



# UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

---

---

FACULTAD DE INGENIERIA

ACTUALIZACIÓN, INSTALACIÓN Y VALIDACIÓN  
DEL SOFTWARE OPERATIVO PARA  
UN MICROSATÉLITE DE ÓRBITA BAJA

T E S I S  
QUE PARA OBTENER EL TÍTULO DE:  
INGENIERO EN TELECOMUNICACIONES  
P R E S E N T A  
MIGUEL ÁNGEL PÉREZ VELÁZQUEZ



ASESOR: Dr. ESAÚ VICENTE VIVAS

MEXICO, D. F. 2004

# Índice

---

## CAPÍTULO 1 INTRODUCCIÓN

1.1	Antecedentes .....	1
1.2	Microsatélite Experimental Mexicano, proyecto SATEX .....	2
1.3	Descripción general del satélite SATEX .....	5
1.4	Características globales del SATEX .....	5
1.5	Subsistemas a cargo del Instituto de Ingeniería de la UNAM .....	7
1.6	Mantenimiento Automatizado de la Computadora de Vuelo .....	9
1.7	Objetivos y delimitación de la presente tesis .....	10

## CAPÍTULO 2 DESCRIPCIÓN DE EQUIPOS Y SUBSISTEMAS UTILIZADOS PARA EL DESARROLLO DE LA TESIS

2.1	Introducción .....	12
2.2	Software de Depuración, Emulación y Validación Operativa (SOFDEVO) .....	13
2.2.1	Estructura general de SOFDEVO .....	14
2.3	Software de Estación Terrena del Proyecto SATEX.....	18
2.3.1	Estructura general del software de Estación Terrena .....	20
2.4	Computadora de Vuelo .....	24
2.5	Software de operación satelital .....	26
2.6	Emulador ROM/RAM TRE-200 .....	27
2.6.1	Modo de empleo del Emulador ROM/RAM .....	27
2.7	Programador Universal UTP-400 .....	30
2.7.1	Modo de empleo del UTP-400 .....	31
2.7.1.1	Modificación del Buffer .....	32
2.7.1.2	Cargar el archivo fuente en el buffer .....	33
2.7.1.3	Programación de memoria EPROM .....	34
2.8	Simulador del satélite (SIMSAT) .....	35

## CAPÍTULO 3 SOFTWARE DE OPERACIONES DEL MICROSATÉLITE

3.1	Introducción .....	37
3.2	Protocolo de comunicación en la Red Interna .....	38
3.2.1	Comandos de red interna .....	40
3.3	Protocolo de comunicación en la Red Externa .....	43
3.3.1	Envío de comandos de respuesta a comandos en línea .....	43
3.3.2	Envío de parámetros de estado y Telemetría .....	44
3.3.3	Envío de imagen .....	48
3.4	Estructura general del software de operación satelital .....	49
3.5	Descripción de los mapas de memoria de la Computadora de Vuelo .....	51
3.6	Software Mínimo de Operación Satelital .....	54

## CAPÍTULO 4 SOFTWARE DE DEPURACIÓN, EMULACIÓN Y VALIDACIÓN OPERATIVA PARA EXPERIMENTOS DEL SATÉLITE

4.1	Introducción .....	56
4.2	Lenguaje de programación utilizado para el desarrollo de SOFDEVO .....	56
4.3	Software asociado con operaciones satelitales detectadas exclusivamente con SOFDEVO .....	58
4.3.1	Avisos generados al detectar comandos satelitales.....	58

4.3.2	Respuestas programables para comandos de red interna asociados con operaciones satelitales .....	59
4.3.3	Respuestas automáticas para comandos de red interna asociados con operaciones satelitales .....	59
4.4	Tramas y protocolo de comunicaciones .....	60
4.5	Software asociado al experimento de comunicaciones ópticas .....	65
4.6	Software asociado con el experimento de la cámara digital de percepción remota .....	65
4.7	Software asociado con el experimento del sistema mínimo de sobrevivencia .....	65
4.8	Software asociado con el experimento de MACV .....	69
4.9	Software asociado con la operación del subsistema de potencia .....	70

**CAPÍTULO 5 ACTUALIZACIÓN DEL SOFTWARE DE OPERACIONES DEL SATÉLITE Y DEL SOFTWARE SOFDEVO**

5.1	Introducción .....	72
5.2	Actualización de la Red Interna de SATEX .....	72
5.3	Dispositivos electrónicos EDAC .....	74
5.3.1	EDAC 29C516E .....	75
5.3.2	Software de control del EDAC .....	79
5.4	Mantenimiento Automatizado de la CV .....	81
5.4.1	Diagnóstico de Computadoras de SATEX .....	85
5.4.2	Reconfiguración y Reinicio .....	89
5.4.3	Restitución de procesadores físicos con falla .....	91
5.5	Software Mínimo de SATEX (SOFMIN) .....	94
5.5.1	Depuración para SOFMIN .....	94
5.6	Compilación del Software de Operación Satelital .....	98
5.7	Actualización de SOFTDEVO .....	101

