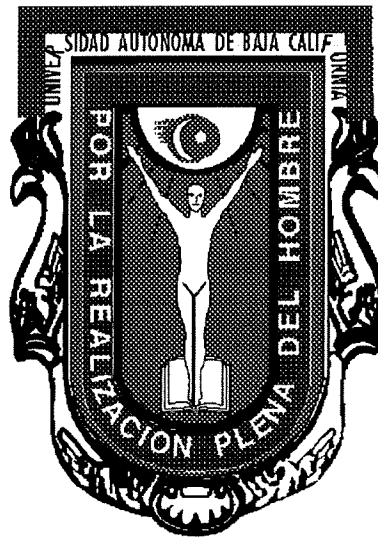


**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA**

**ESCUELA DE INGENIERÍA**



***INTERFACES PARA LA CARGA ÚTIL ÓPTICA DEL  
SATÉLITE EXPERIMENTAL SATEX.1***

**TESIS PROFESIONAL**

**QUE PARA OBTENER EL TÍTULO DE  
INGENIERO EN ELECTRÓNICA  
PRESENTA:**

**HÉCTOR GÓMEZ GUTIÉRREZ**

**DIRECTOR DE TESIS**

**M.C. HÉCTOR MEJÍA VÁZQUEZ**

**Ensenada, B. C. Diciembre de 1996.**

# UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

## ESCUELA DE INGENIERÍA

### *INTERFACES PARA LA CARGA ÚTIL ÓPTICA DEL SATÉLITE EXPERIMENTAL SATEX.1*

#### TESIS PROFESIONAL


QUE PARA OBTENER EL TÍTULO DE  
INGENIERO EN ELECTRÓNICA  
PRESENTA:

**HÉCTOR GÓMEZ GUTIÉRREZ**

DIRECTOR DE TESIS

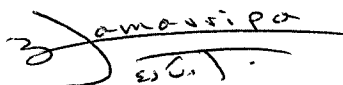
**M.C. HÉCTOR MEJÍA VÁZQUEZ**

APROBADA POR:

  
\_\_\_\_\_  
M.C. HÉCTOR MEJÍA VÁZQUEZ  
Director de Tesis

  
\_\_\_\_\_  
M.I. ARTURO ARVIZU MONDRAGÓN  
Sinodal Propietario

  
\_\_\_\_\_  
M.C. JUAN IVAN NIETO HIPÓLITO  
Sinodal Propietario

  
\_\_\_\_\_  
M.C. JOSE DE JESÚS ZAMARRIPA T.  
Sinodal Propietario

  
\_\_\_\_\_  
M.C. HUMBERTO CERVANTES DE A.  
Sinodal Propietario

Ensenada, B.C.

Diciembre de 1996.

**DEDICATORIA:**

*A mis padres,*

*José Gpe. y Rosa Ma.*

*que les debo todo, y por que siempre me apoyaron incondicionalmente durante mi vida, siendo lo que he logrado por ellos y para ellos les dedico mi trabajo con cariño.*

*A mis hermanos,*

*Gerardo, Karina y Omar*

*son mi familia que quiero mucho y las personas con las que siempre he podido contar.*

*Héctor.*

## **AGRADECIMIENTOS:**

*A mi asesor Héctor Mejía por su paciencia, asistencia, animación, apoyo moral y económico durante la realización de mi tesis.*

*A Arturo Arvizu por ser un amigo cuando necesitaba uno y por brindarme todo tipo de apoyo (BEEP, BEEP).*

*A Guille por darme un levantamiento espiritual y una motivación en la vida.*

*A mi hermana Vero y mi amiga Patricia Daoudal por que por un momento fueron mi segunda familia en Ensenada y por compartir parte de sus vidas conmigo.*

*A Benoit Pelczar mi amigo francés de confianza (cuasi-mexicano) con el que pase buenos momentos.*

*A Juan de Dios Araujo, mi compañero en Ensenada durante toda la carrera, por el despertamiento y por ser amigo.*

*A Rodolfo Campos quien aprecio como amigo que sabe ser.*

*A Jorge Morán y familia por recibirme en su casa y por todo lo demás.*

*A mis compañeros de generación: Alberto, Alvaro, Carlos, Cecy, Diego, Enrique, Gilberto, Javier (Chicali), Roberto, Rigoberto (Rósaro), Sammy, y Victor. También a sus familias por la atención cuando nos juntábamos a estudiar en sus casas.*

*Al Dr. Fco. J. Mendieta por permitirme participar en el proyecto SATEx.1.*

*A mis maestros de la universidad que me enseñaron más que de lo que el temario dictaba.*

*A las secretarias de la UABC y el CICESE siempre atentas y serviciales.*

*Al personal de intendencia y seguridad del CICESE siempre dispuestos a ayudar.*

*A mis amigos del CICESE y la UABC: Rosy Chávez (parece que me toca hacerte inmortal, mucha suerte en lo que te propongas!), Alvaro (no la dejes sola tan seguido!), Oscar (ese honorable transformer por ahí he!), Alejandro (chilaquil, te estamos esperando!), Mario (que esto y que el otro...), Carmen Leal (y familia), Marcos (acuérdate del karma), Mauricio y Lety (¿cuando van para Tijuana pues?), Ernesto (nada más sonsacas y ni vas), Rene (Monsieur Rene), Mauro (¿que le diste Benoit para que no me hiciera caso?), Angel (ya consigüete otro carro ese les trae problemas), Jesus Escoto (integrante del grupo DSP's y compañero de trabajo en SATEx.1), Luis A. Talavera (Tilín, el M.C.), Cecy (en palabras de Rosy, la super-secre!), Zarina (por fin termine!, faltas tu), Gerardo Yepiz (Hey Dude!), Sergio Carnalla (Paisa), Luis Flores (un mundo nos vigila), Horacio y Miguel (las tortugas), Miguel Martínez (Ché), Ruby (¿donde esta el memo pues?), Sergio Peralta, Enrique Pacheco, Ana, Marta, Edith, Carmen Maya, Gonzalo, Chuy, Alfonso (Cachondo), Juan Pablo, Ricardo, Leobardo, Pancho, Héctor (Tocayo), Ricardo, Raúl, Luis, Tamayo, Rosa Marta, Maripaz, Edgar,*

*Miguel Cota, Raymundo, Jaime (Pez Globo), Alfredo (el chilín inside), Rodrigo (Choco), Alex (Magno), Manuel (Kiko), Juan Cabrera, Alberto, Cipriano, Alfonso (Profe), Ricardo (Profe cahuamas) y Amparo, Enrique Castañeda, y a todos aquellos que me brindaron su amistad y con los que compartí algunos momentos durante mi carrera en la UABC y mi estancia en CICESE.*

*Gracias a todos y cada uno de ustedes.*

*Héctor (Greñas).*

---

**CONTENIDO**

<b>1. APUNTAMIENTO ESPACIAL Y LA CARGA ÚTIL ÓPTICA DE COMUNICACIONES DE SATEX.1.....</b>	<b>1</b>
1.1 INTRODUCCIÓN.....	1
1.2 MICROSATÉLITE EXPERIMENTAL SATEX.1.....	2
1.3 APUNTAMIENTO ESPACIAL.....	4
1.3.1 <i>Parámetros orbitales</i> .....	5
1.3.2 <i>Comportamiento de un satélite</i> .....	7
1.3.3 <i>Consideraciones de ruido y perturbaciones atmosféricas</i> .....	8
1.4 CARGA ÚTIL ÓPTICA.....	8
1.5 PLATAFORMA BÁSICA DE CUO.....	12
1.5.1 <i>CCUO</i> .....	12
1.5.2 <i>Módulo de potencia R3.1</i> .....	12
1.5.3 <i>Módulo T1 (láser)</i> .....	13
1.5.4 <i>Módulo R2 foto-detector de avalancha</i> .....	14
1.5.5 <i>Módulo R4 (motores)</i> .....	14
1.5.6 <i>Interfaces</i> .....	14
1.6 OBJETIVOS DEL TRABAJO.....	15
<b>2. PROTOCOLO DE COMUNICACIÓN EN SATEX.1.....</b>	<b>16</b>
2.1 MANDOS INTERNOS.....	17
2.2 MANDOS EXTERNOS.....	20
2.3 PROGRAMAS.....	21
<b>3. ESPECIFICACIONES.....</b>	<b>23</b>
3.1 ESPECIFICACIONES GENERALES.....	23
3.2 ESPECIFICACIONES PARTICULARES.....	24
3.2.1 <i>Monitor mínimo</i> .....	24
3.2.2 <i>Interfaz serie</i> .....	25
3.2.3 <i>Interfaz con motores de pasos</i> .....	25
3.2.4 <i>Interfaz láser</i> .....	26
3.2.5 <i>Interfaz con foto-detector de cuatro cuadrantes</i> .....	26
3.2.6 <i>Relojes de Tiempo Real</i> .....	26
<b>4. DISEÑO DE LAS INTERFACES HACIA LOS ELEMENTOS DE LA CARGA ÚTIL ÓPTICA.....</b>	<b>27</b>
4.1 MONITOR MÍNIMO.....	27
4.2 INTERFACES.....	29
4.2.1 <i>Interfaz serial</i> .....	32
4.2.1.1 <i>Rutina de recepción</i> .....	33
4.2.1.2 <i>Rutina de decodificación</i> .....	36
4.2.1.3 <i>Rutina de transmisión</i> .....	37
4.2.1.4 <i>Rutina de temporización</i> .....	39
4.2.2 <i>Interfaz con motores</i> .....	39

4.2.3	<i>Interfaz con láser.</i>	46
4.2.4	<i>Interfaz con foto-detector de cuatro cuadrantes.</i>	48
4.2.5	<i>Relojes de tiempo real.</i>	49
4.3	ENLACE ENTRE INTERFACES.	51
<b>5.</b>	<b>PRUEBAS Y RESULTADOS SOBRE EL DESARROLLO.</b>	<b>53</b>
5.1	MONTAJE.	55
5.2	PRUEBAS PARA EL MONITOR MÍNIMO.	58
5.2.1	<i>Configuración inicial del microcontrolador.</i>	58
5.2.2	<i>Configuración de periféricos.</i>	59
5.2.3	<i>Autodiagnóstico.</i>	59
5.2.4	<i>Administración de la secuencia lógica del programa.</i>	59
5.2.5	<i>Generación de tableros de efemérides.</i>	60
5.2.6	<i>Recuperación del programa en casos de error por sobreflujo/bajoflujo de pila.</i>	60
5.3	PRUEBAS SOBRE LA INTERFAZ SERIE	60
5.3.1	<i>Reconocimiento y creación de los diferentes tipos de mensajes.</i>	61
5.3.2	<i>Cumplimiento de secuencias temporales de acuerdo con el protocolo II-UNAM.</i>	61
5.3.3	<i>Recuperación del canal de comunicaciones en caso de errores en la transmisión de mensajes y su formato.</i>	62
5.3.4	<i>Reconocimiento y ejecución de instrucciones utilizadas en el protocolo II-UNAM.</i>	63
5.3.5	<i>Telecargado y activación de un nuevo programa y reconocimiento de instrucciones exclusivas de CCUO.</i>	64
5.3.6	<i>Telecargado y activación de un programa.</i>	64
5.3.7	<i>Aborto de un programa.</i>	65
5.3.8	<i>Decodificación del formato hexadecimal Intel MCS-86 para el telecargado de programas.</i>	65
5.4	PRUEBAS SOBRE LA INTERFAZ CON MOTORES	65
5.4.1	<i>Generación de las señales de control cumpliendo con los tiempos requeridos por los manejadores y motores.</i>	65
5.4.2	<i>Inicialización de motores (encendido, centrado y posicionamiento inicial).</i>	67
5.4.3	<i>Lectura de tableros.</i>	68
5.5	PRUEBAS A LA INTERFAZ DEL LÁSER	69
5.5.1	<i>Funcionamiento de la rutina de conversión.</i>	69
5.5.2	<i>Adquisición de parámetros del láser.</i>	69
5.5.3	<i>Activación de la rutina de transmisión en el caso de petición de transmisión de parámetros.</i>	70
5.5.4	<i>Corrección en caso de error por sobrecarrera en la adquisición de las señales del convertidor.</i>	70
5.5.5	<i>Generación de señal de 1 kHz para la modulación del láser.</i>	71
5.6	PRUEBAS A LA INTERFAZ DEL APD	71
5.6.1	<i>Adquisición de señales provenientes del APD.</i>	71
5.6.2	<i>Corrección en caso de error por sobrecarrera en la adquisición de las señales del convertidor.</i>	71
5.7	PRUEBAS AL RELOJ DE TIEMPO REAL	71
5.7.1	<i>Simulación de configuración del reloj desde la estación terrena.</i>	72

5.7.2 Simulación de sincronización del reloj con la ET por medio de un evento externo a CUO.....	72
<b>CONCLUSIONES.....</b>	<b>74</b>
<b>APÉNDICE A.....</b>	<b>76</b>
DETERMINACIÓN DE ÓRBITA.....	76
<b>APÉNDICE B.....</b>	<b>77</b>
<b>GLOSARIO.....</b>	<b>80</b>
<b>BIBLIOGRAFÍA.....</b>	<b>82</b>
<b>ÍNDICE.....</b>	<b>85</b>

---

**LISTA DE FIGURAS**

FIGURA 1. COMPONENTES DE SATEX.1.....	3
FIGURA 2. REPRESENTACIÓN DE LA LÍNEA DE VISTA.....	6
FIGURA 3. PRECESIÓN.....	7
FIGURA 4. SPIN (GIRO).....	7
FIGURA 5. DIAGRAMA GENERAL DEL ENLACE ÓPTICO PARA SATEX.1.....	9
FIGURA 6. DOS VISTAS DIFERENTES DEL PROTOTIPO DE CUO.....	10
FIGURA 7. DIAGRAMA A BLOQUES DE CUO.....	11
FIGURA 8. COMPUTADORA DE LA CARGA ÚTIL ÓPTICA.....	12
FIGURA 9. EN LA FIGURA SE PUEDE APRECIAR EL MÓDULO R3.1 DE POTENCIA Y ACONDICIONAMIENTO.....	13
FIGURA 10. TIEMPO DE ESPERA ENTRE LAS PALABRAS DE CONTROL.....	17
FIGURA 11. PALABRA DE CONTROL (MANDO INTERNO).....	17
FIGURA 12. FORMATO DE UN MANDO EXTERNO.....	20
FIGURA 13. FORMATO PARA SUBIR PROGRAMAS.....	21
FIGURA 14. REGISTRO DE BANDERAS CCUOCON.....	27
FIGURA 15. DIAGRAMA DE FLUJO DE LA RUTINA DE AUTODIAGNÓSTICO CHKRAM().....	28
FIGURA 16. PROGRAMA PRINCIPAL O MONITOR MÍNIMO.....	30
FIGURA 17. DIAGRAMA A BLOQUES DEL MICROCONTROLADOR, ELEMENTOS EXTERNOS E INTERFACES DE CCUO.....	31
FIGURA 18. INTERFAZ ENTRE CP Y CUO.....	32
FIGURA 19. DIAGRAMA A CUADROS DE LA RUTINA DE RECEPCIÓN.....	34
FIGURA 20. SUBRUTINA DE DECODIFICACIÓN DE MENSAJES.....	35
FIGURA 21. DIAGRAMA A BLOQUES DE LA RUTINA DE TRANSMISIÓN.....	38
FIGURA 22. SISTEMA DE GUARDA PARA LA RECEPCIÓN DEL CANAL DE COMUNICACIONES.....	40
FIGURA 23. RELACIÓN ENTRE LAS RUTINAS DE LA INTERFACE DE MOTORES.....	42
FIGURA 24. RUTINA HOME() PARA EL POSICIONAMIENTO INICIAL DE LOS ESPEJOS.....	44
FIGURA 25. RUTINA DE CONTROL PARA LA CADENCIA DE LOS ESPEJOS.....	45
FIGURA 26. DIAGRAMA A CUADROS DE LA RUTINA COMIENZA_BARRIDO().....	47
FIGURA 27. FORMATO DE DATOS EN EL MANDO EXTERNO PARA EL RELOJ DE TIEMPO REAL.....	50
FIGURA 28. MONTAJE DE PRUEBAS PARA LAS INTERFACES DE CCUO.....	57
FIGURA 29. TARJETA DE EMULACIÓN PARA CUO. CON ESTA TARJETA SE OBTIENEN LAS SEÑALES QUE PROVENDRÍAN DEL LÁSER Y APD.....	58
FIGURA 30. SEÑALES DE CONTROL DE PULSOS Y DIRECCIÓN PARA LOS MOTORES.....	66
FIGURA 31. TRAZADO DE LÁSER CON EL MOVIMIENTO DE LOS ESPEJOS UTILIZANDO LOS DATOS DE LOS TABLEROS DE EFEMÉRIDES.....	69
FIGURA 32. SEÑAL DE 1 KHZ PARA LA MODULACIÓN DEL LÁSER.....	72
FIGURA 33. PARÁMETROS ORBITALES.....	76
FIGURA 34. PERIGEO Y APOGEO.....	76
FIGURA 35. VISTA DESDE EL SATÉLITE SOBRE EL BARRIDO TRAZADO EN LA SUPERFICIE DE LA TIERRA PARA LOCALIZAR LA ET.....	77
FIGURA 36. ACERCAMIENTO DE DOS TRAYECTORIAS PARA OBSERVAR EL TRASLAPE ENTRE MANCHAS Y TRAYECTORIAS.....	78
FIGURA 37. BARRIDO DE 10 TRAYECTORIAS.....	79

---

## LISTA DE TABLEROS

TABLERO I. TABLA DE DECODIFICACIÓN DE ORIGEN. ....	17
TABLERO II. TABLA DE DECODIFICACIÓN DEL MENSAJE.....	18
TABLERO III. TABLA DE DECODIFICACIÓN DE FUNCIONES.....	18
TABLERO IV. ESPECIFICACIONES GENERALES DE LA PROGRAMÁTICA.....	23
TABLERO V. ESPECIFICACIONES GENERALES DE LA MECAMÁTICA.....	23
TABLERO VI. ESPECIFICACIONES DEL MONITOR. ....	24
TABLERO VII. ESPECIFICACIONES PARA LA INTERFAZ SERIE. ....	25
TABLERO VIII. ESPECIFICACIONES PARA LA INTERFAZ DE LOS MOTORES DE PASOS. ....	25
TABLERO IX. ESPECIFICACIONES PARA LA INTERFAZ LÁSER. ....	26
TABLERO X. ESPECIFICACIONES PARA LA INTERFAZ CON FOTO-DETECTOR DE CUATRO CUADRANTES.....	26
TABLERO XI. ESPECIFICACIONES PARA EL RELOJ DE TIEMPO REAL. ....	26
TABLERO XII. DESIGNACIÓN DE TERMINALES PARA LAS SEÑALES DE CONTROL.....	46
TABLERO XIII. DESIGNACIÓN DE ENTRADAS ANALÓGICAS. ....	46
TABLERO XIV. DESIGNACIÓN DE SEÑALES A ENTRADAS ANALÓGICAS DEL MICROCONTROLADOR.....	48
TABLERO XV. RUTINAS DE INTERRUPTIÓN UTILIZADAS Y SU JERARQUÍA. ....	51
TABLERO XVI. LISTA DE PRUEBAS PARA EL CUMPLIMIENTO DE LAS ESPECIFICACIONES GENERALES DE LA PROGRAMÁTICA. ....	53
TABLERO XVII. LISTA DE PRUEBAS PARA EL MONITOR MÍNIMO. ....	54
TABLERO XVIII. LISTA DE PRUEBAS PARA LA INTERFAZ SERIE.....	54
TABLERO XIX. LISTA DE PRUEBAS PARA LA INTERFAZ CON LOS MOTORES. ....	54
TABLERO XX. LISTA DE PRUEBAS PARA LA INTERFAZ CON EL LÁSER.....	55
TABLERO XXI. LISTA DE PRUEBAS PARA LA INTERFAZ CON EL APD. ....	55
TABLERO XXII. LISTA DE PRUEBAS PARA LOS RELOJES DE TIEMPO REAL.....	55

---

# 1. APUNTAMIENTO ESPACIAL Y LA CARGA ÚTIL ÓPTICA DE COMUNICACIONES DE SATEX.1.

## 1.1 Introducción.

Los sistemas de comunicaciones ópticas se presentan como una alternativa a la utilización de sistemas de comunicaciones convencionales en la banda de Radio Frecuencia (RF). Lo anterior se debe, por un lado, a su gran potencial de trabajo en altas velocidades requiriendo antenas (ópticas) mucho más pequeñas que las de RF y, por el otro, a una alta directividad que los provee de inmunidad a interferencia y confiabilidad en la comunicación.

Con las comunicaciones ópticas se reducen significativamente los problemas y requerimientos de potencia, volumen y peso en comparación con los sistemas de RF; sin embargo, la realización tecnológica de éstos afronta dificultades que deben ser resueltas con diseños de ingeniería de alta calidad, tales como:

- a) Necesidad de apuntamiento, adquisición espacial y seguimiento extrémadamente finos, los que requieren de subsistemas optomecánicos retroalimentados a partir de la señal de telecomunicaciones o de un faro (beacon) óptico adicional.
- b) Efectos del medio ambiente “ruidoso” en el campo visual del receptor muy superior al ruido de la “ventana” espectral de las microondas.

Por lo anterior, existe una profusa actividad internacional en investigación y desarrollo en los siguientes aspectos:

- a) Análisis de sistemas y diseño.
- b) Métodos de modulación y codificación.
- c) Adquisición, apuntamiento y seguimiento.
- d) Subsistemas de comunicaciones láser.
- e) Experimentación en sistemas ópticos y evaluación del desempeño.

Dentro del área de comunicaciones ópticas a nivel internacional durante más de 10 años se ha estado trabajado en el caso de apuntamiento espacial. En la actualidad existen sistemas de comunicaciones ópticas satélite-satélite mientras que los sistemas Tierra-satélite se encuentran todavía en etapa experimental. México ya ha ingresado en esta área de las comunicaciones. Para el caso de apuntamiento espacial Tierra-satélite se está trabajando en un sistema experimental que formará parte del microsatélite experimental denominado SATEX.1. El presente trabajo está basado en una de las partes que compone este microsatélite y responde a las necesidades del apuntamiento espacial para dicho sistema experimental.

## **1.2 Microsatélite experimental SATEX.1.**

SATEX.1 será un microsatélite experimental de órbita baja diseñado y construido en México por diversas instituciones académicas y de investigación. Realizado, SATEX.1 se convertiría en el segundo microsatélite en órbita construido en México<sup>1</sup>.

Las instituciones participantes en el desarrollo del microsatélite son las siguientes:

- ♦ Instituto Mexicano de Comunicaciones (IMC)
- ♦ Centro de Investigación Matemática (CIMAT).
- ♦ Centro de Investigación y de Estudios Avanzados (CINVESTAV).
- ♦ Departamento de Microelectrónica de la BUAP.
- ♦ Escuela de Ingeniería Aeronáutica y Sección de Graduados, ESIME Unidad Ticomán.
- ♦ Escuela Superior de Ingeniería Mecánica y Eléctrica (ESIME) IPN.
- ♦ Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE).
- ♦ Instituto de Ingeniería (II) y la Facultad de Ciencias (FC) de la UNAM.
- ♦ Departamento de Electrónica y Telecomunicaciones (DET) del Centro de Investigación Científica y de Educación Superior de Ensenada (CICESE).

Las instituciones se encuentran coordinadas por el IMC, dependencia gubernamental de la Secretaría de Comunicaciones y Transportes (SCT) que proporciona el apoyo financiero para SATEX.1 mientras que el resto de las instituciones aportan el recurso humano.

Aunque el principal objetivo es el de diseñar y construir un microsatélite con tecnología nacional, al menos en un 90%, no menos importantes son los objetivos asociados con el desarrollo del satélite como son la generación de infraestructura, utilizable en proyectos futuros, la especialización de personal en el campo aeroespacial y la coordinación de áreas multidisciplinarias y multi-institucionales a nivel nacional en torno a objetivos comunes.

En el siguiente listado podemos observar los proyectos en que se ha dividido SATEX.1 así como las instituciones que los desarrollan.

1. Análisis, diseño y ensamble de la estructura, principalmente con materiales compuestos y duraluminio. ESIME-Ticomán.
2. Subsistema de computadora de a bordo tolerante a fallas, sensores y protocolo. II-UNAM.
3. Análisis térmico y desprendimiento catastrófico. FC-UNAM.
4. Modelado térmico y análisis orbital. CIMAT
5. Estación Terrena (ET). DET-CICESE.
6. Celdas solares y baterías. INAOE.
7. Subsistema de telemetría, comando y control. DET-CICESE.
8. Subsistema de potencia eléctrica incluyendo colectores solares, baterías y circuito de carga (cuatro paneles solares para un total de 30 watts). Departamento de Microelectrónica de la BUAP.

---

<sup>1</sup> El primer satélite construido en México que se encuentra actualmente en órbita estable es el UNAM-SAT II, creado en las instalaciones de la UNAM.

9. Cargas Útiles conformadas por: subsistema de comunicaciones ópticas desarrollado en DET-CICESE; subsistema de comunicaciones en la banda Ka (23 GHz aproximadamente) desarrollado en ESIME Sección de Graduados; cámara de CCD desarrollado en el Departamento de Microelectrónica de la BUAP.

Dentro del microsatélite se cuenta con una red propietaria de comunicaciones. La red se compone de un ducto interno de comunicación serial asincrónica a la que se conectan diferentes computadoras. Una de las computadoras administra el canal y se le denomina computadora principal (CP). Otra computadora llamada redundante (CR) actúa como respaldo de CP en caso de que ésta fallara; las dos conforman una arquitectura tolerante a fallas. Conectadas también a la red están las computadoras de la Carga Útil Óptica (CUO), la del Decodificador de Tonos (DT) y la de la cámara CCD. En la Figura 1 se muestran los diferentes componentes de SATEX.1.

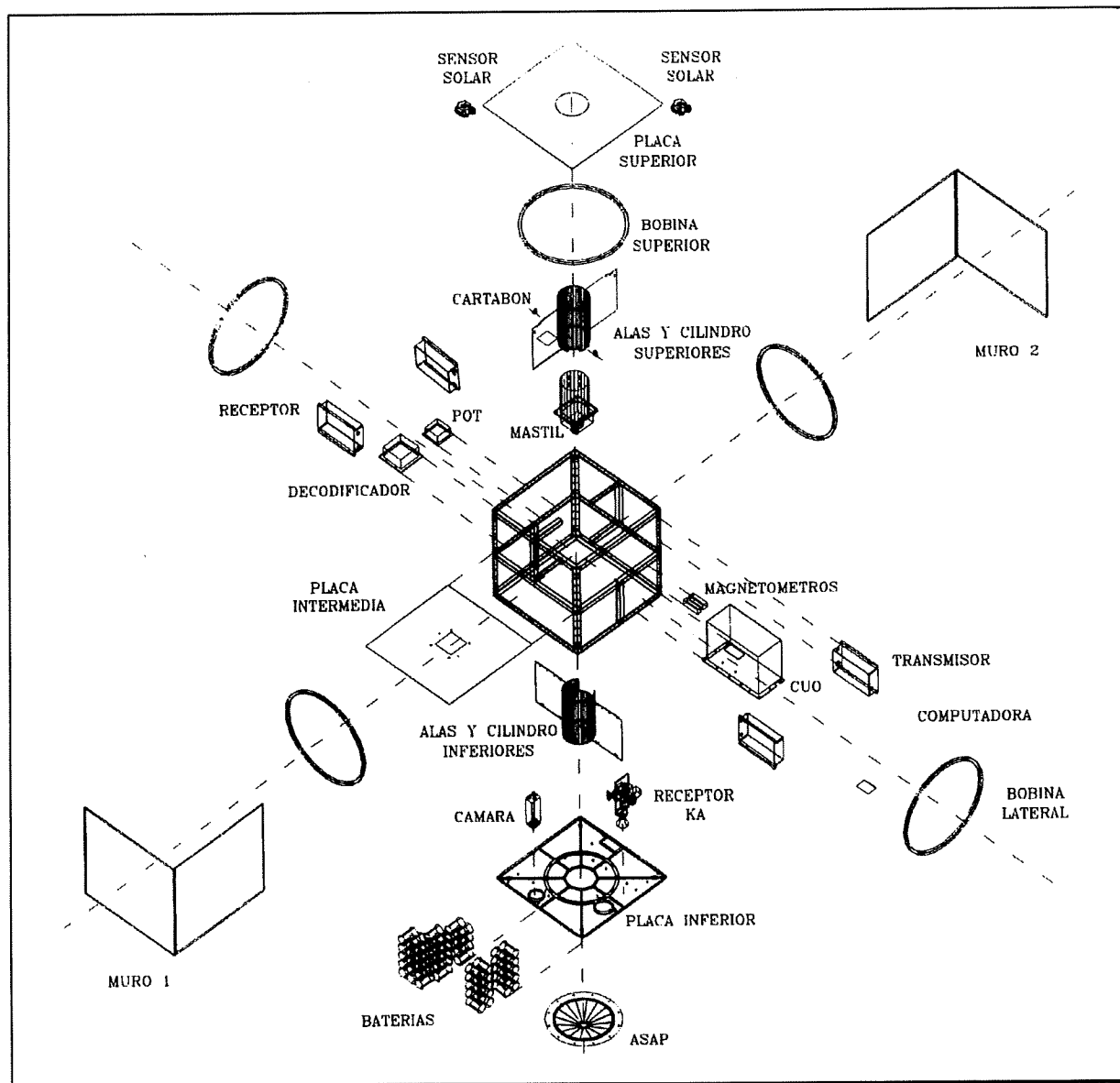


Figura 1. Componentes de SATEX.1.

En la etapa de integración, los diferentes componentes serán colocados dentro del microsatélite para luego ser llevado a laboratorios especializados donde serán sometidos a severas pruebas de resistencia y confiabilidad antes de su lanzamiento.

El peso máximo para el microsatélite será de 47 kg en un volumen conformado por un cubo de 45 cm por lado que estará en una órbita casi circular a 780 kms de altura con una inclinación de 98° con respecto al ecuador y un período de 104 minutos.

El microsatélite será lanzado por un cohete Ariane 5 desde la Guyana Francesa como una carga secundaria. Este lanzamiento fue otorgado gratuitamente a México por parte del Consorcio Europeo Arianespace debido a la utilización de sus servicios en el lanzamiento de los satélites Solidaridad I y II. El espacio designado se encuentra limitado por condiciones de peso, volumen y seguridad.

