000 Km.) y que además estará girando alrededor de la Tierra a una velocidad mucho mayor que la velocidad de rotación de esta (entre 9 y 12 veces mayor).

Por esta razón el satélite será visible desde nuestra estación por un pequeño lapso de tiempo entre 15-40 minutos, que se pueden repetir cada una o dos horas. Estos parámetros dependen de las características de lanzamiento del satélite, (velocidad, altura, inclinación, etc.). El control de las antenas deberá ser capaz de seguir al satélite durante el tiempo en el cual este sea visible, y estar listo a esperar nuevamente el satélite cada vez que este desaparezca.

2.2. MÉTODOS DE SEGUIMIENTO

Existen diferentes métodos de control para lograr el seguimiento de objetos en movimiento, pero no todos son aplicables al seguimiento de satélites. El seguimiento puede realizarse por medio de un software, el cual por medio de desarrollos físicos – matemáticos pueda simular con exactitud y precisión el comportamiento del satélite y al comunicarse con el rotor de las antenas ubicarlas para una correcta comunicación. Este es un control de lazo abierto, ya que el software en ningún momento recibirá señales de conexión o enlace con el satélite y dependerá únicamente de los resultados de sus operaciones.

Otra forma de seguimiento, son los controles de lazo cerrado. Son sistemas que por medio de diferentes tipos de control (control por fase, PID, lógica difusa, etc.) realizados por software o hardware, manipulando las señales enviadas por el satélite, mueven la antena de forma aleatoria o periódica, haciendo un escaneo de la señal del satélite, el cual al ser encontrado, será seguido por las antenas, buscando siempre una mayor intensidad de señal.

Como podemos observar ambos sistemas tienen sus ventajas y desventajas, ya que el primero al ser un sistema de lazo abierto, no recibe información alguna del satélite para asegurarse de si está o no en contacto, y al fallar en cualquier cálculo, generaría una falsa información al rotor y no se lograría hacer el enlace. Por otra parte un control de lazo cerrado, nos garantiza la conexión con el satélite. Pero también puede que el tiempo que este demore en encontrarlo, sea lo bastante grande como para que el satélite salga de la cobertura, y durante ese tramo de tiempo no haya conexión.

Es por este motivo que nuestro control se basará en el primer método, un control de lazo abierto por software, el cual ofrece la flexibilidad, de ser modificado y con la ayuda de radios y la manipulación de las señales, crear un sistema híbrido de lazo cerrado, garantizando aún más la estabilidad del sistema.

3. COMPONENTES PARA LA REALIZACIÓN DEL CONTROL

Observando lo anterior, nos damos cuenta que para la realización del control de las antenas son necesarios varios componentes tales como PC, un hardware de control y comunicación, y actuadores, en nuestro caso un rotor.

El cerebro de nuestro controlador es un microcontrolador de Microchip PIC 18F4550, el cual fue escogido por sus amplias características y básicamente por que es uno de los microcontroladores de microchip que soporta comunicación USB, específicamente USB 2.0.

3.1. EL ACTUADOR PARA EL MOVIMIENTO DE LA ANTENA

Los actuadores, tiene como misión generar movimiento de los elementos terminales a los que están conectados. Un actuador puede emplear energía mecánica, hidráulica o eléctrica. La finalidad es la misma, obtener el movimiento deseado. En el caso del movimiento de las antenas, luego de estudiar las tres opciones, la única que cumple con los requisitos son los actuadores eléctricos.

Figura 4 Actuadores hidráulico, neumático y eléctrico



Tomado de:

http://www.dem.isel.ipl.pt/seccoes/pagspm/proyectos/cartas_bacharelato/images_2003_2004/23351.jpg

<http://www.electra.com.co/images/imgNeumatValvAng.jpg>

<http://www.roboticspot.com/tienda/00200.jpg>

En un actuador neumático su fuente de energía es la presión de aire. Debido a la compresibilidad del aire, los actuadores neumáticos no consiguen buena precisión de posicionamiento. No son buenos para mantener posiciones intermedias, lo cual los hace obsoletos para nuestra aplicación. Por otra parte, el actuador hidráulico funciona de forma similar al neumático con la diferencia que su fuente de energía es la presión de aceites minerales (fluidos). Gracias a que estos fluidos tienen una compresibilidad muy inferior al del aire la precisión obtenida es mucho mayor. Sin embargo estos actuadores son de elevado costo, su instalación es compleja y debe ser a prueba de fugas, y requieren de un constante mantenimiento.

Los actuadores eléctricos son los más aptos para esta aplicación. Requieren de energía eléctrica para su funcionamiento, son precisos, hay gran variedad, y con unos pocos ajustes, se puede obtener la fuerza y precisión necesaria para el movimiento de las antenas. Existen tres grandes grupos de actuadores eléctricos, que son: Motores de corriente continua (DC), Motores de corriente alterna (AC) y Motores pasó a pasó. En el caso del movimiento de las antenas, nuestra necesidad era el control de la rotación tanto del azimut, como el de la elevación de la antena. Por esta razón usamos Rotores. Los rotores, son arreglos entre motores eléctricos, cajas de transmisión reductoras, y circuitos de control, cuya finalidad es prestar un movimiento rotacional preciso y de fácil controlabilidad.

3.1.1. Generalidades del rotor Yaesu G-5500

Figura 5 Rotor Yaesu G-5500 y su unidad de control.



Tomado de:
Yaesu G-5500 Instruction manual.

El rotor Yaesu G-5500 es un dispositivo electromecánico, compuesto por dos partes:

El rotor como tal que consta de los motores eléctricos, las cajas de engranes, los sensores de posición (Potenciómetros). Todo esto encapsulado en un empaque que le brinda la debida protección ante las exigencias climáticas. Este rotor en realidad son dos rotores por separado, uno de Azimut y uno de Elevación que pueden ser instalados conjuntamente como se puede observar en la *Figura 5*, o pueden ser usados por separado obteniendo un resultado de movimiento de Azimut o de Elevación según sea el rotor escogido. Estos rotores que conforman el rotor Yaesu G-5500 están debidamente lubricados y no requieren de mantenimiento según su fabricante.

El rotor de Azimut proporciona aparte del movimiento de 0° a 450°, unos bornes que están conectados al sensor para así obtener el dato de su posición, de acuerdo al voltaje con que alimentemos este sensor así será el voltaje de salida que indicara el Azimut, de igual manera el rotor de elevación cuenta con dichos bornes pero el movimiento que puede realizar es de 0° a 180°.

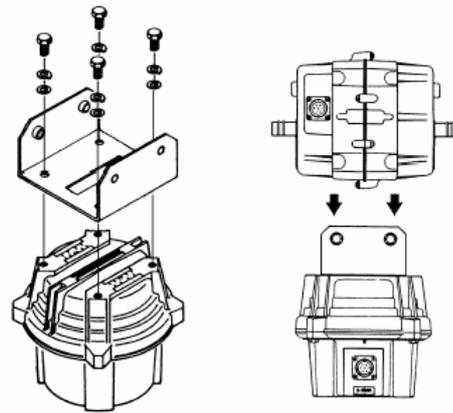
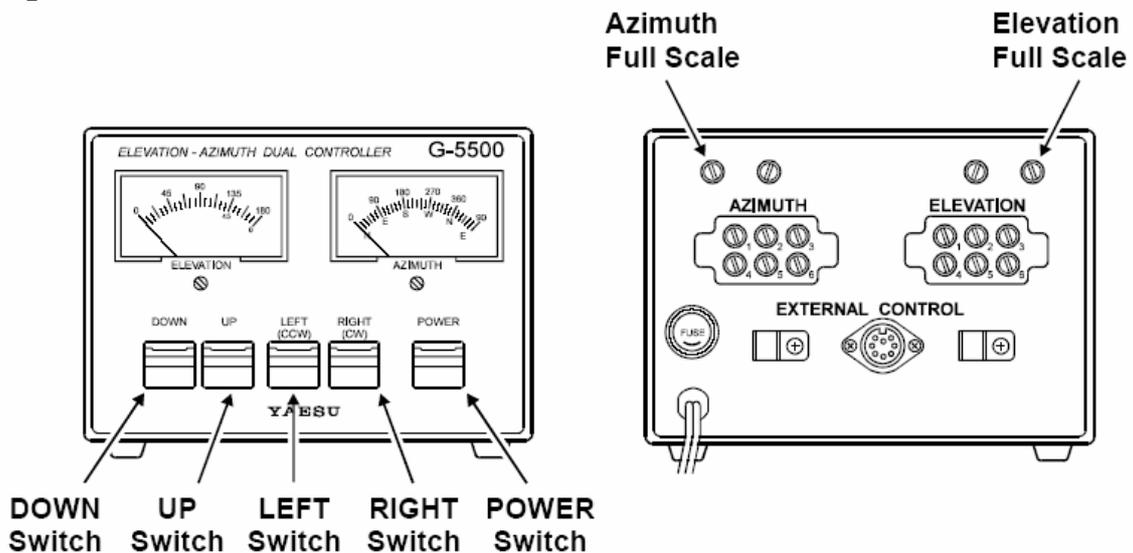


Figura 6 Rotores de Azimut y Elevación
Tomado de:
Yaesu G-5500 Instruction manual.

Figura 7 Unidad de control manual de Yaesu G-5500

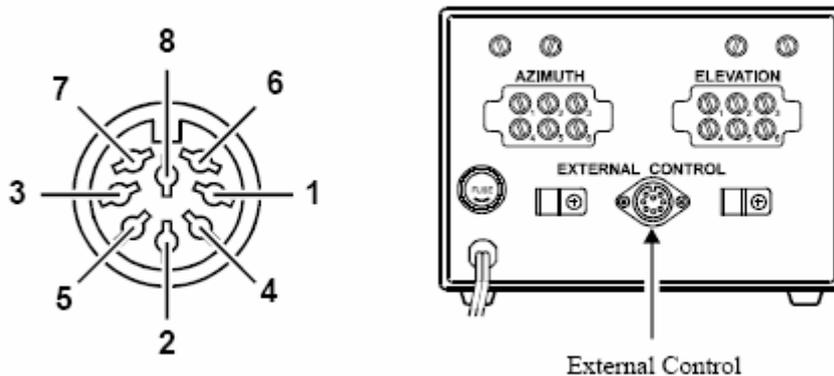


Tomado de:
Yaesu G-5500 Instruction manual.

Aparte del sistema mecánico, este cuenta con una **unidad de control manual** que se encarga de energizar los motores, de componer el circuito de censado alimentado los sensores con que cuentan los rotores de Azimut y de Elevación. Este también cuenta con unos indicadores de posición análogos en los que se puede observar la posición actual de ambos rotores. Posee unos pulsadores que controlan de manera manual el movimiento hacia arriba, abajo, derecha o izquierda. Por ultimo esta unidad tiene un puerto en su parte posterior para poder realizar el control del rotor remotamente desde un PC; este puerto suministra unas salidas de voltaje de 0 a 4.5 voltios aproximadamente que indican el Azimut y la Elevación actual, estos voltajes son debido a la

alimentación que este le suministra a los sensores del rotor. Cuenta con unos pines de entrada que al ser conectados a tierra generan el movimiento de azimut hacia la derecha o izquierda y en elevación hacia arriba o hacia abajo. A su vez entrega una fuente de voltaje de aproximadamente 6 a 13 voltios con una corriente de 200 mA, dicha fuente es muy inestable y con gran contenido de ruido por lo que se ignoró para ser usada como fuente de la interfaz desarrollada.

Figura 8 Puerto de control remoto.



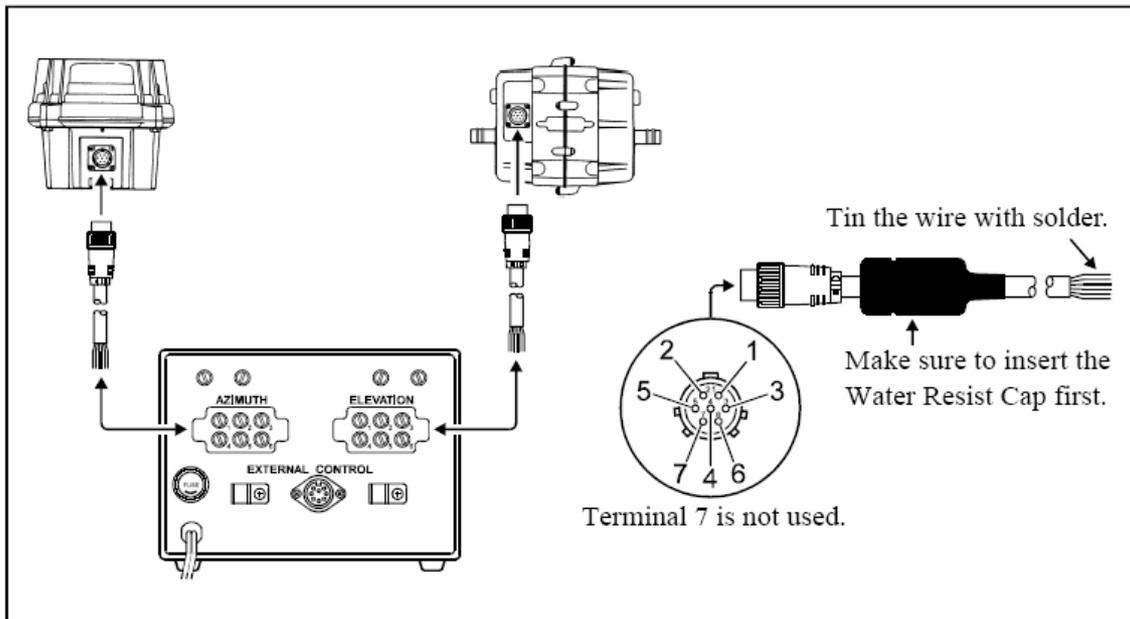
Tomado de:
Yaesu G-5500 Instruction manual.

Cuadro 1 Pines del puerto de control remoto.

Pin	Función
6	Provee de 0 a 4.5 voltios que corresponden de 0° a 450°.
1	Provee de 0 a 4.5 voltios que corresponden de 0° a 180°.
4	Pin que se conecta al pin 8 (GND) para rotar hacia la izquierda.
2	Pin que se conecta al pin 8 (GND) para rotar hacia la derecha.
5	Pin que se conecta al pin 8 (GND) para rotar hacia arriba.
3	Pin que se conecta al pin 8 (GND) para rotar hacia abajo.
7	Provee una fuente de voltaje de 6 a 13 voltios con una corriente de 200 mA.
8	Común (GND).

Estos dos elementos son conectados de la siguiente manera conformando así lo que llamamos Rotor Yaesu G-5500.

Figura 9 Instalación del rotor junto con su unidad de control.



Tomado de:
Yaesu G-5500 Instruction manual.

El diagrama esquemático del Rotor Yaesu G-5500 se encuentra en el Anexo A.