**CAPÍTULO 6 VALIDACIÓN DEL SOFTWARE DE OPERACIONES SATELITAL Y MANTENIMIENTO AUTOMATIZADO DE LA CV CON EL APOYO DE SOFDEVO Y EL SOFTWARE DE ESTACIÓN TERRENA**

6.1	Introducción .....	102
6.2	Instalación de SOFMIN y EPROM a las Tarjetas de Procesamiento .....	102
6.3	Preparación de equipos para la validación del Software de Operación Satelital .....	107
6.4	Inicio de vida de SATEX .....	111
6.5	Validación de fallas simuladas .....	115
6.6	Subir programa .....	119
6.7	Validación del experimento de Mantenimiento Automatizado de la CV .....	121
6.8	Validación del Comando de Diagnóstico Iterado (CDI) .....	128

**CAPÍTULO 7 CONCLUSIONES Y RECOMENDACIONES**

7.1	Conclusiones .....	133
7.2	Recomendaciones .....	134

<b>Bibliografía</b> .....	136
---------------------------	-----

<b>Apéndice A. Comandos de Red Interna</b> .....	A
--	---

<b>Apéndice B. Software de control del EDAC</b> .....	B
---	---

# Capítulo 1

## *Introducción*

---

### **1.1 Antecedentes**

Uno de los resultados más fascinantes y notables obtenidos a partir de los programas espaciales es la tecnología de los satélites artificiales, que aparecieron en la década de los cincuenta. La llegada de estos complejos aparatos electrónicos modificó visiblemente la forma de vida de la mayor parte de la población del mundo ya sea directa o indirectamente. Gracias a dichos aparatos conocemos con más precisión los recursos naturales de la Tierra y los fenómenos meteorológicos; las distancias entre las ciudades y los países se han acortado y ahora pueden intercambiar todo tipo de información casi instantáneamente; y más allá de las capas atmosféricas, podemos observar y comprender mejor el Universo.

Una forma de diferenciar los tipos de satélites, es por la altura a la que se encuentra la órbita por la que circulan. Se pueden diferenciar tres tipos de órbitas según sus altitudes: Órbitas Terrestres Geosíncronas (GEO) situada a 35848 km. de altura, con una latitud de 0 grados, es decir, situada sobre el Ecuador; Órbita Terrestre Media (MEO) situada a una altura de entre 10075 y 20150 km., su posición relativa respecto a la Tierra no es fija; y Órbita Terrestre de Baja altura (LEO) situada a a una altura de entre 400 y 1400 km.

Los satélites localizados en las órbitas bajas son de tres tipos: LEO pequeños (centenares de Kbps) destinados a aplicaciones de bajo ancho de banda; LEO grandes (miles de Kbps) albergan las aplicaciones de los anteriores y otras como telefonía móvil y transmisión de datos; y finalmente los LEO de banda ancha (megaLEO) que operan en la banda de Mbps.

En general, todos los satélites artificiales funcionan bajo el mismo principio y constan de varias partes comunes, independientemente de su objetivo en órbita alrededor de la Tierra. Desde luego hay algunas diferencias fundamentales, por ejemplo, una buena cantidad de celdas solares para alimentarse de energía, antenas para transmitir su información a ciertos puntos del planeta y también para poder recibir instrucciones o cualquier otro tipo de señal desde ellos, así como el medio de propulsión que utilizan para corregir su órbita, posición u orientación con respecto a la Tierra.

SATMEX es la empresa mexicana que está a la vanguardia en las telecomunicaciones en México y cuenta con 31 años de experiencia en transmisión satelital. Estas características han permitido que entienda y se esfuerce por satisfacer las necesidades de sus clientes en todo el continente americano. Actualmente da servicio a clientes en Los Angeles, Nueva York, Miami, Chicago, Sn. Francisco, Washington y Houston. Además cubre México, Argentina, Venezuela, Colombia, Chile, América Central y el Caribe. La empresa fue privatizada en octubre de 1997, cuando las empresas Principia y Loral Space & Communications adquirieron el 75% del capital. El 25% restante pertenece al gobierno federal mexicano, que carece de voto en la empresa.

## CAPÍTULO 1. INTRODUCCIÓN

La Universidad Nacional Autónoma de México incursionó en la investigación satelital desde 1969 con un grupo de radioaficionados mexicanos que se unieron a AMSAT INTERNACIONAL. Años después se formaría el Programa Universitario de Investigación y Desarrollo Espacial (PUIDE), dicho programa permitió la realización de dos experimentos de microsátélites. El primero de ellos fue el llamado UNAMSAT-1 un microsátélite de 10.7 kilos y 10 watts, utilizaba celdas solares y baterías como fuente de energía, su computadora trabajaba con 1 watt y ocupaba menos espacio que una computadora normal; ella organizaría todas las funciones del satélite, incluidas las comunicaciones, desgraciadamente -el 28 de marzo de 1995- una falla en el cohete, que lo pondría en órbita, ocasionó la pérdida del microsátélite. El segundo fue el llamado UNAMSAT-B, un microsátélite gemelo del UNAMSAT-1 que fue puesto en órbita en 1996, pero sólo funcionó correctamente un par de meses.