### **1.3 Apuntamiento espacial.**

El apuntamiento espacial es la parte que permite establecer el enlace óptico por medio del rastreo y seguimiento de una señal de luz en el espacio libre. Como es bien conocido, todo sistema de comunicaciones contiene tres etapas fundamentales: transmisor, receptor y canal de comunicación. Para el caso del apuntamiento espacial, el transmisor y receptor son elementos ópticos montados sobre envoltentes mecánicas manipulados por algoritmos de control. A su vez, el canal de comunicación lo constituye el espacio libre.

Debe entenderse que para esta clase de sistemas de comunicaciones es necesario: primero, establecer el enlace entre el transmisor, a través de una fuente de luz coherente<sup>2</sup>, y el receptor por medio de un elemento foto-sensible; y segundo, empleando algoritmos de control para establecer el seguimiento óptico de la señal de luz.

El rastreo es la búsqueda de la fuente de luz. En esta etapa no se tiene una señal de retroalimentación para guiar la búsqueda por lo que el control de los sistemas de apuntamiento se hace a lazo abierto. Para el caso de SATEX.1 el rastreo se hará por medio de predicciones sobre la trayectoria del microsatélite concentrándose en aquellas en las que éste quede bañado por la luz de la ET.

Cuando se logra establecer el contacto óptico el sistema de apuntamiento trata de mantener el enlace de luz empleando un control de lazo cerrado, por lo que esta etapa se le nombra de seguimiento.

En la primer etapa, la importancia de la determinación de órbita del microsatélite se vuelve evidente considerando lo directivo de la señal a capturar. Así mismo, la determinación de órbita del satélite es importante para la implementación de las interfaces dado que impone especificaciones de diseño.

---

<sup>2</sup> al que se le denomina "faro".

Si bien el objetivo de esta tesis, el cual definiremos ampliamente en la sección 1.6, no es el control del apuntamiento espacial sino el diseño e implementación de la plataforma en que se apoyará dicho control, es muy importante conocer los requerimientos que determinan el apuntamiento espacial de un satélite ya que imponen condiciones, restricciones y/o libertades al diseño de las interfaces; objetivo real de este trabajo.

Los requerimientos más importantes para determinar el apuntamiento de un satélite son:

1. Conocimiento de los parámetros de la órbita. Los parámetros iniciales de la órbita del microsatélite son dados a conocer por la agencia espacial; de ahí se parte para determinar la órbita y generar efemérides<sup>3</sup> de su avistamiento.
2. Conocimiento del comportamiento del satélite (tipos de movimientos y perturbaciones mecánicas).
3. Consideraciones de ruido y perturbaciones atmosféricas (lluvia, nubes, luces de ciudades, luz solar, etc.).

Debe entenderse que de ninguna manera se trata de una lista completa de condicionantes con las que puedan determinarse plenamente el apuntamiento de éste u otros satélites, sin embargo, nos permiten darnos una idea de lo que se requiere para comenzar a generar un modelo del sendero que seguirá el satélite así como de su comportamiento a lo largo de éste.

A continuación se describen brevemente los puntos mencionados.

### 1.3.1 Parámetros orbitales.

Para que la ET pueda bañar de luz al satélite éste deberá estar dentro de su “línea de vista”. Podemos definir la línea de vista como la trayectoria en línea recta entre dos objetos (Figura 2) de manera de obtener un contacto “visual”.

Mientras SATEX.1 se encuentre en órbita, se estima que se tendrá en línea de vista aproximadamente 2 minutos cuando pase por la estación terrena. En este intervalo se tiene que lograr el enlace. Como se puede imaginar, debido al corto tiempo de visualización, es necesario conocer de antemano la trayectoria del satélite y su posición dentro ésta, lo que implica el conocimiento de los parámetros que determinan la órbita de un satélite. Conociendo la órbita de éste es posible: hacer predicciones sobre su aparición dentro de la línea de vista de las antenas ópticas; también podemos preparar el equipo, activándolo solo cuando sea necesario (ahorro de energía); y sobre todo, reducir la búsqueda a una área específica incrementando la posibilidad de lograr el enlace en el menor tiempo posible.

Aquí se puede observar cómo el modelo de la órbita dictamina, tanto las condicionantes para la elaboración de las interfaces, como la preparación del equipo y la implementación de un sistema de tiempo real para el inicio del rastreo.

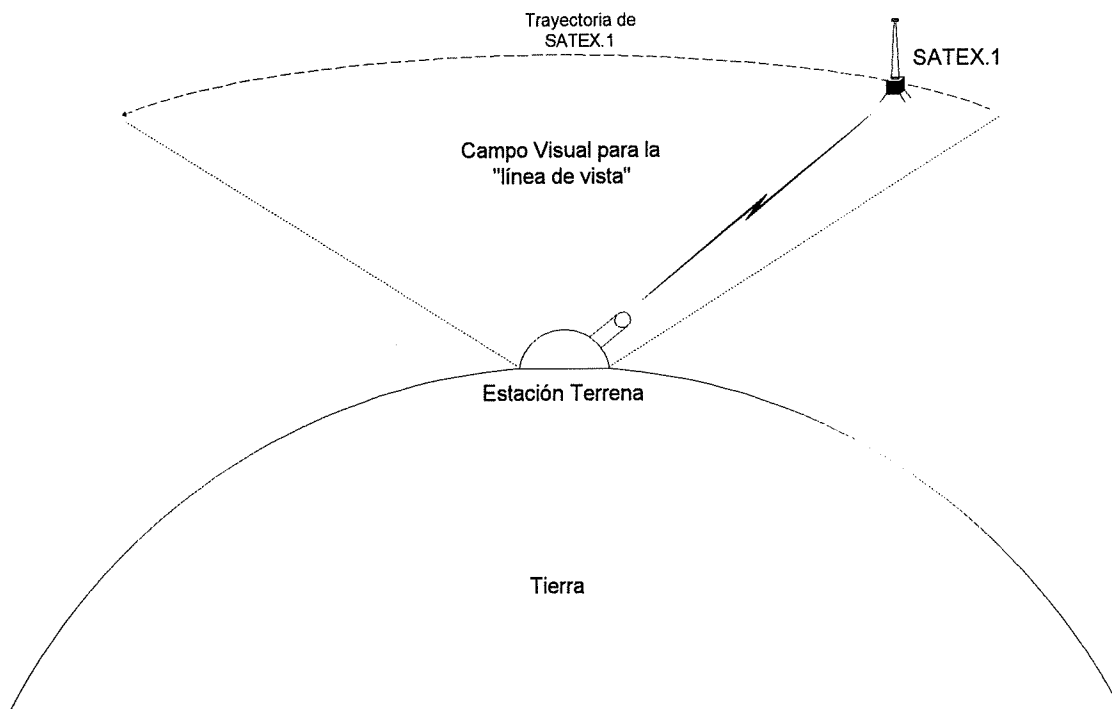
---

<sup>3</sup> Efemérides son un conjunto de datos que dan la posición del satélite en un momento dado.

El modelo de la órbita se plantea en torno a un sistema de coordenadas inercial<sup>4</sup>. La orientación del satélite dentro de dicho sistema se establece a través de diversos elementos orbitales. En el apéndice A se encuentra un resumen de los elementos orbitales más importantes basados en un sistema de coordenadas inercial.

La determinación de órbita comienza con un modelo simplificado al que después se le agregarán correcciones debido a perturbaciones atmosféricas, la rotación de la Tierra, cambios en la forma de la órbita, etc. Es necesario conocer lo más que se pueda sobre los parámetros que determinan la órbita de un satélite y su movimiento para lograr un modelo confiable. El modelo generado será utilizado tanto por la estación terrena como por el satélite para conocer la posición de uno respecto al otro en todo momento, sobretodo en aquellos instantes en que ambos se encuentren en línea de vista y lo más importante: las coordenadas iniciales del apuntamiento dentro de la cobertura de adquisición óptica del satélite.

De todo lo anterior podemos decir, que el modelo de la órbita y el conocimiento sobre la posición del satélite y la ET dentro de la línea de vista son los progenitores de las condiciones iniciales del apuntamiento.



*Figura 2. Representación de la línea de vista. La visualización del microsátélite esta limitada por la curvatura de la Tierra. La figura es una ilustración del concepto de "línea de vista". En realidad el campo visual está delimitado por la tangente a la Tierra en el punto donde se encuentra el observador.*

<sup>4</sup> Se le llama sistema de coordenadas inercial debido a que este sistema mantiene una orientación fija con respecto a las estrellas. En el sistema que se utilizará, llamado Sistema de Coordenadas Inercial de Ascensión Derecha y Declinación, se toma como referencia al centro de la Tierra y el primer punto de la constelación de Aries ( $\Upsilon$ ) para fijar el eje X.

### 1.3.2 Comportamiento de un satélite.

Así como el modelo de la órbita determina las condiciones iniciales del apuntamiento, el comportamiento del satélite determina en gran medida el algoritmo de rastreo en lazo abierto del apuntamiento espacial.

El comportamiento de un satélite es la descripción de como se mueve con respecto a su propio centro de gravedad. Podemos considerar los siguientes tipos de movimientos:

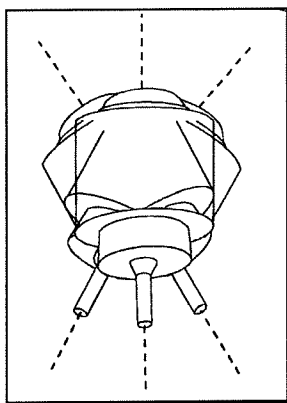


Figura 3. Precesión.

- Actitud. La actitud de un satélite describe la orientación de sus antenas hacia la estación terrena. En el caso de SATEX.1 la orientación mencionada será del montaje telescópico.
- Perturbaciones intrínsecas (movimiento interno de masas). Las perturbaciones intrínsecas en SATEX.1 son ocasionadas por el movimiento de los espejos.
- Precesión. Es un movimiento oscilatorio en torno a su centro de gravedad (ver Figura 3) ocasionado por varias fuerzas externas como el abultamiento de la tierra y la atracción de la luna y el sol.
- Spin o movimiento de rotación. Es un movimiento de giro sobre su propio eje longitudinal. Este giro es necesario para darle al satélite estabilidad y regulación

térmica pero ocasiona problemas para el caso de rastreo y seguimiento del faro óptico.

SATEX.1 cuenta con sensores adecuados para registrar estos movimientos y emplea diseños complejos que minimizan su influencia.

La principal perturbación de un satélite, referente a su desplazamiento alrededor de la Tierra, lo constituye el campo gravitacional de la misma. Debido al abultamiento de la Tierra en el ecuador, la atracción gravitacional es más fuerte en éste y ocasiona distorsiones y corrimientos de la órbita así como del plano orbital. Otras perturbaciones no menos importantes son las ocasionadas por la atracción del Sol y la Luna; el arrastre atmosférico que depende de la geometría del satélite (área expuesta al medio) y la velocidad relativa entre el satélite y el medio. El arrastre atmosférico afecta principalmente la excentricidad de la órbita (ver Apéndice A), por lo que en cada revolución la reduce, tendiendo a llegar a la de un círculo ( $e=0$ ).

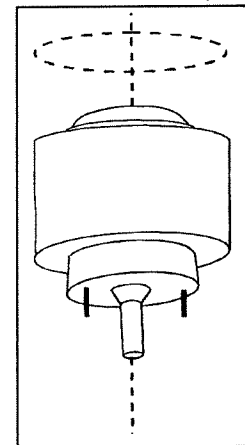


Figura 4. Spin (Giro).

Si se considera que la órbita de un satélite está cambiando constantemente, a una razón que depende de varias condiciones que no se pueden determinar exactamente con anticipación; se vuelve clara la necesidad de actualizar los parámetros orbitales de los cálculos con cierta periodicidad. La actualización dependerá en gran medida del tipo de órbita. Se deben actualizar con más frecuencia los parámetros para las órbitas circulares, como es el caso de SATEX.1, que

para las elípticas de gran altura. Estos cambios implican ajustes en el sistema de apuntamiento<sup>5</sup>. En otras palabras, sabemos que el control de apuntamiento se encuentra en el módulo de CUO y que debe de funcionar de manera autónoma (lo que incluye el cálculo de la trayectoria de rastreo dentro de la línea de vista); sin embargo, debido a la constante variación de los parámetros de la órbita, que solo se pueden obtener desde la Tierra, la interfaz de comunicación por radio se vuelve fundamental para la actualización de los cálculos y por consiguiente del éxito del experimento.

### **1.3.3 Consideraciones de ruido y perturbaciones atmosféricas.**

Además de la dinámica orbital, se encuentran los problemas asociados al ruido del foto-detector y las perturbaciones externas (fuentes de luz como el Sol, la Luna, las luces de la ciudad, etc.), provocando dificultades para el discernimiento de la señal tanto en la etapa de rastreo como la de seguimiento de los faros.

Aunque estos tipos de movimientos y perturbaciones mecánicas y atmosféricas no forman una lista exhaustiva de lo que describe e influye a un satélite; sí logran dar una idea de la magnitud y complejidad del problema asociado al apuntamiento espacial.

A continuación se vera el sistema experimental propuesto por el CICESE para lograr el enlace óptico espacial Tierra-satélite y las partes que lo componen.

## **1.4 Carga Útil Óptica.**

En el Laboratorio de Fibras Ópticas del Departamento de Electrónica y Telecomunicaciones de CICESE se desarrollan proyectos en comunicaciones ópticas. Dentro de estos proyectos se incluye, como ya se había mencionado, el denominado "carga útil óptica de comunicaciones para SATEX.1". Dicho proyecto consiste en implementar un sistema transmisor-receptor para comunicaciones ópticas desde un microsatélite a Tierra utilizando como antena receptora y transmisora montajes telescópicos. Es en este entorno donde se desarrolla el presente trabajo de tesis.

La Carga Útil Óptica (CUO) es un sistema experimental para validar las comunicaciones ópticas en aplicaciones espaciales, lo que significa que no está destinado a reemplazar, por el momento, algún sistema convencional de comunicación Tierra-satélite sino a mostrar los problemas inherentes al establecimiento de este tipo de enlaces ópticos y sobre todo, generar recursos humanos en el estado del arte de las comunicaciones ópticas.

CUO contiene un sistema de control que le permite establecer y mantener el enlace por medios opto-electro-mecánicos mediante el rastreo, adquisición y seguimiento de un faro óptico. El diseño del sistema óptico se encuentra basado en un transceptor óptico desarrollado por la Jet Propulsion Laboratory (JPL).

---

<sup>5</sup> Para lograr la actualización de los parámetros, el experimento se apoya en un enlace por radio entre la ET y el sistema de apuntamiento.

En la Figura 5 se puede apreciar un diagrama general del enlace de comunicaciones ópticas para SATEX.1. De aquí se puede observar algunos detalles importantes del sistema de apuntamiento como son: el sistema se implementa tanto en el satélite como en la ET; el sistema de comunicación de VHF (telemetría, mando y control) se utiliza para comunicarse con la computadora de CUO a través de la CP; se utilizan dos faros ópticos con diferentes frecuencias para las portadoras así el haz de subida se encuentra en el rango de luz visible y el de bajada en el infrarrojo cercano.

Dado que no se menciona claramente en el diagrama, debe aclararse que la única comunicación directa entre la ET y el satélite es con la CP. Ésta se encarga, entre otras cosas, de redirigir los mensajes enviados empleando un ducto propietario y de transmitir a ET los mensajes de las otras computadoras de a bordo.

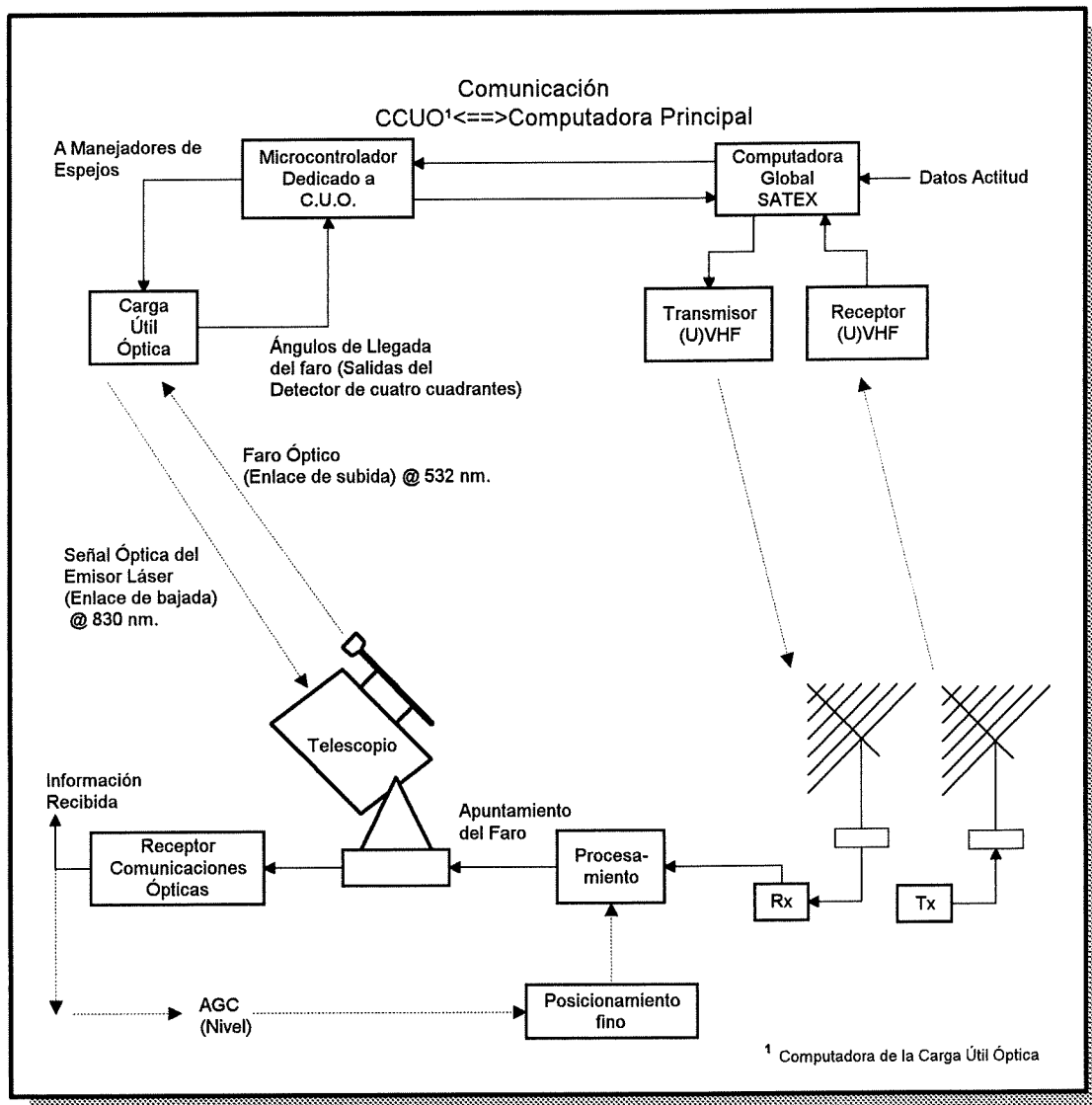


Figura 5. Diagrama general del enlace óptico para SATEX.1.

A su vez, CUO consta de varios módulos que desempeñan diferentes funciones formando en su conjunto el experimento óptico para SATEX.1 (ver Figura 6). Dichos módulos son:

- R3 Computadora de la Carga Útil Óptica (CCUO).
- T1 módulo del láser.
- R2 foto-detector de avalancha.
- R4 módulo de los motores (manejadores y motores).
- Las interfaces, compuestas por la programación del monitor mínimo de CCUO, las rutinas de comunicación, la rutina de adquisición de parámetros; programación de las señales de control, rutinas de temporización, etc. Enlazadas en un esquema de interrupciones.
- Módulo R3.1 que refuerza y aísla las líneas de comunicación y control así como el suministro de la alimentación al resto de los módulos.

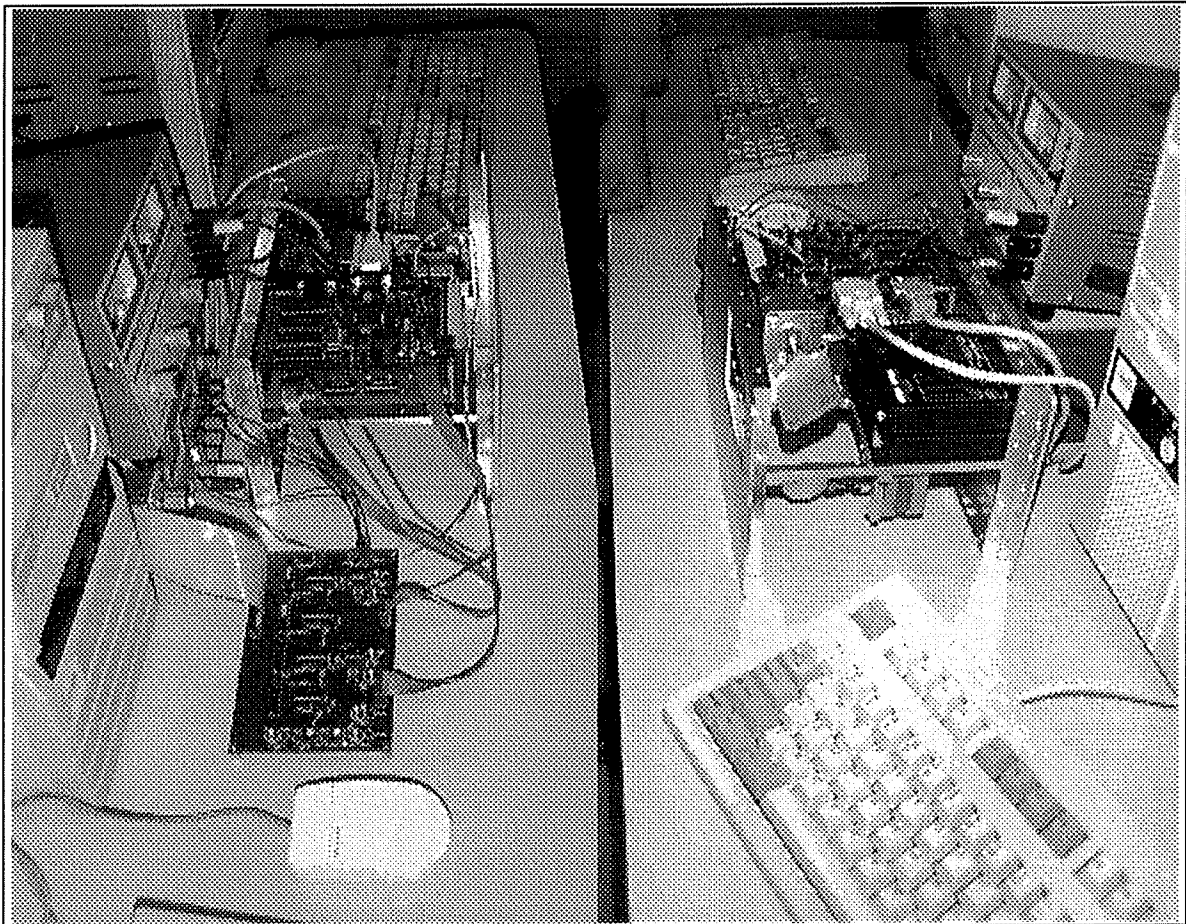


Figura 6. Dos vistas diferentes del prototipo de CUO.

CCUO (módulo R3) contiene toda la inteligencia del sistema. Contendrá un monitor mínimo que se encargará de funciones tales como la configuración del microcontrolador, comunicación serie

(entre CUO y la CP), telecargado y activación de programas. El módulo T1 contiene el láser, manejadores y lentes colimadores. El módulo R2 tiene el foto-detector de avalancha con su circuitería de amplificación, demodulación y filtrado. El módulo R3.1 contiene un par de convertidores para el suministro de potencia así como etapas de aislamiento y reforzamiento de las líneas de control.

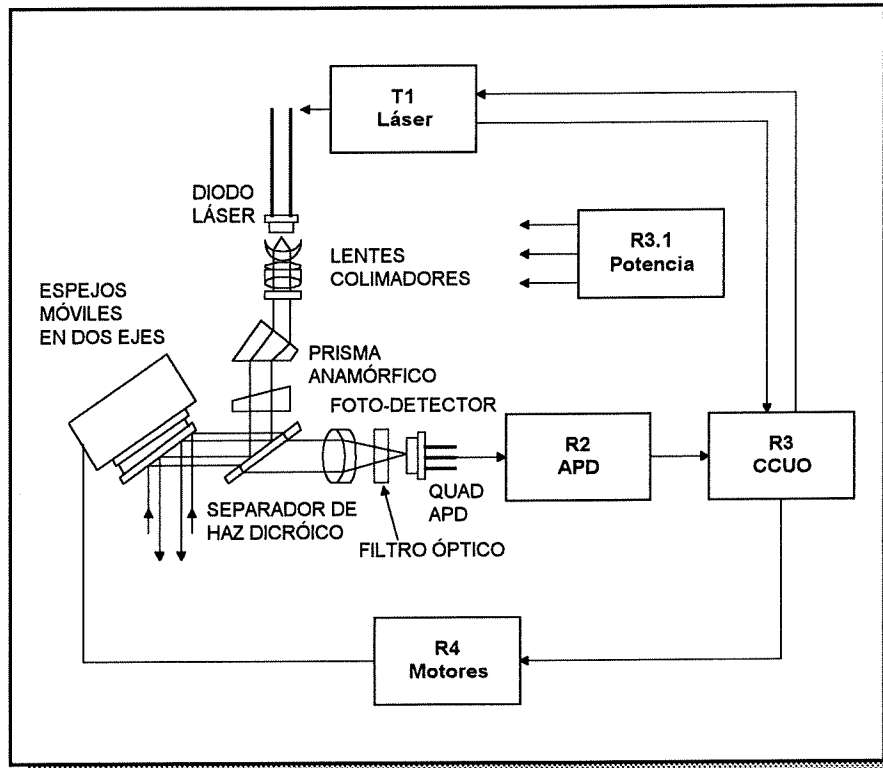


Figura 7. Diagrama a bloques de CUO.

Algunos otros detalles interesantes de mencionar son los referentes a las dimensiones de la plataforma de CUO que son 12x8x6 pulgadas y cuyo peso queda limitado a 8 kg; además, la placa de montura de CUO está construida de aluminio y sirve a su vez como disipadora de calor tanto por conducción como por radiación.

## 1.5 Plataforma básica de CUO

Aquí se describirá la composición básica de cada uno de los módulos que integran a CUO.

### 1.5.1 CCUO.

CCUO se compone de los siguientes elementos:

- Microcontrolador SIEMENS SAB80C166MT3 RISC de 16-bits que opera a 40 MHz. (20 MHz. Interno), temperatura extendida (-40 °C a 85 °C) en un encapsulado P-MQFP100-2.
- Cristal de cuarzo de 40 MHz de nivel TTL de salida, intervalo de temperatura de -55 a 125°C.
- MLE de 32 Koctetos (Memoria de Lectura Exclusiva).
- MAA de 32 Koctetos (Memoria de Acceso Aleatorio). Con la opción de poder agregar hasta 64 Koctetos de MAA en caso de ser necesario.
- Manejador MAX222CPE para satisfacer las características eléctricas de la norma RS-232.
- GAL16V8B-15LD para la lógica de telecargado e inicialización.

Todos los componentes utilizados en esta tarjeta son de grado militar. Los componentes pasivos y el microcontrolador son de montaje superficial. El circuito impreso se diseñó en DET-CICESE y fue construido en E.U.A. con especificaciones militares.

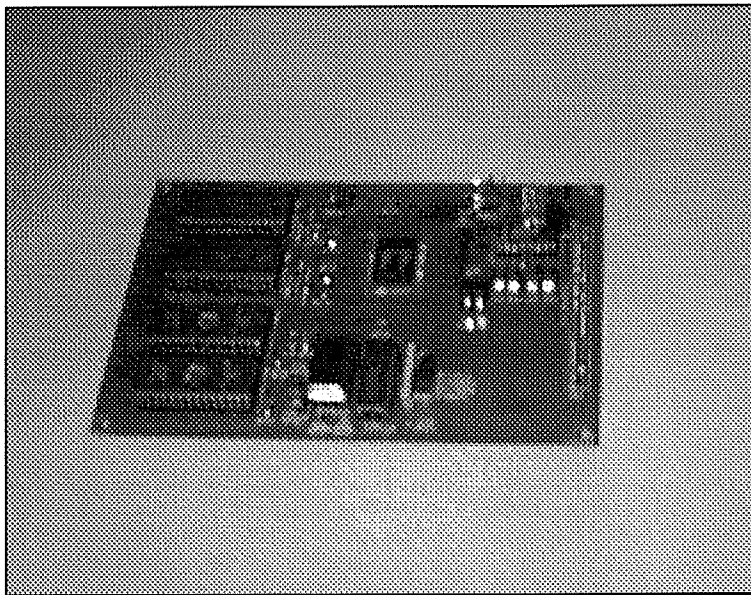


Figura 8. Computadora de la Carga Útil Óptica.

### 1.5.2 Módulo de potencia R3.1.

Este módulo se diseñó para suministrar la potencia a los módulos R3 y R4 así como para aislarlos eléctricamente. En esta tarjeta se colocó un convertidor DC/DC de 28V a 28V para los

manejadores de los motores de pasos y un convertidor DC/DC de 28V a 5V para la computadora y la lógica adicional. También se le agregaron un par de integrados GAL16V8B que funcionan como reforzadores de las líneas de control para los motores de pasos, los manejadores, y una señal de sincronización. Las señales de control están optoacopladas para aislarlas del ruido generado por los motores y los manejadores.