Características del Rotor:

El Rotor Yaesu G-5500 tiene movimiento tanto de azimut como de elevación y en las 4 direcciones. Funciona con alimentaciones de 110-120 o 200-240 VAC. A continuación enumeramos sus características, tomadas del Yaesu G-5500 Instruction manual.

Cuadro 2 Especificaciones Yaesu G-5500

Especificaciones	
Voltaje requerido	110-120 o 200-240 VAC
Voltaje de los motores	24 VAC
Tiempo de rotación	Elevación (180°): 67 segundos Azimut (360°): 58 segundos
Máximo tiempo de continua operación	5 minutos

Torque de rotación	Elevación: 14Kg.-m Azimut: 6Kg-m
Torque de frenado	Elevación: 40Kg-m Azimut: 40Kg-m
Carga Vertical	200Kg
Error de posición	<= 4%
Peso	Rotores: 9Kg Unidad de control: 3Kg

3.1.2. Ventajas del Rotor yaesu G-5500

El rotor yaesu G-5500, nos ofrece innumerables ventajas, no solo por que es un rotor creado para esta función sino también por costos, ya que realizar un montaje similar al presentado por estos rotores, requiere de grandes inversiones y una alta complejidad.

Analizaremos cada una de las posibles opciones de control y explicaremos el por que de la elección del Yaesu G-5500.

Opción 1: Motores AC

Figura 10 Motor AC



Tomado de:

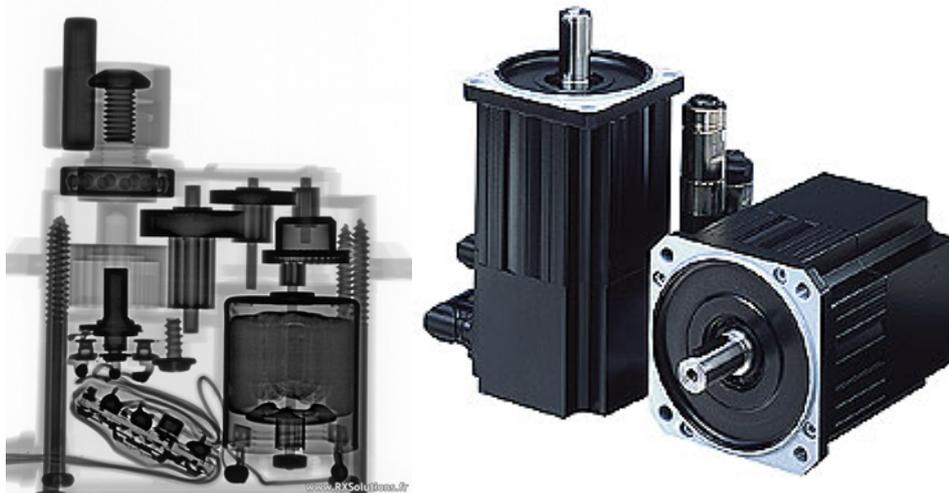
<http://www.wordsun.com/boc709p.jpg>

Son motores de gran potencia y tamaño, sincrónicos o asíncronos, los cuales presentan diferentes características de velocidad y torque que podrían ser bien

utilizadas en aplicaciones donde se necesite manejar cargas altas y trabajo pesado. Un motor AC con un elevado número de polos sería ideal para un movimiento lento y firme necesario para controlar la posición de las antenas. Estos motores, son de fácil consecución en nuestro país lo cual es una ventaja a diferencia de otros tipos de motores. Pero no todo es tan bueno. El control de los motores AC es uno de los más complejos y tediosos, que para hacerlos con un grado de exactitud alto, se necesitaría de una gran inversión y dedicación de tiempo.

Opción 2: Motores DC

Figura 11 Servomotor, estructura interna y apariencia externa.



Tomado de:

[http://www.elmeko.no/akershus/baerum/elmeko.nsf/bilder/Servomotor%20ED-EK.jpg/\\$FILE/Servomotor%20ED-EK.jpg](http://www.elmeko.no/akershus/baerum/elmeko.nsf/bilder/Servomotor%20ED-EK.jpg/$FILE/Servomotor%20ED-EK.jpg)

Son motores, más livianos que los AC, más versátiles y que presentan una facilidad de control mucho mayor. Presentan una menor relación Potencia/tamaño, pero con ayuda de reductores y encoders, se pueden conseguir motores aún más fáciles de controlar llamados Servomotores.

Un servomotor, es un motor DC, que cuenta con una caja reductora, un encoder y un circuito interno de control especialmente diseñados para manejo de cargas pesadas que requieren de precisión como es el caso de la robótica. Estos servomotores son quizá la mejor opción a utilizar para nuestra finalidad. Pero tienen un aspecto en contra: su precio. Un servomotor que tenga características de torque de 10Kg-m, una reducción de 1000 a 3 (para obtener un RPM de 2 vuelta/minuto) y un encoder digital cuesta alrededor de los U\$ 300 en USA y E\$ 280 en Suiza.

Opción 3: El motor paso a paso

Figura 12 Motor paso a paso



Tomado de:

<http://www.eso.org/outreach/eduoff/edu-prog/catchstar/CAS2004/casreports-2004/rep-035/MOTORPaP.jpg>

Los motores paso a paso son motores especialmente diseñados para obtener por medio de impulsos eléctricos un movimiento angular giratorio de gran precisión.

Son motores en los cuales el estator cuenta con varias bobinas, las cuales al ir siendo energizadas ordenadamente dan movimiento al motor, por cada pulso recibido, el rotor gira un determinado número discreto de grados. Estos motores, pese a presentar una mayor facilidad de control que los demás, tiene una gran desventaja, su torque o par. Para lograr un par como el necesitado de al menos 8Kg-m se necesitaría de un motor paso a paso de gran tamaño.

Además de los motores, se necesitaría de una caja reductora, para aumentar aun más su torque y exactitud, en algunos de los motores. También tendríamos que construir los acoples entre los motores y las antenas, así como la protección de todo el circuito, ya que este debe ser impermeable y resistir cualquier tipo de condición atmosférica. Por otra parte se requeriría de la construcción de un freno para evitar que las antenas se muevan cuando los motores no estén en movimiento y evitar daños en la piñonaría de la caja reductora.

Como vemos, la mayoría de los motores eléctricos pueden llegar a cumplir con los requisitos necesarios para nuestro proyecto, pero la relación de costos es mayor y además de ser más complejo el control que utilizando el rotor ya mencionado.

El rotor G-5500 cuenta con motores AC que funcionan a 24V. Como ya se mencionó estos motores tienen un buen par y una velocidad de giro continua. Además el rotor cuenta con reductores que lo hacen aún más fuerte y estable así como con el freno necesario para cuando el rotor no está en movimiento. A parte de todas estas virtudes, trae anexo, una unidad de control, que se encarga de la debida alimentación de los motores y de realizar la lectura de la posición en que se encuentran dando como salida un valor de voltaje correspondiente al ángulo de ubicación.

A continuación mostraremos una tabla donde se observa los costos y características de las diferentes opciones.

Cuadro 3 Comparación entre motores eléctricos y el rotor Yaesu G-5500

Característica	Motor DC	Servomotor	Motor AC	Rotor Yaesu G-5500	Costo aproximado
Caja reductora	Necesita	Incluido	Necesita	Incluido	\$200.000
Encoder	Necesita	Incluido	Necesita	Incluido	\$150.000
Frenos	Necesita	Necesita	Necesita	Incluido	\$100.000
Protección impermeable	Necesita	Necesita	Necesita	Incluido	\$100.000
Montaje y acople	Necesita	Necesita	Necesita	Incluido	\$250.000
Control	Sencillo	Sencillo	Complejo	Sencillo	Cada motor tiene costo diferente
Costo del motor	120.000	600.000	200.000	1'550.000	
Teniendo en cuenta que se necesitarían 2 motores de cada uno, uno para azimut y otro para elevación los costos finales serían:					
Motores DC: $120.000*2 + 200.000*2 + 150.000*2 + 100.000 + 100.000 + 250.000$ = \$1'390.000					
Servomotores: $600.000*2 + 100.000 + 100.000 + 250.000$ = \$1'650.000					
Motores AC: $200.000*2 + 200.000*2 + 150.000*2 + 100.000 + 100.000 + 250.000$ = \$1'550.000					
Además, los frenos, el montaje y el acople se haría aquí en Colombia en forma artesanal, lo cual no garantiza el mejor funcionamiento.					
Rotor Yaesu G-5500: 1'550.000 Con garantía de fabrica de 2 años, fácil control, robusto y no requiere de mantenimiento.					

Por todas estas razones, escogimos el rotor yaesu G-5500. Además su importación se nos facilitó costando 1'550.000 con todo y gastos de envío. Los gastos de envío de los servomotores, los encoder y las cajas reductoras no fueron tenidos en cuenta para el cálculo de gastos.

3.2. EL DISPOSITIVO INTERFAZ ICU G-5500

El ICU G-5500 (Interfaz de Control USCO G-5500) fue diseñado y elaborado cumpliendo con los requerimientos y características para servir de interfaz entre el PC y el Rotor Yaesu G-5500 específicamente. El ICU G-5500 es un dispositivo microcontrolado capaz de capturar los valores análogos de salida del Rotor Yaesu G-5500 procesarlos y determinar las posiciones tanto de Azimut como de Elevación, monitorea la conexión con el Rotor y con el PC lanzando un mensaje de advertencia al detectar que el Rotor o el PC no se encuentran debidamente conectados a el o no presentan respuesta. La comunicación con el PC es realizada mediante el puerto USB 2.0, obteniendo

una gran velocidad de transmisión. Hacia el Rotor la conexión se hizo mediante un conector DB9 el cual se conecta al puerto de control remoto con que cuenta el control manual del Yaesu G-5500. Los pines están conectados así:

Cuadro 4 Correspondencia de pines entre el DB9 y el puerto de control externo de Yaesu G-5500

Pin DB9	Pin del puerto de control externo del Yaesu G-5500	Función
1	8	GND
2	3	Pin que se conecta a GND para rotar hacia abajo.
3	4	Pin que se conecta a GND para rotar hacia la izquierda.
4	2	Pin que se conecta a GND para rotar hacia la derecha.
5	5	Pin que se conecta a GND para rotar hacia arriba.
6	6	Provee de 0 a 4.5 voltios que corresponden de 0° a 450°.
7	1	Provee de 0 a 4.5 voltios que corresponden de 0° a 180°.
8	7	Provee una fuente de voltaje de 6 a 13 voltios con una corriente de 200 mA.
9	NC	No tiene función.

Como unidad de proceso del ICU G-5500 se escogió el microcontrolador PIC18F4550 de la familia Microchip, el posee 40 pines, 5 puertos I/O, modulo ADC con 13 canales análogos, modulo USB 2.0 que puede trabajar a Low Speed (1.5Mb/s) o a Full Speed (12Mb/s), 32K de memoria d programa, 256 de memoria EEprom.

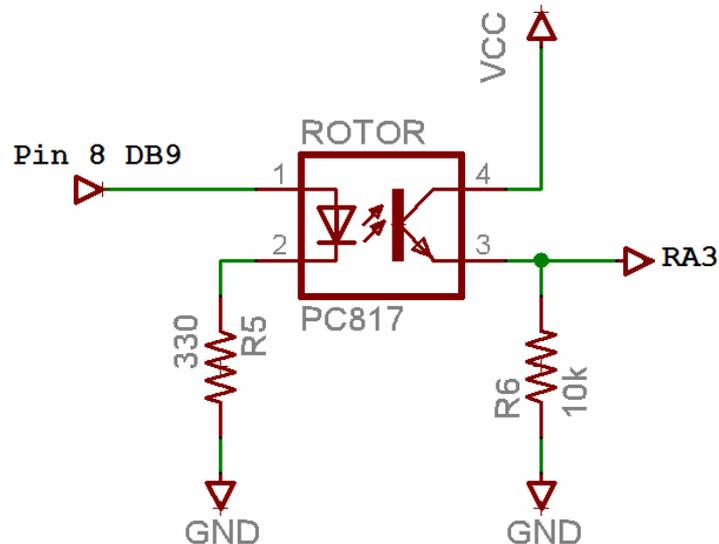
Cuadro 5 Características del microcontrolador PIC18F4550.

Device	Program Memory		Data Memory		I/O	10-bit A/D (ch)	CCP/ECCP (PWM)	SPP	MSSP		EAUSART	Comparators	Timers 8/16-bit
	Flash (bytes)	# Single-Word Instructions	SRAM (bytes)	EEPROM (bytes)					SP ^I ™	Master I ² C™			
PIC18F2455	24K	12288	2048	256	24	10	2/0	No	Y	Y	1	2	1/3
PIC18F2550	32K	16384	2048	256	24	10	2/0	No	Y	Y	1	2	1/3
PIC18F4455	24K	12288	2048	256	35	13	1/1	Yes	Y	Y	1	2	1/3
PIC18F4550	32K	16384	2048	256	35	13	1/1	Yes	Y	Y	1	2	1/3

El sistema monitorea la conexión con el rotor mediante la fuente de 13 a 6 voltios (pin 8 DB9) que entrega el Yaesu G-5500, esta señal es usada como la alimentación del Led de un optocoplador PC817, el cual es activado al ser energizado el led de disparo, el fototransistor es saturado conectando el VCC a la resistencia R6, así se obtiene un alto cuando el rotor se encuentra encendido

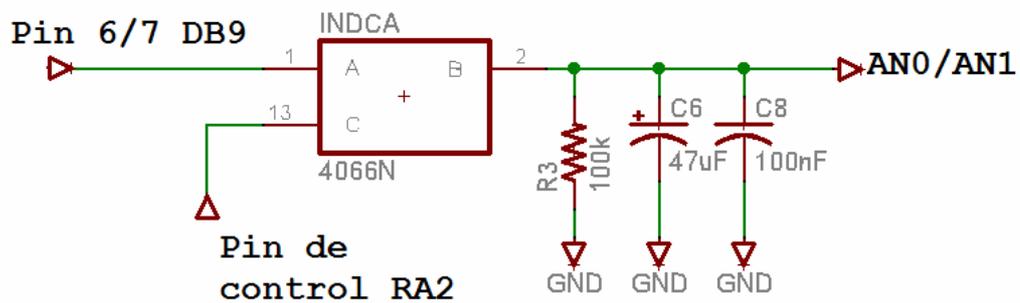
y conectado al ICU G-5500, de lo contrario se obtendrá un bajo. Esta señal de salida de este circuito se conecta al microcontrolador por el pin 5 (RA3).