Gracias a los resultados y la investigación que se generaba en la UNAM, en 1994, el ahora desaparecido Instituto Mexicano de Comunicaciones (IMC), financió un ambicioso proyecto con el fin de diseñar, construir y validar un microsátélite experimental, con la intención de desarrollar tecnología espacial 100% mexicana, dicho proyecto tiene el nombre de Satélite Experimental (SATEX).

### **1.2 Microsatélite Experimental Mexicano, proyecto SATEX**

El proyecto SATEX (SATélite EXperimental) es un esfuerzo mexicano para establecer un programa de desarrollo de tecnología espacial. El SATEX es el primer microsátélite de lo que se espera sea una serie de satélites que nos permitan satisfacer necesidades puntuales en diversas áreas de interés del país. El SATEX es un microsátélite de 50Kg de masa, que lleva como cargas útiles: un experimento óptico denominado Carga Útil Óptica (CUO), un sistema de radio comunicaciones en banda Ka y una cámara de CCD; todo esto soportado por una plataforma diseñada y ensamblada por instituciones nacionales.

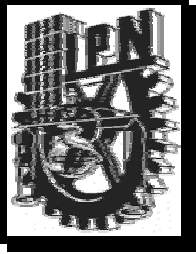
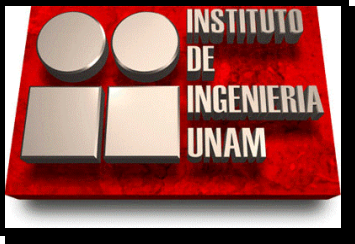


La importancia del desarrollo de la actividad espacial se da por el hecho de que ésta es generadora de tecnología de punta, volcando hacia otros campos de la ciencia y la industria una derrama tecnológica esencial para el desarrollo autónomo. Además de que las actividades espaciales ofrecen oportunidades para validar nuevas tecnologías de telecomunicaciones, así como la formación de recursos humanos.

De esta forma, el objetivo de la misión es el diseño, construcción y validación de un microsátélite experimental en un ambiente universitario y multi-institucional, que permita el desarrollo de una misión científica enfocada a la investigación en telecomunicaciones y la generación de una plataforma multimisión con capacidad evolutiva, adaptable a diversos requerimientos. Las instituciones que actualmente participan en el proyecto son: El Instituto de Ingeniería y el de Geografía de la Universidad Nacional Autónoma de México (UNAM), el Centro de Investigación en Matemáticas (CIMAT), el Centro de Investigación y Educación Superior de Ensenada (CICESE), el Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE), el Centro Nacional de Metrología (CENAM), así como la Escuela de Aeronáutica, el CITEDI y la Sección de Graduados de la ESIME del Instituto Politécnico Nacional (IPN); quien también ejerce la coordinación general del proyecto. Cada una de las instituciones anteriores se encargan de diferentes partes que conforman al SATEX, dichas partes se desglosan como sigue:

## CAPÍTULO 1. INTRODUCCIÓN

	<h3>CICESE</h3> <p>(Centro de Investigación Científica y de Estudios Superiores de Ensenada)</p> <p><a href="http://www.satex.org.mx">http://www.satex.org.mx</a></p> <p>Subsistemas que desarrolla:</p> <ul style="list-style-type: none"><li>✓ Equipos de Comunicaciones: Radio1 y Radio2.</li><li>✓ Experimento de Comunicaciones Ópticas (CUO).</li><li>✓ Sistema Mínimo de Supervivencia (SIMS).</li><li>✓ Hardware de estación terrena.</li></ul>
	<h3>CIMAT</h3> <p>(Centro de Investigación en Matemáticas)</p> <p><a href="http://www.cimat.mx">http://www.cimat.mx</a></p> <p>Subsistemas que desarrolla:</p> <ul style="list-style-type: none"><li>✓ Modelado de la dinámica orbital del vehículo, de campo magnético terrestre y de estabilización en tres ejes.</li><li>✓ Algoritmos de estabilización del satélite.</li><li>✓ Validación terrestre del sistema de estabilización.</li></ul>
	<h3>CITEDI</h3> <p>(Centro de Investigación en Tecnología Digital)</p> <p><a href="http://www.citedi.mx">http://www.citedi.mx</a></p> <p>Subsistemas que Desarrolla:</p> <ul style="list-style-type: none"><li>✓ Sistema de Potencia.</li><li>✓ Celdas Solares.</li><li>✓ Baterías.</li></ul>
	<h3>INAOE</h3> <p>(Instituto de Astrofísica Óptica y Electrónica)</p> <p><a href="http://www.inaoe.mx">http://www.inaoe.mx</a></p> <p>Subsistema que desarrolla:</p> <ul style="list-style-type: none"><li>✓ Transmisor en banda Ka (en estación terrena a 23 GHz).</li></ul>