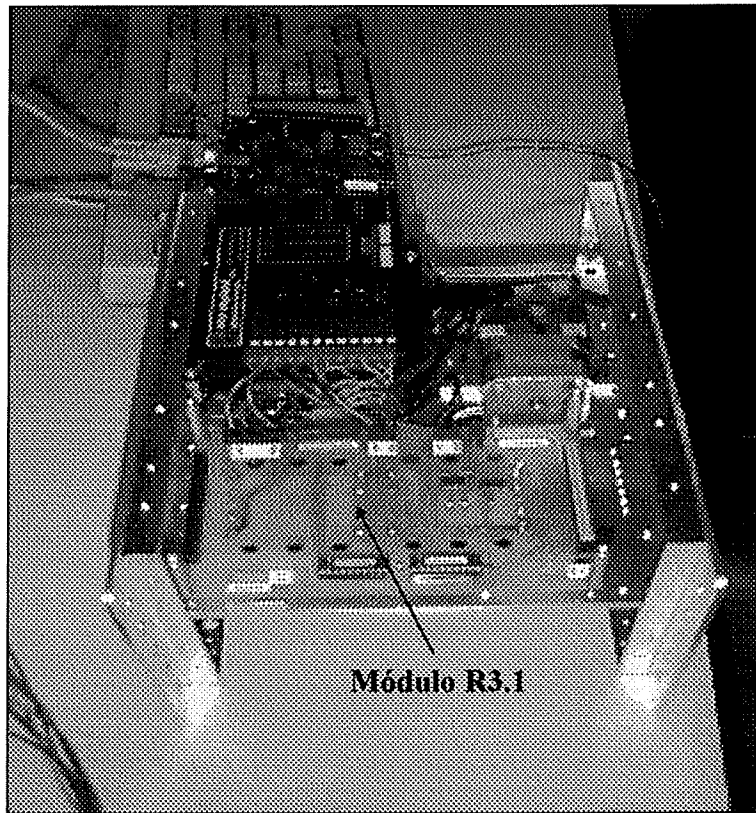


Figura 9. En la figura se puede apreciar el módulo R3.1 de potencia y acondicionamiento.

Por medio de esta tarjeta se puede activar y desactivar los manejadores de los motores de pasos y tener control sobre la corriente suministrada a los motores. Lo anterior permite ahorrar energía mientras el satélite se encuentre fuera de línea de vista así como controlar el par de los motores de pasos.

### 1.5.3 Módulo T1 (láser).

En este módulo se encuentra el láser y la óptica. El haz es generado por un diodo emisor de luz en el infrarrojo cercano (200 mW @ 830 nm) dirigidos por un par de espejos que se encuentran en el módulo R4 (motores y manejadores). También se encuentra en T1 unos lentes colimadores que tienen la función de convertir un haz divergente en un haz paralelo y así evitar su dispersión. Para el control del láser se ocupan dos módulos que controlan la temperatura y la corriente del mismo.

#### **1.5.4 Módulo R2 foto-detector de avalancha.**

El foto-detector de avalancha o APD (Avalanche Photo-Diode) es un foto-detector de cuatro cuadrantes con un campo visual de 400-1100 nm que se usa para la detección del láser de subida tanto en la fase de rastreo como en la de seguimiento. Se utilizan filtros ópticos para reducir el ruido ambiental y mejorar la relación señal a ruido (*ver* Figura 7).

#### **1.5.5 Módulo R4 (motores).**

Son parte de este módulo los espejos, manejadores de los motores y los motores en sí. Los espejos dirigen, inicialmente, al haz de bajada durante el rastreo pero una vez que se logra el enlace mantienen también al haz de subida centrado en el foto-detector, ya que los dos usarán la misma trayectoria (*ver* Figura 7).

Los motores empleados son motores de pasos de 200 pasos por revolución (1.8° por paso). Proporcionan un par máximo de 23 onzas-pulgada con un tamaño de cuadro NEMA 17 (40 mm).

Se utilizan manejadores de potencia para manipular los motores de pasos. La ventaja de estos manejadores es que se puede tener hasta 10 micropasos por cada paso del motor, obteniendo un control más fino. Los manejadores se encargan de eliminar los efectos de resonancia de baja frecuencia, típicos en esta clase de motores.

Los espejos presentan características adecuadas a los requerimientos de la óptica. Se trata de un par de espejos cuyos recubrimientos están diseñados específicamente para la longitud de onda que se maneja.

#### **1.5.6 Interfaces.**

Contempla el objetivo de este trabajo y será tratado en lo que resta de éste.

## **1.6 Objetivos del trabajo.**

Diseñar e implementar un monitor mínimo para CUO que realice las funciones de comunicación, telecargado y configuración inicial del microcontrolador. Trabajando exclusivamente en el módulo de CCUO.

Elaborar interfaces para la carga útil óptica de SATEX.1 que sirven para enlazar los diferentes dispositivos que lo componen. Esto implica la configuración y administración de los diferentes periféricos del microcontrolador.

El trabajo se dividió en seis partes:

- Monitor mínimo.
- Interfaz serie.
- Interfaz con motores.
- Interfaz con láser.
- Interfaz con APD.
- Reloj de tiempo real.

El monitor mínimo funcionará como el administrador de la secuencia lógica del programa y se encarga de la inicialización del microcontrolador. Para la interfaz serie es necesario describir el protocolo de comunicaciones empleado en SATEX.1 mismo que fue desarrollado por el Instituto de Ingeniería de la UNAM. Lo anterior formará la base fundamental de la implementación de dicha interface. La sección de interfaz con los motores mostrará el fuerte vínculo que existe entre la mecánica y la programación del microcontrolador para lograr el control de la posición de los motores y con ello el posicionamiento del haz del faro de subida sobre el foto-detector. Por su parte, la interfaz con el láser permite realizar telemetría sobre los parámetros de potencia, corriente y temperatura de esta fuente de luz coherente así como de su modulación. Para el foto-detector de cuatro cuadrantes, la interfaz captura y codifica la información producida por la impresión de la huella del faro de subida sobre el foto-detector de avalancha. En el caso del reloj de tiempo real se logra sincronizarlo y configurarlo lo que permite programar el evento de inicio de cobertura o modo de rastreo del faro de subida.

---

## 2. PROTOCOLO DE COMUNICACIÓN EN SATEX.1.

El protocolo de comunicación para la red empleada por SATEX.1 se diseñó de manera exclusiva para este proyecto. La responsabilidad del diseño e implementación del protocolo y la red fue del Instituto de Ingeniería de la UNAM.

A criterio de dicha Institución, el empleo de un protocolo de comunicaciones existente resultaba impráctico para las necesidades de SATEX.1, por lo tanto, se optó por diseñar un protocolo diferente que satisficiera dichas necesidades. Como resultado, se obtuvo un protocolo para red interna del satélite que permite la comunicación entre nodos. El protocolo, al que denominaremos por las siglas II-UNAM, permite transmitir un mensaje seleccionando al destinatario de la información así como su remitente. Como restricción del protocolo, se tiene que los equipos conectados dentro de la red del satélite no puedan transmitir al ducto a menos que se los demande CP o CR, o en el eventual caso de detectar un error en la comunicación al intentar recibir un mensaje. La CP es la única que puede estar transmitiendo por el ducto constantemente ya que funge como administrador del canal de comunicaciones internas mientras que CR<sup>6</sup> se encuentra de respaldo.

La comunicación dentro de la red interna se logra con mensajes definidos por el protocolo II-UNAM que pueden ser de tres tipos:

- Mandos internos (palabras de control).
- Mandos externos.
- Programas.

En cada uno de estos mensajes se define el destinatario. En los mandos internos se puede definir, además, el remitente del mensaje.

La unidad receptora deberá reconocer los diferentes tipos de mensajes y ejecutar rutinas predefinidas. A veces se requiere transmitir más de un octeto para completar el mensaje, razón por la cual el protocolo define tres tipos de mensajes con diferente formato y longitud que permiten transmitir información adicional.

Así pues, la diferencia entre estos tipos de mensaje es su longitud y formato. Los mandos externos son normalmente enviados desde la estación terrena pero pueden ser utilizados para enviar información por cualesquier unidad conectada a la red. Los mandos internos solo utilizan un octeto por tener funciones predefinidas. El formato para programas permite enviar una cantidad mayor de información de manera continua.

---

<sup>6</sup> Cuando se menciona CP queda implícito CR ya que son redundantes, por lo que de aquí en adelante solo se mencionará CP.

El protocolo especifica que deberá existir un tiempo de espera de 30 ms entre las dos palabras de control (“salir de bajo consumo de energía” y la función) que se envían al inicio del mensaje; el resto de la información se puede mandar de manera continua.

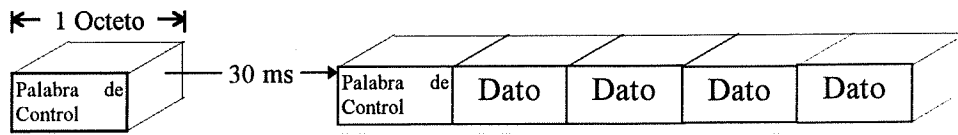


Figura 10. Tiempo de espera entre las palabras de control.

A continuación describiremos cada una de las diferentes opciones de mensaje dentro de la red.

### 2.1 Mandos internos.

La palabra de control para las comunicaciones internas consta de un octeto y tiene el siguiente formato:

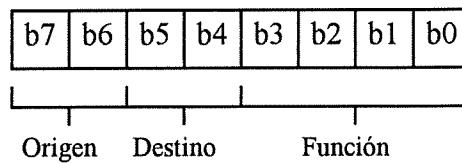


Figura 11. Palabra de control (mando interno).

b6 Y b7 dan el origen del mensaje como se indica en el Tablero I

Tablero I. Tabla de decodificación de origen.

b7	b6	Origen
0	0	Computadora Principal (CP/CR).
0	1	Computadora de la Cámara.
1	0	Computadora de la Carga Útil Óptica.
1	1	Decodificador de Tonos (8751).

mientras que b5 y b4 indican el destino del mensaje (Tablero II)

Tablero II. Tabla de decodificación del mensaje.

b5	b4	Destino
0	0	Computadora Principal (CP/CR).
0	1	Computadora de la Cámara.
1	0	Computadora de la Carga Útil Óptica.
1	1	Decodificador de Tonos (8751).

La función puede ser instrucción o mando para el destinatario y se decodifica con los bits b3, b2, b1, b0, teniendo las opciones siguientes:

Tablero III. Tabla de decodificación de funciones.

b3	b2	b1	b0	Función
0	0	0	0	Solicita al destino ejecutar su rutina de diagnóstico.
0	0	0	1	Diagnóstico correcto.
0	0	1	0	Diagnóstico incorrecto.
0	0	1	1	CP o CR piden a la cámara transferir imagen.
0	1	0	0	Reset.
0	1	0	1	Transferencia de parámetros.
0	1	1	0	Retransferir bloque de una imagen.
0	1	1	1	Desactivar interrupción por puerto serie para comunicación interna, por un tiempo dado.
1	0	0	0	Retransmisión.
1	0	0	1	Sincronizar temporizadores con algún evento.
1	0	1	0	Avisa a CCUO que ET detectó enlace óptico.
1	0	1	1	Aviso de que CP o CR está en mal estado.
1	1	0	0	Transferir nuevo programa.
1	1	0	1	Transferir mandos.
1	1	1	0	Poner en bajo consumo de potencia.
1	1	1	1	Salir de bajo consumo de potencia o ejecutar rutina de interrupción por puerto serie.

La descripción de las funciones que se pueden incluir en la palabra de control es como sigue:

*Solicita al destino ejecutar su rutina de diagnóstico.* Esta instrucción es un mando interno y se enviará rutinariamente por CP en cada período orbital después de pasar por México a todas las unidades para que realicen un autodiagnóstico.

*Diagnóstico correcto.* Mensaje que sirve como respuesta para indicar que no se encontraron errores como resultado del autodiagnóstico pedido.

*Diagnóstico incorrecto.* Mensaje para indicar que se reportó error en el sistema después de ejecutar el autodiagnóstico.

*Reset.* Se le pide con este mando a la unidad receptora que reestablezca su sistema.

*Transferencia de parámetros.* Mando que demanda a CCUO realice telemetría del láser. Los datos son enviados a CP con el formato de mandos externos y con destino hacia ET.

*Desactivar interrupción por puerto serie para comunicación interna por un tiempo dado.* Ya que la cámara CCD utiliza otro protocolo de comunicación diferente, esta instrucción permite que la cámara envíe sus datos a través del ducto sin interferir con otros dispositivos.

*Sincronizar temporizadores con algún evento.* Para aplicaciones en las que la exactitud no es un factor relevante, es posible sincronizar los temporizadores de CCUO con algún evento mediante el empleo de este mando. Para la sincronización del reloj de tiempo real y el inicio del rastreo del láser para localizar la ET se emplea un esquema de circuitería y programación más complejo.

*Avisa a CCUO que ET detectó enlace óptico.* Sin comentarios.

*Aviso de que CP o CR está en mal estado.* Mensaje que se envía a ET para informar que una de las computadoras está en mal estado.

*Transferir nuevo programa.* Con este mensaje preparamos a la unidad receptora para que reciba un nuevo programa. El programa monitor se encarga de recibir al nuevo programa pero no lo ejecuta.

*Transferir mandos.* Mensaje interno para avisar que se transmitirá un mensaje externo. En los mensajes externos se pueden incluir instrucciones que no se encuentran definidas en el protocolo.

*Poner en bajo consumo de potencia.* Mensaje interno para poner a un dispositivo en estado de mínimo consumo de energía.

*Salir de bajo consumo de potencia o ejecutar rutina de interrupción por puerto serie.* Mensaje interno para reactivar a los dispositivos que se encuentren en un estado de bajo consumo de energía y así prepararlos para recibir un mensaje.

Un detalle importante del protocolo está relacionado a mandos enviados por la estación terrena y recibidos por CP en los cuales el mensaje lleva otro destino diferente a CP. En este caso el mensaje es retransmitido hacia su destinatario a través de la red interna con el mismo formato que fue recibido. Este formato será explicado más adelante.

Procedimiento para transmitir un mensaje :

1. Enviar palabra de control “Salir de bajo consumo de potencia” (para todas las unidades).
2. Enviar palabra de control de la función requerida (*ver* Tablero III).

Los mandos internos son los más sencillos de los tres tipos de mensajes dado que cada palabra de control se encuentra predefinida en el protocolo y no requiere de mayor decodificación. El mando interno siempre es enviado antes de un mando externo o un programa nuevo, e indica si sigue más información y su longitud.

### 2.2 Mandos externos.

En mandos externos se reciben 3 paquetes de 28 octetos. Cada paquete contiene la misma información. El hecho de que se reciban tres veces es para asegurar que se recibe correctamente el mensaje sin error en la transmisión. Se tiene como verificación de error la suma de prueba de los datos que viene incluida en la transmisión dentro de los 28 octetos. Si acaso uno de los paquetes difiere de los demás se pedirá al remitente que retransmita la secuencia inmediatamente después de concluido el mensaje.

Procedimiento para transmitir un mensaje externo en la red interna :

1. Enviar palabra de control “Salir de bajo consumo de potencia” (para todas las unidades).
2. Enviar palabra de control “Transferir mando” para unidad receptora.
3. Enviar mensaje de 28 octetos.
4. Repetir los pasos 1 a 3 dos veces más.

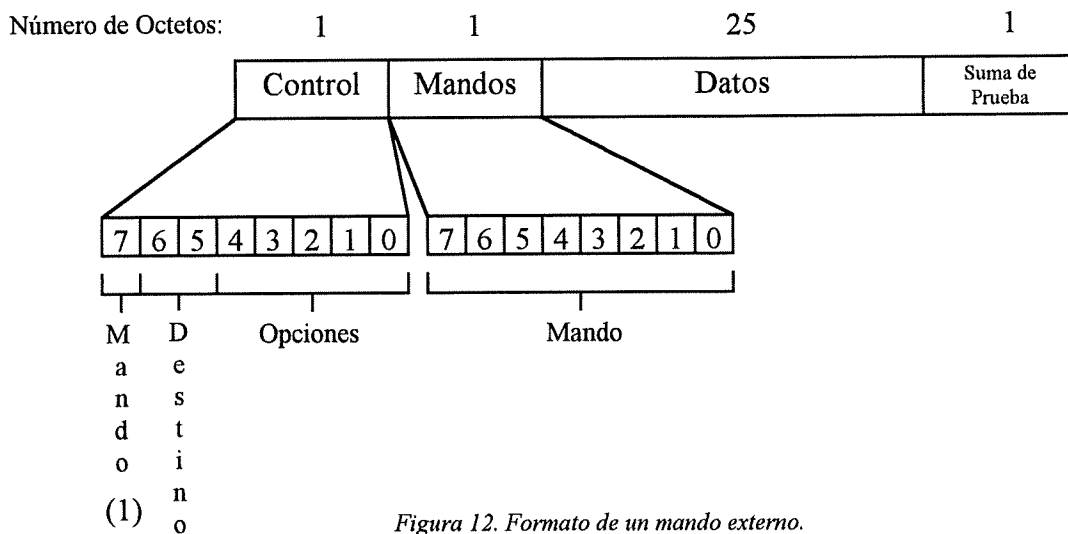


Figura 12. Formato de un mando externo.

### 2.3 Programas.

Un programa nuevo puede tener una longitud hasta de 64 paquetes de 503 octetos cada uno. Cada paquete se transmite una sola vez e incluye una suma de verificación para validar los datos. Si se encuentra un error en uno de los paquetes se indicará con la petición de una retransmisión completa del programa al terminar el envío de los paquetes.

Para este caso, en que se puede transmitir varios bloques seguidos, no es necesario enviar los octetos de control “salir de bajo consumo de potencia” y “desactivación de canal de recepción”, entre cada paquete del programa, solo es necesario volverlos a transmitir para el próximo mensaje una vez que se ha terminado de enviar todos los paquetes del programa.

Procedimiento para transmitir un programa en la red interna :

1. Enviar palabra de control “Salir de bajo consumo de potencia” (para todas las unidades).
2. Enviar palabra de control “Transferir nuevo programa”.
3. Enviar programa en bloques de 500 octetos con el formato de programas.

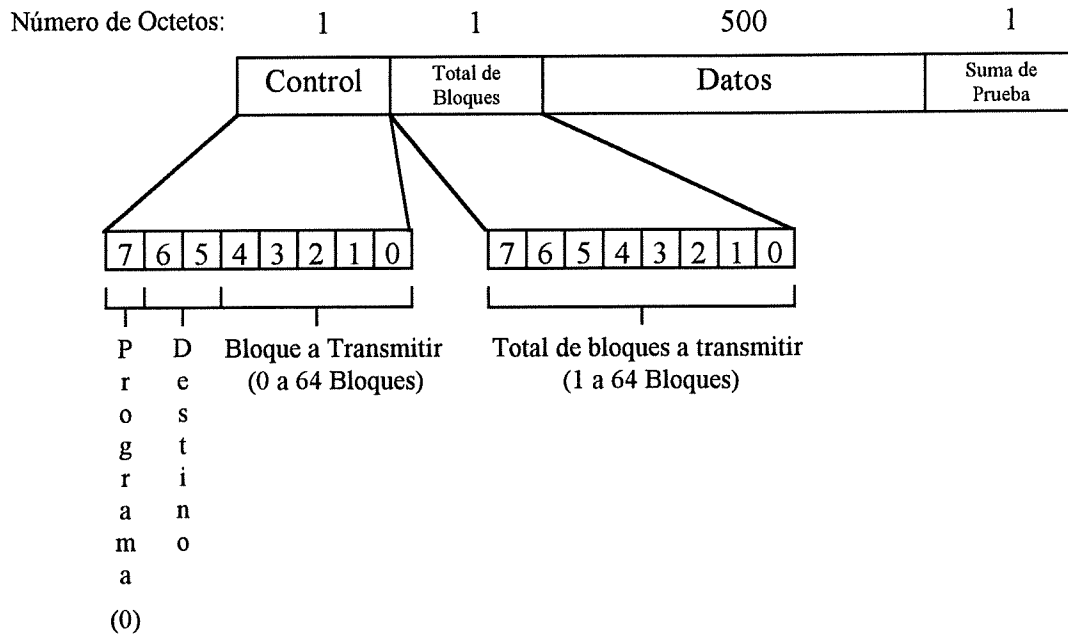


Figura 13. Formato para subir programas.

Para ciertos mandos internos o externos, se espera una respuesta de la unidad receptora. Es entonces cuando CCUO puede transmitir algo en la red interna además de algún mensaje de error en la recepción del mensaje. CCUO deberá estar en escucha el resto del tiempo.

El protocolo define cuantos datos serán enviados dependiendo de la palabra de control que se transmitió como encabezado de un paquete. En el caso de transmisión de un programa nuevo se debe especificar cuantos bloques o paquetes serán enviados.

Como se había mencionado, se debe transmitir una palabra de control a cada unidad conectada a la red antes de enviar el mando o bloque. Esta palabra de control sirve para alertar a las unidades que a continuación se transmitirá un mensaje. La segunda palabra de control es el mensaje en sí e indica para quién va dirigido. Todas las otras unidades, para las cuales no es el mensaje, deberán de esperar 5 minutos con su canal de recepción desactivado para evitar alguna confusión mientras se transmite el mensaje. Esto quiere decir que si un paquete se termina de transmitir antes de 5 minutos, un próximo mensaje para otro destinatario no se debe de enviar si no hasta que transcurra el tiempo especificado.

CCUO puede recibir los tres tipos de mensaje. Es importante distinguir entre los tres debido a la diferencia de formatos y su longitud.

## 3. ESPECIFICACIONES.

Este capítulo contiene una colección de tableros que listan las especificaciones que debe cumplir el monitor y las interfaces de la Carga Útil Óptica del satélite experimental SATEX.1.

Las especificaciones se plantean básicamente de acuerdo a la funcionalidad deseada para cada interfaz incluyendo el programa monitor. Para esto, dividiremos las especificaciones de acuerdo a la cobertura funcional en dos grupos: *especificaciones generales* y *especificaciones particulares*.

### 3.1 Especificaciones generales.

Dada su formación experimental, que lo sitúa dentro de las fronteras de la tecnología disponible y utilizable, el desarrollo de este muy particular tipo de sistema no se sujeta a normatividades establecidas. Sin embargo, como parte integrante de un sistema global aplicado a una situación real, deberá cumplir con un mínimo de conceptos y/o normas interdisciplinarias como son:

*Tablero IV. Especificaciones generales de la programática.*

<b>Programática</b>
Apegada a la normatividad ANSI C.
De cumplimiento con el protocolo de comunicación para la red interna de SATEX.1 desarrollado por II-UNAM.
En tiempo real.
Tolerante a fallas.
Versátil y fácil de mantener.

*Tablero V. Especificaciones generales de la mecánica.*

<b>Mecánica</b>
De cumplimiento con el estándar RS-232 limitado a las señales de recepción, transmisión y común (niveles de voltaje y parámetros de comunicación).
Con reforzamiento de señales digitales de control.
Con optoacoplamiento de señales de habilitamiento de la etapa de potencia.

### 3.2 Especificaciones particulares.

De nuevo, haciendo énfasis en la funcionalidad, podemos especificar de manera particular cada uno de los siguientes grupos en que se divide la programática:

- Monitor mínimo.
- Interfaz serie.
- Interfaz con motores.
- Interfaz con láser.
- Interfaz con APD.
- Interfaz de tiempo real.

#### 3.2.1 Monitor mínimo.

El monitor mínimo debe ser el administrador de la secuencia lógica del programa, basando su funcionamiento en el empleo de banderas de estado. Dichas banderas deben actualizarse de acuerdo a la presentación de los eventos. Éstos últimos deberán atenderse mediante el llamado de rutinas por programática o interrupción.

Las funciones que deberá cumplir el monitor mínimo se detallan a continuación.

*Tablero VI. Especificaciones del monitor.*

<b>Monitor mínimo</b>
Configuración inicial del microcontrolador.
Configuración de periféricos de acuerdo a las necesidades de los módulos R2 (APD), R4 (motores), y T1 (láser).
Autodiagnóstico.
Comunicación serial para recepción y transmisión de datos entre la Computadora de la Carga Útil Óptica y la Computadora Principal o la Computadora Redundante.
Manejo del centinela (watchdog).
Decodificación y ejecución de mandos.
Administración de la secuencia lógica del programa.
Telemetría de los módulos R2 y T1.
Manejo de los lazos - abierto y cerrado - de control de los espejos para lo cual deberá: <ul style="list-style-type: none"> <li>• Calcular la trayectoria de lazo abierto.</li> <li>• Medir la potencia óptica detectada proveniente del módulo R2.</li> <li>• Controlar el posicionamiento de los motores de pasos (módulo R4).</li> </ul>

Aunque las interfaces y el programa monitor se encuentran altamente vinculados es conveniente especificar a las interfaces en forma independiente como se desarrolla a continuación.

### 3.2.2 Interfaz serie.

Las especificaciones para esta interfaz son dadas por el protocolo de comunicaciones creado en el Instituto de Ingeniería de la UNAM de acuerdo a las necesidades de SATEX.1.

*Tablero VII. Especificaciones para la interfaz serie.*

<b>Interfaz serie</b>
Capacidad de reconocimiento y creación de los diferentes tipos de mensajes a transmitirse a través del ducto interno del satélite así como el cumplimiento de secuencias temporales de acuerdo con el protocolo II-UNAM.
Programación tolerante a errores de comunicación tanto en su forma como de duración de mensaje evitando que el puerto de comunicación se bloquee en caso de error, permitiendo la solicitud de retransmisión al remitente del mensaje de acuerdo al protocolo II-UNAM.
Los parámetros de la comunicación deben ser: <ul style="list-style-type: none"> <li>• Velocidad : 9600 bauds.</li> <li>• Longitud : 8 bits de datos + 1 bit de paro.</li> <li>• Paridad : Ninguna.</li> </ul>
Reconocimiento y ejecución de instrucciones utilizadas en el protocolo II-UNAM.
Reconocimiento de instrucciones exclusivas de CCUO.
Telecargado de un programa.
Ejecución de un programa nuevo.
Aborto de un programa.
Decodificación del formato hexadecimal Intel MCS-86 para el telecargado de programas.
Recepción y transmisión de datos manejados por interrupción.
Utilización de temporizadores para la implementación de los tiempos de guarda establecidos por el protocolo y para apoyo de la programación tolerante a errores.

### 3.2.3 Interfaz con motores de pasos.

*Tablero VIII. Especificaciones para la interfaz de los motores de pasos.*

<b>Interfaz con módulo R4</b>
Generación de pulsos para ambos motores respetando temporizadores.
Generación de la señal de sentido de giro para cada motor.
Lectura de tableros de efemérides que contienen posiciones relativas a la posición anterior, comenzando de una posición conocida.
Colocación inicial de los motores en una posición de referencia.
Cadencia de los motores máxima de 1 kHz.

### 3.2.4 Interfaz láser.

*Tablero IX. Especificaciones para la interfaz láser.*

<b>Interfaz con módulo T1</b>
Adquisición de parámetros del láser.
Tolerancia a errores de sobrecarrera en la adquisición.
Activación de la rutina de transmisión una vez obtenidos los datos.
Generación de señal de 1 kHz para la modulación del láser.

### 3.2.5 Interfaz con foto-detector de cuatro cuadrantes.

*Tablero X. Especificaciones para la interfaz con foto-detector de cuatro cuadrantes.*

<b>Interfaz con módulo R2</b>
Adquisición de las señales provenientes del fotodetector de cuatro cuadrantes.
Activación de la rutina de transmisión a petición explícita.
Tolerancia a errores de sobrecarrera en la adquisición.
Muestreo automático del APD cuando los espejos están en movimiento.

### 3.2.6 Relojes de Tiempo Real.

*Tablero XI. Especificaciones para el reloj de tiempo real.*

<b>Relojes de Tiempo Real</b>
Capacidad de configuración de reloj desde la estación terrena. <sup>7</sup>
Sincronización de reloj (inicio de cobertura) con estación terrena por medio de un evento externo a CUO.
Adecuada resolución para los períodos de inicio de cobertura, fin de cobertura, y tiempo base requeridos.

<sup>7</sup> Incluidos con los datos de configuración se enviarán parámetros orbitales de SATEX.1 que deben ser extraídos para el cálculo de órbita que realizará CCUO antes de comenzar el rastreo de la estación terrena.

## 4. DISEÑO DE LAS INTERFACES HACIA LOS ELEMENTOS DE LA CARGA ÚTIL ÓPTICA

El trabajo se dividió en seis partes:

- Monitor mínimo.
- Interfaz serie.
- Interfaz con motores.
- Interfaz con láser.
- Interfaz con foto-detector de cuatro cuadrantes.
- Reloj de tiempo real.

Los programas que aquí se describen son incluidos en un disco, con formato para MS-DOS, en esta tesis. El disco lleva la etiqueta de “*Interfaces para la Carga Útil Óptica de SATEX.1*”. Los programas fueron elaborados siguiendo el formato ANSI C y compilados con el programa CrossView C-166 Cross-Compiler v5.0 r0 para el microcontrolador SAB 80C166 de Siemens.

### 4.1 Monitor mínimo.

El monitor mínimo o programa principal (rutina *main()*) se diseñó con el propósito de tener un control sobre las tareas que realiza CCUO. Como se vió en la sección 3.2.1, la interfaz maneja la configuración inicial del microcontrolador y lleva la secuencia lógica del programa; también se encarga de la inicialización de variables y periféricos del microcontrolador.

El monitor mínimo se encontrará en un ciclo infinito revisando el estado del registro CCUOCON. Es en este registro donde se le indica al monitor la función o tarea a realizar. En la Figura 14 vemos el contenido del registro CCUOCON. La mayoría de las banderas manejan un nivel lógico de 1 para indicar que existe la condición citada, CCUOCON.6 es la excepción por que indica si llega un programa o mando.

15	14	13	12	11	10	9	8
		Retransmisión de parámetros	Operaciones de punto flotante correctas	Centrado de motores	Próximo Dato		Sincronizado
7	6	5	4	3	2	1	0
Enlace no establecido	Cálculo de trayectoria	Mando=1 Programa=0	Mensaje RXD	Mensaje TXD	Error de conversión A/D	Error de comunicación	Mal funcionamiento

Figura 14. Registro de banderas CCUOCON.

Las primeras tres banderas del registro son para indicar errores.

CCUOCON.0 es usado por la rutina de autodiagnóstico *chkram()* para señalar si detectó un error en la memoria del sistema. La prueba solo se realiza a la MAA excluyendo la memoria interna. Un diagrama de flujo sobre el procedimiento de la rutina de autodiagnóstico lo podemos observar en la Figura 15. En general esta bandera solo indica si la memoria es confiable.

CCUOCON.1 es manejada por las rutinas relacionadas con la recepción de datos. Sirve para indicar un error en la transmisión. El error puede ser detectado por la mecánica o por programática (ver sección 4.2.1). Cuando se fija esta bandera el monitor mínimo activará la rutina de transmisión.