Figura 13 Circuito de monitoreo de conexión con el Rotor Yaesu G-5500



Para Capturar los valores de voltaje que son entregados por los sensores de posición y convertirlos a su correspondiente valor digital, las señales provenientes de los pines 6 y 7 del DB9 son conectadas a un circuito integrado 74hc4066 el cual es un switch análogo de estado sólido, este actúa de una manera similar a un relé con la ventaja de no tener un sistema mecánico, evitando ruido además de poseer un velocidad mayor de respuesta. Este integrado permite al PIC controlar la conectividad de voltajes análogos a sus pines de conversión llegado el caso de que el voltaje se acerque al valor límite de entrada desconectando así esta señal para evitar posibles averías del microcontrolador. El 74hc4066 posee unos pines de control para la conexión entre sus pines de entrada y salida, además sus especificaciones hablan de una impedancia de 80ohm la cual se presenta ante la señal que se le conecte como entrada. Posee 4 switch bidireccionales de los cuales se usaron tan solo 2. A las salidas de estos switch se realizó un filtro para las interferencias que puedan venir en la señal y un filtro para el rizado que aunque es minúsculo afecta enormemente el valor de resultado digital, con este filtro no se logra atenuar totalmente el problema pero lo hace mucho menor. Y por ultimo para hacer mínima la caída de tensión sobre el 74hc4066 se conecta en serie a este una resistencia de 100kohm.

Figura 14 Circuito de captura de posición.



Este dispositivo cuenta con un terminal para realizar la programación del PIC en el circuito evitando la desconexión del PIC, reduciendo el peligro de averiarlo al ser desprendido de su zócalo.

Ahora para el disparo de los motores, la interfaz maneja una serie de optocopladores MTC6 conectados a los pines asignados al control del PIC. El PIC alimenta el diodo Led interno del optocoplador disparando el fototransistor con un alto de los pines del Micro. Este fototransistor se conecta al Yaesu G-5500 mediante los pines 2, 3, 4, 5 del DB9 para una rotación hacia abajo, izquierda, derecha y arriba respectivamente. La función del fototransistor es cortocircuitar los pines con tierra, lo que hace internamente en el Yaesu G-5500 es conectar a tierra la base del transistor PNP el cual dispara los relevos encargados de alimentar a los motores.

Figura 15 Conexión de disparo y control del Rotor.

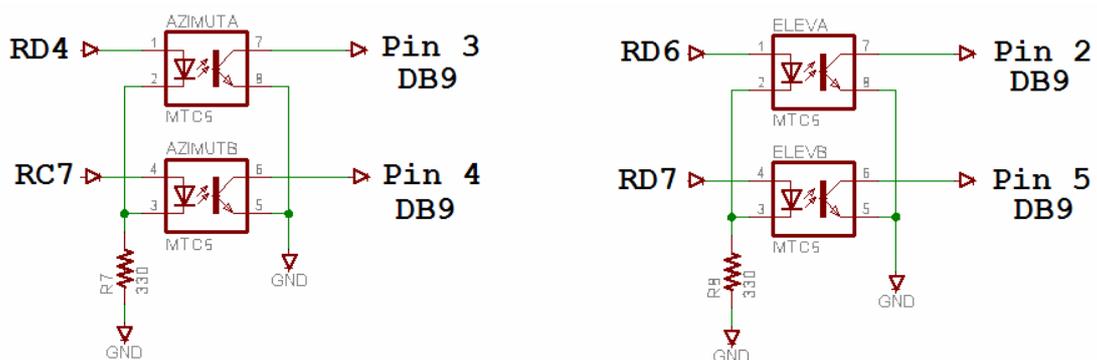
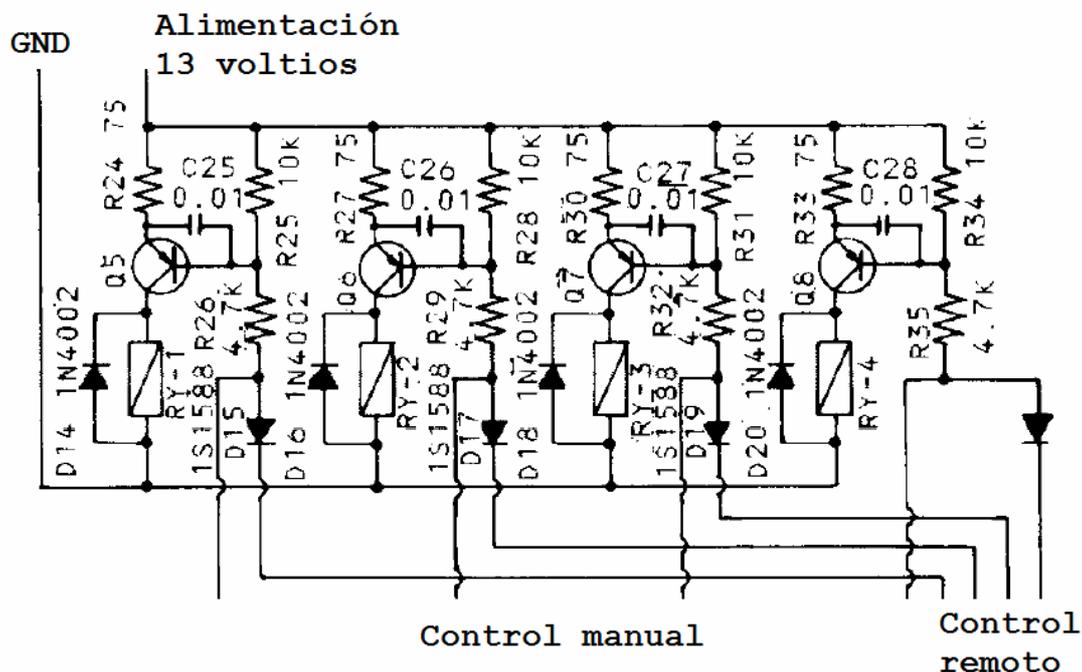


Figura 16 Control interno para el disparo de los relevos que alimentan los motores en el Yaesu G-5500

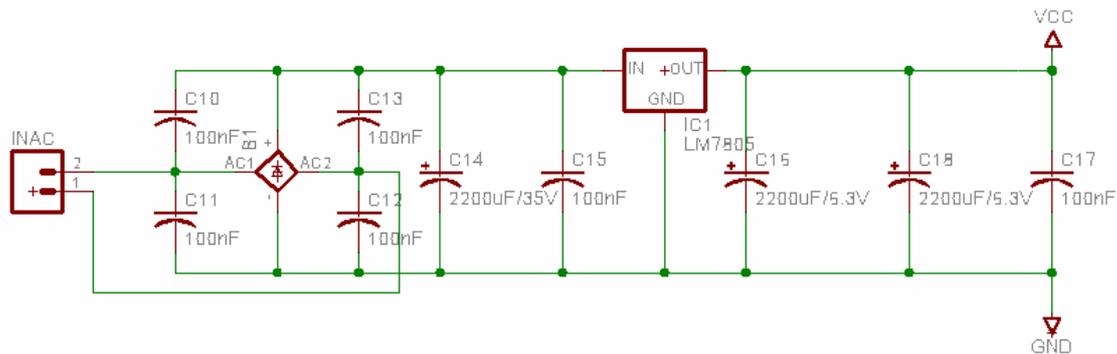


Al los pines llamados control remoto se conectan mediante el DB9 las señales de control provenientes del microcontrolador.

El ICU presenta también un panel de monitoreo de las posiciones visualizadas en un display LCD de 16 caracteres de dos líneas, y una serie de Leds indicando el estado de conexión del rotor, del PC y hacia donde se esta realizando la rotación o si no se esta rotando. Además se dispusieron 3 pulsadores que permiten entrar en menú de selección del dato que se desea visualizar, o si se desea pasar de modo automático de seguimiento al modo manual y viceversa.

Este se alimenta de la red domiciliaria a 110 voltios AC/60Hz, esta alimentación es reducida a 12 voltios rms mediante un transformador *Magon 508*, este voltaje es sometido a una rectificación de onda completa, filtrado y regulado para obtener una alimentación de 5 voltios aproximadamente. La fuente como el circuito en general cuenta con un grupo de condensadores encargados de eliminar las interferencias que pueda presentar la red. En la salida del regulador de voltaje (LM7805) se conectaron dos capacitares de 3300uF/6.3 voltios para evitar que el PIC se reinicie al sufrir las caídas de voltaje que ocurren cuando los rotores llegan a sus topes de rotación, esto hace que el circuito se apague por un instante de tiempo muy pequeño pero suficiente para que el PIC sufra un reset.

Figura 17 Fuente del ICU G-5500



Por ultimo el código en Basic para el Microcontrolador es Anexado como ANEXO F.

3.2.1. Generalidades y Características

El ICU G-5500 presenta las siguientes características:

- Comunicación USB 2.0 para la conexión con el ordenador.
- Comunicación paralela mediante un conector DB9 con el Yaesu G-5500 controller.
- Display LCD donde se despliegan los mensajes de alerta, los valores de azimut y elevación actual, los valores de azimut y elevación entregados por el PC.
- Un menú de información y de control de modo (Automático-Manual), que operado mediante los pulsadores encontrados en el panel Frontal. Dicho menú contiene:
 - Posición actual
 - Posición deseada
 - Modo automático
 - Modo manual
 - Calibrar
 - Salir del menú
- Posee la opción de realizar la calibración de escala automáticamente para así realizar la conversión de voltaje a su respectivo ángulo en grados de una manera mas aproximada.
- Por ultimo posee 2 Leds bicolor indicadores del estado de conexión con el PC y el Rotor indicando con el color rojo un error de conexión y con el color amarillo un estado de conexión OK. 2 Leds tricolor mas indican si se esta moviendo el rotor y en que dirección, indicando en color rojo que se encuentra inmóvil, con Azul que esta realizando una rotación hacia la derecha para el Rotor de azimut y hacia arriba para el rotor de elevación, y con el color amarillo indica que se mueve hacia la izquierda en el caso del azimut y hacia abajo en el caso de la elevación.

3.2.2. Instalación y Funcionamiento

Como podemos ver, el dispositivo ICU-G-5500 es un dispositivo sencillo, que se encargará de comunicarse vía USB con nuestro ordenador. Al ser un dispositivo de interfaz humana HID, este no necesita de instalación sobre sistema operativo Windows XP; es decir es un dispositivo “Plug & Play”.

El software que se encarga de controlar el rotor, fue diseñado en Visual Basic 6, sin embargo, la programación HID soporta diversos lenguajes tales como C++ y C#. Nuestro software, es un programa de prueba, el cual cumple con los requisitos necesarios para poder observar el funcionamiento del sistema completo y hacer simulaciones de seguimientos de satélites por medio del software de seguimiento WinOrbit. La comunicación entre nuestro software y WinOrbit se hace por medio del protocolo de intercambio dinámico de datos DDE.

Que es WinOrbit? ¹

WinOrbit es un software que se encarga de computar la posición y visibilidad de satélites artificiales terrestres por medio de la descripción general matemática del movimiento del satélite, y el conocimiento de algunos parametros fundamentales del satélite a seguir. Es un software Libre, diseñado para sistemas desde Windows 3.1 en adelante, para la realización de proyectos estudiantiles, radioaficionados o de investigación. Sus principales funciones son:

- Exhibición gráfica de la posición de satélites en modo tiempo real y en modo simulación.
- Exhibición tabular de la posición de satélites en ambos modos.
- Generación de tablas (calendarios astronómicos), para predicción y análisis de órbitas basados en los satélites.
- Salida de datos para programas clientes para controlar el seguimiento de satélites ya sea por medio de antenas, radios, telescopios, etc.

Que es el protocolo DDE?²

El protocolo de Intercambio Dinámico de datos o DDE (Dynamic Data Exchange) es uno de los métodos de comunicación entre procesos o programas soportado por Microsoft Windows que permite intercambiar datos entre sus aplicaciones. De esta forma, dos programas que sean compatibles y estén instalados bajo aplicación Windows pueden realizar una comunicación DDE enviándose mensajes entre ellos. Entre estos dos programas, uno es el servidor y el otro es el cliente. El servidor es aquel que tiene acceso a información útil para otros programas. El cliente, es aquel que necesita y

¹ Tomado de Web oficial de WinOrbit <http://www.sat-net.com/winorbit/>

² Basado en el texto: Laboratorio Virtual para la enseñanza de Automatización e instrumentación industrial, Jesús A. Calderón-Vielma, Universidad de los Andes, Mérida-Venezuela.

obtiene datos desde el servidor. En nuestro caso, la aplicación Servidor es WinOrbit y el software nuestro es la aplicación Cliente.

Una conversación DDE se inicia con el programa que actúa como cliente. Este transfiere un mensaje a todos los programas que se están ejecutando en ese momento en Windows. Dicho mensaje indica una categoría general de datos que el cliente necesita. Un servidor DDE que posee dichos datos puede responder a este mensaje. En este instante comienza la conversación. Un único programa puede ser cliente para un programa, y servidor para otro, pero esto requiere dos conversaciones DDE distintas. Un servidor puede entregar datos a múltiples clientes y un cliente puede obtener datos desde múltiples servidores, pero esto requiere múltiples conversaciones DDE.

La ventaja de la comunicación DDE es que no se necesita de codificación especial para transferir datos, ya que fue creada, para facilitar la comunicación entre aplicaciones Windows. El software de monitoreo Aplicación ICU-G-5500, es una aplicación sencilla la cual de manera grafica, nos muestra en tiempo real, la posición de azimut y elevación actual de la antena, así como los valores de azimut y elevación entregados por el WinOrbit.

Figura 18 Aplicación ICU-G-5500.



En la parte izquierda, el programa nos muestra el valor del azimut (por medio de la brújula) y la elevación actual de la antena. Además nos permite cambiar el valor de la tolerancia para el seguimiento de la antena entre ± 1 , 2, 3, 4, 5 y 10 grados. Esto nos permite hacer más o menos preciso el movimiento de las antenas, dependiendo de las necesidades del proyecto.

A la derecha, se observan los datos tomados del WinOrbit. Los valores de azimut (en la brújula), elevación, el nombre del satélite, la frecuencia del enlace de subida y la frecuencia del enlace de bajada. Este software es de monitoreo y sirve para intercambiar datos entre WinOrbit y el hardware ICU-G-5500, ya que el control lo hace dicho hardware.

Descripción del programa en Visual Basic

El programa consta de tres fases en las cuales realiza las labores de monitoreo y envío de datos hacia el controlador.

Como primera medida, el software, al ser cargado, realiza la comunicación DDE con WinOrbit. Éste debe estar abierto y la comunicación DDE habilitada. Luego envía una trama hacia el control, para saber si este está conectado o no y saber si la computadora ya lo reconoció como dispositivo HID. En el momento en el que el controlador le responde con la trama de conexión afirmativa, se inicia el proceso de transmisión y recepción de datos.