## CAPÍTULO 1. INTRODUCCIÓN

	<p><b>IPN</b> (Instituto Politécnico Nacional) <a href="http://www.ipn.mx">http://www.ipn.mx</a></p> <p>El IPN es el encargado de la coordinación general del proyecto y es responsable de los siguientes subsistemas:</p> <p>A través de la Escuela de Ingeniería Aeronáutica:</p> <ul style="list-style-type: none"><li>✓ Estructura del satélite.</li><li>✓ Integración y pruebas.</li></ul> <p>A través de la sección de graduados de la Escuela Superior de Ingeniería Mecánica y Eléctrica, ESIME:</p> <ul style="list-style-type: none"><li>✓ Bobinas de torque magnético (BTM).</li><li>✓ Experimento en banda Ka.</li></ul>
	<p><b>UNAM</b> (Universidad Nacional Autónoma de México) <a href="http://pumas.iingen.unam.mx">http://pumas.iingen.unam.mx</a></p> <p>La UNAM, a través del Instituto de Ingeniería, tiene a cargo el desarrollo de los siguientes subsistemas:</p> <ul style="list-style-type: none"><li>✓ Computadora de Vuelo.</li><li>✓ Sensores de corriente, temperatura y magnetómetros.</li><li>✓ Hardware de acondicionamiento y Multicanalización de sensores.</li><li>✓ Protocolos para telemetría y comando.</li><li>✓ Hardware y software para la red interna del satélite.</li><li>✓ Software de vuelo.</li><li>✓ Software de Estación Terrena.</li><li>✓ Participación en el experimento de la cámara digital.</li><li>✓ Experimento de arquitectura de computadoras "Computadora Semivirtual Tolerante a Fallas".</li></ul>
	<p><b>UNAM</b> (Universidad Nacional Autónoma de México) <a href="http://pumas.iingen.unam.mx">http://pumas.iingen.unam.mx</a></p> <p>La UNAM, a través del Instituto de Geografía, tiene a cargo el desarrollo de los siguientes subsistemas:</p> <ul style="list-style-type: none"><li>✓ Fabricación de una mesa suspendida en aire para la validación de los algoritmos de estabilización del satélite.</li><li>✓ Sensores finos de Sol.</li></ul>
	<p><b>CENAM</b> (El Centro Nacional de Metrología)</p> <ul style="list-style-type: none"><li>✓ Pruebas de Vibración.</li><li>✓ Pruebas de compatibilidad Electromagnética para equipos electrónicos.</li></ul>

## CAPÍTULO 1. INTRODUCCIÓN

Cada institución participante desarrolla los subsistemas que tienen comprendidos, además de depurarlos y actualizarlos, ya que la tecnología en software y hardware avanza a pasos agigantados.

### 1.3 Descripción general del Satélite SATEX

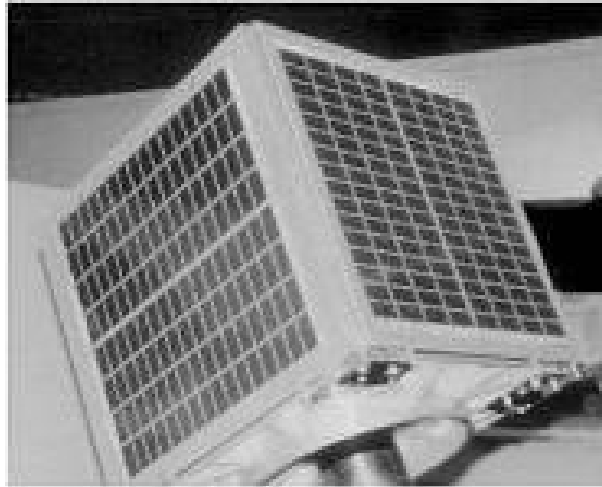


Figura 1.1. Maqueta a escala del SATEX

El satélite es un cubo de 50 centímetros por lado exterior, que se colocará en órbita heliosincrónica a 800 km de altura y 98 grados de inclinación. El periodo de recorrido de la órbita tiene una duración de 101 minutos, con un periodo de sol de 67 minutos y un periodo de eclipse de 34 minutos. El satélite básicamente se divide en dos partes: la plataforma y la carga útil. Se define a la plataforma como todo aquello que permite que el satélite funcione. En cuanto a la carga útil —se definió como experimental en el área de las telecomunicaciones—: un experimento de comunicaciones ópticas, comunicaciones en la banda Ka y VHF y una cámara digital que permitirá validar la operación del satélite al registrar fotografías del planeta.

También cabe señalar que el IINGEN propuso un experimento para automatizar el proceso de mantenimiento de la Computadora de Vuelo (CV) la UNAM con lo cual se implantó una arquitectura de cómputo semivirtual tolerante a fallas.