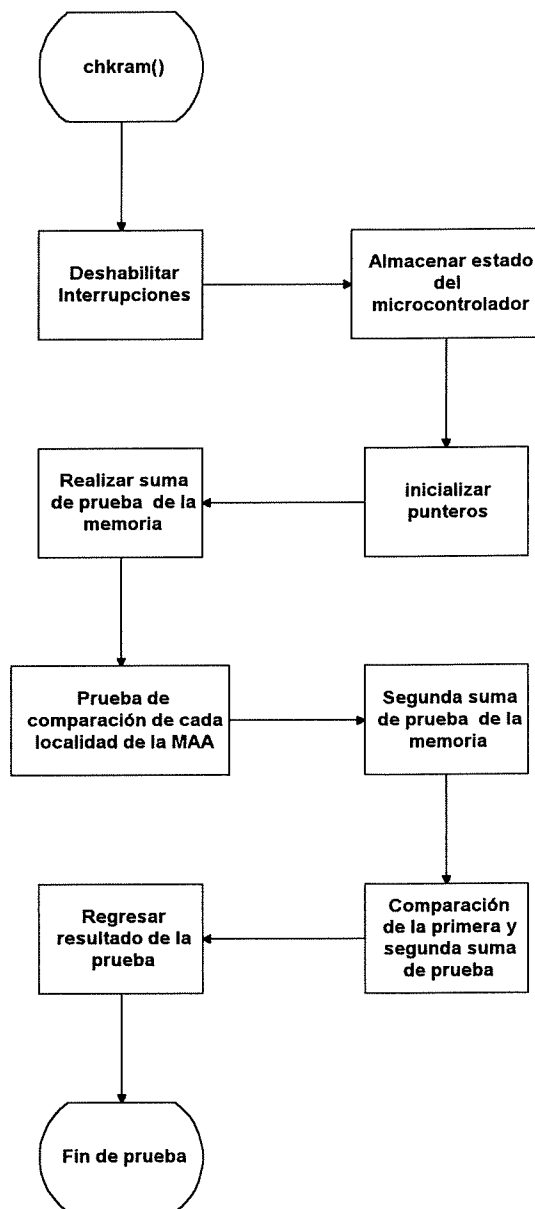


Figura 15. Diagrama de flujo de la rutina de autodiagnóstico *chkram()*.

CCUOCON.2 indica al programa principal que el convertidor A/D sufrió un error de sobrecarrera, por lo que debe volverse a tomar las mediciones antes de utilizar los resultados.

Después de recibir en su totalidad un mensaje la rutina de recepción activa la bandera CCUOCO.4 para que el programa principal llame la rutina de decodificación *decod\_men()* y el mensaje sea interpretado.

Activando CCUOCON.6 el monitor mínimo llama la rutina *trayectoria()* para comenzar el cálculo de las efemérides. Esta bandera es activada después de recibir datos para el RTRIC y configurar sus temporizadores. La rutina *trayectoria()* requiere de un parámetro orbital actualizado que es enviado junto con los datos del RTRIC (ver sección 4.2.5).

Si al finalizar el rastreo de la estación terrena no se logra establecer contacto la rutina *cadencia()* fija la bandera CCUOCON.7 para indicar la situación. Al mismo tiempo esta bandera sirve para reestablecer las variables y temporizadores utilizados durante rastreo y para desactivar la potencia a los motores (conservación de energía).

CCUOCON.8 se activará una vez que se logra sincronizar el RTRIC. La activa la rutina *ser\_CAPIN()*.

CCUOCON.11 le sirve a la rutina *home()* (ver sección 4.2.2) para llevar los espejos a una posición conocida.

Con CCUOCON.12 se puede saber si CUO realizó el cálculo de las efemérides sin problemas. Es activada dentro de la rutina *trayectoria()* al finalizar los cálculos. En caso de que no sea activada después de que se pide a CUO que realice los cálculos, indicaría que se tuvo un error debido a las operaciones de punto flotante.

En la Figura 16 se puede observar un diagrama de flujo de el programa principal o monitor mínimo.

## **4.2 Interfaces.**

La Figura 17 muestra las interfaces en SATEX.1.

Cada interfaz trabaja individualmente y están diseñadas para que la interferencia con otras operaciones sea mínima. Esto se logra con la administración de los recursos del microcontrolador y programando de manera modular. Las interfaces son administradas por un programa principal que trabaja en un ciclo infinito.

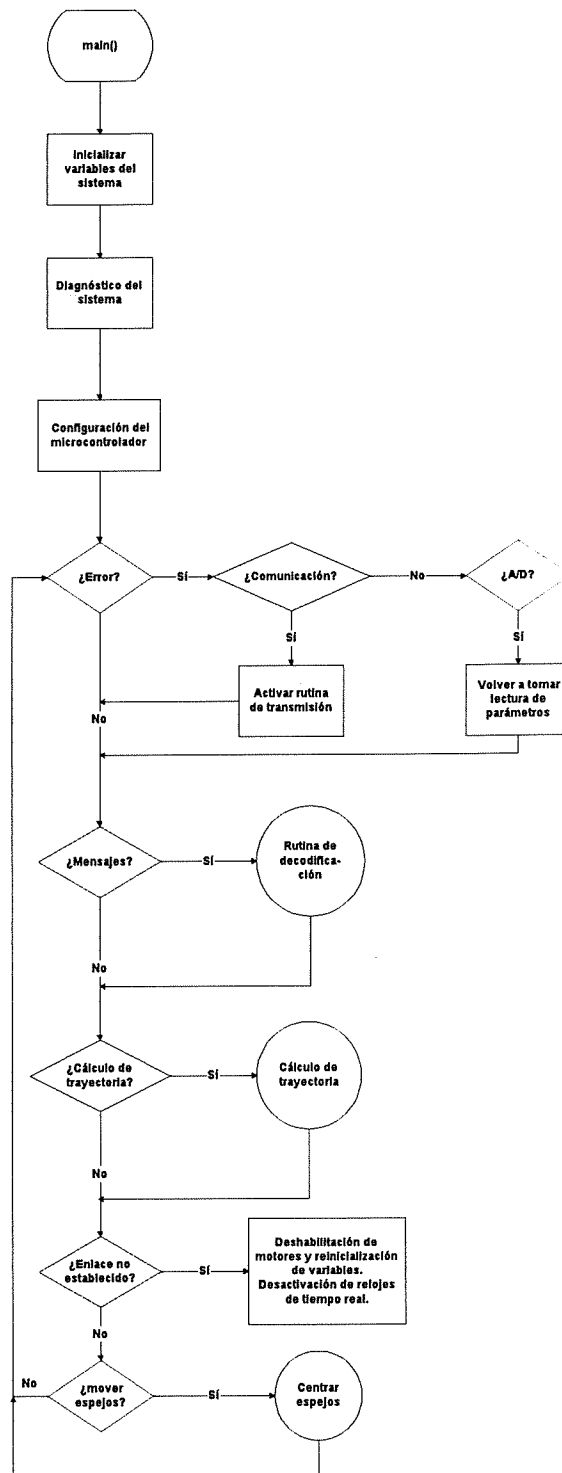


Figura 16. Programa principal o monitor mínimo.

En general, los diferentes periféricos del microcontrolador con los que trabajan las interfaces son configurados al inicializar el microcontrolador. Sin embargo en ciertas situaciones surge la necesidad de reconfigurar algún periférico para atender diferentes casos.

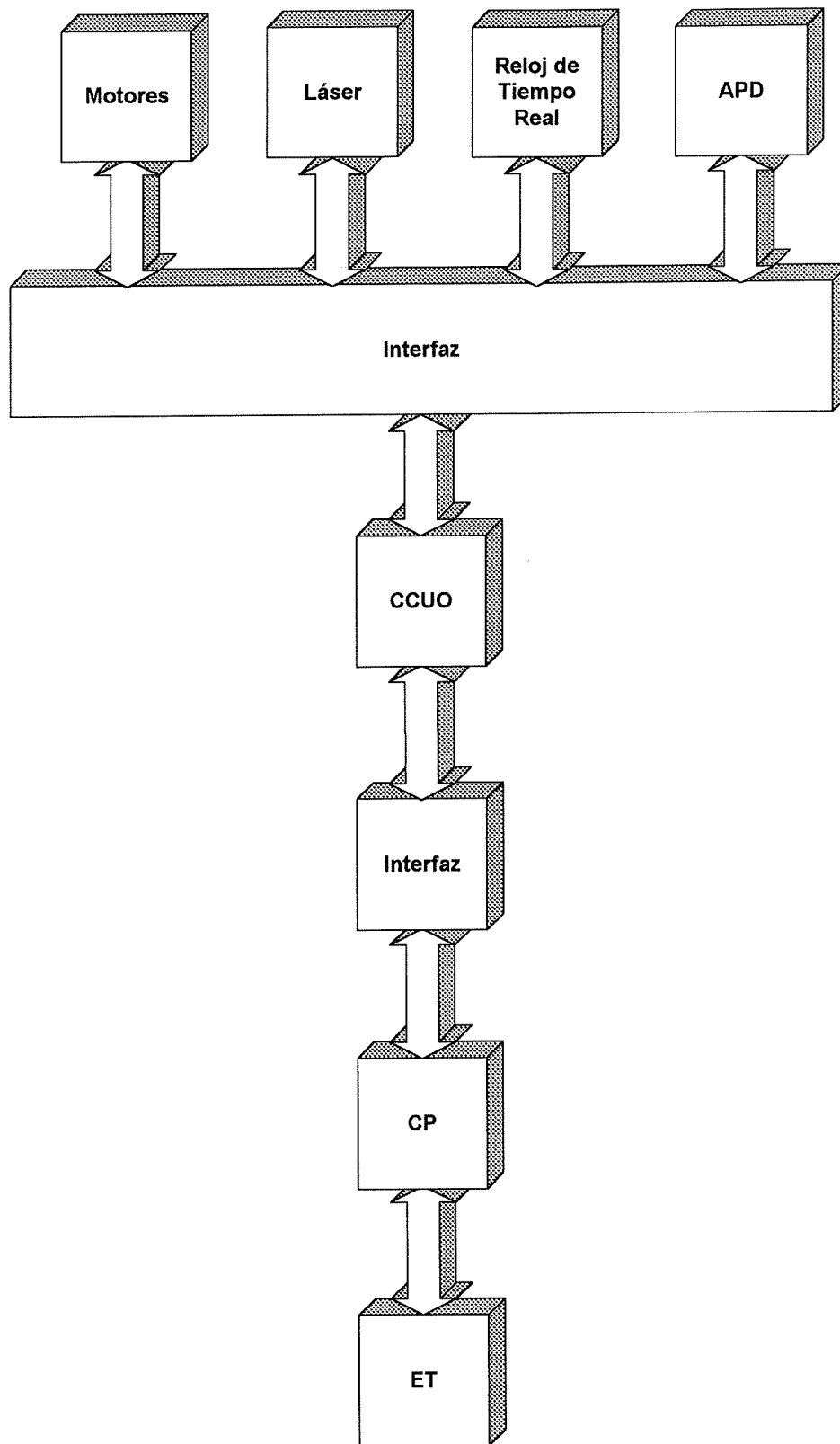


Figura 17. Diagrama a bloques del microcontrolador, elementos externos e interfaces de CCUO.

### 4.2.1 Interfaz serial.

La interfaz serie es probablemente el más importante de las interfaces dentro de la computadora de CUO. Con el podemos comenzar un programa, cambiarlo, abortarlo, así como mandar instrucciones y datos específicos. La interfaz se compone de tres partes. La rutina de transmisión, la de recepción, y la de temporización. La rutina de recepción a su vez se divide en dos más; una para manejar el protocolo de SATEX.1 y otra para decodificar la información de los mensajes. La rutina de transmisión solo se compone de una parte que consiste en la transmisión de los datos de acuerdo al protocolo. La rutina de temporización da los tiempos de guarda especificados por el protocolo en la transmisión y evita el bloqueo del canal de recepción.

Cuando se inicializa el microprocesador o se comienza un programa se inicializan variables que se requieren en las tres rutinas y se configuran los diferentes periféricos que utiliza la interfaz.

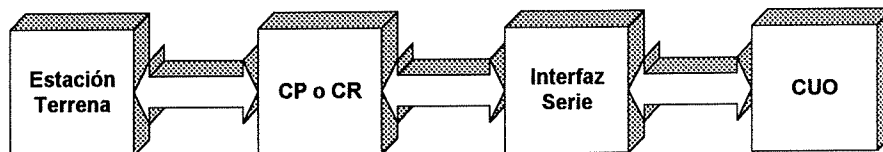


Figura 18. Interfaz entre CP y CUO.

Cuando la estación terrena envía un mensaje la CP lo recibe y decodifica. Si el mensaje no es para CP es retransmitido hacia la red interna del satélite por CP. Cada nodo dentro de la red detecta el mensaje pero nada más es atendido por su destinatario.

La interfaz se encarga de manejar principalmente el protocolo de comunicaciones internas pero debe de reconocer el protocolo de comunicaciones externo. También debe identificar otros formatos que utiliza exclusivamente CUO.

Cada mensaje lleva un formato dependiendo de que tipo es. Existen tres tipos de mensajes: mando interno, mando externo, y programa. Cada mensaje lleva un formato y longitud diferente. La interfaz puede distinguir si el mensaje es para CUO y que tipo de mensaje es.

Los mandos internos requieren un mínimo de datos para transmitir su mensaje. La interfaz reconoce cada una de las palabras de control pertinentes a CUO e instrucción a la computadora a realizar la acción peticionada. Las palabras de control se encuentran descritas en el Capítulo 2 *Protocolo de Comunicación en SATEX.1*.

Un mando externo requiere de más datos para transmitir su mensaje. Los mandos externos provienen de la estación terrena y mantienen su formato aún dentro de la red interna. Dentro de los mandos externos se pueden incluir instrucciones al microprocesador de CUO que no se encuentran definidas en el protocolo de comunicaciones internas. También es posible enviar y recibir información hasta de una longitud máxima de 25 octetos con formato de mando externo.

Con este formato se envía a CCUO los datos que requiere el reloj de tiempo real y se recibe del mismo los parámetros del láser.

Los programas requieren de un formato semejante al de los mandos externos pero con un mayor número de datos. Cada programa es dividido en bloques de 500 octetos antes de ser transmitido y cada bloque es enviado por separado con el formato para programas. El formato especifica en cada bloque cuantos bloques se transmitirán y el número de bloque que se transmite.

En el caso particular de CCUO, un nuevo programa debe ser localizado en cierta parte de la memoria para no interferir con la secuencia del programa actual. Como el protocolo de comunicaciones II-UNAM no está diseñado para acceder a localidades de memoria se utiliza el formato hexadecimal Intel MSC-86 con el que se puede especificar en que parte de la memoria queremos que sea puesto el nuevo programa. La decodificación de formato Intel se hace inmediatamente después de recibir un bloque de información, cada bloque lleva consigo una dirección para su localización en memoria de manera que el programa es colocado ordenadamente en la memoria antes de recibir el próximo bloque. Se puede lograr esto debido a la velocidad relativamente lenta con que se reciben los datos. Colocado el nuevo programa en memoria, se puede enviar una instrucción a la computadora de CUO para que lo ejecute.

#### ***4.2.1.1 Rutina de recepción.***

Para la recepción se utilizó uno de los puertos serie disponibles en el microcontrolador y al que se le denomina S1. La recepción se maneja por medio de interrupción. El microcontrolador tiene designado un vector de interrupción para el canal la recepción de S1 (2Eh). Cuando un octeto es recibido se activa la interrupción y entra una rutina que prepara el programa para recibir más datos (si es necesario) y fija dos banderas que indicarán que se tiene un mensaje y de que tipo es. Estas banderas son empleadas por el programa principal para decodificar el mensaje. Se utilizó el esquema de interrupción que permite por un lado continuar con otras actividades mientras se reciben datos y por el otro permite a la rutina de recepción interrumpir las actividades del procesador en caso de ser necesario.

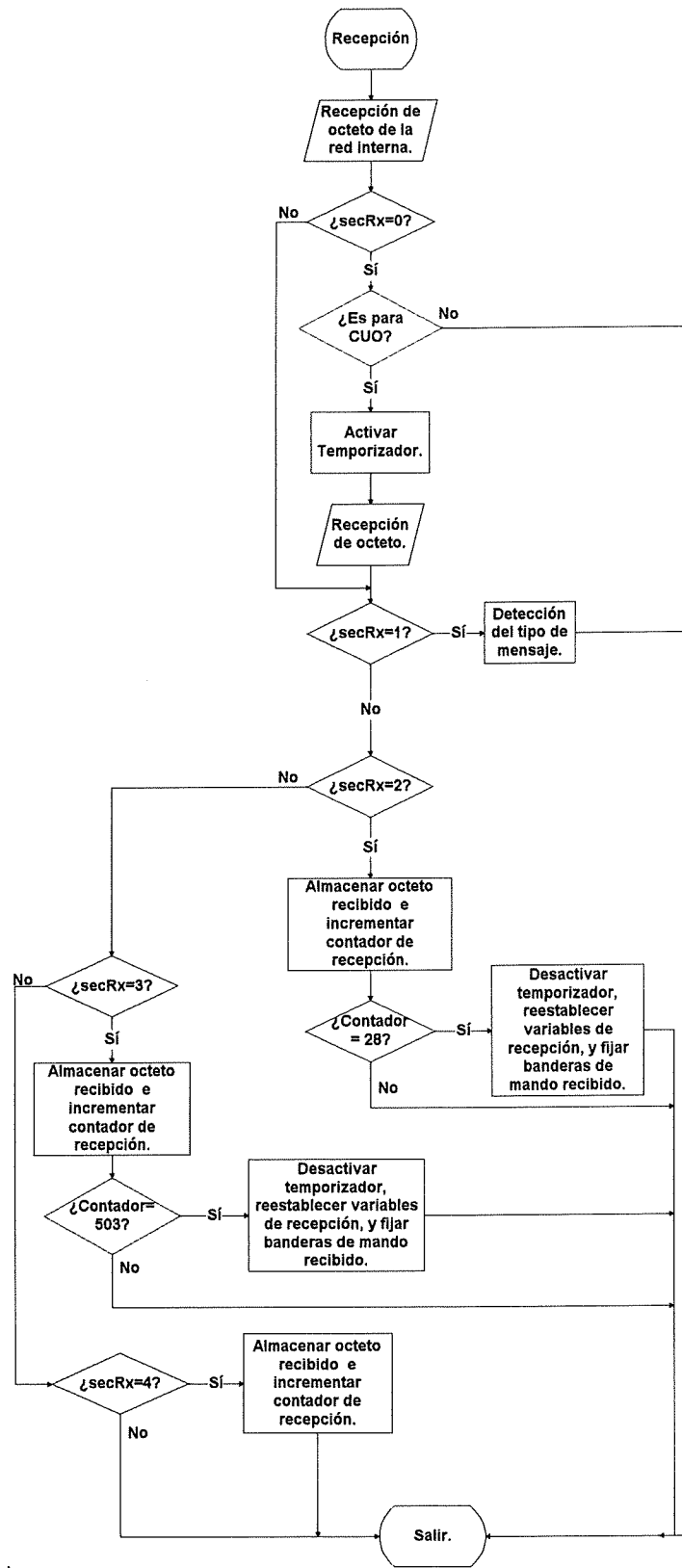


Figura 19. Diagrama a cuadros de la rutina de recepción.

En la Figura 19 se muestra el diagrama de flujo de la rutina de recepción. Nótese en este diagrama como la variable “secRx” determina el flujo del programa de recepción. En el primer caso se revisa si el mensaje es para CCUO, en caso de serlo se espera otro dato más de acuerdo al protocolo. La variable secRx se configura para que al llegar el próximo dato éste sea atendido por el siguiente caso que determinará que tipo de mensaje es. Si es mando interno entonces no se esperan más datos y la variable de recepción es reinicializada. Si es mando externo o programa entonces se esperan más datos y se configura secRx para que seleccione el caso de acuerdo con el tipo de mensaje.

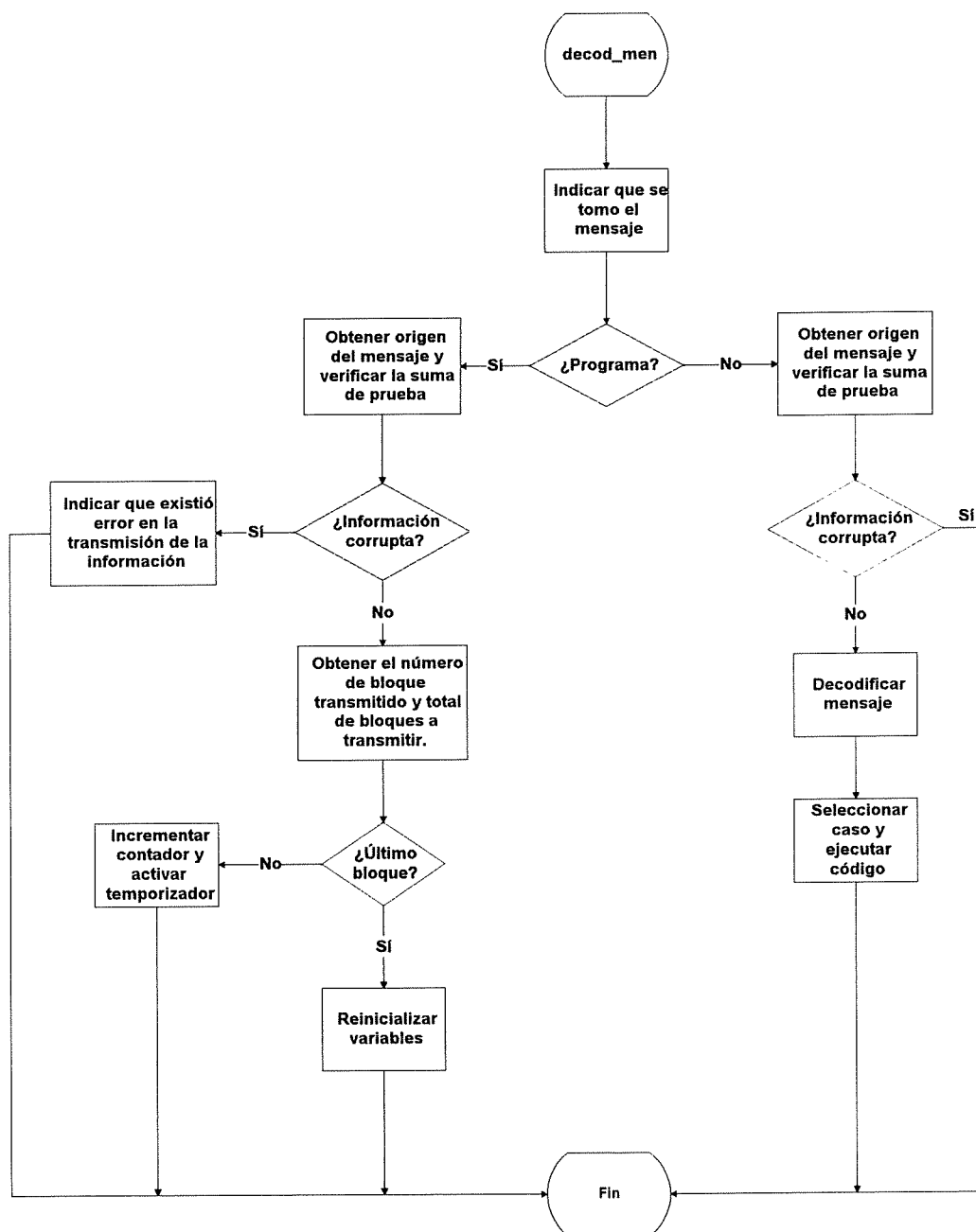


Figura 20. Subrutina de decodificación de mensajes.

Como se tiene tres tipos de mensajes la rutina incluye casos para poder atender a la longitud de cada uno de ellos. En esta parte de la rutina obtenemos que tipo de mensaje es y una suma de los octetos recibidos para verificación de su validez a la que llamaremos “suma de prueba”. El contenido del mensaje se almacena en un espacio de memoria temporal y se activa una bandera indicando que el mensaje está listo para ser interpretado. La velocidad de recepción es de 9600 bauds. El receptor utiliza el vector de interrupción 2Eh.

#### **4.2.1.2 Rutina de decodificación.**

Como resultado de la activación de la bandera de mensaje listo, el programa principal llama la subrutina de decodificación *decod\_men()*. En la Figura 20 tenemos un diagrama a cuadros de la subrutina de decodificación. Nótese como dependiendo del tipo de mensaje se direcciona el flujo del programa dentro de la rutina.

Dividido en dos partes, *decod\_men()* extrae información del mensaje recibido y responde a las instrucciones enviadas. Una parte atiende a los mandos internos y externos y la otra a un programa nuevo. Cada mando se reconoce como un caso en la subrutina y cada caso contiene el código a ejecutar de acuerdo al mando.

Lo primero que realiza la subrutina de decodificación es guardar la dirección del remitente. Lo anterior se hace por si es necesario responder al remitente por causa de un error o bien por solicitud explícita como en el caso de una “transmisión de parámetros”.

Como segundo paso se revisa la suma de prueba realizada en la rutina de recepción. Si la suma resulta nula se procede a decodificar el mensaje. En caso contrario se descarta el mensaje y se indica que la información llegó corrupta para que de esa manera se solicite un mensaje de retransmisión.

En esta rutina el temporizador T2 es utilizado como centinela y se activa al recibir el primer dato de un mensaje. Si después de 1 segundo no se ha recibido el siguiente dato se reinician las variables de la rutina y se activa una bandera que indica un error. La bandera de error a su vez activará la petición de la retransmisión el mensaje. El temporizador solo es activado cuando se espera más información.

Como los programas se reciben en bloques, *decod\_men()* se encarga tanto de llevar el conteo de los bloques recibidos y leer el encabezado del programa para conocer el número del bloque que se está recibiendo así como el total de bloques que se pretende recibir.

#### 4.2.1.3 Rutina de transmisión.

La transmisión de datos se efectúa solamente en los casos de detectarse error en la recepción, o por petición expresa de la ET como en los casos de “*diagnóstico*” y “*transferencia de parámetros*”.

Cualesquiera que sea el caso se procede a la activación de la bandera CCUOCON.3 y configuración de la variable “sectx”. La bandera CCUOCON.3 indica que se ha transmitido el mensaje (ver sección 4.1).

La variable ‘sectx’ sirve para identificar, dentro de la rutina de transmisión, el mensaje a enviar. El mensaje a transmitir comprende alguna de las siguientes opciones:

- Diagnóstico Correcto.
- Diagnóstico Incorrecto.
- Retransmisión de Mando.
- Retransmisión de Programa.
- Transmisión de Parámetros.
- Mando externo con parámetros del láser.

Todos son de un octeto de longitud excepto en el mando externo que consta de 28 octetos. Para este último caso los 28 octetos son leídos de un arreglo que se genera con anterioridad y es enviado tres veces. El arreglo contiene el formato de un mando externo por lo que solo es necesario agregarle el destino en el encabezado del mensaje. Con cada transferencia de parámetros se incluye una suma de prueba, incluida dentro de los 28 octetos, aplicada sobre los datos transmitidos. El mensaje completo será enviado tres veces y de acuerdo con el protocolo II-UNAM.

Todas las variables son reestablecidas cuando se termina de enviar el mensaje por completo.

La rutina de transmisión solo puede llamarse desde el programa principal a través de la activación de la bandera CCUOCON.1 y se auto-inhíbe después de enviar el mensaje.

La transmisión se apoya en el empleo de un temporizador (T2) para generar el tiempo de espera después del envío del primer octeto de control (ver Capítulo 2).

La velocidad para la transmisión de los datos es de 9600 bauds por segundo con 8 bits de datos sin paridad. El transmisor utiliza el vector de interrupción 2Dh para su rutina y es precedido en prioridad por el centinela, receptor, y CAPIN.

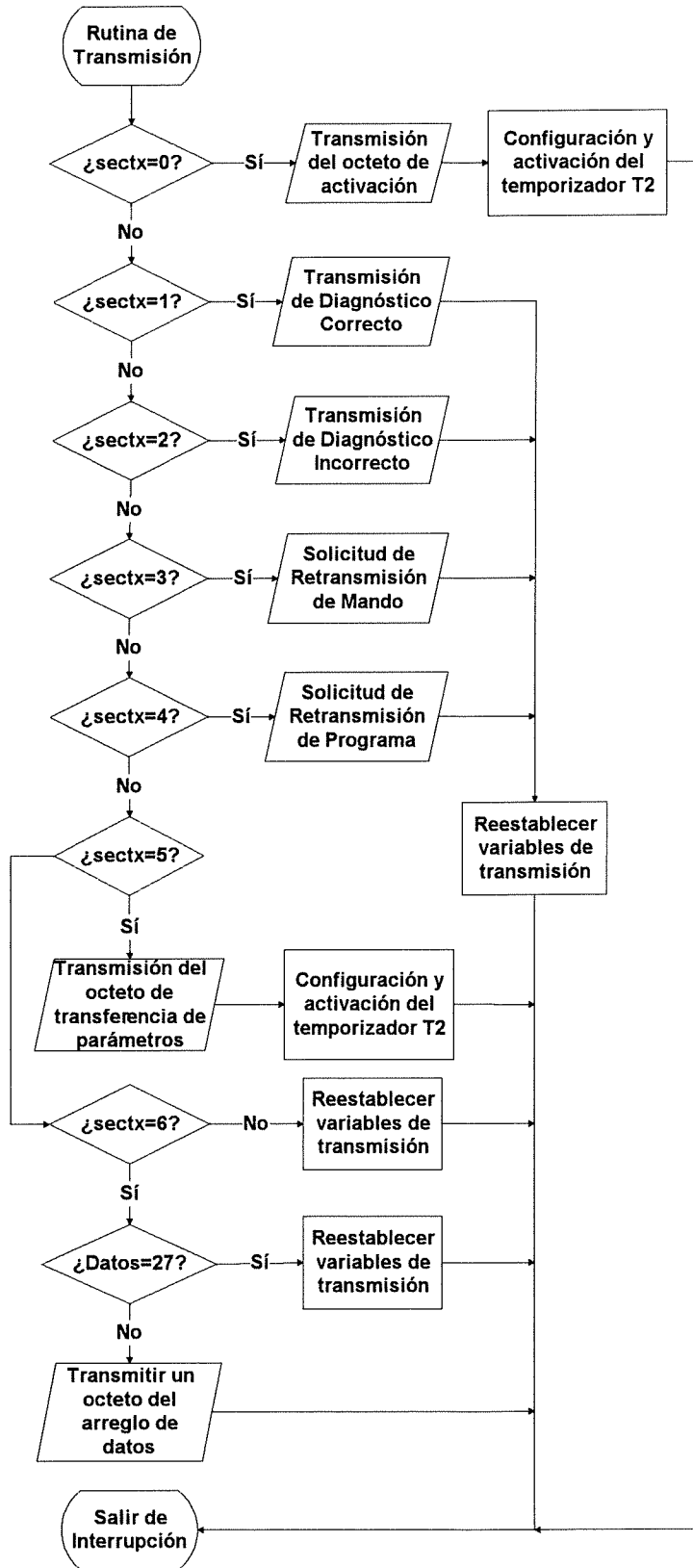


Figura 21. Diagrama a bloques de la rutina de transmisión.