El programa toma los valores del WinOrbit como palabras, por lo tanto para poderlos enviar directamente hacia el micro, es necesario primero convertirlo a dos bytes, proceso que se hace antes de ser enviados. Primero se espera la petición del micro, si éste envía en un 1 en el byte de control el PC se alista para recibir el rango de azimut, en caso de ser un 2 se recibirá el rango de elevación, 3 es para recibir el azimut y elevación actual y 4 para que los datos se comparen y que todo haya llegado bien.

Por último, como tercera fase, el PC envía los valores que han sido tomados por WinOrbit. Cuando el control envía un 5 en su byte de control, el PC se alista para enviar los datos de azimut y elevación tomados del WinOrbit y la tolerancia que puede ser cambiada por el usuario.

El formato para la comunicación de control se lista a continuación:

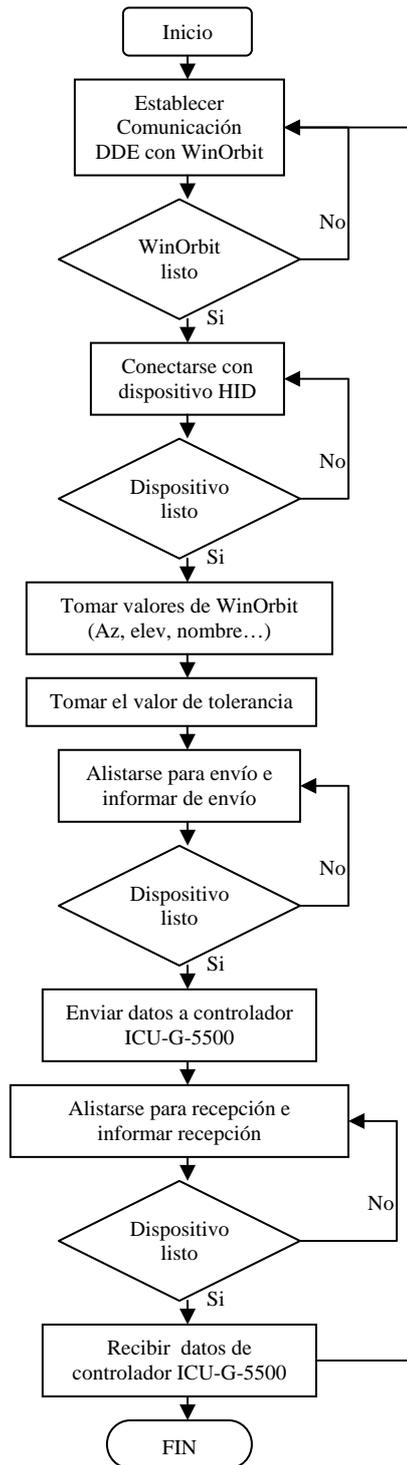
El tamaño del buffer con el que se realiza la comunicación USB es de 8 Bytes. Buffer[0:7]= Hardware y Buffer[1:8]= PC.

Cuadro 6 Descripción de la comunicación PC-Hardware

DATO ENVIADO POR EL HARDWARE	DATO DE RESPUESTA DEL PC	ACCION DESENCADENADA
BUFFER(0)= 0	BUFFER(8)=50	Comprobación de conexión PC-Hardware de control
BUFFER(0)= 2	BUFFER(2)=0:1	Calibrado
BUFFER(0)= 3 BUFFER(3,4)=Azimut actual. BUFFER(5,6)=Elevación actual.	No hay datos de respuesta.	Software imprime en pantalla la posición actual del rotor.
BUFFER(0)= 5	BUFFER(4,5)=Azimut deseado. BUFFER(6,7)=Elevación deseado. Datos de WinOrbit.	Entrega el PC al Hardware los datos a donde se debe ubicar el rotor en tiempo real.
BUFFER(0)=7 BUFFER(1)=1	Datos de WinOrbit para hacer comparación.	Modo automático.
BUFFER(0)=7 BUFFER(1)=0	Datos de controles manuales. BUFFER(3:4)=0 (STOP) BUFFER(3)=1 (UP) BUFFER(3)=2 (DOWN) BUFFER(4)=1 (LEFT) Buffer(4)=2 (RIGHT)	Modo manual.

A continuación se muestra el diagrama de flujo, donde se observa la secuencia que realiza el programa en el proceso de envío y recepción de datos. Este proceso se repite continuamente y se actualiza cada 1mseg es decir a una frecuencia de 1 kHz.

Figura 19 Diagrama flujo del programa en Visual Basic



Descripción del programa en el Microcontrolador

El micro controlador, por su parte debe llevar a cabo un mayor número de tareas. La interfaz de control ICU-G-5500 es la encargada de realizar el control de las antenas, debe estar verificando la conexión con el computador y la conexión con el rotor, también debe mostrar los datos en un LCD, y responder a interrupciones manuales realizadas desde su hardware por medio de los botones.

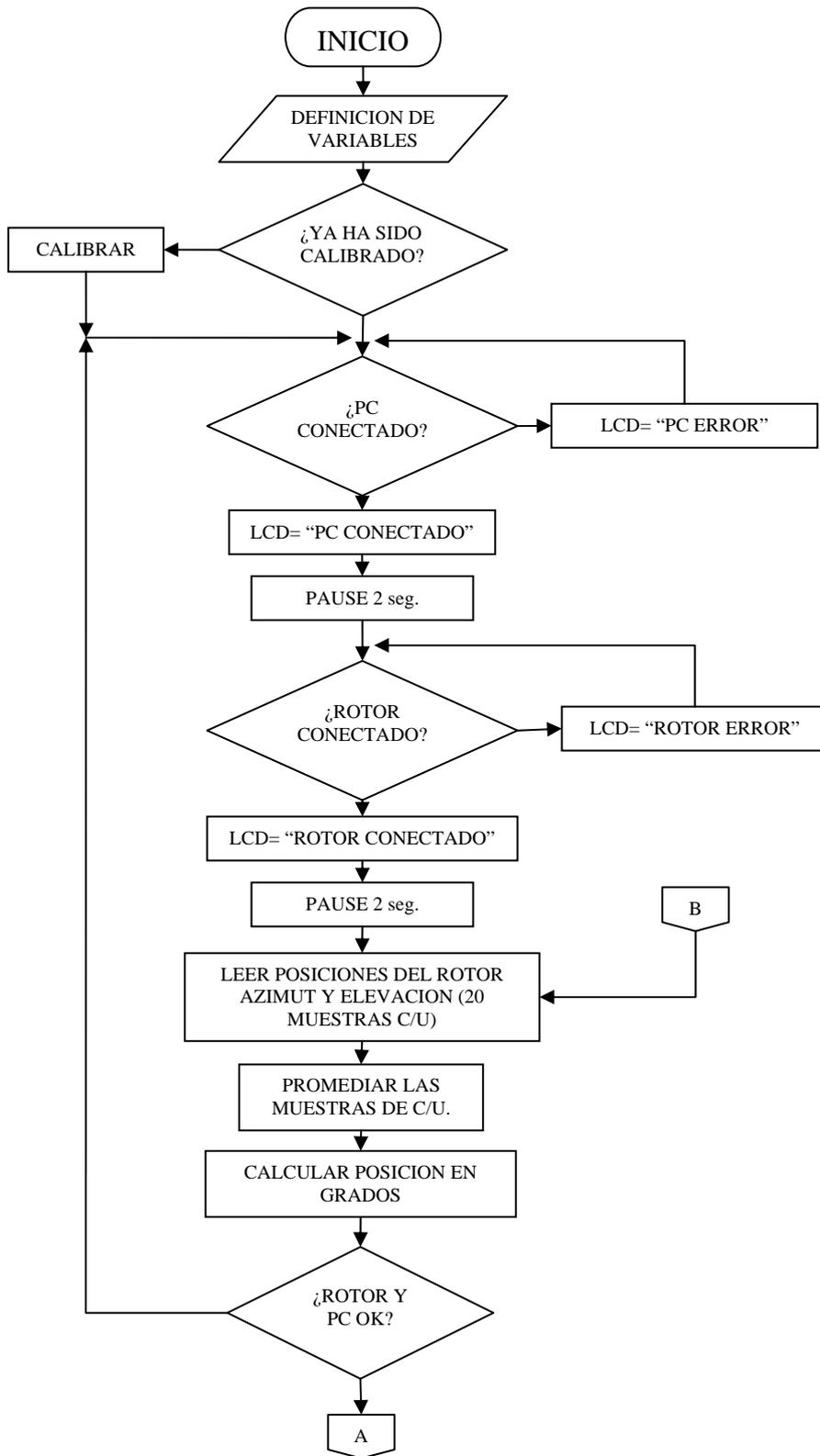
En primer lugar, al encender la interfaz, el microcontrolador se encarga de asignarle un valor inicial a cada uno de sus puertos. Esto se hace para evitar algún movimiento al hacer reset en el dispositivo y para tener un punto de partida. Luego de hacer esto, el controlador pasa a calibrar el rotor. Como sabemos el voltaje entregado por el rotor que nos indica su posición tanto en azimut como elevación puede ser variado por medio de unos potenciómetros externos para su calibración. Por esta razón decidimos dejarlos en un punto fijo y realizar la calibración de forma interna en el microcontrolador. Lo que se hace es lo siguiente:

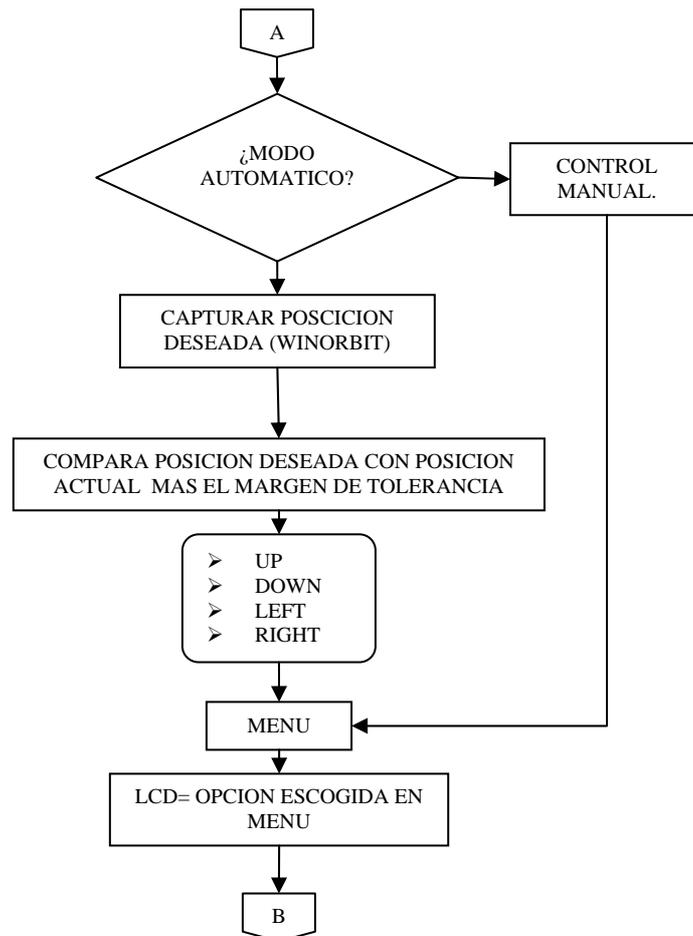
Se lleva el rotor en su posición de azimut hasta cero, cuando se reciba un valor igual o cercano a cero voltios el movimiento del rotor se detiene y se toma el dato del valor obtenido. Este dato es almacenado en la EEprom del micro. Luego se hace girar en azimut al rotor hasta llegar a su valor máximo. Esto se hace esperando un tiempo determinado. Sabemos que el rotor tarda 58 segundos en ir de cero a 360° según sus especificaciones. Por tanto para ir de 0 a 450° tardará aprox. 73 segundos. En nuestras medidas observamos que fueron entre 73 y 74 seg., este ultimo es el tiempo que dejamos de calibración al rotor para ir de 0 a 450°. Cuando han pasado este tiempo, se toma nuevamente la lectura enviada por el rotor y es guardada también en la EEprom. Luego de calibrar el azimut, hacemos el mismo proceso con la elevación. Se lleva de 0 a 180° y se toman los valores medidos en los extremos. Luego de tener estos valores medidos, obtenemos el rango de azimut y elevación. Esto se hace restando el valor máximo obtenido menos el valor mínimo, dividido por 450 en el caso del azimut y 180 en el caso de la elevación. Con esto sabremos cuantos milivoltios equivalen a cada grado en el rotor y no importará en que valor esta calibrado el potenciómetro del rotor.

Todos estos valores son almacenados en la EEprom, de tal forma que en caso de apagar y reiniciar el sistema no se perderán y la calibración no se tendrá que volver a hacer a menos de que se le de la orden de hacerlo.

Luego de esto, el micro comienza la rutina de control. Los datos recibidos desde el WinOrbit los llamamos posiciones deseadas, y los datos tomados del rotor se conocen como posiciones actuales. El controlador inicia, digitalizando las señales tomadas del rotor y enviándolas hacia el ordenador. Este a su vez las recibe y las pone en pantalla.

Figura 20 Diagrama flujo del programa para el microcontrolador





Seguido, el micro procede a pedir la información obtenida de WinOrbit al PC. Al recibir estos datos, el microcontrolador compara la posición actual con la posición deseada. Dependiendo de hacia donde se deba mover el rotor, el control enviará señales a los circuitos de salida, para que estos a su vez muevan el rotor a la posición deseada. La precisión para alcanzar el punto deseado dependerá de la tolerancia escogida en el software del computador.

3.3. COMUNICACIÓN USB³

El puerto USB (Universal Serial Bus) es una interfase plug & play entre la PC y ciertos dispositivos tales como teclados, ratones, scanner, impresoras, módems, placas de sonido, cámaras, etc.). Una característica importante es que permite a los dispositivos trabajar a velocidades mayores, en promedio a unos 12 Mbps, esto es más o menos de 3 a 5 veces más rápido que un dispositivo de puerto paralelo y de 20 a 40 veces más rápido que un dispositivo de puerto serial.

Como Funciona:

³ Basado en documentos de <http://www.usb.org/developers/docs/>

Trabaja como interfaz para transmisión de datos y distribución de energía, que ha sido introducida en el mercado de PC's y periféricos para mejorar las lentas interfaces serie (RS-232) y paralelo. Esta interfaz de 4 hilos, 12 Mbps y "plug & play", distribuye 5V para alimentación, transmite datos y está siendo adoptada rápidamente por la industria informática.

Es un bus basado en el paso de un testigo. El controlador USB distribuye testigos por el bus. El dispositivo cuya dirección coincide con la que porta el testigo responde aceptando o enviando datos al controlador. Este también gestiona la distribución de energía a los periféricos que lo requieran.