### 1.4. Características globales del SATEX

Masa:	55 Kg.
Estabilización:	<ul style="list-style-type: none"><li>✓ Gradiente gravitacional de 6 metros de longitud con masa terminal de 2.2 Kg.</li><li>✓ Seis bobinas de par magnético, dos en cada eje ortogonal del satélite.</li><li>✓ Dos magnetómetros triaxiales.</li><li>✓ Cuatro sensores finos de sol bidimensionales.</li><li>✓ Un sensor burdo de sol.</li></ul>



## CAPÍTULO 1. INTRODUCCIÓN

Cargas útiles:	<ul style="list-style-type: none"><li>✓ Experimento de comunicaciones ópticas, receptor infrarrojo.</li><li>✓ Experimento de comunicaciones en banda Ka, receptor de 3.5GHz.</li><li>✓ Cámara Digital en el espectro visible.</li><li>✓ Comunicaciones de respaldo por tonos (sistema mínimo de sobrevivencia).</li><li>✓ Mantenimiento automatizado a la Computadora de Vuelo.</li></ul>
Orbita:	Baja (LEO).
Altitud:	800 km. aproximadamente, dependiendo del lanzamiento.



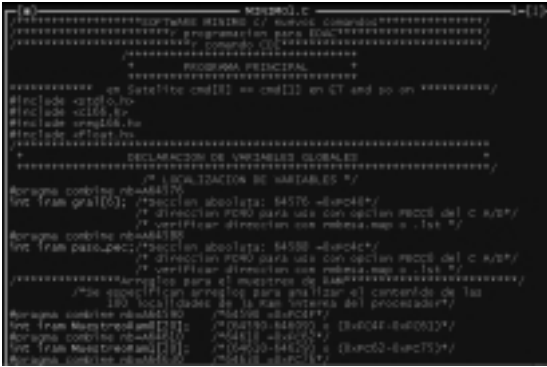
El Microsatélite Experimental SATEX (MES) es un satélite de órbita baja que será ubicado en una órbita polar, ello implica que se desplazará continuamente por todo el planeta, de tal forma que visitará a su Estación Terrena (ET) de supervisión y de control varias veces por día. Además, cabe señalar que los tiempos de visita serán a su vez variables, presentándose tiempos de visita mayores (alrededor de quince minutos) cuando el satélite sobrevuele el Cenit de la ET, en tanto que fuera de éste los tiempos se reducen en la medida en que se reduzca su ángulo de vuelo respecto al horizonte.

El problema de la comunicación Tierra-Satélite se torna significativo en vista de que el satélite se encontrará disponible a la ET por tiempos reducidos y variables. Durante los tiempos de visita, el satélite enviará primeramente su telemetría a Tierra y posteriormente recibirá los comandos que definan operaciones por ejecutar en línea durante la visita satelital, además de recibir las misiones o tareas que ejecutará cuando se encuentre fuera del alcance de la estación terrena. Solo de esta forma se podrán programar experimentos satelitales para realizarse en los tiempos de visita satelital y en sitios geográficos diferentes a los de la Estación Terrena.

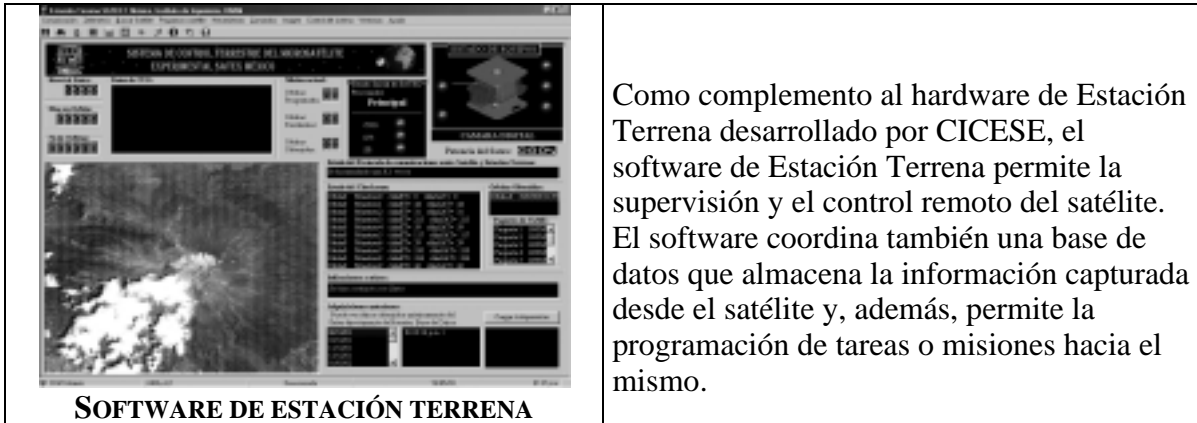
Una vez orbitado el microsatélite, el acceso a él quedará determinado por las capacidades del software de su CV y por el software de captura de telemetría y control de operaciones que estará instalado en la estación terrena. La ubicación de la ET está planeada para ubicarse en la Ciudad de Ensenada, BC, debido a que CICESE es responsable del experimento de comunicaciones ópticas y Baja California es un sitio de baja nubosidad y baja precipitación pluvial, lo cual la convierte en un sitio idóneo para realizar comunicaciones ópticas. No obstante, se espera que durante el progreso del proyecto se encuentre financiamiento para instalar a su vez una Estación Terrena en la Ciudad de México, en Ciudad Universitaria, para recepción de telemetría y para control satelital.