#### **4.2.1.4 Rutina de temporización.**

Ésta es utilizada por las rutinas de transmisión y recepción. La rutina se apoya en el temporizador T2 del módulo GPT1. El temporizador T2 esta programado para que cuente con una resolución de 25.6  $\mu$ s de manera descendente y que de el tiempo pedido (30 ms, 1 seg) en cada bajoflujo. Es utilizado para implementar los tiempos de espera requeridos por el protocolo de comunicaciones de SATEX.1 y para asegurar que no se bloquee el canal de recepción..

El temporizador es activado en cuanto se recibe o se transmite una palabra de control. La desactivación se hace, para el caso de recepción de datos, 1 segundo después de recibido un dato o bien al finalizar la recepción de un paquete; y, para el caso de transmitir, después de 30 ms de haber enviado la palabra de control “*salir de bajo consumo de potencia*”. El tiempo de espera de 1 segundo en la recepción de datos es suficiente para obtener hasta más de 500 octetos a 9600 bauds, como en el caso de un bloque de un programa nuevo.

En caso de que falle la comunicación y no se reciba el siguiente dato o se trunque el mensaje (i.e. el mensaje no cumplió con el protocolo, se recibieron menos octetos de lo esperado) entra una rutina de servicio a interrupción del temporizador T2 para reestablecer el canal de recepción, y se solicita la retransmisión del último mensaje a través de la bandera CCUOCON.3.

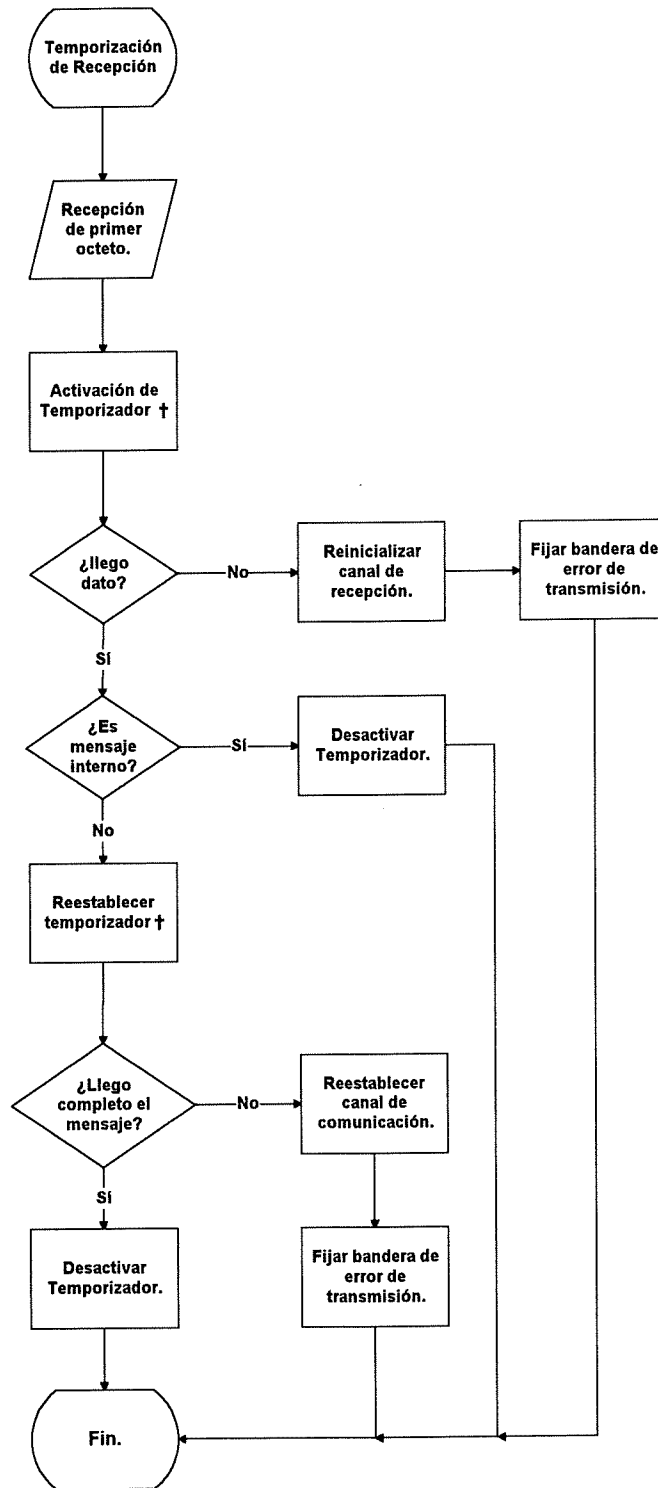
#### **4.2.2 Interfaz con motores.**

Los motores de pasos y sus manejadores requieren de señales de control tanto para el movimiento de los motores como para el suministro de energía (habilitación/deshabilitación) y control del par.

La interfaz con motores se encarga de la generación de estas señales proporcionando el control de giro y los pulsos que mueven los motores así como las señales EN28V, ENMX y ENMY para el suministro de energía y control del par. Cabe mencionar que el diseño de la interfaz satisface los lineamientos para el control de lazo abierto, es decir, la etapa de rastreo hasta que se establece el enlace.

Para lograr sus objetivos, la interfaz con los motores se vale de cuatro rutinas. Dos de las rutinas son para la temporización, una para el conteo de pasos y lectura de los tableros, y la cuarta para el posicionamiento inicial de los motores. Las rutinas son dependientes unas de las otras y en conjunto funcionan para dar movimiento a los espejos.

De manera general, la secuencia de las rutinas para realizar el rastreo es como sigue: después de que CUO recibe el mando de sincronización (el cual contiene información para la configuración de los relojes de tiempo real del sistema y de inicio de cobertura, así como las condiciones iniciales para el cálculo de la trayectoria) se colocan los espejos en su posición inicial (rutina *home()*), se generan los tableros de efemérides y CCUO queda a la expectativa de recibir una señal de sincronización proveniente de la ET la cual iniciará a los relojes de tiempo real. Cuando el reloj



† Cada vez que se activa el temporizador, o se reestablece, durará un minuto contando hasta que llegue a un bajoflujo. Si no se ha terminado de recibir el mensaje completo una rutina del temporizador se encargará de reestablecer el puerto serie.

Figura 22. Sistema de guarda para la recepción del canal de comunicaciones.

de inicio de cobertura lo indique una rutina llamada *comienza\_barrido()* desactiva este reloj de tiempo real, configura temporizadores y activa el rastreo. Durante el rastreo la rutina *motores()* se encarga de leer los tableros y llevar el conteo de los pasos de los motores. La relación entre las diferentes rutinas la podemos ver en la Figura 23. Esta interfaz emplea un mayor número de recursos del microcontrolador que el resto de las interfaces. Ahora describiremos con más detalle la función de cada rutina. Se verá la dependencia entre las rutinas y la estrecha relación entre la programática y la mecánica del microcontrolador.

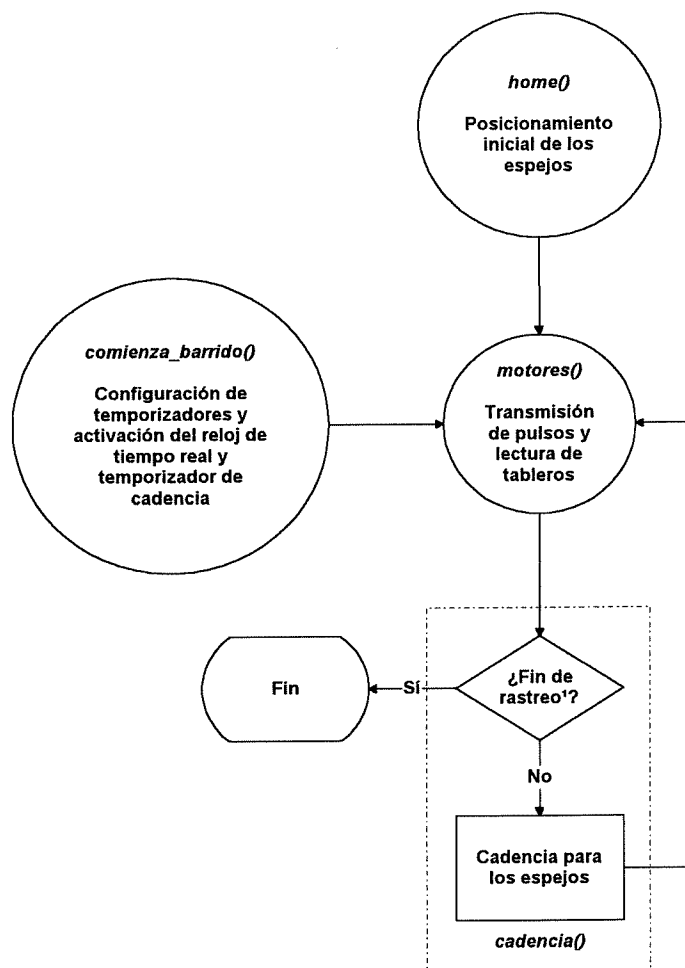
Con la llegada de la señal de sincronización se comienza el cálculo de las efemérides. Antes de los cálculos, los motores y manejadores son activados y los espejos se colocan en una posición conocida. La activación se hace por medio de las líneas de control EN28V, ENMX y ENMY. La primer línea es para el control de la fuente de energía para los motores y manejadores, las siguientes son para el control de la corriente (y torque) de los motores de los espejos X y Y respectivamente. El movimiento de los espejos se hace 20 ms después de encender (activar) los manejadores para cumplir con las especificaciones de funcionamiento del fabricante [22]. La colocación de los espejos en una posición conocida se realiza llevándolos a un tope mecánico colocado en una posición extrema y luego llevándolos a una posición central. Para el primer movimiento, en que los ejes de los motores se encuentran con el tope mecánico, es necesario reducir la corriente de operación de los motores al mínimo ( $ENMX=ENMY=0$ ) para disminuir su torque y evitar algún choque severo que pudiera dañar alguna parte o desajustar el mecanismo. Una vez en la posición conocida se habilita la corriente a los motores para tener el máximo par configurado. Lo anterior se realiza por medio de la rutina *home()* la cual es llamada secuencialmente (cuatro veces) desde el programa principal *main()*. En la primera llamada los espejos se llevarán hacia un extremo llegando al tope mecánico; también se activa la bandera CCUOCON.11 para indicar que falta el posicionamiento central de los espejos. El programa principal detecta esta bandera y llama a *home()* de nuevo<sup>8</sup> para colocar a los espejos en la posición central. En esta segunda vez se activa otra bandera (CCUOCON.6) para indicarle al programa principal que puede comenzar el cálculo de las efemérides. La tercera llamada se hace al terminar el cálculo de los tableros y sirve para colocar los espejos en su posición inicial (primer dato de los tableros). La cuarta y última llamada a *home()* se realiza al finalizar el posicionamiento inicial y después de cambiar los datos de los tableros de efemérides por movimientos relativos; en esta parte solo se obtienen los siguientes datos de los tableros en preparación para el inicio del rastreo. El primer dato leído de los tableros, en la tercera llamada de *home()*, contiene la posición inicial y absoluta de los espejos (con referencia al centro del campo visual) mientras que el segundo dato leído por la cuarta llamada ya es con posición relativa a la posición actual de los espejos. El cálculo de las posiciones relativas se hace entre la tercera llamada de la rutina *home()* y la cuarta. En el diagrama a cuadros de la Figura 24 muestra como funciona la rutina *home()*. Se debe mencionar que la rutina *home()* no es la encargada de mover los motores, solo configura las variables que utilizan las rutina que mueven los motores para que sean colocados en la posición deseada. Después de configurar las variables inicia el movimiento de los motres activando el

---

<sup>8</sup> La activación de la bandera CCUOCON.11 es solo una de las dos condiciones se tienen que cumplir para que el programa principal llame por segunda vez a *home()*. La segunda condición es que los espejos ya se encuentren en el extremo (tope). Esto último se logra detectando la desactivación del temporizador de cadencia T1. Como se verá más adelante el temporizador T1 regula los movimientos de los espejos.

temporizador T1 en las primeras tres llamadas; en la cuarta llamada a *home()* solo se configuran variables y la activación del movimiento queda a cargo del reloj de inicio de cobertura.

Una vez que se tiene a los espejos en su posición inicial, y se tiene a la mano los datos de los tableros, el sistema esta listo para comenzar el rastreo.



<sup>1</sup> La rutina *home()* configura las variables que utiliza *cadencia()* para determinar cuando se llega al final de los tableros. Al mismo tiempo se emplea otra variable para detectar si el movimiento es de posicionamiento inicial o rastreo. En el primer caso un fin de rastreo significaría fin de movimiento pero sin activar bandera CCUOCCON.7 que indica que no se logró establecer el enlace.

Figura 23. Relación entre las rutinas de la interface de motores.

Para indicar el comienzo del rastreo el Reloj de Tiempo Real de Inicio de Cobertura (RTRIC) genera una interrupción. Esta interrupción la atiende la rutina *comienza\_barrido()* la cual inicializa a los temporizadores encargados de la cadencia y señal de modulación para el módulo T1. Para lograr lo anterior, los temporizadores T3 y T4 que conformaban el RTRIC -el cual ha cumplido su cometido al indicar el inicio del rastreo- se reutilizan en modo de recarga [18] para generar el oscilador de 1 kHz. Por otra parte, la rutina *comienza\_barrido()* configura al

temporizador T1 en modo de recarga [18] para emplearlo como temporizador de la cadencia de los motores (los valores iniciales para esta configuración provienen de los cálculos de las efemérides). Finalmente los temporizadores T3 y T1 son habilitados. El diagrama a cuadros de esta rutina lo podemos ver en la Figura 26.

Con la activación del temporizador T1 se establece la cadencia de la lectura de los tableros. La cadencia es una especificación para mantener una velocidad angular constante. El tipo de barrido que se seleccionó para el rastreo de la ET es de forma circular con círculos que disminuyen en tamaño conforme el microsatélite se acerca a la ET. Cada círculo generado debe llevar la misma velocidad angular de manera de llevar un control sobre el área cubierta por el rastreo. El círculo exterior lleva la mayor velocidad por lo que se convierte en el patrón para el resto de los círculos. Esto implica que los círculos internos se trazan varias veces sobre la superficie terrestre. El apéndice B contiene gráficas para describir el barrido producido por la lectura de los tableros.

La rutina *cadencia()* atiende las interrupciones del temporizador T1 y lleva un control sobre la activación del temporizador T0 para la generación de los pulsos a los motores y el rastreo. Como ya sabemos, el temporizador T1 indica la cadencia de la posición de los espejos. Al iniciar el barrido, el primer movimiento comienza cuando lo indique el temporizador T1, los datos necesarios como la dirección de giro y número de pasos a dar (posición) ya fueron obtenidos por la rutina *home()* en preparación al rastreo.

En cada interrupción generada por el temporizador T1 se activa el temporizador T0 que da la base de tiempo para la generación de los pulsos como se verá más adelante. Mientras el tiempo de cadencia transcurre la rutina *motores()* lleva a los espejos a su lugar y termina desactivando al temporizador T0 y leyendo los siguientes datos de los tableros de efemérides para inicializar variables en preparación para el próximo movimiento.

El tiempo de cadencia permite realizar el movimiento y lectura de tableros, por lo tanto las rutinas *comienza\_barrido()*, *cadencia()* y *motores()* se consideran como rutinas de tiempo real. Cuando se llega al último dato de los tableros la rutina *cadencia()* desactiva al temporizador T1 y fija la bandera CCUOCON.7, terminando así el rastreo de la ET e indicando que no se logró establecer contacto con ésta.

En el eventual caso de establecer contacto con la ET entraría en función otra rutina para un control a lazo cerrado de los espejos. Podemos observar la rutina *cadencia()* en el diagrama a cuadros de la Figura 25.

Antes de continuar se explicará como se generan los pulsos para los motores de pasos. Para la generación de los pulsos se emplean dos comparadores (CC1 y CC2) asociados a un temporizador base (T0). El temporizador T0 lleva la base de tiempo con un período del doble del deseado para los pulsos y opera en modo de recarga automática empleando el registro TOREL [18]. La razón del doble del período es por el modo en que funcionan los registros comparadores. El nivel de comparación para los registros CC1 y CC2 se fijó a medio valor del período del temporizador T0, con el cual se comparan. Con esto, y configurando a los comparadores a operar

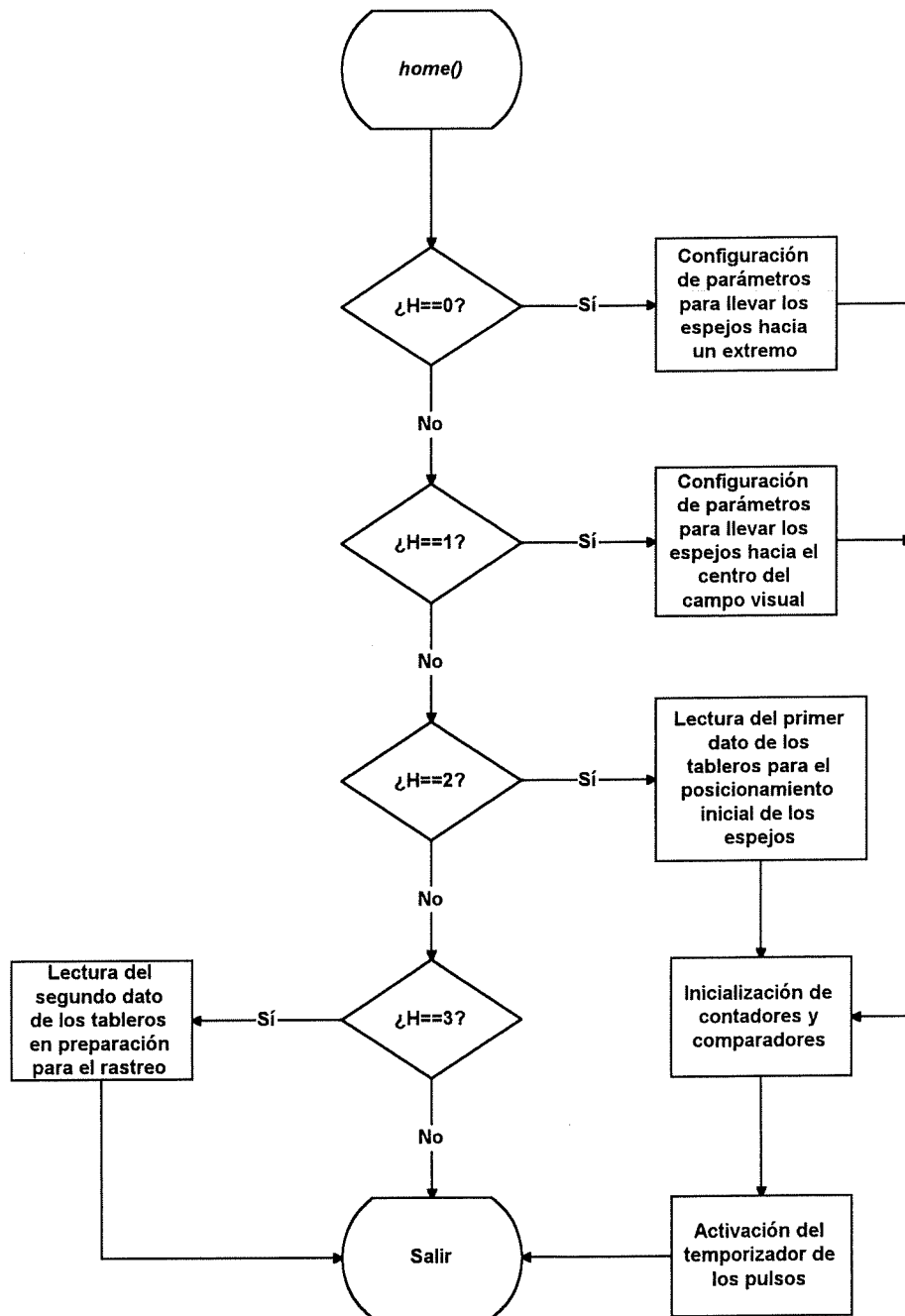


Figura 24. Rutina home() para el posicionamiento inicial de los espejos.

en modo 3 [18], se obtiene una señal con un ciclo de trabajo del 50% en las terminales de salida de CC1 y CC2 del microcontrolador (P2.1 y P2.2 respectivamente). Las señales producidas llegan a los manejadores de los motores de pasos que se encargan de transmitir el tren de pulsos a los motores.

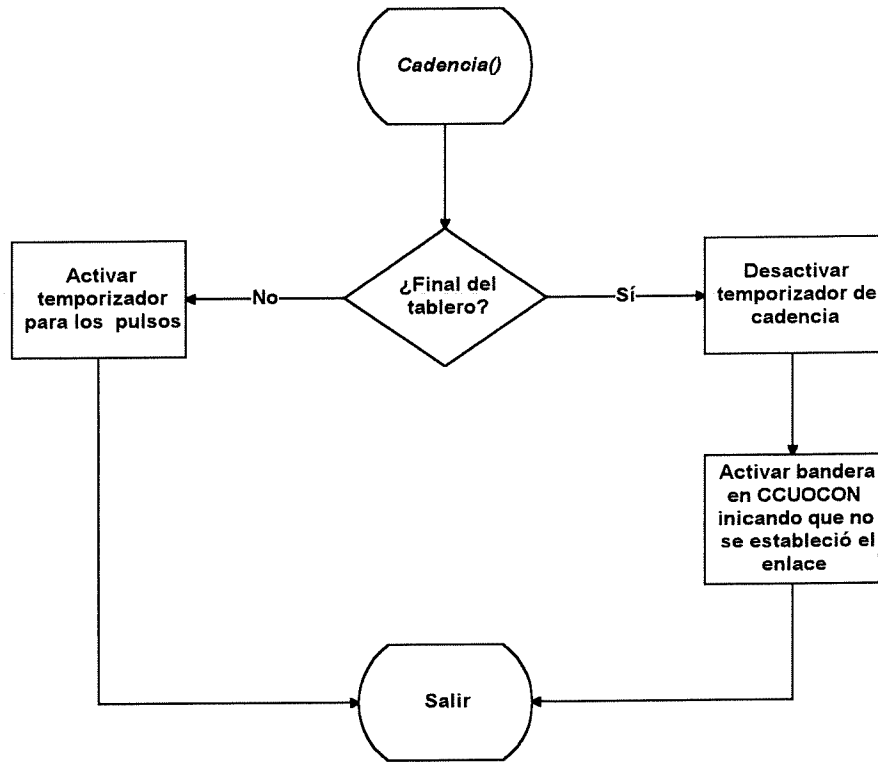


Figura 25. Rutina de control para la cadencia de los espejos.

Como se mencionó, mientras la rutina *cadencia()* lleva el tiempo de cadencia la rutina *motores()* se encarga del movimiento de los espejos y la lectura de los tableros de efemérides. Los tableros contienen la posición relativa de los espejos y su dirección de giro. Cada dato es contenido en un formato de carácter con signo donde el signo indica el sentido de giro. La rutina *motores()* atiende las interrupciones generadas por el temporizador T0 y lleva el conteo de estas<sup>9</sup>. Cuando un espejo llega a la posición indicada, se detiene su generador de pulsos<sup>10</sup>. Al momento en que los dos espejos están en la posición indicada la rutina desactiva el temporizador T0 y lee los próximos datos de los tableros para configurar los comparadores, la dirección de giro (que debe establecerse 100  $\mu$ s antes de comenzar el movimiento) e inicializa variables.

La designación de terminales para las señales de control se muestra en el Tablero XII.

<sup>9</sup> Recuerde que a cada interrupción de T0 corresponde el envío de un pulso a los motores por lo que al contabilizar las interrupciones se cuentan los pasos dados por los motores.

<sup>10</sup> El período del temporizador T0 se fija especificando un valor de recarga con el cual comienza a contar de manera ascendente hasta su sobreflujo. Para lograr que algún comparador deje de producir pulsos su valor de comparación se fija por debajo del que comienza el temporizador T0, ocasionando que nunca encuentre este valor y no produzca una salida.

Tablero XII. Designación de terminales para las señales de control.

Línea	Terminal
Pulsos para eje X	P2.1
Pulsos para eje Y	P2.2
Dirección eje X	P2.3
Dirección eje Y	P2.4
EN28V	P2.5
ENMX	P2.6
ENMY	P2.7

#### 4.2.3 Interfaz con láser.

Para conocer el funcionamiento del módulo T1 (láser) de CUO, esta interfaz realiza telemetría de sus parámetros. Estos parámetros son transmitidos a la estación terrena a petición de la misma. Mediante el mando “*transferencia de parámetros*” la interfaz recibe las señales por medio del convertidor analógico-digital en cinco de sus diez entradas analógicas del puerto 5, y ordena los datos para su transmisión. Otra función que cumple es la generación de una señal de 1 kHz para la modulación del láser en el módulo T1 durante y después del rastreo de la ET.

Dentro de los mandos internos se existe uno que solicita a CCUO el envío de los parámetros del láser. Cada parámetro es muestreado por el convertidor del microcontrolador y almacenado en un tablero para su posterior lectura por la rutina de transmisión.

Tablero XIII. Designación de entradas analógicas.

Terminal	Canal	Mnemónico	Función
P5.4	AN4	TLAS	Temperatura del láser
P5.5	AN5	ILAS	Corriente del láser
P5.6	AN6	PLAS	Potencia del láser
P5.7	AN7	IPEL	Corriente del Peltier
P5.8	AN8	VMOD	Voltaje de Modulación

Para el convertidor A/D se utiliza el modo de conversión 2 [18]. Cada canal es muestreado, convertido, y almacenado en memoria. Los resultados se ordenan en memoria para cumplir con el protocolo de comunicación, y se les ubica en un área específica. El convertidor utiliza el vector de interrupción 28h que atiende la rutina *convertidor()* para almacenar los resultados cuando termine cada conversión.

Después de transmitido los resultados permanecen en memoria y solo se actualizan hasta la recepción de un nuevo mando “*transferencia de parámetros*”.

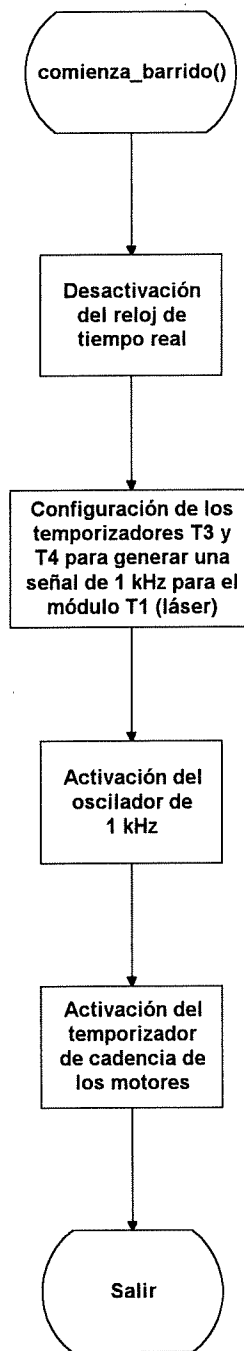


Figura 26. Diagrama a cuadros de la rutina `comienzo_barrido()`.

El láser es modulado con una señal de 1 kHz que proporciona el microcontrolador empleando el temporizador T3.

Esta interfaz comparte la rutina `convertidor()` con la interfaz del detector de cuatro cuadrantes.

#### 4.2.4 Interfaz con foto-detector de cuatro cuadrantes.

El APD es un foto-diodo de avalancha que responde a luz visible con longitudes de onda correspondientes al verde. Se utiliza para detectar la posición del faro de subida. El APD ofrece cuatro señales analógicas para detectar la posición del faro óptico de la estación terrena. La conversión de las señales del APD son coordinadas con el movimiento de los motores para el control de los mismos. Las señales son tomadas por el microcontrolador en cuatro de sus diez entradas analógicas para conversión analógica-digital. Después de procesar las señales, los resultados son localizados en lugares específicos de memoria para ser utilizados en el control de lazo cerrado de los motores o para la transmisión de parámetros hacia la estación terrena.

El convertidor A/D funciona en el modo 3 de conversión [18]. En este modo realiza una conversión de cada canal de manera continua y los resultados son almacenados en memoria con la rutina de interrupción *convertidor()* que se comparte con la interfaz del láser. La rutina *convertidor()* puede, o no, activar la transmisión de parámetros dependiendo si es solo una lectura de posición del faro de subida durante el control de lazo cerrado, o una petición de la estación terrena para la transmisión de parámetros.

Para la transmisión de parámetros la rutina almacena los datos en un arreglo de acuerdo al formato de un mensaje externo. Dentro de esta información se incluye el contenido de CCUOCON.

El APD es muestreado cada vez que se van a mover los espejos para ver si se detectó el faro de subida, y durante el enlace para la rutina de control de lazo cerrado. La rutina *convertidor()* solo espera una indicación para tomar mediciones del APD.

En caso de algún error de sobrecarrera en el convertidor, se tiene la rutina *err\_convertidor()* que ayuda a descartar los resultados y comenzar de nuevo el proceso.

La configuración del convertidor se hace durante la inicialización del sistema.

Tablero XIV. Designación de señales a entradas analógicas del microcontrolador.

Terminal	Canal	Mnemónico	Función
P5.0	AN0	APDA	Cuadrante A del APD
P5.1	AN1	APDB	Cuadrante B del APD
P5.2	AN2	APDC	Cuadrante C del APD
P5.3	AN3	APDD	Cuadrante D del APD

#### 4.2.5 Relojes de tiempo real

Todos los relojes de tiempo real se basan en interrupciones para la generación de los tiempos. En unos casos se emplea una combinación de mecánica con programática.