Emplea una topología de estrellas apiladas que permite el funcionamiento simultáneo de 127 dispositivos a la vez. En la raíz o vértice de las capas, está el controlador anfitrión o host que controla todo el tráfico que circula por el bus. En la actualidad, la mayoría de Main Boards (tarjeta madre o placa principal), incluyen un controlador USB integrado en el chipset. Esta topología permite a muchos dispositivos conectarse a un único bus lógico sin que los dispositivos que se encuentran más abajo en la pirámide sufran retardo. A diferencia de otras arquitecturas, USB no es un bus de almacenamiento y envío, de forma que no se produce retardo en el envío de un paquete de datos hacia capas inferiores.

El sistema de bus serie universal USB consta de tres componentes:

- * Controlador
- * Hubs o Concentradores
- * Periféricos

Estándares

La tecnología USB, se creó, al ver la necesidad de encontrar una manera rápida de conectar diferentes dispositivos con las computadoras, y que soportara hacer esto incluso sin la necesidad de reiniciar el ordenador. Aunque inicialmente, se pensaba utilizar en servidores, para la integración de dispositivos telefónicos. La idea nació de las grandes compañías fabricantes de software y computadoras, tales como Intel, Macintosh, HP, entre otras. A continuación mostramos como a lo largo de la evolución del USB, se han ido actualizando los estándares:

- **USB 0.9:** Primer borrador, publicado en Noviembre de 1995.
- **USB 1.0:** Publicada en 1996 establece dos tipos de conexión: La primera, denominada **velocidad baja** ("Low speed"), ofrece 1.5Mbps, y está pensada para periféricos que no requieren un gran ancho de banda, como ratones o joysticks. La segunda, denominada **velocidad completa** ("Full speed"), es de 12Mbps, y está destinada a los dispositivos más rápidos.
- **USB 1.1:** Publicada en 1998, añade detalles y precisiones a la norma inicial; es el estándar mínimo que debe cumplir un dispositivo USB.

- **USB 2.0:** Su versión final fue publicada en Abril del 2000; es una extensión de la norma compatible con las anteriores. Permite velocidades de hasta 480Mbps, denominada **alta velocidad** ("High speed").

Cables y conectores:

El puerto USB maneja 4 hilos. Cada hilo tiene una función específica y están distribuidos de la siguiente manera:

Cuadro 7 Descripción de funciones de los pines del conector USB

Pin	Nombre del Pin	Descripción	Color
1	V Bus	+5V alimentación para periféricos	Rojo
2	D-	Datos -	Azul
3	D+	Datos +	Amarillo
4	GND	Tierra	Verde

El hecho de que el bus trabaje dos líneas de datos se debe a su codificación. El bus serie USB es síncrono, y utiliza el algoritmo de codificación NRZI ("Non Return to Zero Inverted").

Figura 21 Conectores USB Tipo A y B



Tomado de:

http://www.zator.com/Hardware/H2_5_3.htm

Hay tres tipos de conectores certificados para el Puerto USB. Dos de estos son para periféricos comunes (impresoras, Mouse, teclados, memorias, etc.) y uno reducido, el mini-USB utilizado frecuentemente en dispositivos de pequeño tamaño y alta tecnología (celulares, Palm, cámaras digitales, etc.).

El conector tipo A está diseñado para concentradores, computadoras es decir, dispositivos Host (que tienen capacidad de soportar periféricos). El conector tipo B Es el que comúnmente se encuentra en los periféricos o dispositivos 'esclavo'.

3.3.1. USB 2.0

Que es USB 2.0:

Finalizado en el 2001, el Puerto serial universal USB 2.0 es un reacondicionamiento completo del protocolo de entrada - salida USB 1.0-1.1

que le permite muchas mas altas velocidades. La meta del nuevo bus serial es ensanchar toda la gama de periféricos externos que se pueden utilizar en una computadora. Una unidad de disco duro vía USB, puede bloquear fácilmente el USB 1.0, mientras que puede ser mucho más manejable en un USB 2.0

Que le sucedió al USB 1.1

USB 1.1 brindaba un máximo de transferencia de datos de 12Mbits/seg. Esto ahora es obsoleto, pero ambas de sus velocidades (1.5Mbps y 12Mbps) están siendo adoptadas y soportadas por USB 2.0, lo cual es llamado USB ORIGINAL. Sin embargo muchos de estos dispositivos están etiquetados como full-speed. Note que esto puede ser un poco engañoso ya que es fácil confundir full-speed, con Hi-Speed. La explicación es sencilla, full-speed se refiere a USB 1.1 a 12Mbps y Hi-speed se refiere a USB2.0 que alcanza velocidades de 480Mbps, que son mucho mayores.

Por el momento los 1.5Mbps habían sido más que suficientes para dispositivos como Mouse y teclados, pero poco a poco el mercado fue cambiando, creando dispositivos usb bluetooth, cámaras de video que transmiten vía usb y dispositivos de un mayor rendimiento que requerían de al menos 12Mbps.

Que tan rápido es USB 2.0

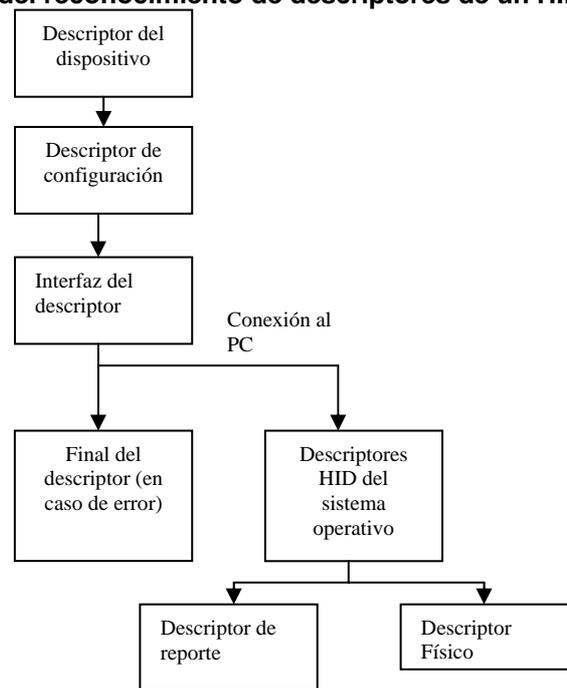
El usb 2.0 tiene una tasa de transmisión de información bruto de 480Mbps, y es considerado 40 veces mayor a su interfaz predecesor usb 1.1. Originalmente fue pensado para trabajar velocidades tan rápidas como 240Mbps, pero en Octubre de 1999, cuando un pequeño cambio en la programación del protocolo hizo bombear información un poco encima de la velocidad de 480Mbps.

En la realidad la tasa eficaz de transmisión estándar es de 40MBps o 320Mbps para una transmisión limpia sin usar Hubs usb.

3.3.2. Dispositivos de Interfaz Humana (HID)

Un dispositivo de interfaz humana (HID) nos permite fácilmente conectar un dispositivo externo a nuestro PC usando el puerto USB. Una de las mayores ventajas del HID es que no se necesita de drivers en los dispositivos, ya que estos son entregados por el sistema operativo (Windows, MacOS y/o Linux), razón por la cual, rápidamente desplazaron a dispositivos más lentos y tediosos de instalar como los dispositivos RS 232 y paralelos.

Figura 22 Diagrama del reconocimiento de descriptores de un HID.



Tomado de:

Device class definition for Human Interface Devices (HID), version 1.1, descargado de <http://www.usb.org/developers/docs/>

La interfaz HID funciona con un simple concepto. Los sistemas operativos tienen guardados, en sus bases de datos, todas las características acerca de los diferentes dispositivos HID, que son los llamados drivers. Igualmente, cada dispositivo que es creado y programado para una interfaz HID, guarda en su ROM (Read Only Memory) las características propias de su naturaleza, que harán que la PC lo reconozca, características comunes a otros dispositivos (descriptores del dispositivo), y características propias de el (descriptores de configuración). Por ejemplo en una memoria USB, la característica común es que es un dispositivo de almacenamiento como todas las memorias, y características propias a el, serían la maraca, el formato de almacenamiento, etc.

En el momento de conectar el dispositivo al puerto USB, el PC lo reconoce como dispositivo de interfaz humana y compara los datos que éste trae en su ROM con los de la base de datos (drivers). De esta forma el sistema operativo reconoce si el dispositivo es una cámara, un Mouse, una memoria, etc. La primera vez que el dispositivo es conectado la PC tarda unos segundos instalando el dispositivo, es decir reconociéndolo, luego guarda esos datos de tal forma que la próxima vez que se conecte este dispositivo, el PC lo reconozca inmediatamente. Estos datos son guardados en dos bases de datos del sistema una llamada Descriptor de reporte y la otra Descriptor físico. A estas características de cada dispositivo se les llama descriptores.

3.4. COMPATIBILIDAD Y APLICABILIDAD

Este sistema, al ser creado como un dispositivo HID, es compatible con cualquier sistema operativo que soporte este tipo de interfaz. Además el software creado para su monitoreo y control, esta hecho en Visual Basic 6 de Microsoft lo cual hace que sea 100% compatible con sistemas operativos Windows.

La aplicación es múltiple ya que en nuestro caso, estamos utilizando este para el control de antenas para satélites CUBESAT, pero también puede ser utilizado para controlar visores telescópicos, o sintonizar diferente satélites geoestacionarios. El sistema esta diseñado para funcionar con rotores yaesu g-5500, pero igualmente puede ser modificado para soportar otro tipo de rotores.

4. IMPLEMENTACION Y PROYECCION

La implementación de este proyecto, esta enfocada a una futura estación terrena la cual estará ubicada en alguna de las sedes de la universidad Surcolombiana. El propósito es llevar a cabo comunicaciones entre la base terrena USCO y la base terrena Universidad Distrital. El objetivo, es incentivar a los estudiantes en campos de investigación, que son un poco desconocidos para la comunidad estudiantil en Colombia, como lo son las comunicaciones satelitales, y en base a los avances que se logren con este proyecto, poder pensar en un futuro, en lanzar nuestro propio satélite, ya sea un satélite tipo CUBESAT, o cualquier otro tipo de satélite que nos ayude a dar un paso más en este campo.

La implementación de esta base terrena, esta propuesta para finales del presente año. El tiempo de vida del satélite en el espacio es menor o igual a 3 años, tiempo durante el cual, gran cantidad de estudiantes e ingenieros, podrán hacer estudios, tomar datos, analizar diferentes aspectos tanto en el campo del control en la base terrena, como en el campo de las telecomunicaciones.

5. COSTOS

Definiremos los costos del presente proyecto, teniendo en cuenta los costos físicos de este, así como los costos del trabajo realizado. Hay que tener en cuenta que el software desarrollado por nosotros, es una versión básica, y que el programa completo que deberá manejar este sistema, esta siendo desarrollado por otros estudiantes como tesis de grado.

Figura 23 Costos de la implementación del control físico de las antenas.

CANTIDAD	DESCRIPCIÓN	PRECIO
1	Computador para monitoreo	\$4'000.000
1	Software ICU G-5500	\$200.000

1	Rotor Yaesu G-5500	\$1'550.000
1	Interfaz ICU G-5500	\$600.000
	Cables y conectores	\$300.000
1	Mástil	\$500.000
TOTAL		\$7'150.000

El valor de los cables y conectores, puede variar de acuerdo con la distancia que hay desde el cuarto de control hasta la antena. El presupuesto esta basado para una distancia máxima de 30 metros. Igualmente, el mástil puede variar de tamaño y valor.

CONCLUSIONES

- Pensando en la estabilidad y velocidad a la hora de sintonizar el satélite, se realizó la interfaz utilizando el puerto USB, el cual es de última generación y es capaz de manejar altas velocidades de transmisión.
- El control realizado aun al ser un control muy aproximado es impreciso, debido a que es un sistema de lazo abierto, es decir, el sistema no tiene una señal de realimentación por parte del satélite para determinar si existe un total contacto y una máxima potencia de transferencia. Esto aunque lo puede hacer impreciso lo hace de una alta velocidad de respuesta, aspecto que se ve altamente afectado si se hiciese en lazo cerrado ya que el sistema empezaría a realizar ubicaciones aleatorias en el firmamento hasta encontrar el satélite, acción que puede tardar demasiado o puede llegar a no lograrse. Para que esto sea de la mejor manera sería conveniente combinar juntos sistemas, tanto el de lazo abierto donde la posición es calculada por un software y la característica de lazo cerrado, que al encontrar al satélite en una posición determinada por el software, realice ahora el seguimiento buscando siempre la mayor intensidad de potencia que emite el satélite.
- En el mercado se encuentran diferentes interfaces para controlar rotores desde el PC, las cuales observamos, tienen grandes defectos y desventajas con respecto a la nuestra, ya que sus circuitos de salida son basados en relevos y sus conexiones hacia la computadora son hechos por medio del puerto paralelo.
- En vista de las limitaciones de tecnología, herramientas y conocimientos con que contamos para poder realizar un diseño mecánico e implementarlo se decidió adquirir el yaesu G-5500 obteniendo todas las características necesarias de torque, velocidad y demás que requiere para poder mover las antenas, esto nos dice que para nuestro caso como estudiantes de ingeniería electrónica es propicio dedicarnos a lo nuestro, el control, debido a que los resultados que hubiésemos podido alcanzar con un sistema mecánico artesanal no hubiese sido el mismo al logrado.
- Inicialmente, se pensó en crear un software de predicción, similar al WinOrbit. Debido a faltas de recursos y de capacitación para poder hacerlo, utilizamos este ya mencionado software. Nos parece innecesario realizar un software de este tipo, ya que éste a parte de ser gratuito, presta la función de protocolo DDE para comunicarse con cualquier otro software bajo Windows.

RECOMENDACIONES

- A futuro, podría pensarse en un control, basado no solo en el software de predicción, sino también midiendo la intensidad de señal recibida por la antena, para así lograr una mejor transmisión de datos, además teniendo en cuenta la baja potencia de la señal emitida por el satélite, la recepción debe ser lo más eficiente posible.
- Para mejorar la velocidad de conmutación en los circuitos del rotor, se podría cambiar el sistema de encendido/apagado del rotor, ya que este es realizado mediante contactos electro mecánicos (relevos), y cambiarlos por dispositivos de conmutación de estado sólido, tales como transistores de potencia, tiristores o triacs, que además generan mucho menor ruido de conmutación.
- El dispositivo interfaz ICU G-5500 está diseñado para trabajar con el rotor propuesto para este proyecto. En caso de querer comercializarlo, podrían modificarse algunos de sus circuitos, para así poder soportar diferentes tipos de rotores como lo son los circuitos de captura de posición.