El sistema de potencia de SATEX se encuentra constituido por cuatro paneles solares colocados en cuatro de los lados del cubo, de 50 cm de arista, para generar la potencia que se suministrará a los subsistemas electrónicos mediante el sistema de baterías, este sistema es controlado por un microprocesador de tipo militar, que además es el encargado de la conversión de voltajes (mediante el uso de convertidores DC-DC), distribución de corriente y administración de energía. El microsatélite cuenta con cuatro Cargas Útiles principales que son: Un sistema de comunicaciones ópticas, un Detector de Tonos, Una Cámara Digital y una arquitectura de cómputo semivirtual tolerante a fallas.

**1.5. Subsistemas a cargo del Instituto de Ingeniería de la UNAM**

 <p><b>COMPUTADORA DE VUELO</b></p>	<p>La CV contiene redundancia triple de procesadores, cada procesador se encuentra protegido contra un efecto denominado “Latch-up” que se origina por radiación espacial. El diseño de la tarjeta de cada procesador contiene 1 Mb adicional de memoria RAM y además contiene un dispositivo EDAC (Error Detection And Correction) entre el microcontrolador y la memoria RAM el cual permite corregir un bit erróneo por localidad de memoria.</p>
 <p><b>MÓDULO ACONDICIONAMIENTO Y MULTICANALIZACIÓN DE SENSORES</b></p>	<p>La CV realiza la lectura de 48 sensores por medio de un módulo de multiplexaje cuyas salidas se envían a los canales Analógico/Digital (A/D) del procesador. Adicionalmente, el módulo referido realiza el acondicionamiento eléctrico de las señales de los sensores. Este acondicionamiento consiste básicamente en un proceso de filtrado, amplificación y recorte de señales. Este hardware se diseñó en circuitos impresos adicionales y constituye un equipo más desarrollado para el satélite.</p>
 <p><b>SOFTWARE DE LA COMPUTADORA DE VUELO</b></p>	<p>El software de la CV coordina las funciones de automatización asignadas al satélite, como el encendido o apagado de los equipos de radiofrecuencia, cargas útiles, liberación del gradiente gravitacional, además de otras funciones como la adquisición de datos de sensores (telemetría), coordinación de procesos de diagnóstico y detección de fallas en las diferentes computadoras del satélite y comunicaciones con Tierra. Por medio de ésta se reciben comandos en línea y misiones que especifican al satélite la ejecución de tareas a realizar en tiempos específicos.</p>

## CAPÍTULO 1. INTRODUCCIÓN



### ESTUDIO DE CONFIABILIDAD DE LA COMPUTADORA DE VUELO

Con la finalidad de obtener parámetros cuantitativos sobre la vida útil del hardware de la CV y acentuar las ventajas que se obtienen al incorporar técnicas de redundancia y tolerancia a fallas, se realizó un estudio de confiabilidad con base en normas militares, modelos serie, modelos paralelo y la ley exponencial de fallas [Johnson, 1989] y [Mejía, 2002].

Además, los estudios de confiabilidad se extendieron al análisis de la arquitectura de cómputo semivirtual tolerante a fallas, para la cual se desarrollaron modelos probabilísticos de Markov y programación versátil e interactiva en Matlab que permite evaluar confiabilidades por unidades de hardware y por tiempos variables.

Para poder validar el hardware y software desarrollado por el IIUNAM, fue necesario desarrollar dos herramientas:

- **Simulador del satélite (SIMSAT)**



Figura 1.2. Imagen de la construcción del simulador del satélite (SIMSAT).

Es un equipo que simula a diversos módulos, sensores y actuadores del satélite. Además, contiene el hardware de la red interna principal y redundante por lo cual resulta posible conectar a través de él hasta cuatro cargas útiles, incluyendo a la CV, y de esta manera validar el software de comunicación entre computadoras.

## CAPÍTULO 1. INTRODUCCIÓN

- **Software de depuración y validación operativa (SOFDEVO)**

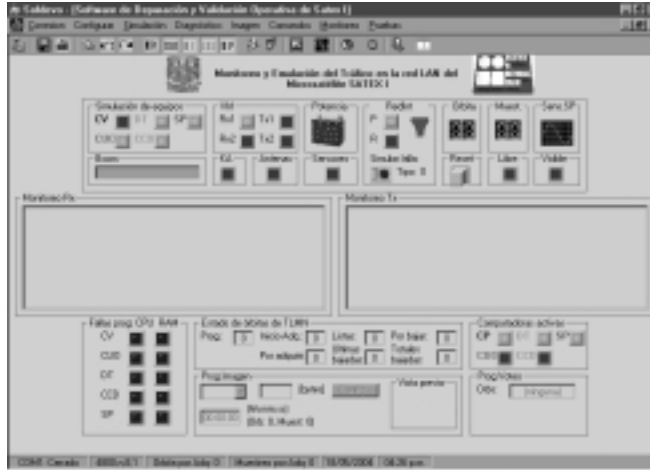


Figura 1.2. Pantalla principal del software SOFDEVO.