Un reloj de tiempo real utilizado para los motores de pasos (ver sección 4.2.2) se creó con el temporizador T6. Este reloj tiene un período de 0.01 segundos que se calculó como resolución suficiente para la base de tiempo de los motores.

El temporizador T1 se configura como contador de bajoflujos del temporizador T6 y sirve para generar la cadencia de los motores (ver sección 4.2.2).

El temporizador T0 funciona como base de tiempo para la generación de los pulsos para los motores de pasos (ver sección 4.2.2). Se configura para un período de 1 ms que sirve de base para los comparadores de CC1 y CC2 responsables de generar los pulsos.

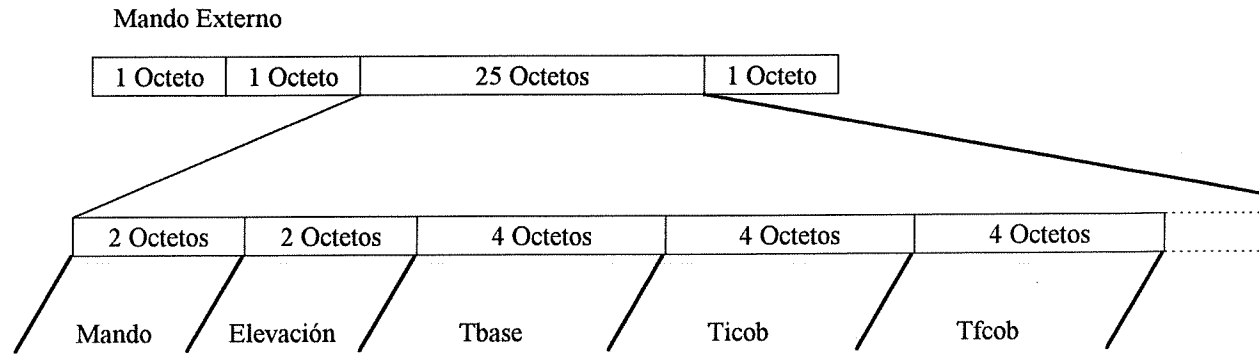
El temporizador T2 se emplea para cumplir con los tiempos de espera del protocolo de comunicaciones y como centinela del canal de recepción. El período de resolución es variado dinámicamente de acuerdo a las necesidades. Este temporizador funciona como un reloj de tiempo real dando los intervalos de 30 ms, 1 segundo y 5 minutos. Para contabilizar el espacio de 5 minutos se emplea una combinación de programática con mecánica. La rutina de interrupción que atiende al temporizador T2 dispone de un contador de segundos.

El temporizador T5 está destinado a dar servicio al centinela (watchdog) del microcontrolador. Se programa para tener un período de 400 ms. En cada período se genera una interrupción que atiende una rutina para dar servicio al centinela por medio de la instrucción `_srwtd()` y recargar el temporizador. Por el momento el centinela se encuentra deshabilitado de acuerdo al archivo de configuración `CSTART.ASM`, por lo que este temporizador queda disponible para cuando se desee implementar el centinela.

El reloj de tiempo real de inicio de cobertura (RTRIC) se utiliza para determinar cuando se debe de iniciar la búsqueda del faro de la ET. El reloj debe ser sincronizado con la hora de la estación terrena antes de comenzar el rastreo. Los datos de inicialización del reloj son enviados con tiempo ya que también se incluyen datos para realizar los cálculos que generan los tableros.

El RTRIC que se desea debe tener también una resolución mínima de 0.01 segundos y contar hasta horas. Los temporizadores del microcontrolador son de 16 bits con una mínima resolución de 51.2  $\mu$ s siendo solo posible obtener de un temporizador un período máximo de 3.36 segundos. Fue necesario, entonces, emplear dos temporizadores concatenados para obtener así un temporizador de 32 bits con una base de tiempo de 0.01 segundos y lograr un período superior a los 105 minutos. Este temporizador de 32 bits lo formaron los temporizadores T3 y T4.

Los datos de inicialización para el RTRIC son enviados como un mando externo (ver Capítulo 2). El programa principal se encarga de detectar que se trata de datos para el RTRIC y lo decodifica. Los datos son enviados dentro de los 28 octetos de información de un mando externo como se ve en la Figura 27 (datos tiempo real).



*Ticob* = Tiempo de inicio de cobertura.

*Tfcob* = Tiempo de fin de cobertura.

*Tbase* = Tiempo base.

*Figura 27. Formato de datos en el mando externo para el reloj de tiempo real.*

En la parte de *mando* de la Figura 27 se envía el carácter 'S' para indicar que son datos para el RTRIC. El parámetro de elevación se requiere para los cálculos del barrido del espejo. Como se ve solo se ocupan 16 de los 28 octetos, el resto de los octetos no contienen información válida.

Con los datos de inicialización se actualiza el RTRIC y se espera una señal de sincronización que activará el reloj. La señal de sincronización le llega al microcontrolador por la terminal P3.2 (CAPIN) que es una entrada para atender eventos externos. La llegada de la señal de sincronización activará una interrupción. Al ser atendida la interrupción por la rutina *ser\_CAPIN()* se verifica el estado de la señal de sincronización. La señal deberá durar en estado alto durante 1 ms. Si la señal cumple con esta condición se activará el RTRIC con el flanco descendente de la señal de sincronización y se levantará la bandera CCUOCON.8 indicando que se ha sincronizado el reloj con la ET.

La señal de sincronización sufre un retraso desde el momento en que es transmitida en tierra hasta el momento en que es recibida por SATEX.1. Considerando lo anterior se incluye una corrección en el tiempo de inicio para compensar el retraso. El tiempo de retraso se calculará por medio de la carga útil de comunicaciones por radio.

Iniciados los relojes de tiempo real el RTRIC indicará a CCUO después de un cierto intervalo el instante de comenzar el barrido. El RTRIC se programa para que realice una cuenta regresiva. Al momento de llegar a un bajoflujo se activa una interrupción que comienza el barrido. El temporizador T3 se reconfigurará en ese momento para dejar de ser el RTRIC y convertirse en el reloj de 1 kHz que requiere el láser.

### 4.3 Enlace entre interfaces.

Las interfaz forman el acoplo entre los diferentes elementos de CUO y el microcontrolador, así como con las unidades externas. Las interrupciones son parte importante de los interfaces ya que en su mayoría funcionan con rutinas de servicio a interrupción. Cada interrupción necesita un nivel de jerarquía dentro del esquema de las interrupciones. Toman prioridad las interrupciones designadas con mayor número en el intervalo de 0-Fh. Se tiene entonces 15 niveles de interrupción sin contar el nivel cero que es un número especial para deshabilitar las peticiones a interrupción. Se puede subdividir cada nivel de interrupción en cuatro subniveles más para dar un total de 60 niveles de interrupción. En el Tablero XV se tienen las rutinas de interrupción y su nivel de prioridad.

Tablero XV. Rutinas de interrupción utilizadas y su jerarquía.

Rutina o función	Número de interrupción	Registro de interrupción	Valor del registro de interrupción	ILVL	GLVL
Centinelas (Watchdog)	25h	T5IC	7Fh	15	3
Recepción <sup>11</sup>	2Eh	S1RIC	7Bh	14	3
Serv CAPIN	27h	CRIC	3Ah	14	2
Transmisión	2Dh	S1TIC	39h	14	1
err de Rx	2Fh	S1EIC	38h	14	0
Temporizador de RxTx	22h	T2IC	37h	13	3
comienza barrido	24h	T4IC	35h	13	1
Cadencia	21h	T1IC	34h	13	0
Generador de pulsos para el eje X	11h	CC1IC	33h	12	3
Generador de pulsos para el eje Y	12h	CC2IC	32h	12	2
motores	20h	TOIC	31h	12	1
Error de sobrecarrera en el convertidor	29h	ADEIC	30h	12	0
Convertidor A/D	28h	ADCIC	6Fh	11	3

Como se aprecia en el Tablero XV las rutinas de mayor prioridad son las de servicio a centinela y de recepción. El sistema del centinela es activado al momento de inicializar el microcontrolador y permite el reestablecimiento del microcontrolador en caso de que se pierda ejecutando código erróneo. La comunicación presenta el segundo nivel de importancia porque con ella podemos abortar un programa, iniciarlo, o cambiarlo. Las demás rutinas no requieren de un alto nivel de prioridad porque sus funciones no necesitan de tiempos de respuesta rápidos. En el caso de los

<sup>11</sup> Los vectores de interrupción utilizados para la comunicación son los del canal S1 del microcontrolador.

motores no se ocupa un alto nivel de prioridad tomando en cuenta ciertas consideraciones: durante el movimiento de los motores no se espera recibir algún mando que no sea el de “*Reset*”, “*Desactivar interrupción por puerto serie*” y “*Salir de bajo consumo de potencia*”.

## 5. PRUEBAS Y RESULTADOS SOBRE EL DESARROLLO.

A continuación se describe el conjunto de pruebas aplicadas al diseño e implementación de las interfaces para CUO. Se comprobará el funcionamiento adecuado de las interfaces en base a las especificaciones descritas en el Capítulo 3.

Aunque cada interfaz cumple con sus requerimientos específicos funcionales, todos deben cumplir con una serie de conceptos y/o funciones que los hacen modulares y compatibles entre sí. Estos conceptos se presentan como una lista de pruebas a especificaciones generales, mientras que los requerimientos específicos se presentan como una lista de pruebas a especificaciones particulares.

### *Lista de pruebas a especificaciones generales.*

*Tablero XVI. Lista de pruebas para el cumplimiento de las especificaciones generales de la programática.*

<b>Programática</b>
<ul style="list-style-type: none"> <li>a) Apegada a la normatividad ANSI C.</li> <li>b) De cumplimiento con el protocolo de comunicación para la red interna de SATEX.1 desarrollado por II-UNAM.</li> <li>c) En tiempo real.</li> <li>d) Tolerante a fallas.</li> <li>e) Versátil y fácil de mantener.</li> </ul>

Estas primeras especificaciones se comprueban durante las pruebas a especificaciones particulares. El apego a la normatividad ANSI C se vuelve evidente cuando los programas elaborados son compilados con el paquete de desarrollo CrossView sin ningún problema, ya que este paquete en sí trabaja con dicha norma. La tolerancia a fallas se implementa utilizando rutinas y mecánica para detectar errores y tomar medidas de corrección. La versatilidad y mantenimiento se logra con la modularidad de las interfaces. El desarrollo en tiempo real es de acuerdo a las necesidades de los eventos externos con los que interactúan las interfaces.

**Lista de pruebas a especificaciones particulares.***Tablero XVII. Lista de pruebas para el monitor mínimo.*

<b>Monitor mínimo</b>
<ul style="list-style-type: none"> <li>a) Configuración inicial del microcontrolador.</li> <li>b) Configuración de periféricos.</li> <li>c) Autodiagnóstico.</li> <li>d) Administración de la secuencia lógica del programa.</li> <li>e) Generación de tableros de efemérides.</li> <li>f) Recuperación del programa en casos de error por sobreflujo/bajoflujo de pila.</li> </ul>

*Tablero XVIII. Lista de pruebas para la interfaz serie.*

<b>Interfaz serie</b>
<ul style="list-style-type: none"> <li>a) Reconocimiento y creación de los diferentes tipos de mensajes.</li> <li>b) Cumplimiento de secuencias temporales de acuerdo con el protocolo II-UNAM.</li> <li>c) Recuperación del canal de comunicaciones en caso de errores tanto de formato como de duración de mensaje.</li> <li>d) Retransmisión al remitente del mensaje de acuerdo al protocolo II-UNAM.</li> <li>e) Reconocimiento y ejecución de instrucciones utilizadas en el protocolo II-UNAM.</li> <li>f) Reconocimiento de instrucciones exclusivas de CCUO.</li> <li>g) Telecargado y activación de un programa.</li> <li>h) Ejecución de un programa nuevo.</li> <li>i) Aborto de un programa.</li> <li>j) Decodificación del formato hexadecimal Intel MCS-86 para el telecargado de programas.</li> <li>k) Recepción y transmisión de datos manejados por interrupción.</li> <li>l) Utilización de un temporizador del microcontrolador para la implementación de los tiempos de guarda establecidos por el protocolo.</li> </ul>

*Tablero XIX. Lista de pruebas para la interfaz con los motores.*

<b>Interfaz con motores</b>
<ul style="list-style-type: none"> <li>a) Generación de las señales de control para los manejadores (dirección y pulsos).</li> <li>b) Cumplimiento de los tiempos requeridos por los manejadores y motores en las señales de control.</li> <li>c) Inicialización de motores antes del barrido (encendido, centrado y posicionamiento inicial).</li> <li>d) Lectura de tableros</li> </ul>

Tablero XX. Lista de pruebas para la interfaz con el láser.

<b>Interfaz con láser</b>
a) Funcionamiento de la rutina de conversión. b) Adquisición de parámetros del láser. c) Activación de la rutina de transmisión. d) Corrección en caso de error por sobrecarrera en la adquisición de las señales del convertidor. e) Generación de señal de 1 kHz para la modulación del láser.

Tablero XXI. Lista de pruebas para la interfaz con el APD.

<b>Interfaz con APD</b>
a) Adquisición de señales provenientes del APD a petición externa. b) Corrección en caso de error por sobrecarrera en la adquisición de las señales convertidor.

Tablero XXII. Lista de pruebas para los relojes de tiempo real.

<b>Reloj de Tiempo Real</b>
a) Configuración del reloj desde la estación terrena. b) Sincronización del reloj con la ET por medio de un evento externo a CUO.

Algunos de los puntos mencionados se subdividen en otros varios para lograr una prueba completa. Cada uno de estos subpuntos se verá al tratar la prueba. Todas las pruebas se realizaron en el laboratorio de Fibras Ópticas del Departamento de Electrónica y Telecomunicaciones del CICESE bajo la supervisión del M. C. H. Mejía.

## **5.1 Montaje.**

El montaje para realizar las pruebas de desempeño sobre las interfaces incluye los diferentes módulos que componen la plataforma de CUO así como computadoras e instrumentos de medición. También requiere de algunos componentes externos como la CP y el decodificador de tonos (DT). Sin embargo estos últimos no se encontraban disponibles razón por la cual tuvieron que ser emulados a través de dispositivos y programas sobre todo en lo referente a las funciones de comunicación, respetando el protocolo II-UNAM. Tampoco se tenían disponibles los módulos R2 y T1 por lo que también se emularon con una tarjeta llamada "Simvolt".

Antes de presentar las pruebas y resultados describiremos los componentes involucrados, así como el montaje de éstos en el laboratorio.

El sistema de pruebas comprende los siguientes dispositivos:

- Dos computadoras 486 compatibles con IBM equipadas con ratón y Windows 3.1 o mayor versión.
- Tarjeta de puertos denominada “Puertos”.
- Tarjeta de emulación de los módulos T1 y R2 y la señal de sincronización, denominada “Simvolt”.
- Tarjeta RMB-166 de Rigel.
- Carga Útil Óptica con los módulos R3, R3.1 y R4.
- Fuente de alta potencia, 28 Volts @ 2 Amps: Leader LPS-164A o semejante.
- Fuente de voltaje de salida múltiple, 5 V @ 1 Amp y  $\pm 12$  V @ 0.3 Amp: HP6236B o semejante.
- Paquete depurador CrossView de Boston Systems Office Tasking.
- Paquete generador de secuencia de pruebas “TESTCASE.C” desarrollado en CICESE.
- Par de cables serial DB9 macho-hembra en configuración DTE-DCE.
- Osciloscopio Digital Hewlett Packard 54504A.
- Analizador Lógico/Osciloscopio Hewlett Packard 1660AS.

El montaje de la plataforma de pruebas es como sigue:

1. En una de las computadoras, a la que llamaremos “Supervisor de la CCUO” o SCCUO, se instala el depurador Cross-View y en la otra, a la que nombraremos “Emulador de la Estación Terrena-Computadora Principal” o EETCP, el paquete “TESTCASE.C” y la tarjeta “Puertos” [15].
2. Se interconectan las tarjetas “Simvolt” y “Puertos”.
3. A la tarjeta RMB-166 se le configura en modo de telecargado de configuración y se le conectan los cables de la tarjeta “Simvolt”; de manera semejante se le conectan los cables seriales de las computadoras acuerdo al siguiente orden: De SCCUO al puerto S0 y de EETCP al puerto S1.
4. Se polarizan las tarjetas, en la secuencia RMB-166, “Simvolt” y finalmente el módulo 3.1, con los voltajes correspondientes.
5. Empleando SCCUO se telecarga el programa “CASE” en la CCUO y se corre el programa “TESTCASE.C” en EETCP.

El montaje para las pruebas lo podemos observar en la Figura 28, la computadora de la izquierda es SCCUO mientras la computadora de la derecha es EETCP. Cada computadora se comunica con CCUO por uno de los dos puertos serie del microcontrolador. La plataforma prototipo de CUO es el dispositivo que se encuentra entre las dos computadoras personales. La tarjeta que reemplazó a los módulos del láser y el foto-detector se puede ver en la Figura 29.



*Figura 28. Montaje de pruebas para las interfaces de CCUO.*

La tarjeta "Simvolt" se diseñó para estas pruebas y entrega las señales que vendrían de R4 y T1. Las señales de voltaje producidas son modificables, cuatro de emulación del APD las producen cuatro convertidores D/A leyendo la tarjeta "Puertos" de EETCP y utilizando un programa llamado PUERTOS.C. Cada convertidor es direccionable y lo que se le envía a uno queda almacenado en un cerrojo de entrada. Las otras señales se modifican desde "Simvolt" variando la resistencia de un potenciómetro que tiene cada salida.

Para la comunicación serie CCUO-CP se utilizó un programa llamado "TESTCASE.C". El programa transmite todos los mandos del protocolo de comunicaciones y los mandos exclusivos así como la señal de sincronización proveniente del DT.

Al final de cada prueba se escribirán los resultados y se determinará si el elemento en prueba funciona satisfactoriamente.

Todos los resultados deben estar de acuerdo a las especificaciones citadas en el Capítulo 3.

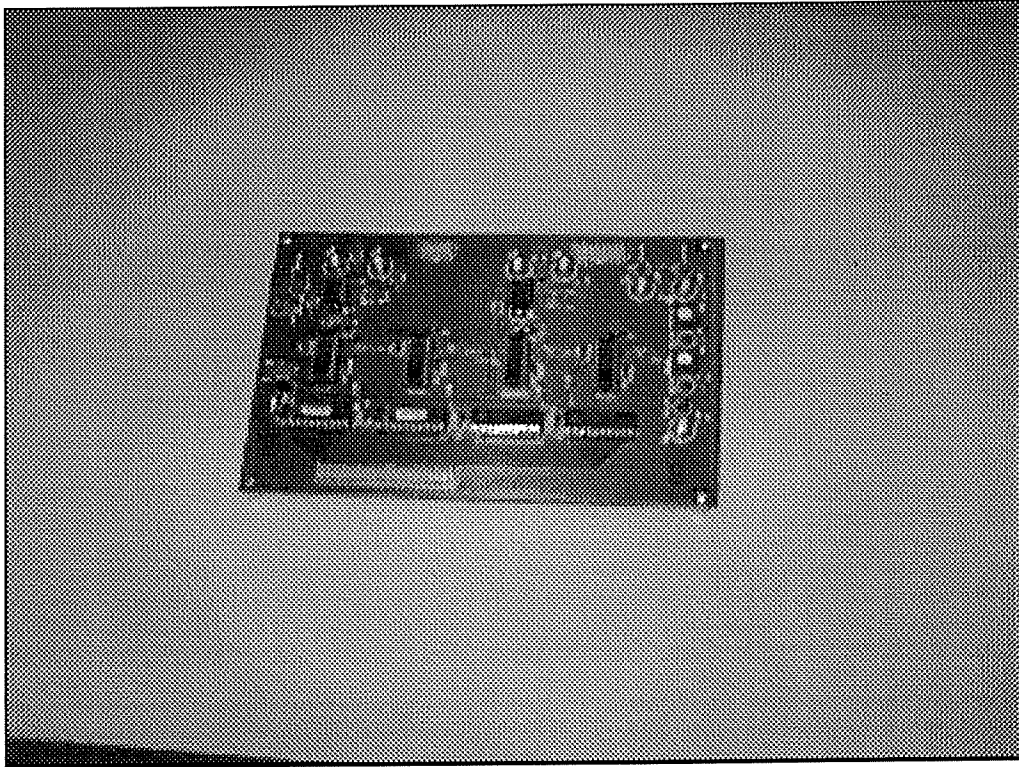


Figura 29. Tarjeta de emulación para CUO. Con esta tarjeta se obtienen las señales que provendrían del láser y APD.

## **5.2 Pruebas para el monitor mínimo.**

Dada la gran interactividad entre la circuitería y la programación, se propone que el monitor mínimo actúe como el administrador de la secuencia lógica del programa dentro de un esquema mezclado de los conceptos de tarea e interrupción plana [1]. Por lo anterior, el funcionamiento del monitor deberá basarse en el empleo de registros compartidos -banderas de estado- que deberán actualizarse de acuerdo a la presentación de los eventos -secuencialmente o por interrupción- y que serán atendidos por tareas mediante el empleo de procedimientos y/o rutinas de servicio a interrupción.

### **5.2.1 Configuración inicial del microcontrolador.**

Esta prueba se realiza cargando el programa principal CASES.C, una vez compilado, a CCUO por medio del paquete CrossView en SCCUO. Antes del cargado, el programa CrossView inicializa el procesador como se describe en el archivo CSTART.ASM. En este archivo se especifican características como el tiempo de espera y retardo para lectura de memoria, tipo de ducto, tamaño de pila, inicialización de punteros y registros de control.

Como segundo paso el programa principal se ejecuta y después de realizar ciertas acciones entrará en un ciclo infinito. Es entonces cuando, empleando el paquete CrossView, es posible verificar el

estado del procesador para verificar que haya quedado de acuerdo a lo descrito en el archivo de configuración.

### 5.2.2 Configuración de periféricos.

Después de la inicialización del microcontrolador, se carga el programa principal, CASES.C, el cual finaliza la configuración de periféricos del microcontrolador de acuerdo al procedimiento CONFIG.C.

La configuración de los periféricos es de acuerdo a los requerimientos del resto de las interfaces.

### 5.2.3 Autodiagnóstico.

El autodiagnóstico se ejecuta al iniciar el programa principal (monitor), por demanda externa y en cada reinicialización del microcontrolador. La rutina para el autodiagnóstico se llama *chkram()*. El autodiagnóstico consiste en una revisión de la MAA excluyendo la memoria interna del microcontrolador (1 kocteto). La memoria interna contiene los punteros, registros y la pila. La memoria es puesta a prueba almacenando un dato y luego volviéndolo a leer para verificar si es el mismo que se escribió. El diagnóstico se realiza con cuatro tipos diferentes de datos (FFFFh, 0000h, 5555h y AAAAh), pero antes de escribir algo en memoria el contenido de esa localidad a probar se resguarda en un registro de la memoria interna para ser reestablecido después de la prueba.

También se realiza una suma de prueba antes de comenzar las lecturas y escrituras a memoria, y una vez más al terminar. Los dos resultados de la suma de prueba se comparan y deberán ser idénticos.

Si se detecta algún error en la memoria, la función de autodiagnóstico regresa un valor de 1. El resultado de la rutina *chkram()* sirve para establecer el valor de una bandera en el registro CCUOCON indicando el estado de la memoria y para activar la transmisión del mensaje adecuado al remitente.

### 5.2.4 Administración de la secuencia lógica del programa

El programa principal *main()* se encuentra en un ciclo infinito revisando constantemente el estado de las banderas de CCUOCON. Recordemos que en este registro se indica al procesador la tarea que deseamos realizar (ver Capítulo 4, sección 4.1).

El llamado de las rutinas se hace por interrupción para aquellas en donde el tiempo de respuesta es crítico (tiempo real), y por programática desde *main()*, para aquellas en donde no se requiere un tiempo de respuesta rápido. Con interrupciones tenemos la flexibilidad que ofrece el control por mecánica en combinación con la programática para la secuencia de eventos.

### 5.2.5 Generación de tableros de efemérides.

El cálculo de las efemérides comienza cuando se recibe la señal de sincronización. La rutina que se encarga de realizar los cálculos se llama *trayectoria()*.

Los resultados de los cálculos de las efemérides se almacenan en dos tableros: uno para el eje X (azx[]) y otro para el eje Y (azy []). Inicialmente, la posición absoluta de los espejos se calcula, pero una vez que se tienen los resultados, se efectúan otros cálculos para obtener la diferencia entre cada posición y poder realizar movimientos relativos de los espejos.

Por medio de SCCUO se puede observar el contenido de los tableros y verificar su creación. La prueba final sobre la validez de los tableros es el movimiento de los espejos con dichos datos.

### 5.2.6 Recuperación del programa en casos de error por sobreflujo/bajoflujo de pila.

Para el caso de sobreflujo/bajoflujo de pila, se almacenó directamente en la tabla de vectores, en el espacio correspondiente a estas trampas, la instrucción de reinicialización. Con esto se asegura la recuperación del control del programa.

En errores de punto flotante, existe una función del paquete de desarrollo de CrossView que almacena el estado del procesador (pila, registros, banderas) antes de realizar las operaciones de punto flotante [1]. En caso de error, por tales operaciones, la función reestablece el estado del procesador a como se encontraba antes de comenzar las operaciones de punto flotante y regresa un valor para indicar el error.

## 5.3 Pruebas sobre la interfaz serie

La comunicación interna de SATEX.1 se realiza a través de un ducto diseñado de acuerdo a las necesidades del proyecto. Este ducto se emplea para la comunicación interna del microsatélite entre computadoras. CCUO cuenta con dos puertos de comunicación serie de los cuales utiliza solo uno en el microsatélite y dos durante las pruebas. La comunicación se realizará entre CCUO y la EETCP.

El programa de emulación "TESTCASE" puede desplegar los caracteres recibidos y los enviados. Con esto se verifica la recepción de los caracteres que envía CCUO.

CCUO almacena los datos recibidos por el puerto serie en un área designada de memoria para luego ser interpretados. Utilizando el programa CrossView en SCCUO se puede ver el contenido de la memoria de CCUO y verificar la recepción de caracteres del programa de emulación.

Esta prueba confirma que las rutinas de transmisión y recepción de CCUO funcionan para la transmisión y recepción de datos.

Las rutinas de recepción y transmisión de datos trabajan a base de interrupciones para dar mayor flexibilidad al programa. Aunque estas rutinas ocupan la más alta prioridad dentro de las

interrupciones, no interfieren de manera crítica en algunas de las otras funciones que el procesador realice (ésto depende del tipo de instrucción que se reciba, pero en general la recepción y transmisión no interfieren notablemente debido a que se apoyan en la mecánica).

### **5.3.1 Reconocimiento y creación de los diferentes tipos de mensajes.**

Para detectar que CCUO transmita de acuerdo al protocolo, el programa de emulación "TESTCASE" despliega los datos que envía CCUO, verificando la transmisión correcta de acuerdo al protocolo. Como se tienen tres tipos de formatos para los mensajes, de los cuales CCUO debe transmitir los tres, las pruebas se realizan probando secuencialmente la interpretación de CCUO a todos los mandos del protocolo concernientes a CUO.

La rutina de comunicaciones atiende cada tipo de mensaje por separado. Como verificación de que distingue los tres tipos se utiliza el paquete CrossView. Cada mensaje enviado se almacena en un área de memoria designada para éste propósito. Con CrossView se puede observar dicha área de memoria designada para los mensajes, y ver que se reciban en su totalidad.

### **5.3.2 Cumplimiento de secuencias temporales de acuerdo con el protocolo II-UNAM.**

Según lo visto en la sección 4.1.1.4, las condicionantes de tiempo dentro del protocolo II-UNAM se pueden dividir en dos partes: las relacionadas con la transmisión de mensajes y la preventiva para evitar problemas en el ducto de SATEX.1.

La primer parte establece los parámetros de comunicación serie y formato temporal de los mensajes.

En cuanto a parámetros, la velocidad de comunicación se estableció a 9600 bps con un formato sin paridad de 8 bits de datos y uno de paro. Estas especificaciones se programaron tanto en EETCP como en CCUO. El hecho de que las dos computadoras lograron comunicarse es prueba de que los parámetros quedan establecidos y los programas funcionan.

Dentro del formato de los mensajes se especifica que debe de existir una separación temporal, entre el primer octeto de un mensaje transmitido al ducto y el resto del mensaje, de 30 ms. El resto del mensaje se transmite de manera continua. Por medio de un analizador lógico/osciloscopio se revisó que se cumpliera el espacio de tiempo en la transmisión de los datos.

Respecto a la segunda parte o condicionante preventiva se pide que el puerto de comunicaciones de CCUO sea desactivado por un espacio de 5 minutos cada vez que lo solicite CP (ver Capítulo 2). La desactivación del canal de comunicación se verifica enviándole a CCUO el mando de "*desactivar canal de comunicación*" y luego intentando establecer comunicación con esta. CCUO no debe responder por un espacio de 5 minutos, lo cual hizo en repetidas veces que se intentaba y con completa recuperación de la comunicación después de los 5 minutos.

Con esta parte se verifica el funcionamiento de la interfaz en tiempo real al cumplir con las especificaciones temporales del protocolo.

### 5.3.3 Recuperación del canal de comunicaciones en caso de errores en la transmisión de mensajes y su formato.

El programa puede identificar varios tipos de errores en la comunicación. Los errores pueden detectarse por programática o mecánica. La programática se encarga de detectar errores en el formato del protocolo y valida los datos; la mecánica detecta errores en la transmisión de los datos tales como: sobrecarrera, de cuadro e información corrupta. Cuando CCUO, por medio de esta interfaz, detecte un error descartará el mensaje y pedirá su retransmisión al remitente.

La detección de errores por mecánica detecta errores de sobrecarrera, de cuadro (formato de trama) y valida la autenticidad de los datos. Cada uno de estos errores genera una interrupción que es atendida por una sola rutina. La rutina de interrupción de error en la comunicación no tiene otra función más que la de activar una bandera para indicar el error y preparar al interfaz para que al finalizar la recepción del mensaje se pida su retransmisión.