GLOSARIO

- Azimut: Es el ángulo que forma el círculo vertical que pasa por un punto de la esfera celeste o del globo terráqueo con el meridiano. Es una de las dos coordenadas del sistema altacimutal o sistema de coordenadas astronómicas horizontales.
- Cubesat o CUBESAT: Cube: cubo, sat: satélite
- DDE: Protocolo de Intercambio Dinámico de datos o DDE (Dynamic Data Exchange) es uno de los métodos de comunicación entre procesos o programas soportado por Microsoft Windows que permite intercambiar datos entre sus aplicaciones.
- Elevación o altura: distancia angular de un cuerpo celeste sobre el horizonte, es decir, el ángulo entre el plano del horizonte y la línea que une al observador y al cuerpo celeste, o el arco de circunferencia vertical que se extiende entre el cuerpo celeste y el horizonte.
- Encoder: Codificador angular de posición, se encargan de convertir un movimiento angular, en una señal eléctrica codificada.
- Geoestacionario: Satélite ubicado en la órbita GEO que gira alrededor a la misma velocidad de rotación de la tierra
- GITUD: Grupo de Investigación en Telecomunicaciones Universidad Distrital
- HID: Human Interfaz Device, Dispositivos de Interfaz Humana
- ICU G-5500: Dispositivo de Interfaz de Control Usco G-5500, encargado de comunicar el actuador de las antenas con el software de monitoreo.
- IEEE_UD: Grupo IEEE Universidad Distrital
- LCD: Liquid Cristal Display, Pantalla de Cristal Líquido.
- Mbps: Tasa o tasa de transmisión de datos medida en número de bits por segundo.
- MBps: Tasa o tasa de transmisión de datos medida en número de Bytes por segundo.
-

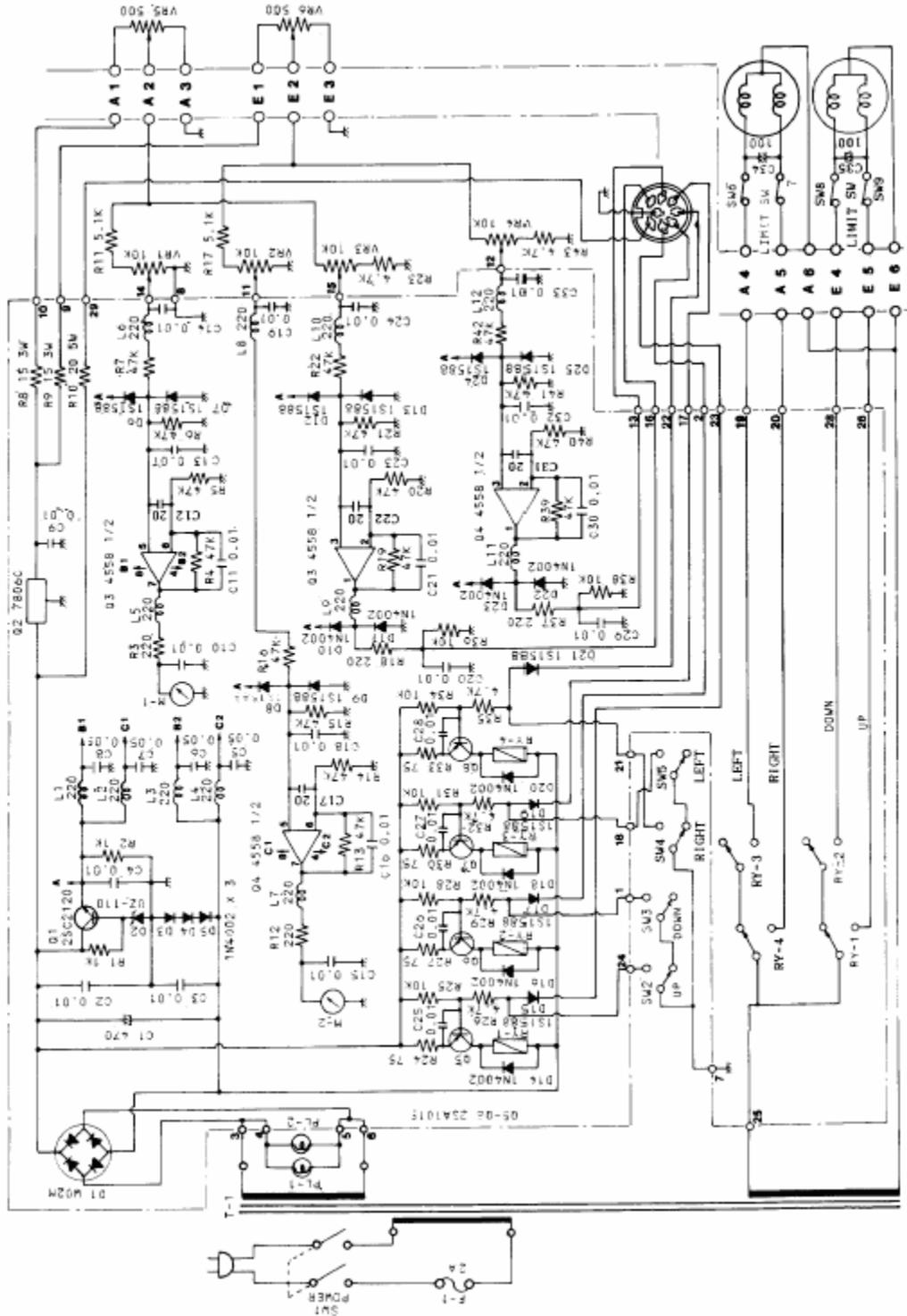
- Órbita LEO: Low Earth orbit, Órbita terrestre baja, comprendida entre 600 y 1200 Km.
- PC: Personal Computer o computador Personal, ordenador, computadora.
- Plug & Play: Tecnología integrada en sistemas operativos, para el reconocimiento y utilización inmediata de dispositivos periféricos
- RF: Radio Frequency, Señales moduladas en la banda de señales de Radio Frecuencia.
- USB: Universal Serial Bus, Bus o puerto Serial Universal

BIBLIOGRAFÍA

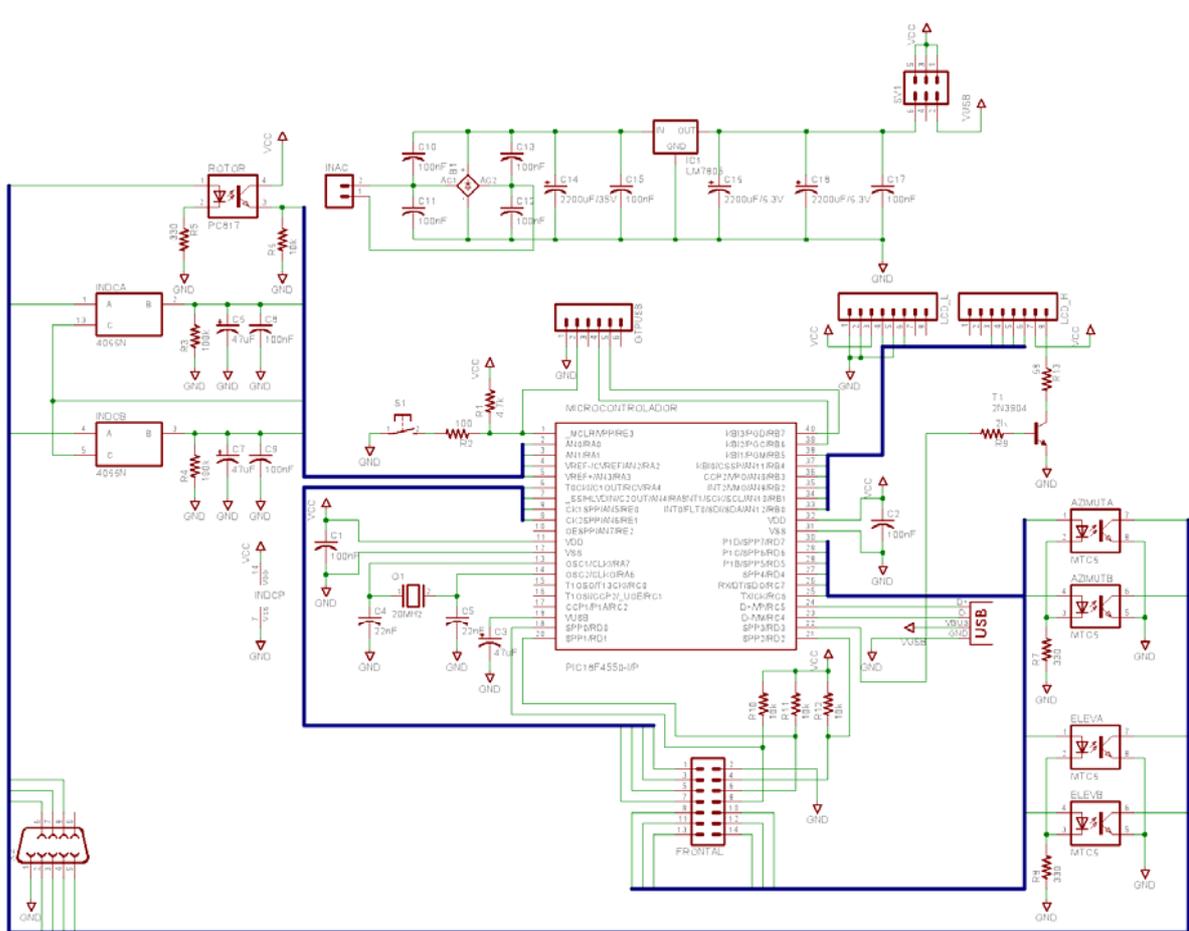
- FUNDAMENTOS DE ROBÓTICA, Antonio Barrientos, Luis Felipe Peñín, Carlos Balaguer, Rafael Aracil. Editorial McGraw-Hill.
- INSTRUCTION MANUAL G-5500, Vertex Standard Co, LTD.
- DEVICE CLASS DEFINITION FOR HUMAN INTERFACE DEVICES (HID), version 1.11. 1996-2001 USB Implementers' Forum—All rights reserved.
- TRACKING, TELEMETRY, AND COMMAND TEAM PDR REPORT, UH CubeSat Project, College of Engineering. University of Hawaii at Manoa
- A TECHNICAL INTRODUCTION TO USB 2.0,
<http://www.usb.org/developers/docs/>
- GROUND STATION SUBSYSTEM, The Ground Station Team is responsible for construction of MEROPE's ground control in Bozeman.
- <http://www.astromia.com>
- <http://www.mecanique.co.uk/>
- http://picmania.garcia-cuervo.com/USB_0_Desencadenado.htm
- <http://www.todopic.com.ar/foros/index.php?topic=13406.20>
- <http://www.astromia.com/glosario/mecanicaceleste.htm>
- http://ar.geocities.com/seti_argentina/mecacel/mecesa.html
- <http://www.xe1rcs.org.mx/programas/diversos.html>
- <http://celestrak.com/NORAD/elements/master.asp>

ANEXOS

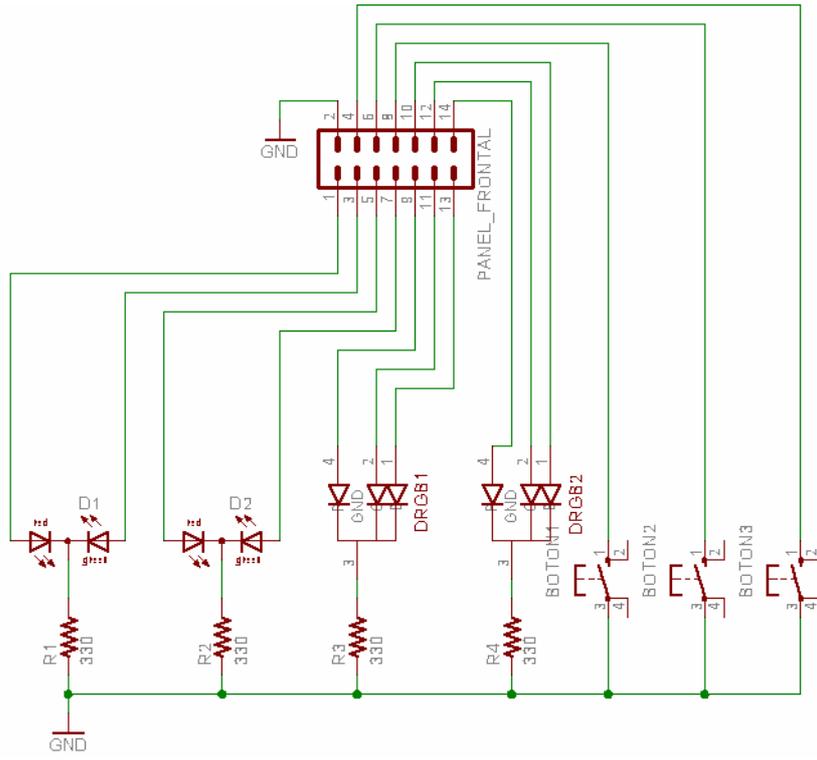
ANEXO A Diagrama esquemático del Rotor Yaesu G-5500



ANEXO B Esquemático completo del dispositivo interfaz ICU G-5500. Circuito principal y circuito frontal.

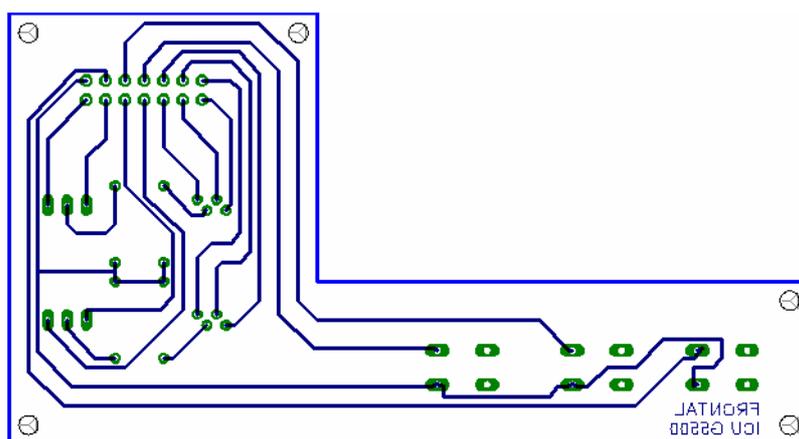
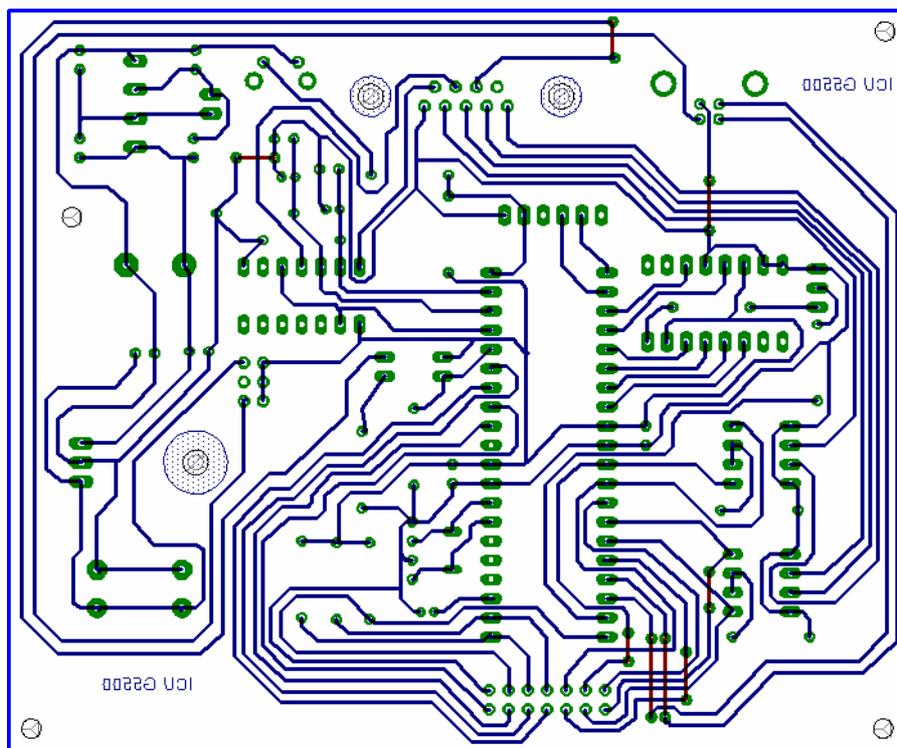


Circuito principal microcontrolado.