El SIMSAT permite validar de forma visual muchas de las tareas que realiza el satélite; sin embargo, existen ciertas operaciones que no son visibles con esta herramienta. Para tales casos fue desarrollado SOFDEVO.

SOFDEVO permite depurar y validar las operaciones que realiza el satélite por medio del monitoreo del tráfico de comunicaciones en los canales de comunicación de red interna. El software desarrollado en un lenguaje visual en ambiente Microsoft Windows (Visual Basic 6.0), monitorea los canales de red de las computadoras del satélite y también permite emular los procesos de comunicaciones de red para cualquier carga útil del satélite, incluyendo a la CV. SOFDEVO permite no sólo validar y depurar el software de la CV, sino el software de cualquier computadora del satélite, sin la necesidad de disponer físicamente de tales equipos, ayudando así a reducir los tiempos de integración del satélite.

### **1.6 Mantenimiento Automatizado de la Computadora de Vuelo (MACV)**

Para automatizar el proceso de mantenimiento de la CV, la UNAM implantó una arquitectura de cómputo semivirtual tolerante a fallas. La organización topológica de las computadoras del satélite se controla por medio de comandos de software (desde la estación terrena), de esta forma se arma una arquitectura semivirtual que contiene una parte física formada por la CV y una parte virtual formada por las computadoras de carga útil (experimentos), además, cuenta con procesadores redundantes en ambas partes de la arquitectura.

En cuanto a la detección de fallas, la arquitectura utiliza procesos de voto mayoritario, de resultados de diagnóstico de sus nodos computacionales [Torres, 2002].

Esta arquitectura persigue realizar de forma automatizada las tareas de diagnóstico, detección y reconfiguración ante fallas de la computadora más crítica del satélite, es decir, la CV.

## CAPÍTULO 1. INTRODUCCIÓN

Para conformar este experimento se desarrolló hardware y software especializado.

- **Hardware:** Se cuenta con redundancias en ciertos sistemas críticos para el funcionamiento del microsatélite, como la CV y la red interna de comunicaciones entre computadoras.
- **Software:** Se ubica en algoritmos que conducen procesos de diagnóstico, detección de fallas, así como la conmutación de equipos en caso de una posible falla.

En cuanto al sistema de detección de fallas en equipos denominado voto, permite evaluar el estado de cada una de las computadoras que integran al satélite.

Para satisfacer los requisitos de la técnica utilizada para detectar fallas, el algoritmo de voto requiere de la participación de 4 nodos computacionales, por ello inicialmente la arquitectura utiliza al Sistema de Potencia (SP) que siempre estará encendido; el sistema mínimo de sobrevivencia o Detector de Tonos (DT) quien dará el mantenimiento a la CV realizando la conmutación entre procesadores en caso de que falle alguno; la Computadora de Vuelo (CV) y la Carga Útil Óptica (CUO), dejando a la cámara digital (CCD) como procesador de refacción para la parte virtual en caso de fallas en la carga útil óptica (CUO).

Este modo de operación será utilizado como un experimento adicional del microsatélite, el cual constituye una más de las innovaciones y aportaciones del proyecto SATEX en el área de microsatélites.

La arquitectura semivirtual tolerante a fallas se activará por software cada 10 minutos y durante su conformación requiere la ayuda de al menos tres computadoras de carga útil (experimentos) del satélite en un lapso de tiempo de milisegundos.

### ***1.7 Objetivos y delimitación de la presente tesis***

El objetivo principal de esta tesis es actualizar y concluir el software operativo del Microsatélite SATEX y así mismo desarrollar un software mínimo que permita validar el experimento de mantenimiento automático del satélite SATEX.

La programación desarrollada para la CV se puede dividir en tres módulos principales, la primera es la relacionada con las tareas de control y automatización de funciones a bordo, la segunda referente al control de comunicaciones de la red interna de SATEX y la última relacionada con el software de comunicaciones con la Estación Terrena [VICENTE, 1998].

En el diseño de la CV existe un decodificador de mapa de memoria que admite el uso ya sea de memoria ROM y RAM, o bien, sólo el uso de memoria tipo RAM. El mismo decodificador permite direccionar un programa grabado de forma previa al ensamble y lanzamiento del satélite, el cual se utilizará durante la fase de inicio de vida del satélite o bien después de que se realice el apagado de la CV por razones de protección. Adicionalmente, el programa grabado previamente se utilizará como software de respaldo en caso de que el programa cargado en memoria RAM presente errores incorregibles por efecto de la radiación.