Este tipo de detección de errores se probó aumentando la velocidad de transmisión de la EETCP y esperando un mensaje de retransmisión de parte de CCUO. La prueba solo verifica el funcionamiento de la rutina de atención a error por sobrecarrera; pero, debido que los otros dos tipos de errores utilizan el mismo vector de interrupción, y por lo tanto la misma rutina, se puede confiar en que pasando esta prueba, que es una prueba de la rutina y no de las funciones del microcontrolador, se garantiza la detección y atención de los otros tipos de errores. El mensaje de error fue recibido en EETCP por lo que se pasó la prueba.

Cuando se envía un mensaje su formato queda especificado por el protocolo II-UNAM (ver Capítulo 2), en el eventual caso de que no cumpliera este formato el programa de comunicaciones quedaría bloqueado en espera de más información. Para esto se estableció un temporizador que reestablece el puerto de comunicaciones en caso de error por formato. El temporizador es activado cuando se recibe un dato por el canal de comunicaciones, y desactivado al terminar el mensaje. Si no se recibe el mensaje completo el temporizador no se desactiva y al pasar 1 segundo se genera una interrupción que llama a una rutina que reestablece la rutina de recepción y activa el envío de un mensaje de error. El espacio de tiempo de 1 segundo es suficiente para recibir hasta el más exigente de los mensajes (e.g. envío de programa).

Como prueba, se le envió a CCUO cada tipo de mensaje, pero no en su totalidad. Se espera que CCUO responda con un mensaje de error y la completa restauración del puerto de comunicaciones. El mensaje de retransmisión se recibió en la EETCP y logró comunicarse normalmente con CCUO después del evento.

Una última prueba sobre la validez de la información es por medio de una suma de prueba. La suma de prueba solo se realiza en los mensajes de tipo externo y programa, para los internos solo los dos métodos mencionados anteriormente son utilizados como verificación de la validez de la información. Al momento de transmitir un mensaje se incluye, dentro de este, una suma de prueba de los datos enviados. Al momento de recibirlos se realiza una segunda suma de prueba y el resultado se compara con la que se hizo al enviar el mensaje. Si los resultados son diferentes el mensaje se descarta y se pide la retransmisión del mismo. A manera de prueba se envió una suma

diferente a la que debería corresponder dentro de los mensajes tipo externo y programa, y CCUO respondió detectando el error y solicitando la retransmisión del mensaje.

En todos los casos de error CCUO debe conocer el remitente del mensaje para poder responder, pero, recordando el formato de los mensajes (*ver* Capítulo 2) veremos que únicamente los mandos internos llevan consigo información del remitente. El problema se resuelve al observar que cada tipo de mensaje debe ser precedido por un octeto de activación, que es un mando interno. Con este primer octeto se puede obtener el remitente y se almacena en una variable global para utilizarlo al momento de transmitir una respuesta en el caso que sea necesario.

Se hizo la prueba cambiando la dirección del remitente para ver si CCUO distinguía los cambios. En todos los casos CCUO respondió dirigiendo correctamente el mensaje al remitente

### **5.3.4 Reconocimiento y ejecución de instrucciones utilizadas en el protocolo II-UNAM.**

Las siguientes instrucciones pertenecen al juego correspondientes al protocolo II-UNAM y solo se incluyen las que en un momento dado conciernen a CUO. Para esta prueba se utilizan las dos computadoras SCCUO y EETCP, con los programas de depuración y emulación, respectivamente. La forma de verificar el funcionamiento depende de la instrucción, pero para todas, la respuesta a la instrucción se seguía paso a paso con ayuda del depurador CrossView en SCCUO. De las más fáciles de verificar son las que requieren de una transmisión del resultado por demanda. El resultado es visto en EETCP para verificar el funcionamiento correcto.

Con las instrucciones que no se espera una respuesta transmitida, se verifica, como se había mencionado, paso a paso y observando la información que maneja en sus registros y en la memoria con el depurador CrossView en SCCUO.

La lista de instrucciones para CUO del protocolo II-UNAM es:

1. Diagnóstico\*.
2. Transferencia de parámetros\*.
3. Desactivación de puerto serie (5 min.).
4. Retransmisión\*.
5. Sincronización de temporizadores.
6. Transferencia de nuevo programa.
7. Transferencia de mandos.
8. Bajo consumo de potencia.
9. Salir de bajo consumo de potencia (Atención).
10. Reset.
11. Corre programa.

Las instrucciones marcadas con un asterisco (\*) ocasionan que el procesador transmita el resultado por demanda al remitente. El resto no requieren de una respuesta.

### 5.3.5 Telecargado y activación de un nuevo programa y reconocimiento de instrucciones exclusivas de CCUO.

Como complemento al juego de instrucciones del protocolo II-UNAM, se agregan dos instrucciones más para CCUO. La transmisión de estas instrucciones se apega por completo al protocolo de comunicaciones incluyéndolas dentro del formato de *mando externo* (ver Capítulo 2). El programa monitor, por medio de la interfaz de comunicación serie, detecta el tipo de mensaje y decodifica su contenido extrayendo la instrucción del cuerpo del mensaje.

Las dos instrucciones adicionales son:

1. 'G' Cambio de tablero de vectores y activación del nuevo programa.
2. 'S' Configuración del reloj de tiempo real para su sincronización.

La instrucción denominada 'G' se emplea para activar el nuevo programa transmitido, cambiando en última instancia la tabla de vectores. La instrucción 'S' se usa para enviar los datos al reloj de tiempo real y prepararlo para la sincronización con la ET.

Para la instrucción 'G' la verificación de su funcionamiento se realiza comprobando que efectivamente se inicio el nuevo programa con la nueva tabla de vectores de interrupción. Al nuevo programa se le incluyen funciones no incluidas en la original para probar que efectivamente se inicio. La prueba más común fue la programación de algún temporizador para generar una señal oscilatoria de salida, así como el reconocimiento del resto de las instrucciones.

Con la instrucción 'S' se verifica, paso a paso, que los datos de configuración para el reloj de tiempo real lleguen y sean extraídos correctamente. Con la configuración de los temporizadores para el reloj de tiempo real, basta utilizar un cronómetro para verificar que el reloj de tiempo real responde a la señal de sincronización activando los motores después del tiempo programado (ver sección 4.1.5 Relojes de tiempo real).

### 5.3.6 Telecargado y activación de un programa.

CCUO tiene la capacidad de recibir un nuevo programa para actualizar sus funciones y los parámetros que utiliza. El programa enviado se divide en bloques de 500 octetos. Para solventar la carencia del protocolo para el direccionamiento de datos en áreas específicas de memoria, estos son codificados en formato hexadecimal Intel MCS-86. Cada bloque recibido es decodificado y puesto en el área de memoria especificada (ver Interfaz Serial, Capítulo 4).

El nuevo programa contiene el código actualizado, menos la tabla de vectores nueva. Los vectores de interrupción se cargan hasta el último momento, antes de comenzar el programa nuevo. Con la instrucción 'G' se cambia la tabla de vectores justo antes de reestablecer el microcontrolador por programática.

Las pruebas se realizaron enviando un programa con la configuración para un temporizador.

### 5.3.7 Aborto de un programa.

Para abortar un programa se envía un mando de reinicialización a CCUO. Después de este mando, CCUO debe reestablecerse por completo.

La instrucción fue transmitida repetidamente y CCUO se recuperó completamente en cada ocasión mostrando que respondía a todas las instrucciones.

### 5.3.8 Decodificación del formato hexadecimal Intel MCS-86 para el telecargado de programas.

El mensaje tipo programa viene codificado con el formato hexadecimal Intel MCS-86. Este formato nos sirve para indicar específicamente donde queremos que se almacene el programa en la memoria de manera que no interfiera con el programa existente. La decodificación y almacenamiento se hace a la vez y después de recibir un bloque completo (500 octetos).

La rutina *decod\_men()* se encarga de realizar la decodificación y almacenamiento llamando a la rutina *intelahex()*.

Con el programa CrossView se verifica el almacenamiento y decodificación correcta del mensaje localizando el programa en el espacio de memoria especificado.

## 5.4 Pruebas sobre la interfaz con motores

La interfaz para motores sirve como base para el control a lazo abierto de éstos a través de los manejadores. Con dicha interfaz podemos variar el par (a dos niveles), la velocidad, el sentido de giro de los motores y el encendido y apagado de los motores y manejadores. La interfaz cumple con la especificación general de tiempo real respondiendo a la cadencia requerida durante el rastreo de los motores.

### 5.4.1 Generación de las señales de control cumpliendo con los tiempos requeridos por los manejadores y motores.

Las señales de control deben de cumplir las especificaciones del fabricante del equipo y el apuntamiento espacial, principalmente en cuestiones temporales.

Con respecto a las características eléctricas, los manejadores son directamente compatibles con el nivel TTL. El acoplamiento se realiza con la tarjeta R3.1 y unos optoacopladores integrados del manejador.

El nivel lógico de las señales que indican el sentido de giro de los ejes de los motores debe establecerse 100  $\mu$ s antes de transmitir los pulsos que harán que el rotor gire. La relación entre el sentido de giro de los ejes y el nivel lógico de la señal de dirección depende las conexiones realizadas en los devanados de los motores [22].

Como es conocido, al manejar motores de pasos, a un pulso le corresponde un paso, es decir, un desplazamiento angular conocido de la flecha. Para el tipo de manejadores utilizados en este trabajo, un paso es generado durante la transición negativa del pulso. Las especificaciones del fabricante indican que dicha transición debe ocurrir en 200 ns o menos. El microcontrolador y reforzador cumplen con esta especificación. En cuanto a velocidad se refiere, el ancho de pulso en cualesquiera de los dos estados lógicos de la señal de los pulsos, debe ser no menos de 1  $\mu$ s resultando en una frecuencia pico de 500 kHz. La frecuencia máxima de los pulsos quedará determinada por la masa inercial de los espejos y sus adaptadores [15]. En la Figura 30 se puede observar una gráfica de las señales de dirección y pulsos capturadas por un osciloscopio. Se puede ver en la señal de dirección, o sentido de giro, como cambia de acuerdo a las especificaciones, antes de que se transmitan los pulsos. También se puede observar que la señal de pulsos está por debajo del máximo de frecuencia para que respondan los motores; específicamente el valor para esta señal se encuentra condicionada a las estimaciones hechas de acuerdo a los más recientes

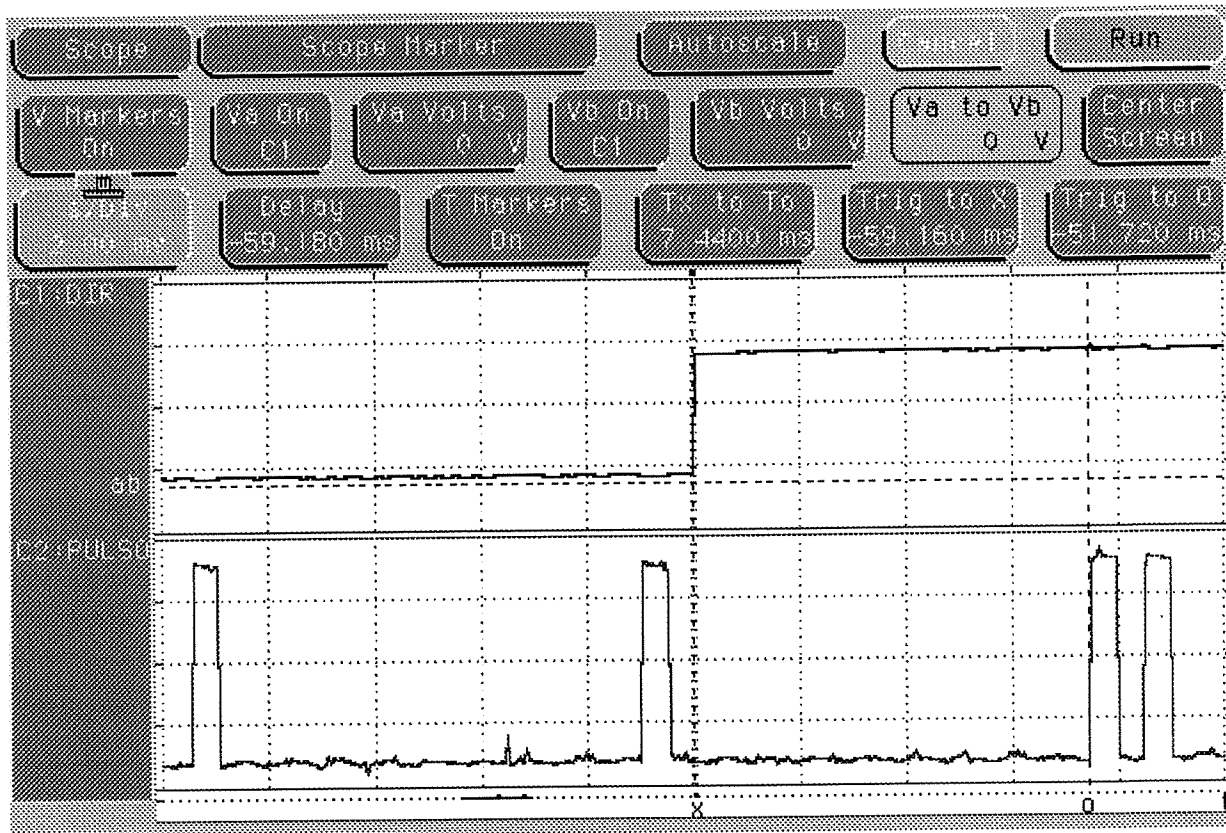


Figura 30. Señales de control de pulsos y dirección para los motores.

parámetros orbitales. Realizando repetidos cálculos de efemérides, con diferentes valores, se determinó que la máxima frecuencia para la señal de los pulsos nunca sería alcanzada. Esto es debido a que la colocación de los espejos se realiza por medio de movimientos relativos, los

cuales se comprobó<sup>12</sup> que no pasarían de un determinado número de pasos entre cada posición, pudiendo completarse el movimiento en un tiempo mucho menor al requerido por la función de rastreo del sistema de apuntamiento. Por lo anterior, se observó que no son necesarias altas tasas de velocidad beneficiando al control de los espejos con más torque.

#### **5.4.2 Inicialización de motores (encendido, centrado y posicionamiento inicial).**

Esta rutina activa los manejadores de los motores, a los motores mismos y coloca los espejos en una posición conocida. A continuación se describen estas etapas.

Como sabemos los espejos van sujetos a las flechas o ejes de los motores. Antes de comenzar el rastreo para la localización de la estación terrena, los espejos deben ser colocados en una posición conocida, Por lo que se trasladan a una posición inicial seleccionada como el centro del campo visual.

La colocación se realiza a cabo en dos etapas. Primero se trasladan los espejos hasta el extremo de su campo visual donde se encuentran con un tope mecánico. De aquí ya se sabe la posición de los espejos y en la segunda etapa se le envía el número correcto de pulsos para colocarlos en la posición central. Durante la primera etapa, la corriente de los devanados se limita de manera de reducir el par generado de los motores evitando un fuerte impacto con los topes mecánicos

La comprobación de la colocación de los espejos se realizó con un diodo láser. Cada vez que se mandaba inicializar los espejos (centrado y posición central) el haz del láser reflejado por uno de los espejos coincidía con una referencia de prueba. El ajuste mecánico de los espejos para los topes y centrado se realizará cuando el prototipo contenga todas sus partes funcionales.

Previo al centrado, los motores y manejadores se encuentran desactivados. Es tan solo poco antes de comenzar los movimientos que la interfaz activa la fuente para los dos. Según las especificaciones del fabricante [22], la activación de la potencia a los motores debe ocurrir de 10 a 20 ms antes de tratar de moverlos.

A manera de comprobación de que la interfaz con los motores funciona de manera adecuada para su utilización por futuros algoritmos de control del sistema de apuntamiento espacial tanto en lazo cerrado como lazo abierto se realizó una simulación de rastreo para el peor caso<sup>13</sup> (el análisis queda fuera del propósito de esta tesis pero los resultados de éste sí deben poder ser manipulados por la interfaz). Con los espejos en su posición central, y los tableros de efemérides generados, los espejos son llevados a su posición inicial. Esta posición se encuentra como primer dato en los tableros de efemérides y se obtiene antes de cambiar los valores absolutos de los tableros por valores relativos. Este primer movimiento del rastreo es el único movimiento que probablemente requiera el mayor número de pasos, razón por la cual se realiza antes de comenzar el rastreo a manera de preparación. El número de pasos enviados por el programa fueron contabilizados con

---

<sup>12</sup> El análisis sobre los aspectos críticos del control de apuntamiento quedan fuera del contexto de este trabajo de tesis.

<sup>13</sup> Basta mencionar que el control en lazo abierto impone mayores compromisos de velocidad y par a los motores.

el analizador de estados lógicos; por medio de este se pudo comprobar que la interfaz de los motores enviaba el número preciso de pasos que se le indicaban.

### 5.4.3 Lectura de tableros.

Después de generar los tableros de efemérides y colocar los espejos en una posición inicial, se calculan los movimientos relativos de los espejos. Estos valores almacenados en tableros separados para cada eje, no contienen valores superiores a 5.

La rutina se elaboró de manera que pudiera leer estos tableros y extraer el sentido de giro y el número de pulsos en ese sentido. A como se menciona en el Capítulo 4 en la sección 4.1.2 Interfaz con Motores, los datos de giro y número de pasos vienen en un formato de caracteres con signo, donde el signo indica el sentido de giro para los motores.

Era necesario probar que la rutina leía los tableros en su totalidad y que el sentido de giro y número de pasos era obtenido y enviado correctamente. Para comenzar, la frecuencia de los pulsos transmitidos a los motores se verificó con un osciloscopio digital (*ver* Figura 30); siendo la correcta (1 kHz). La fijación de la señal de sentido de giro para los motores fue observada corriendo paso a paso esa parte del programa y verificando los resultados en las terminales de los manejadores de los motores. Con un dato negativo leído del alguno de los tableros se fijaba un '1' lógico en la terminal correspondiente, ocasionando un giro en sentido antihorario. Si el dato era positivo se enviaba un '0' lógico para indicar un movimiento en pro de las manecillas del reloj. El analizador de estados lógicos sirvió para comprobar que una vez que leía un dato de un tablero la interfaz, enviaba los pulsos correspondientes con el giro apropiado. El sentido de dirección se tiene que fijar cuando menos 100  $\mu$ s antes de enviar los pulsos de los pasos de acuerdo a las especificaciones del fabricante. En la Figura 30 podemos observar como cambia la señal de dirección antes de transmitir los pulsos. Este tiempo observado es mucho mayor que el requerido por el fabricante, pero recordemos que el movimiento lleva una cierta cadencia y es precisamente este tiempo el que se observa. La cadencia se puede medir de una secuencia de pulsos a otra (separados por más de 7 ms). En este tiempo se aprovecha para configurar las variables con los siguientes datos de los tableros.

La prueba verificó que los datos eran leídos correctamente y en su totalidad.

Para comprobar la validez de los datos en los tableros y el funcionamiento adecuado de la interfaz, se empleó un diodo láser colocado en la forma que llevará originalmente CUO, y se observó en una superficie plana el trazado del láser debido al movimiento de los espejos durante la lectura de los tableros<sup>14</sup>. En la Figura 31 se puede apreciar el trazado del láser en la superficie, lográndose producir la figura deseada para el rastreo. El apéndice B describe a mayor detalle el barrido del faro óptico sobre la superficie de la tierra.

---

<sup>14</sup> Los datos generados por las ecuaciones para los tableros han sido previamente probados en un programa de simulación. Se conocen los resultados que genera y que debe reproducir CCUO.

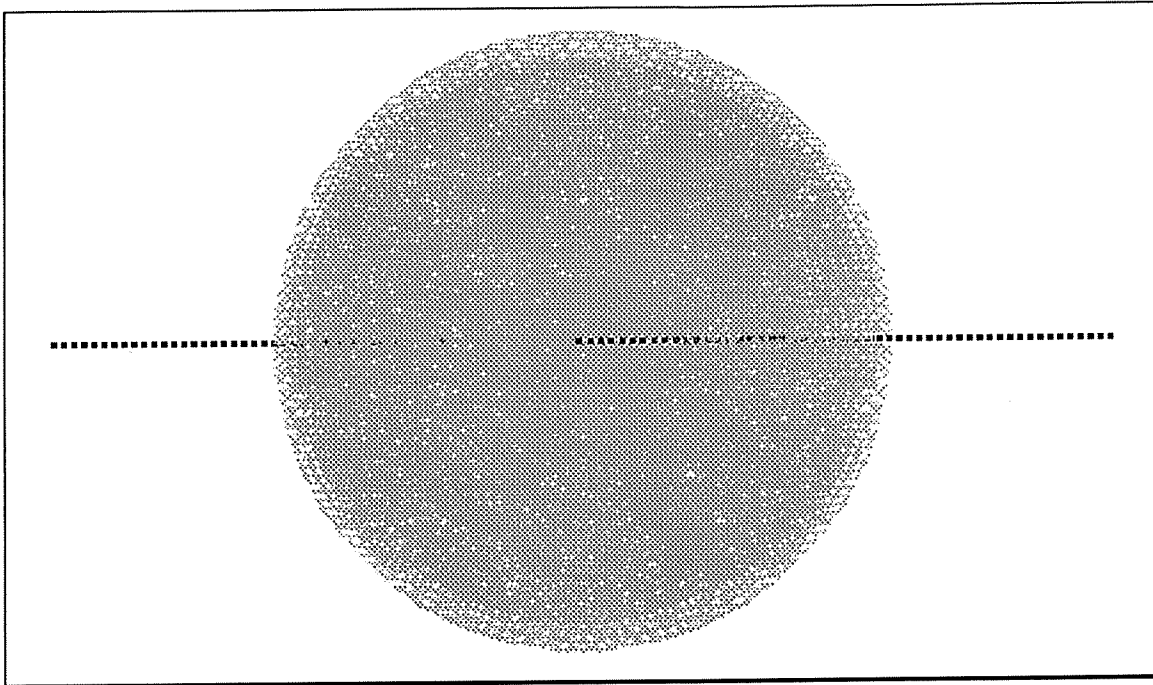


Figura 31. Trazado de láser con el movimiento de los espejos utilizando los datos de los tableros de efemérides.

## 5.5 Pruebas a la interfaz del láser

La interfaz para el láser convierte las señales analógicas de los parámetros del láser y envía una señal de 1 kHz para la modulación del mismo. Los parámetros del láser serán emulados por la tarjeta "Simvolt". Se ocuparán para la prueba las computadoras SCCUO y EETCP, con los paquetes de depuración y emulación, respectivamente. También se requiere del osciloscopio digital para la verificación de la generación de la señal de 1 kHz para el módulo T1.

### 5.5.1 Funcionamiento de la rutina de conversión.

La rutina de conversión *convertidor()* fue probada repetidas veces para verificar la precisión y exactitud de la conversión de los parámetros<sup>15</sup>. Se esperaba cuando menos una resolución mínima de 6 bits significativos que requiere la rutina de control de lazo cerrado. Se pudo comprobar que se cumplía con la resolución y que esto se podía mejorar reduciendo las fuentes de ruido para lo cual las medidas de corrección serían implementadas al ensamblar el prototipo en su totalidad (con los módulos T1 y R2).

### 5.5.2 Adquisición de parámetros del láser.

La adquisición se inicia con la recepción del mando interno "Transferencia de parámetros" (ver Capítulo 2) que hace la petición del envío de los parámetros del láser. El convertidor muestrea y

<sup>15</sup> Esto al mismo tiempo sirvió para la caracterización del convertidor A/D.

convierte cada señal para los canales especificados<sup>16</sup> y almacena los datos en una localidad de memoria designada. Se debe mencionar que la conversión no es solo de los parámetros del módulo T1, si no también del APD. Lo anterior es debido al tipo de conversión que se seleccionó (modo 2 [18]) y a que las señales del APD ocupan los primeros canales (del 0 al 3) de las entradas analógicas del convertidor mientras que las del láser las cinco subsecuentes (*ver* Capítulo 4, interfaces láser y APD).

Antes de enviar los parámetros, los resultados son almacenados en un área de memoria específica y acomodados de acuerdo al formato de un mando externo (*ver* Capítulo 2). Con SCCUO se verifica los resultados de las conversiones en la memoria deteniendo el programa antes de ser transmitidos. Con lo anterior revisamos la conversión de los parámetros y el formato con que se almacenan.

### **5.5.3 Activación de la rutina de transmisión en el caso de petición de transmisión de parámetros.**

Una vez que el convertidor obtiene los parámetros y los almacena, la rutina *convertidor()* debe hacer un llamado a la rutina de transmisión para que lea los datos del arreglo y los transmita al remitente. Con la EETCP verificábamos la transmisión de los datos observándolos en la pantalla al recibirlos y comparándolos con los del arreglo en la memoria de CCUO por medio de SCCUO. Los datos llegaban de acuerdo al formato del protocolo II-UNAM para un mando externo y con el orden descrito en el archivo CASES.C (*ver* Capítulo 4, Interfaz láser).

### **5.5.4 Corrección en caso de error por sobrecarrera en la adquisición de las señales del convertidor.**

El error por sobrecarrera ocurre cuando un dato del convertidor no es leído antes de que se produzca el próximo resultado. Si esto llega a ocurrir los datos que se obtengan estarán recorridos en el arreglo que se almacenan, lo que produciría errores en la rutina de control del sistema de apuntamiento espacial.

La rutina de corrección de sobrecarrera *err\_convertidor()* se probó evitando la lectura de un canal durante la conversión por medio de programática. Al inducir el error, la rutina *err\_convertidor()* se activa indicando por medio de una bandera el error, evitando con esto la transmisión o lectura de los datos hasta volver a tomar lecturas coherentes.

Por medio de SCCUO vimos que la rutina *err\_convertidor()* efectivamente era llamada por la interrupción que genera el error, y que los datos no eran transmitidos. También se verificó que los parámetros eran de nuevo muestrados y convertidos.

---

<sup>16</sup> El número de canales a convertir define si es una petición de los parámetros del láser (que requiere su transmisión) o, el muestreo del APD para la detección y localización del faro de subida (*ver* Capítulo 4, Interfaz láser).

### **5.5.5 Generación de señal de 1 kHz para la modulación del láser.**

El oscilador de 1 kHz se activa al momento de que inicia el rastreo de la estación terrena. Los temporizadores utilizados son T3 y T4 en un arreglo de contador y recarga (ver Capítulo 4, Relojes de tiempo real, sección 4.2.5). La salida de esta señal oscilatoria es por la terminal P3.3 del microcontrolador [18]. Es aquí donde se verifica la existencia de esta señal por medio del osciloscopio. En la Figura 32 se puede observar la señal de 1 kHz para el módulo T1.

## **5.6 Pruebas a la interfaz del APD**

La interfaz con el APD comparte la rutina *convertidor()* con la interfaz del láser. La diferencia entre los dos es la manera en que manejan los datos una vez que se han convertido y guardado. Aquí se emplea la computadora SCCUO durante las pruebas.

### **5.6.1 Adquisición de señales provenientes del APD.**

Especificando 4 canales a convertir, la rutina *convertidor()* entiende que se trata de la lectura de los 4 primeros canales, correspondientes a las señales del APD (ver Capítulo 4, Interfaz APD). En este caso no se activa la transmisión de los parámetros como el de la interfaz láser.

La prueba se hacía verificando que CCUO lee las señales del APD y las almacena en los espacios de memoria designados usando el paquete CrossView en SCCUO.

### **5.6.2 Corrección en caso de error por sobrecarrera en la adquisición de las señales del convertidor.**

Ver prueba 5.5.4.

## **5.7 Pruebas al reloj de tiempo real**

El reloj de tiempo real sirve para indicar el comienzo del rastreo de la estación terrena. Este reloj debe ser configurable debido a la variación en los parámetros orbitales del microsatélite. En todo caso, el reloj se debe sincronizar con la estación terrena para asegurar la inicialización correcta del rastreo.

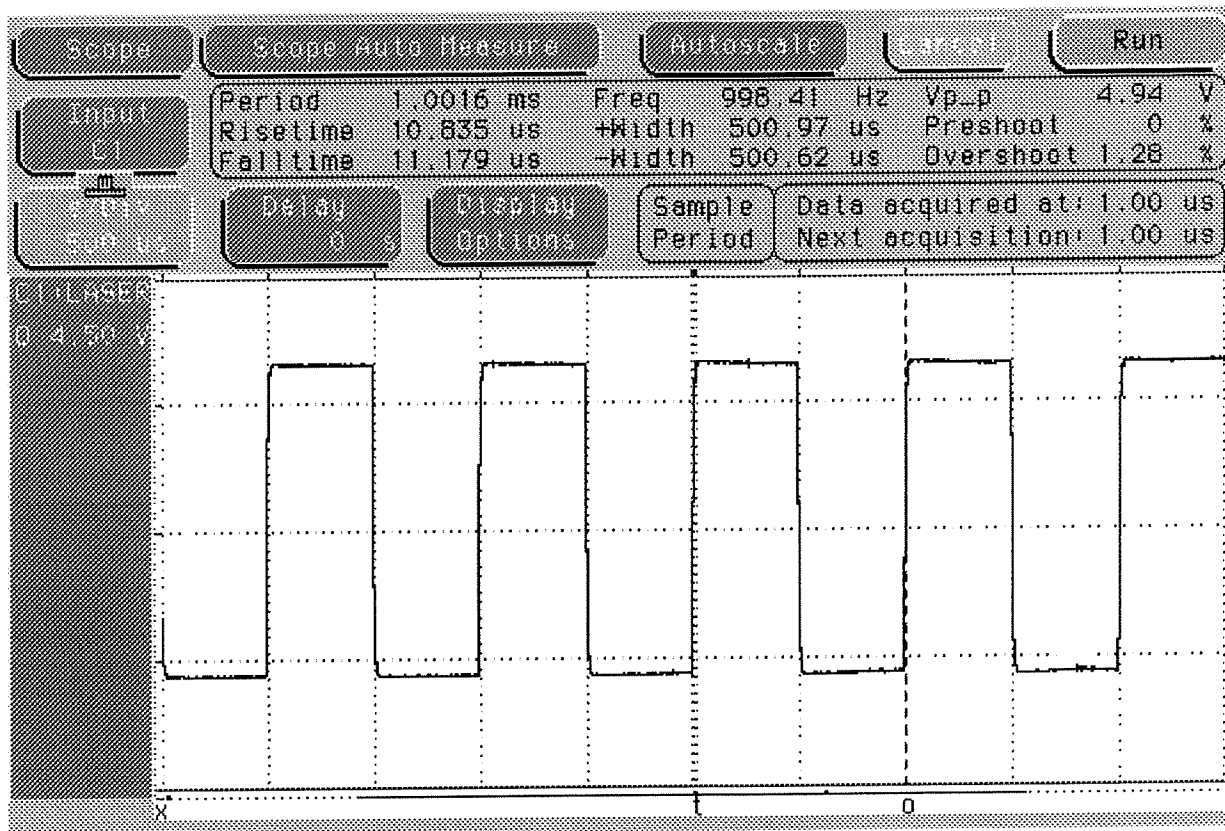


Figura 32. Señal de 1 kHz para la modulación del láser.