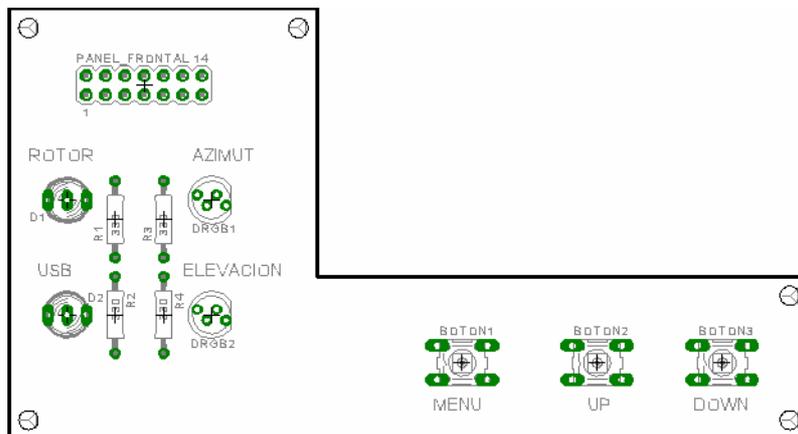
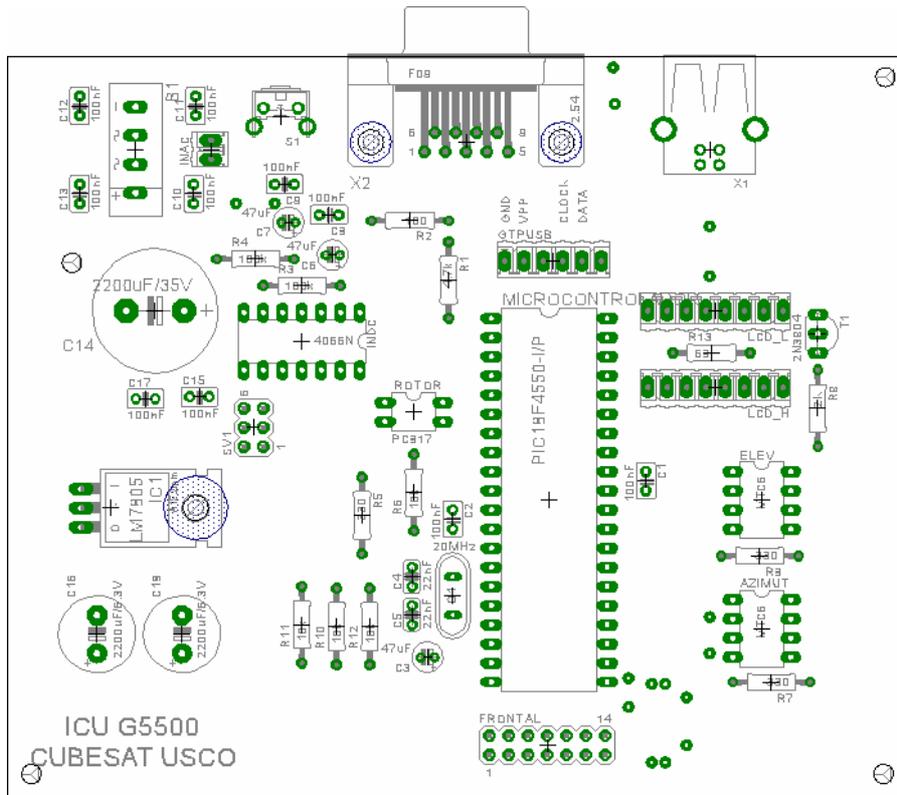


Esquemático del circuito frontal de control e información.

ANEXO C Circuitos impresos de la interfaz ICU-G-5500. Circuitos impresos principal y frontal.



ANEXO D Distribución de los elementos en las placas finales del ICU G-5500.



ANEXO E Código en Basic para el Microcontrolador.

Archivo de declaración de variables y definiciones.

```
*****
!* Name : Variables y configuración de pines.BAS *
!* Author : Control de movimiento antena CUBESAT-USCO *
!* Date : 20/04/2007 *
!* Version : 1.0 *
!* Notes : *
!* : *
*****

DEFINE OSC 48 'Oscilador al que trabajara internamente el pic para.
              'la comunicación USB.
DEFINE LCD_DREG PORTB 'Puerto encargado de manejar el LCD-> PORTB.
DEFINE LCD_DBIT 0 'Bus de datos para el LCD en RB0:RB3.
DEFINE LCD_RSREG PORTB 'Bit 4 (RB4) como RS.
DEFINE LCD_RSBIT 4
DEFINE LCD_EREG PORTB 'Bit 5 (RB5) como E.
DEFINE LCD_EBIT 5
DEFINE LCD_BITS 4 'Bus de datos de 4 bits.
DEFINE LCD_LINES 2 'Modulo LCD de 2 lineas.

DEFINE ADC_BITS 10 'Set number of bits in result
DEFINE ADC_CLOCK 3 'Set clock source (rc = 3)
DEFINE ADC_SAMPLEUS 50 'Set sampling time in microseconds

*****
'Buffers para la comunicación USB *
*****
USBBufferSizeMax con 8 ' Máximo tamaño del buffer USB
USBBufferSizeTX con 8 ' Buffer Usb de entrada
USBBufferSizeRX con 8 ' Buffer Usb de salida
' the USB buffer...
USBBuffer Var Byte[USBBufferSizeMax]
USBBufferCount Var Byte
*****
'Designación de nombres para los pines del PIC18F4550 *
*****
VALOR_AZIMUT VAR PORTA.0 'Entrada analógica 1, señal de la posición Azimut.
VALOR_ELEVACION VAR PORTA.1 'Entrada analógica 2, señal de posición Elevación.
CONTROL VAR PORTA.2 'Bit de control de las señales análogas.
ROTOR VAR PORTA.3 'Bit del estado de conexión del rotor.
LED_CONEX_USB VAR PORTA.4 'Indica que La conexión esta establecida con el PC.
LED_NOCONEX_USB VAR PORTA.5 'Indica que La conexión no esta establecida con el
PC.

'PORTB Puerto que maneja el Modulo LCD.
'PORTB.0:3 B0:B3 Bus de 4 bits.
'PORTB.4 RS.
'PORTB.5 E.

AZIMUT_STOP VAR PORTC.6 'Indicador de parada de motor de azimut.
AZIMUT_RIGHT VAR PORTC.7 'Azimut hacia la derecha.

BOTON_1 VAR PORTD.0 'Botón multipropósito 1.
BOTON_2 VAR PORTD.1 'Botón multipropósito 2.
```

```

BOTON_3      VAR PORTD.2 'Botón multipropósito 3.
LUZ_LCD      VAR PORTD.3 'Bit de encendido de la luz de modulo LCD.
AZIMUT_LEFT  VAR PORTD.4 'Azimut hacia la izquierda.
ELEVACION_STOP VAR PORTD.5 'Led indicador de rotor de elevación quieto.
ELEVACION_DOWN VAR PORTD.6 'Elevación hacia abajo.
ELEVACION_UP  VAR PORTD.7 'Elevación hacia arriba.

LED_ROTOR_ERROR VAR PORTE.0 'Indica que el rotor esta conectado.
LED_ROTOR_NOERROR VAR PORTE.1 'Indica que el rotor no esta conectado.
*****
'Variables usadas en el cuerpo del programa.
*****
PC_CONECTADO  VAR BYTE   'Variable que presenta el estado de conexión con el PC.
TOLERANCIA    VAR WORD   'Valor en grados de tolerancia.
i             VAR BYTE   'Contador auxiliar para los ciclos FOR.
INICIO        VAR BYTE   'Variable que lleva el estado de encendido.
EPROM1        VAR BYTE
B1            var byte
B2            var byte
B3            VAR BYTE
AUTOMATICO    VAR BYTE
DIV_AZ        VAR WORD
DIV_EL        VAR WORD

AD_ADON       VAR ADCON0.0 'Bit de habilitación del conversor AD.
AD_GO_DONE    VAR ADCON0.1 'Arrancador de la conversión.
AD_CANAL      VAR ADCON0.2 'Seleccionador del canal AN0 o AN1.
ELEVACION_ACTUAL VAR WORD  'Elevación actual del rotor.
ELEVACION_DESEADA VAR WORD  'Elevación suministrada por el PC.
AZIMUT_ACTUAL  VAR WORD  'Azimut actual del rotor.
AZIMUT_DESEADO VAR WORD  'Azimut suministrado por el PC.
RESTA         VAR WORD   'Variable auxiliar para realización de restas.
PROMEDIO      VAR WORD   'Variable que lleva el promedio de 5 muestras consecutivas.
AUX1          VAR WORD[10] 'Variable que ayuda a calcular el promedio.
AUX2          VAR WORD[10] 'Variable que ayuda a calcular el promedio.
MIN_AZ        VAR WORD
MAX_AZ        VAR WORD
MIN_EL        VAR WORD
MAX_EL        VAR WORD
*****
'Configuración de los pines de entrada y salida del PIC18F4550 *
*****
'Configuración de entradas.
INPUT VALOR_AZIMUT
INPUT VALOR_ELEVACION
input rotor
INPUT BOTON_1
INPUT BOTON_2
INPUT BOTON_3
'Configuración de salidas.
OUTPUT CONTROL
OUTPUT LED_ROTOR_ERROR
OUTPUT LED_ROTOR_NOERROR
OUTPUT LED_NOCONEX_USB
OUTPUT LED_CONEX_USB
OUTPUT AZIMUT_STOP
OUTPUT AZIMUT_RIGHT

```

```

OUTPUT AZIMUT_LEFT
OUTPUT ELEVACION_STOP
OUTPUT ELEVACION_UP
OUTPUT ELEVACION_DOWN
OUTPUT LUZ_LCD
OUTPUT PORTB
*****
'Iniciación de los puertos                                     *
*****
PORTA=0
LATA=0
PORTB=0
LATB=0
CMCON=$07           'Configuración de entradas digitales y no de comparación.
PC_CONECTADO=0
TOLERANCIA=10
AZIMUT_DESEADO=0
ELEVACION_DESEADA=0
TOLERANCIA.BYTE1=0
MIN_AZ=0
MAX_AZ=0
MIN_EL=0
MAX_EL=0
INICIO=0
B1=0
B2=9
B3=0

AUTOMATICO=1
ADCON1=%00001101    'Configuro el RA0 y RA1 como entradas análogas y los
                    'demás canales como i/o digitales, Vref+=VCC, Vref-=GND,
ADCON2=%10111010    'Resultado del AD justificado a la derecha, TAD=20,F=Fosc/32
                    'para un cristal de 20MHz.

```

Código del archivo del programa main().

```
include "variables.bas"

usbinit ' initialise USB...
! *****
! * Inicia con poner en un estado de inicio todos los      *
! * bits de salida, esto se hace cuando se enciende el     *
! * sistema, cuando el micro sufra un reset o detecte     *
! * que ha perdido la conexión con el PC o el ROTOR.      *
! *****

RECONOCIMIENTO:
  low CONTROL          'Deshabilita la entrada de las señales análogas de posición.
  LOW LED_ROTOR_NOERROR
  LOW LED_CONEX_USB
  LOW AZIMUT_RIGHT
  LOW AZIMUT_LEFT
  LOW ELEVACION_UP
  LOW ELEVACION_DOWN

  HIGH LED_ROTOR_ERROR
  HIGH LED_NOCONEX_USB
  HIGH AZIMUT_STOP
  HIGH ELEVACION_STOP
  HIGH LUZ_LCD

  GOSUB ENCENDIDO
  GOSUB LEER_RANGOS
  GOSUB VERIFICAR_CONEXION_A
! *****
! * En esta rutina se mantendrá el microcontrolador siempre *
! * que no exista una novedad de desconexión, o una orden  *
! * de calibración                                         *
! *****

ProgramStart:
  GOSUB LECTURAS_ANALOGAS
  GOSUB VERIFICAR_CONEXION_B
  USBBUFFER[0]=1
  GOSUB DOUSBOUT
  GOSUB DOUSBIN
  AUTOMATICO=USBUFFER[1]
  IF (AUTOMATICO==1) THEN
  GOSUB COMPARACION_DE_POSICION
  else
  GOSUB MANUAL
  ENDIF
  USBBUFFER[0]=2
  GOSUB DOUSBOUT
  GOSUB DOUSBIN
  IF (USBUFFER[1]==1) THEN CALIBRADO
  GOSUB MENU
  GOTO PROGRAMSTART
! *****
! * Rutina que lee la EEprom para determinar si el sistema ya*
! * posee valores de calibración si no es así procede a      *
! * llevar el rotor a sus valores extremos para tomar medidas*
! * y almacenarlas debidamente.                               *
```

! *****

ENCENDIDO:

```
if (INICIO==0) then          'Esto se hace para saber si el sistema
                             'ya había pasado por esta rutina para así
                             'no repetir los mensajes de inicio, esto debido
                             'a que en un error de conexión de software
                             'o hardware se reiniciara el programa desde la
                             'etiqueta reconocimiento.
LCDOUT $FE,1,"CONTROL ICU G5500",$FE,$C0,"1. Inicio"
pause 2000
INICIO=1                    'El sistema ya esta encendido.
LCDOUT $FE,1,"1. Inicio ok!",$FE,$C0,"2. calibrado?"
pause 2000
ENDIF
read 1,EPROM1              'Se observa si ya hay datos de calibración.
IF (EPROM1==255) THEN
LCDOUT $FE,1
```