### 5.7.1 Simulación de configuración del reloj desde la estación terrena.

El reloj de tiempo real es configurado después de la inicialización de CCUO. Ésto en cuanto al tipo de conteo (ascendente/descendente), período y jerarquía de interrupción (ver Capítulo 4, Reloj de tiempo real, sección 4.2.5). La terminal del microcontrolador denominada CAPIN (P3.2) se utiliza para recibir la señal de sincronización del DT. La interrupción para esta terminal es deshabilitada inicialmente hasta recibir un mando (específicamente “*Sincronizar temporizadores con algún evento*”, ver Capítulo 2) con los datos iniciales del reloj. Una vez recibidos los datos y configurado el reloj de tiempo real la interrupción para CAPIN es habilitada.

Antes de continuar con el programa se verifica la correcta configuración de los temporizadores por medio de SSCUO. Todo estuvo de acuerdo a lo especificado.

### 5.7.2 Simulación de sincronización del reloj con la ET por medio de un evento externo a CUO.

CAPIN esta configurado para atender una transición positiva en la señal del DT. Se pide como medida de seguridad que la señal dure en estado alto cuando menos 1 ms para validarla. En caso de ser válida, CAPIN se configura para atender la transición negativa de la señal.

El programa de emulación en EETCP envía la señal para CAPIN a través de la tarjeta "Simvolt" donde es reforzada antes de llegar a CCUO.

El espacio de tiempo entre la transición positiva y la negativa debe ser suficiente para que CCUO calcule los datos de la tabla de efemérides, esto en el caso de que la señal de sincronización (en su primera etapa, transición positiva) haya sido enviada cuando menos un minuto después de enviar los datos al reloj. Al momento de recibir la transición negativa se activa el RTRIC. El rastreo comenzará una vez que expire el tiempo programado del reloj.

Como prueba del funcionamiento del reloj de tiempo real se varió el tiempo en que debía comenzar el rastreo. En todas las variaciones el rastreo comenzó al momento indicado.

---

## CONCLUSIONES

El presente trabajo de tesis representa parte del proyecto Carga Útil Óptica de comunicaciones para el microsatélite experimental SATEX.1. Éste consistió en la elaboración, implementación y prueba de interfaces para la carga útil óptica de SATEX.1. Las interfaces realizadas se conformaron de acuerdo a lineamientos tanto de mecánica como programática. Cabe recalcar que el diseño de este trabajo es de tipo experimental, por lo que un aspecto significativo de éste fue la creación de una plataforma que presente características de alta versatilidad y facilidad de mantenimiento que le permitirán la adhesión de nuevas funciones e incluso de rutinas de control aún no establecidas.

Entre las características más importantes de las interfaces tenemos el empleo del sistema de interrupciones del microcontrolador combinado con la programática para cumplir con la especificación de funcionamiento en tiempo real. Ésto resta uso del procesador permitiendo realizar otras funciones mientras se espera un evento. Las rutinas de interrupción se crearon utilizando el menor código posible para lograr una entrada y salida rápida ocasionando el menor disturbio. Se combinó programación de alto nivel con ensamblador en casos donde se necesitaba tener mayor control sobre el procesador y el flujo del programa.

El trabajo también se puede notar por la administración dinámica de los recursos del microcontrolador. Durante el diseño de las interfaces la falta de un mayor número de temporizadores forzó la administración dinámica de los mismos. Esto se pudo lograr debido a que algunos de los temporizadores designados a cierta función no permanecen activos todo el tiempo. Durante ese tiempo de inactividad los temporizadores son utilizados en otras funciones, lo que requiere de una configuración dinámica. En algunos casos, un temporizador realiza hasta tres ciclos de temporización diferente.

Para lograr los objetivos del monitor mínimo, fue necesario agregar instrucciones o mandos que no se incluían en el protocolo II-UNAM. Estos mandos deben ser transparentes para el protocolo pero no para la interfaz serie. Como resultado se agregaron tres nuevos mandos que puede reconocer CCUO y que son totalmente transparentes para el sistema de comunicaciones utilizando el protocolo II-UNAM. Estas nuevas instrucciones son exclusivas para CUO.

Las pruebas comprobaron que el diseño elaborado de las interfaces era funcional y de acuerdo a especificaciones.

Con las interfaces se logrará establecer una comunicación entre CCUO y los diferentes módulos que conforman a CUO así como la comunicación externa con las computadoras de SATEX.1 y la ET. El trabajo quedará como parte del microsatélite y servirá para las pruebas finales del funcionamiento de CUO.

Aunque el trabajo fué diseñado para el proyecto SATEX.1, los programas creados se pueden utilizar para el monitoreo de sistemas remotos y de difícil acceso. Las entradas analógicas para el convertidor A/D del SAB 80C166, en el sistema mínimo, se conectarían a otros sistemas y el programa enviaría los resultados de las mediciones. Las rutinas de los motores se pueden emplear en otra aplicación que requiera del control a lazo abierto de motores y manejadores como los empleados en este proyecto. La comunicación serie se puede realizar empleando el programa de emulación "TESTCASE.C".

## Apéndice A

### Determinación de órbita.

En el siguiente listado se presentan los parámetros más importantes para la determinación de una órbita, basados en un sistema de coordenadas inerciales.

- ♦ Inclinación Orbital ( $i$ ).
- ♦ Época ( $T_0$ ).
- ♦ Argumento de Perigeo<sup>17</sup> ( $\omega$ ).
- ♦ Excentricidad ( $e$ ).
- ♦ Semieje Mayor ( $a$ ).
- ♦ Velocidad Media ( $N_0$ ).
- ♦ Anomalía Media ( $M_0$  o  $MA$ ).
- ♦ Ascensión Derecha del Nodo Ascendente ( $\Omega$ ).

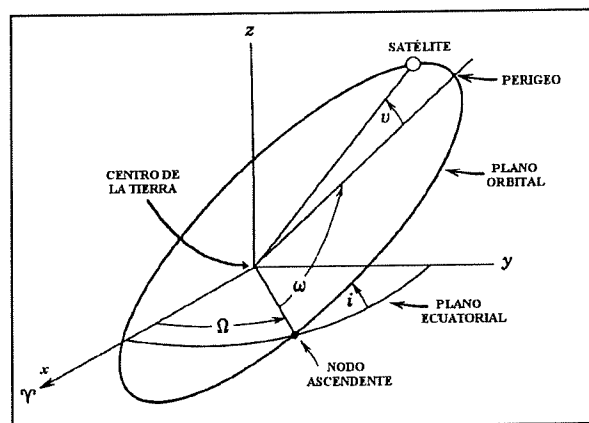


Figura 33. Parámetros Orbitales.

La inclinación orbital  $i$  se mide con respecto al plano ecuatorial y se da en el rango de  $0^\circ$  a  $180^\circ$  (ver Figura 33); lo que nos indica es la inclinación del plano orbital con respecto al plano ecuatorial. La época es el tiempo (hora y fecha) en el cual se midieron los parámetros orbitales, se usa para determinar si los parámetros están actualizados. La Ascensión Derecha del

Nodo Ascendente ( $\Omega$ ) se mide a partir del eje de referencia del sistema de coordenadas inercial hasta donde el satélite cruza el plano ecuatorial de sur a norte (nodo ascendente, (ver Figura 33) y nos sirve para conocer la orientación del plano orbital. El Argumento de perigeo se mide desde el nodo ascendente

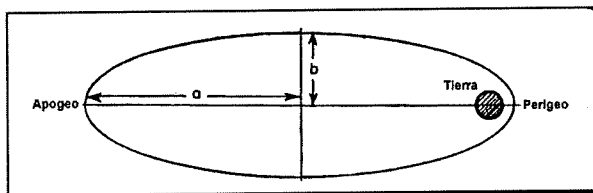


Figura 34. Perigeo y Apogeo<sup>18</sup>.

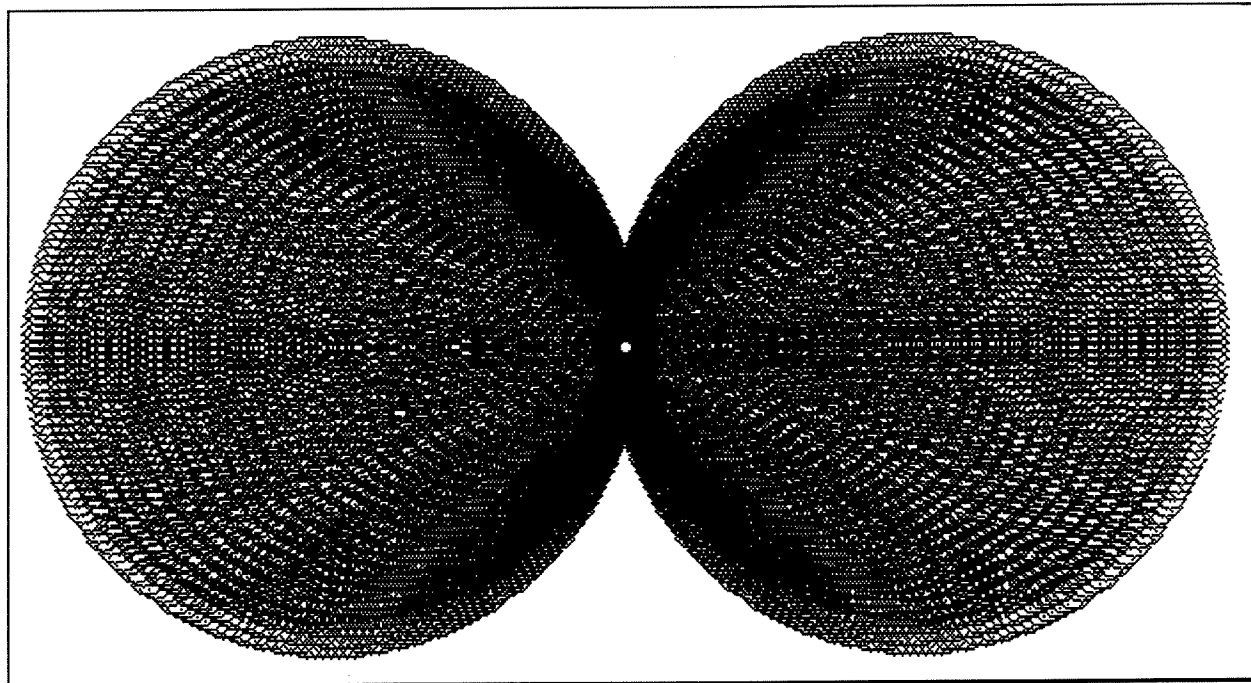
sobre el plano ecuatorial hasta el perigeo (ver Figura 33), con el podemos determinar el giro que a sufrido la órbita dentro del plano orbital. La excentricidad nos define la forma de la órbita. El semieje mayor es uno de los elementos de una elipse (ver Figura 34) que nos define el modelo básico de la órbita. La velocidad media es el promedio de velocidad del satélite dentro de su órbita; se conoce que la velocidad de un objeto en órbita alrededor de la Tierra (o cualesquier planeta) varía conforme se acerca y se aleja de la Tierra, sufriendo un mayor incremento de velocidad al acercarse a la Tierra y siendo mas lento al alejarse. La anomalía media es la posición del satélite dentro de su órbita medida desde el perigeo en sentido antihorario visto desde el norte. Las anomalía media se puede especificar en grados o en 256-avos de círculo.

<sup>17</sup> Perigeo es el punto más cercano a la Tierra que alcanza un objeto en órbita alrededor de esta.

<sup>18</sup> Apogeo es el punto más lejano a la Tierra que alcanza un objeto en órbita alrededor de esta.

## Apéndice B

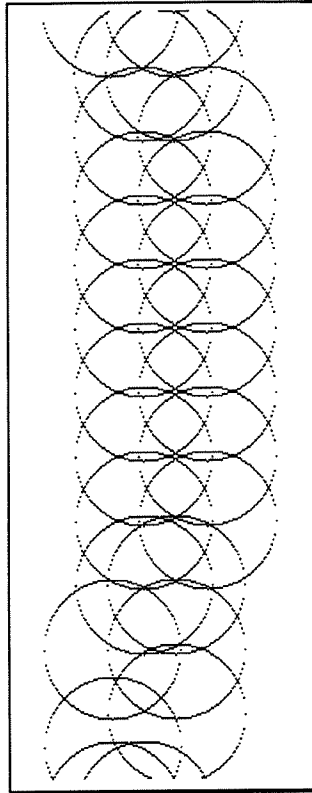
En este apéndice se describe por medio de ilustraciones el rastreo que realiza CUO para localizar la ET. Se pretende dar una idea más clara sobre el patrón de barrido que generará SATEX.1 para dicho rastreo. La primer figura describe de manera general el tipo de barrido utilizado, las segundas ofrecen un mayor detalle acerca del barrido.



*Figura 35. Vista desde el satélite sobre el barrido trazado en la superficie de la tierra para localizar la ET. Esta ilustración es solo una simulación de un caso particular para ejemplificar el patrón de prueba sobre una superficie plana, no muestra la curvatura de la tierra.*

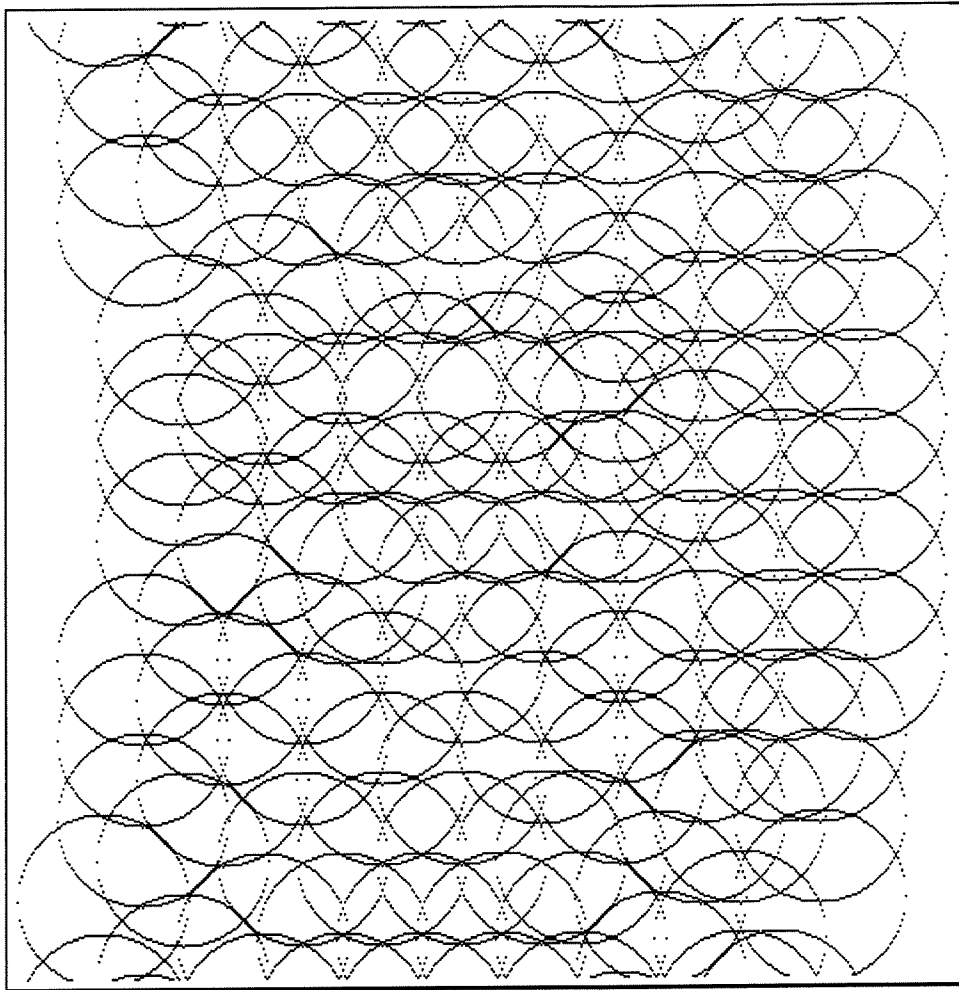
Como se puede ver en la Figura 35 el barrido se realiza trazando trayectorias en forma de círculos concéntricos de diferente diámetro conforme SATEX.1 se acerca a la ET. Se había mencionado en el Capítulo 1 que el inicio de cobertura (o rastreo) comienza cuando el microsátélite entre en línea de vista de ET; pero ésta no es la única condición para que el rastreo inicie, también se tiene que considerar el campo de vista del sensor óptico de CUO [14]. Esto nos restringe el número de veces que se puede lograr establecer un enlace óptico; refiriéndonos a que no todas las órbitas serán útiles para este propósito. El inicio de cobertura, indicado por RTRIC, comienza cuando ET entra dentro del campo visual de CUO. La primer trayectoria de adquisición trazada supone alcanzar la ET; al terminarla el microsátélite se encontrará más cerca de la estación terrena por lo que la próxima trayectoria a trazar se reduce en diámetro para compensar el movimiento de traslación de manera que las manchas que se trazan sobre la superficie terrestre tengan un traslape de un radio de las mismas para asegurar una cobertura completa del área de rastreo. Se debe

señalar que este tipo de barrido es un caso especial cuando el microsatélite pasa por encima de la estación terrena. En otros casos donde no pase directamente por encima se implementará otro tipo de barrido.



*Figura 36. Acercamiento de dos trayectorias para observar el traslape entre manchas y trayectorias.*

La Figura 36 es un acercamiento para mostrar el traslape que se forma entre cada mancha (círculos) del láser en la superficie de la Tierra y entre cada trayectoria de adquisición. Esta figura es una ampliación de dos trayectorias. Podemos notar el traslape de un radio de la mancha del láser entre cada paso (cada mancha avanza un radio dentro de la trayectoria de adquisición con cierta cadencia de tiempo [14]) y el traslape entre las dos trayectorias trazadas por las manchas. El traslape procura enmascarar algunos de los movimientos o perturbaciones del microsatélite y asegurar una cobertura completa. En la Figura 37 podemos ver parte de un trazado de 10 trayectorias de adquisición con el traslape mencionado y la cobertura del área que se genera.



*Figura 37. Barrido de 10 trayectorias.*

---

## GLOSARIO

**A/D.** Analógico/Digital.

**APD.** Avalanche Photodetector.

**APOGEO.** Punto más cercano a la tierra que alcanza un objeto en órbita alrededor de ésta.

**CCD.** Charge Coupled Device.

**CCUO.** Computadora de la Carga Útil Óptica.

**CICESE.** Centro de Investigación Científica y de Educación Superior de Ensenada.

**CIMAT.** Centro de Investigación Matemática.

**CINVESTAV.** Centro de Investigación y de Estudios Avanzados.

**CP.** Computadora Principal.

**CPU.** Central Processing Unit (Unidad de Procesamiento Central).

**CR.** Computadora Redundante.

**CUO.** Carga Útil Óptica.

**D/A.** Digital/Analógico.

**DET.** Departamento de Electrónica y Telecomunicaciones.

**DT.** Decodificador de Tonos.

**EFEMÉRIDE.** Localización en espacio y tiempo de un objeto.

**ESIME.** Escuela Superior de Ingeniería Mecánica y Electricia.

**UNAM.** Universidad Nacional Autónoma de México.

**FC-UNAM.** Facultad de Ciencias de la UNAM.

**II-UNAM.** Instituto de Ingeniería de la UNAM. Las siglas también se utilizan en esta tesis para referirse al protocolo de comunicaciones de SATEX.1.

**IMC.** Instituto Mexicano de Comunicaciones.

**INAOE.** Instituto Nacional de Astrofísica, Óptica y Electrónica.

**JPL.** Jet Propulsion Laboratory.

**MAA.** Memoria de Acceso Aleatorio (RAM).

**MLE.** Memoria de Lectura Exclusiva (ROM).

**NEMA.** National Electrical Manufacturers Association.

**PERIGEO.** Punto más lejano a la tierra que alcanza un objeto en órbita alrededor de esta.

**R2.** Módulo del foto-detector de avalacha.

**R3.** Módulo de la computadora de la carga útil óptica.

**R3.1.** Módulo de potencia de la carga útil óptica.

**R4.** Módulo de los motores.

**RF.** Radio Frecuencia.

**RTRIC.** Reloj de tiempo real de inicio de cobertura.

**SATEX.1.** Satélite Experimental 1

**SCT.** Secretaria de Comunicaciones y Transportes.

**T1.** Módulo del láser.

**UAP.** Universidad Autónoma de Puebla.

**VHF.** Very High Frequency.

**SCCUO.** Supervisor de la Computadora de la Carga Útil Óptica.

**EETCP.** Emulador de la Estación Terrena-Computadora Principal.

---

## BIBLIOGRAFÍA.

- [1] Boston System Office/Tasking, 80166 C Cross-Compiler User's Guide.
- [2] Control de Apuntamiento y Seguimiento Espacial. Carga Útil Óptica. Proyecto Satex.1. Diseño de la Computadora e Interfaces.
- [3] Data Converter Reference Manual. Vol II. 1992. Analog Devices. USA.
- [4] Davidoff, Martin R., *The Satellite Experimenter's Handbook*, ARRL, Newington CT, 1990.
- [5] Díaz, Héctor. ESIME Ticoman. Comunicado por facsímil dirigido al Dr. Javier Mendieta, mayo de 1994. 3 pp.
- [6] General Scanning, *Digital Scanning Controller DSC, User Manual*, General Scanning, 50 pp., 1991.
- [7] General Scanning, *G100 and G300 Series Galvanometer Optical Scanners*, Información Técnica, General Scanning, 1991.
- [8] General Scanning, *Mounted Mirrors for Optical Scanners*, Información Técnica, General Scanning.
- [9] Investigación Hoy. Julio 1994 Edición #54. Instituto Politécnico Nacional.
- [10] Logsdon, Tom, *The Navstar Global Positioning System*, Van Nostrand Reinhold, New York, 1992.
- [11] Lopez Lopez, Mario; Diseño y Especificación del Sistema de Comunicación en VHF para el Control y Comando de un Satélite Experimental; Tesis, 1995. Facultad de Ciencias Físico-Matemáticas, Colegio de Electrónica. BUAP.
- [12] Ludwiszewski, Alan, *Standards of Oscillatory Scanners*, General Scanning, 21 pp.
- [13] Mejía Vázquez Héctor. Diseño de un Sistema de Control para Movimientos Coordinados en Tres Ejes. 1991, CICESE. 153 pp.
- [14] Mejía Vázquez, Héctor; Mendieta Jimenez, Francisco J.; Tovmasian, Susana y Gómez Gutiérrez, Héctor. 1994. Control de Apuntamiento y Seguimiento Espacial. Carga Útil Óptica. Proyecto SATEX.1. Informe Técnico. Comunicaciones Académicas, Serie de Electrónica y Telecomunicaciones, CICESE. 64pp. CIET9411.

- [15]Mejía Vázquez, Héctor; Héctor Gómez Gutiérrez; Fco. J. Mendieta Jiménez. 1996. Control de apuntamiento espacial y seguimiento espacial. Carga Útil Óptica. Diseño de la computadora e interfaces. Informe Técnico Comunicaciones Académicas. Serie Electrónica y Telecomunicaciones, CICESE, 167 pp. CTETT9513.
- [16]Mendieta, F. J.; Goanach, Sylviane, Conception and Experimentation of an Optical Free Space Link for the SATEX, Informe Técnico, Comunicaciones Académicas, Serie Electrónica y Telecomunicaciones, CICESE 49 pp. CIETT9406.
- [17]Mensajero Académico 22. Julio 1995. CICESE.
- [18]SIEMENS. Microcomputer Components. SAB 80C166/83C166. 16-Bit CMOS Single-Chip Microcontrollers for Embedded Control Applications. User's Manual. Edition 6.90. Germany, 1990.
- [19]Miles, Howard, *Artificial Satellite Observing*, American Elsevier, New York, 1974.
- [20]Mounted Mirrors for Optical Scanners. Product Specifications. General Scanning. USA.
- [21]PACE Incorporated. Operation & Maintenance Manual. Manual No. 5050-0219 Rev. C, USA.
- [22]New England Affiliated Technologies. Precision Positioning Components and Systems. Product Catalog. 1992, USA.
- [23]Ramón Escobal, Pedro, *Methods of Orbit Determination*, John Wiley & Sons, New York, 1976.
- [24]Siemens, Using The SAB80C166 Bootstrap Loader For A Rom-Less System, Family Application Notes, USA.
- [25]SLO-SYN AC Synchronous, Gearmotors and DC Stepper Motors. Superior Electric. USA. 119 pp.
- [26]SLO-SYN Motion Controls. Catalog MSC 190. Superior Electric. USA. 119 pp.
- [27]SLO-SYN. CAMAS Torque Finder. Stepper Motor Software. MKT 100. Superior Electric.
- [28]Step Motor, Control and Load Interaction in a Step Motor System. Technical paper 80 MB-1. Superior Electric, 1986. USA. 11 pp.
- [29]The C Programming Language, Second Edition, Kernighan/Ritchie, Prentice Hall 1988.
- [30]The Guide to Position Sensing. Graseby Optronics. Tutorial. USA. 15 pp.

[31] Vivas, Essau. II UNAM. Comunicado por facsímil dirigido al Dr. Javier Mendieta, mayo de 1994. 15 pp.

# ÍNDICE

## A

Actitud.....	7
Anomalía Media .....	76
APD.....	14, 48
Apéndice A.....	76
Apéndice B.....	77
Apuntamiento espacial.....	4
Argumento de Perigeo.....	76
Ariane 5.....	4
Arianespace .....	4
arrastre atmosférico.....	7
Ascensión Derecha del Nodo Ascendente.....	76
Avalanche Photo-Diode .....	14

## C

cadencia.....	43
<i>cadencia()</i> .....	43
CAPIN.....	50
Carga Útil Óptica.....	8
carga útil óptica de comunicaciones para SATEX.1	8
Cargas Útiles	
cámara CCD, subsistema de comunicaciones	
banda Ka, subsistema de comunicaciones	
ópticas .....	3
CCUO.....	10, 12
CCUOCON.....	27
<i>chkram()</i> .....	28, 59
<i>comienza_barrido()</i> .....	41
comparadores.....	43
CC1, CC2.....	43
Conclusiones.....	74
Contenido .....	i
convertidor A/D .....	48
<i>convertidor()</i> .....	46
CrossView .....	56
CUO .....	8

## D

<i>decod_men()</i> .....	36
determinación de órbita.....	6, 76
Diseño de las interfaces.....	27

## E

EETCP .....	56
Emulador de la Estación Terrena-Computadora	
Principal.....	56
EN28V .....	39
Enlace entre interfaces .....	51
ENMX.....	39

ENMY .....	39
------------	----

## É

Época .....	76
-------------	----

## E

<i>err_convertidor()</i> .....	70
Especificaciones .....	23
Especificaciones generales.....	23
Especificaciones particulares .....	24
Estación Terrena	
ET .....	2
Excentricidad .....	76

## F

faro.....	1, 8, 9, 48
foto-detector de cuatro cuadrantes.....	14, 48
foto-diodo de avalancha .....	48

## G

Glosario.....	80
---------------	----

## H

<i>home()</i> .....	41
---------------------	----

## I

II-UNAM .....	16
Inclinación Orbital .....	76
instrucciones exclusivas de CCUO.....	64
Intel MSC-86 .....	33
Interfaces.....	29
Interfaz con foto-detector de cuatro cuadrantes .....	48
Interfaz con láser .....	46
Interfaz con motores .....	39
Interfaz serial .....	32
interrupciones.....	51

## L

láser .....	13, 26, 46
línea de vista .....	5
Lista de figuras .....	iv
Lista de pruebas a especificaciones generales.....	53
Lista de pruebas a especificaciones particulares .....	54
Lista de tableros .....	vi

## M

<i>main()</i> .....	27, 59
Mandos externos .....	20
Mandos internos.....	17
Mecamática .....	23

Microcontrolador .....	12	temporizadores T3 y T4 .....	42
Módulo de potencia R3.1 .....	12	Relojes de tiempo real.....	49
módulo R2 .....	11	Resultados .....	53
Módulo R2 foto-detector de avalancha .....	14	Rutina de decodificación.....	36
módulo R3 .....	10	Rutina de recepción.....	33
módulo R3.1 .....	11	Rutina de temporización.....	39
Módulo R4 (motores) .....	14	Rutina de transmisión.....	37
módulo T1 .....	11	<b>S</b>	
Módulo T1 (láser) .....	13	SATEX.1 .....	2
Monitor mínimo.....	27, 58	SCCUO .....	56
Montaje .....	55	secRx.....	35
<i>motores()</i> .....	41	sctx.....	37
<b>N</b>		seguimiento.....	4
niveles de interrupción.....	51	Semieje Mayor.....	76
<b>O</b>		<i>ser_CAPIN()</i> .....	50
Objetivos del trabajo .....	15	Simvolt.....	57
oscilador de 1 kHz .....	42	sistema de coordenadas inercial .....	6
<b>P</b>		Solidaridad	
Parámetros orbitales.....	5	I, II .....	4
perturbaciones atmosféricas .....	8	Spin.....	7
Perturbaciones intrínsecas.....	7	Supervisor de la CCUO .....	56
Plataforma básica de CUO .....	12	<b>T</b>	
Precesión .....	7	temporizador de 32 bits .....	49
programa principal .....	27, 59	temporizador T0.....	43, 49
Programas.....	21	temporizador T1 .....	43, 49
Programática.....	23	temporizador T2 .....	36, 39, 49
Protocolo de comunicación.....	16	temporizador T5 .....	49
Pruebas .....	53	temporizador T6 .....	49
<b>R</b>		temporizadores concatenados	
rastreo.....	4	T3, T4.....	49
Reloj de Tiempo Real de Inicio de Cobertura (RTRIC).....	42, 49	<i>trayectoria()</i> .....	29
		<b>V</b>	
		Velocidad Media .....	76