CALIBRADO:

```
USBBUFFER[0]=4
GOSUB DOUSBOUT
IF (ROTOR==1)THEN
HIGH CONTROL
low AZIMUT_STOP
HIGH AZIMUT_LEFT
MIN_AZ=1023
LCDOUT $FE,1,"CALIBRANDO",$FE,$C0,"Azimut",$7E,"0 gra."
WHILE (MIN_AZ>2)
  ADCIN 0,MIN_AZ
  USBSERVICE
WEND
PAUSE 2000
LOW AZIMUT_LEFT
ADCIN 0,MIN_AZ
HIGH AZIMUT_RIGHT
LCDOUT $FE,1,"CALIBRANDO",$FE,$C0,"Azimut",$7E,"450 gra."
PAUSE 38000
PAUSE 35000
USBSERVICE
low AZIMUT_RIGHT
HIGH AZIMUT_STOP
ADCIN 0,MAX_AZ
LOW ELEVACION_STOP
HIGH ELEVACION_DOWN
LCDOUT $FE,1,"CALIBRANDO",$FE,$C0,"Elevacion",$7E,"0 gra."
MIN_EL=1023
WHILE (MIN_EL>2)
  ADCIN 1,MIN_EL
  USBSERVICE
WEND
PAUSE 2000
LOW ELEVACION_DOWN
ADCIN 1,MIN_EL
HIGH ELEVACION_UP
LCDOUT $FE,1,"CALIBRANDO",$FE,$C0,"Elevacion",$7E,"180 gra."
PAUSE 35000
USBSERVICE
```

```

        PAUSE 35000
        USBSERVICE
        LOW ELEVACION_UP
        HIGH ELEVACION_STOP
        ADCIN 1,MAX_EL
        USBSERVICE
        gosub escribir_rangos
    ELSE
        LCDOUT $FE,2,"ESPERANDO",$FE,$C0,"CONECTAR ROTOR"
        USBSERVICE
        GOTO CALIBRADO
    ENDIF
    B1=0
    B2=9
    GOTO RECONOCIMIENTO
ENDIF
USBSERVICE
RETURN
! *****
! * Escribe datos de los rangos de Azimut y Elevación *
! * en la memoria EEprom *
! *****
ESCRIBIR_RANGOS:
    EPROM1=0
    WRITE 1,EPROM1
    WRITE 2,MIN_AZ.BYTE0
    WRITE 3,MIN_AZ.BYTE1
    WRITE 4,MAX_AZ.BYTE0
    WRITE 5,MAX_AZ.BYTE1
    WRITE 6,MIN_EL.BYTE0
    WRITE 7,MIN_EL.BYTE1
    WRITE 8,MAX_EL.BYTE0
    WRITE 9,MAX_EL.BYTE1
    write 10,(MAX_AZ-MIN_AZ)/450
    WRITE 11,(MAX_EL-MIN_EL)/180
RETURN
! *****
! * Lee los datos de los rangos de Azimut y Elevación *
! * Obtenidos para realizar la calibración de las lecturas *
! *****
LEER_RANGOS:
    READ 2,MIN_AZ.BYTE0
    READ 3,MIN_AZ.BYTE1
    READ 4,MAX_AZ.BYTE0
    READ 5,MAX_AZ.BYTE1
    READ 6,MIN_EL.BYTE0
    READ 7,MIN_EL.BYTE1
    READ 8,MAX_EL.BYTE0
    READ 9,MAX_EL.BYTE1
    READ 10,DIV_AZ
    READ 11,DIV_EL
RETURN
! *****
! * Observa si tanto el rotor como el PC se encuentran *
! * conectados al sistema dando alarmas en LCD. *
! *****
VERIFICAR_CONEXION_A:

```

```

LCDOUT $FE,1          'Borra LCD
CON_PC:
IF (PC_CONECTADO<>50) THEN      'Verifica que el programa responda al hardware.
  USBSERVICE          'Mantiene la comunicación USB.
  LCDOUT $FE,2,"ERROR DE PC",$FE,$C0,"CONTROL G5500"
  usbbuffer[0]= 0      'Verifica conexión con software (PC-USB)
  gosub dousbout
  GOSUB DOUSBIN
  PC_CONECTADO=USBBUFFER[7]
  GOTO CON_PC
ENDIF
LCDOUT $FE,1,"CONTROL G5500",$FE,$C0,"CONEXION OK!"
PAUSE 3000            'Conexión con el software oK!.
lcdout $FE,1          'Borra LCD
LOW LED_NOCONEX_USB   'Led indicador error de conexión con PC apagado
HIGH LED_CONEX_USB    'Led indicador de conexión con PC encendido
while (ROTOR==0)      'Comprobación de que el rotor G5500 este conectado.
  LCDOUT $FE,2,"ERROR DE ROTOR",$FE,$C0,"YAESU G5500"
  HIGH LED_ROTOR_ERROR 'Rotor no conectado.
  USBSERVICE          'Mantiene la comunicación usb viva.
WEND
LCDOUT $FE,1,"YAESU G5500",$FE,$C0,"CONEXION OK!"
PAUSE 3000            'Rotor conectado.
lcdout $FE,1          'Borra LCD.
LOW LED_ROTOR_ERROR   'Apaga el led indicador de rotor desconectado.
HIGH LED_ROTOR_NOERROR 'Enciende led de rotor conectado.
gosub leer_rangos     'Lee los rangos para calibración ya almacenados.
RETURN

! *****
! * Observa si tanto el rotor como el programa de control *
! * se encuentran conectados al sistema mientras se esta *
! * realizando el seguimiento. *
! *****

VERIFICAR_CONEXION_B:
usbbuffer[0]= 0      'Verifica conexión con software (PC-USB)
gosub dousbout
GOSUB DOUSBIN
PC_CONECTADO=USBBUFFER[7]
USBSERVICE          'Mantiene la comunicación usb viva.
IF (PC_CONECTADO<>50)OR(ROTOR==0)THEN RECONOCIMIENTO
  'Salta a reconocimiento si algo falla en
  'la conexión.

RETURN
! *****
! * Lectura del puerto A (RA0,RA1), obteniendo valores *
! * Correspondientes a las posiciones DE elevación y azimut *
! *****

LECTURAS_ANALOGAS:
  AZIMUT_ACTUAL=0      'Inicializa el AZIMUT ACTUAL para una nueva lectura
  ELEVACION_ACTUAL=0   'Inicializa el ELEVACION ACTUAL para una nueva
lectura
  HIGH CONTROL          'Habilita las entradas análogas, (IC 4066)
  FOR i=1 TO 20
CANAL1: adcin 0,AUX1[i] 'Lee canal 0 (Azimut).
  AZIMUT_ACTUAL=AZIMUT_ACTUAL+AUX1[i]

```

```

CANAL2: ADCIN 1,AUX2[i]           'Lee canal 1 (Elevación).
      ELEVACION_ACTUAL=ELEVACION_ACTUAL+AUX2[i]
      PAUSE 2
      USBService
      NEXT i
      AZIMUT_ACTUAL=AZIMUT_ACTUAL/20
      ELEVACION_ACTUAL=ELEVACION_ACTUAL/20
      GOSUB VERIFICAR_CONEXION_B   'Verifica que no haya error en la conexión.
      LOW CONTROL                 'Desconecta entradas análogas.
      AZIMUT_ACTUAL=AZIMUT_ACTUAL/DIV_AZ
      ELEVACION_ACTUAL=ELEVACION_ACTUAL/DIV_EL
      USBBUFFER[0]=3              'Envío de datos análogos leídos
      USBBUFFER[3]=AZIMUT_ACTUAL.BYTE0
      USBBUFFER[4]=AZIMUT_ACTUAL.BYTE1
      USBBUFFER[5]=ELEVACION_ACTUAL.BYTE0
      USBBUFFER[6]=ELEVACION_ACTUAL.BYTE1
      GOSUB DOUSBOUT

return
! *****
! * El micro se comunica por el puerto UBS con el PC      *
! * Los valores del azimut y elevación actual normalizados *
! * los valores del azimut y elevación deseados          *
! * y la tolerancia en grados para hacer las comparaciones *
! *****

LECTURA_PC:
      usbbuffer[0]= 5              'Indica al pc que debe entregar az y el deseados
      gosub dousbout
      gosub DOUSBIN

      TOLERANCIA.BYTE0=USBUFFER[2]      'Valor de error en grados permisible
      AZIMUT_DESEADO.BYTE0= USBUFFER[3] 'AZIMUT DESEADO
      AZIMUT_DESEADO.BYTE1= USBUFFER[4]
      ELEVACION_DESEADA.BYTE0=USBUFFER[5]'ELEVACION DESEADA
      ELEVACION_DESEADA.BYTE1=USBUFFER[6]
      usbservice

RETURN
! *****
! * En este código se realiza la comparación de la posición *
! * actual de la antena con la posición del satélite para   *
! * Realizar los ajustes pertinentes                          *
! *****

COMPARACION_DE_POSICION:
GOSUB LECTURA_PC
IF (AZIMUT_DESEADO > AZIMUT_ACTUAL-TOLERANCIA)AND(AZIMUT_DESEADO <
AZIMUT_ACTUAL+TOLERANCIA) THEN
      HIGH azimut_stop
      LOW AZIMUT_LEFT
      LOW azimut_RIGHT
ELSE
      IF AZIMUT_DESEADO <= AZIMUT_ACTUAL-1 THEN
            low azimut_stop
            LOW AZIMUT_RIGHT
            HIGH azimut_LEFT
      ENDIF
      IF AZIMUT_DESEADO >= AZIMUT_ACTUAL+1 THEN
            low azimut_stop
            HIGH AZIMUT_RIGHT

```

```

        LOW azimuth_LEFT
    ENDIF
ENDIF
USBSERVICE
IF (ELEVACION_DESEADA > ELEVACION_ACTUAL-
TOLERANCIA)AND(ELEVACION_DESEADA < ELEVACION_ACTUAL+TOLERANCIA) THEN
    HIGH elevacion_stop
    LOW ELEVACION_DOWN
    LOW elevacion_UP
ELSE
    IF ELEVACION_DESEADA <= ELEVACION_ACTUAL-1 THEN
        low elevacion_stop
        LOW ELEVACION_UP
        high elevacion_DOWN
    ENDIF
    IF ELEVACION_DESEADA >= ELEVACION_ACTUAL+1 THEN
        low elevacion_stop
        HIGH ELEVACION_UP
        LOW elevacion_DOWN
    ENDIF
ENDIF
RETURN
! *****
! * receive data from the USB bus *
! *****

DoUSBIn:
    USBBufferCount = USBBufferSizeRX      ' RX buffer size
    USBService      ' keep connection alive
    USBIn 1, USBBuffer, USBBufferCount, DoUSBIn ' read data, if available
    return

! *****
! * wait for USB interface to attach *
! *****

DoUSBOut:
    USBBufferCount = USBBufferSizeTX      ' TX buffer size
    USBService      ' keep connection alive
    USBOut 1, USBBuffer, USBBufferCount, DoUSBOut ' if bus available, transmit data
    return

! *****
! * Rutina de selección de opciones del Menú, operación que *
! * se realiza sin detener el seguimiento del satélite. *
! *****

MENU:
IF (B1==0) THEN      'Se testea si a sido presionado el pulsador de menú.
    IF (BOTON_1==0) THEN
        B3=B2
        B1=1
        B2=1
        PAUSE 300
    ENDIF
ELSE
    IF (BOTON_2==0) THEN 'Desplaza el menu hacia arriba.
        IF B2>1 THEN
            B2=B2-1
            pause 200
        ENDIF

```

```

ENDIF
IF (BOTON_3==0) THEN 'Desplaza el menú Hacia abajo.
  IF B2<6 THEN
    B2=B2+1
    pause 200
  ENDIF
ENDIF

if (BOTON_1==0) THEN
  B2=B2+6
  B1=0
  pause 200
ENDIF
ENDIF
! *****
'Mensajes que despliegan en el Display dependiendo del contador que
' lleva la cuenta de la posición en el menú donde se esta.
! *****
SELECT CASE B2
CASE 1
  LCDOUT $FE,2,$7E,"1. P. Actual    ",$FE,$C0," 2. P. Deseada    "
CASE 2
  LCDOUT $FE,2,$7E,"2. P. Deseada    ",$FE,$C0," 3. M. Automatico  "
CASE 3
  LCDOUT $FE,2,$7E,"3. M. Automatico ",$FE,$C0," 4. M. Manual    "
CASE 4
  LCDOUT $FE,2,$7E,"4. M. Manual    ",$FE,$C0," 5. Calibrar    "
CASE 5
  LCDOUT $FE,2,$7E,"5. Calibrar    ",$FE,$C0," 6. Salir del Menu "
case 6
  LCDOUT $FE,2,$7E,"6. Salir del Menu ",$FE,$C0,"          "
CASE 7
  LCDOUT $FE,2,"El. Act: ",#ELEVACION_ACTUAL," "
  LCDOUT $FE,$C0,"Az. Act: ",#AZIMUT_ACTUAL," "
CASE 8
  LCDOUT $FE,2,"El. Des: ",#ELEVACION_DESEADA," "
  LCDOUT $FE,$C0,"Az. Des: ",#AZIMUT_DESEADO," "
case 9
  AUTOMATICO=1
  LCDOUT $FE,1,"El: ",#ELEVACION_ACTUAL,$7E,#ELEVACION_DESEADA," "
  LCDOUT $FE,$C0,"Az: ",#AZIMUT_ACTUAL,$7E,#AZIMUT_DESEADO," "
CASE 10
  AUTOMATICO=0
  LCDOUT $FE,1,"El: ",#ELEVACION_ACTUAL,$7E,#ELEVACION_DESEADA," "
  LCDOUT $FE,$C0,"Az: ",#AZIMUT_ACTUAL,$7E,#AZIMUT_DESEADO," "
CASE 11
  goto CALIBRADO
CASE 12
  B2=B3
end select
RETURN

```

```
! *****  
'Rutina encargada de controlar el rotor de forma manual desde el pc.  
! *****
```

MANUAL:

```
  USBBUFFER[0]=7  
  GOSUB DOUSBOUT  
  GOSUB DOUSBIN  
  SELECT CASE USBBUFFER[1]  
  CASE 0  
    LOW AZIMUT_RIGHT  
    LOW AZIMUT_LEFT  
    LOW ELEVACION_UP  
    LOW ELEVACION_DOWN  
    HIGH AZIMUT_STOP  
    HIGH ELEVACION_STOP  
  CASE 1  
    HIGH ELEVACION_UP  
    LOW ELEVACION_STOP  
  CASE 2  
    HIGH ELEVACION_DOWN  
    LOW ELEVACION_STOP  
  CASE 3  
    HIGH AZIMUT_RIGHT  
    LOW AZIMUT_STOP  
  CASE 4  
    HIGH AZIMUT_LEFT  
    LOW AZIMUT_STOP  
  END SELECT
```