## CAPÍTULO 1. INTRODUCCIÓN

Como se ha señalado, el software operativo de la CV del Microsatélite coordinará las funciones de automatización asignadas al satélite, como el encendido o apagado de los equipos de radiofrecuencia, cargas útiles, liberación del gradiente gravitacional, además de otras funciones como la adquisición de datos de sensores (telemetría), coordinación de procesos de diagnóstico y detección de fallas en las diferentes computadoras del satélite y comunicaciones con Tierra. Por medio de este se reciben comandos en línea y misiones que especifican al satélite la ejecución de tareas a realizar en tiempos específicos.

El software operativo de la CV es una parte esencial del esquema de tolerancia a fallas propuesto para SATEX. Adicionalmente integra software dedicado para conformar experimentalmente la arquitectura de cómputo semivirtual a bordo de SATEX, cuyo propósito principal es realizar mantenimiento automatizado a la CV. La arquitectura realiza la ejecución de voto entre los procesadores a bordo, la reconfiguración automática en caso de errores no corregibles en memoria RAM, la conmutación automática de canales de comunicación ya sea en la red interna o en la comunicación con Tierra en el caso de detectar alguna falla, etc. [VICENTE, 2001].

El Software operativo de vuelo fue desarrollado en lenguaje C ANSI y con algunas utilerías propias del microcontrolador SIEMENS SAB80C166. El compilador usado fue el “TASKING Cross C Compiler Ver. 3.5.6” [TASKING, 1993] de la compañía TASKING [www.tasking.com], dicho compilador proporciona las extensiones de lenguaje necesarias para el control de puertos, periféricos y registros del microcontrolador.

En la presente tesis se desarrollaron los cambios requeridos en el software de operaciones para alcanzar la versión de vuelo oficial. De igual forma se desarrollaron los cambios necesarios en el software SOFDEVO que ha realizado el Instituto de Ingeniería en los últimos años para realizar la emulación de experimentos microsatelitales desde el punto de vista de tráfico de comunicaciones por medio de una red de área local.

Asimismo, de SOFDEVO se extrajeron los códigos fuente relativos a cada experimento para enviarlos e instalarlos en los equipos microsatelitales que se desarrollan en diferentes estados de la República.

Las adiciones de software cubren funciones de: comunicaciones Satélite-Tierra, comunicaciones entre CV y experimentos, medios de seguimiento de pruebas en tiempo real, control de un dispositivo EDAC (*Error Detection And Correction*) instalado en la CV y control de tiempo orbital. Adicionalmente se realizaron las pruebas pertinentes para validar el software operativo del satélite, así como las pruebas de validación en laboratorio para certificar la operación del sistema de automatización del satélite.

## Capítulo 2

### *Descripción de equipos y subsistemas utilizados para el desarrollo de esta tesis*

---

#### **2.1 Introducción**

En el presente capítulo se explicarán las características globales del Hardware y Software desarrollado por el Instituto de Ingeniería de la UNAM para la instrumentación del satélite, con el fin de identificar todos los equipos que se necesitaron para cubrir los objetivos de la presente tesis, además se indican sus detalles operativos fundamentales.

Se presenta la descripción general de: la Computadora de Vuelo (CV); Software de Estación Terrena (SET); Simulador del Satélite (SIMSAT); Software de Depuración, Emulación y Validación Operativa (SOFDEVO); Electrónica de Acondicionamiento de Sensores y Voto (EAS y V), Emulador ROM/RAM (Emulador TRE-200); Programador de Memorias Universal UPT-400 (Programmer & Tester UPT-400) y una descripción general del software operativo del microsatélite SATEX, el cual se verá más a detalle en el capítulo 4.

Al contar con información de los procesos que realizará el microsatélite durante su proceso de construcción se tiene una gran ayuda para depurar errores, además de que contribuye a reducir el tiempo de actualización del software de control. Por estos motivos, en el IIUNAM surgió la idea de diseñar una herramienta que permitiera la visualización de los procesos lógicos que se llevan a cabo dentro del SATEX. Esta herramienta es SOFDEVO, Software de Emulación, Depuración y Validación de operaciones del SATEX.

Respecto al SIMSAT, este se desarrolló para simular varios periféricos del satélite, como sensores, actuadores, hardware de red, conectores de red; etcétera.

Considerando los altos costos involucrados en el desarrollo de un microsatélite (económico, coyuntural, seguridad, etcétera) desde el inicio del proyecto, la UNAM adoptó la filosofía de prevención de fallas en todas las etapas de desarrollo de la instrumentación del satélite que quedaron bajo su responsabilidad. Parte de esta metodología incluyó la identificación de riesgos en las fases de diseño e implantación de todo el sistema, así como la adopción de técnicas para evitar fallas y de tolerancia a las mismas para permitir la operatividad del satélite aún ante casos de fallas catastróficas en equipos, componentes y software.

## 2.2 *Software de Depuración, Emulación y Validación Operativa (SOFDEVO)*

Al ser SATEX un proyecto en el cual los equipos se desarrollan en diversos lugares de la Republica Mexicana, resulta sumamente complicado hacer pruebas preliminares durante el desarrollo del proyecto. En este sentido SOFDEVO se planteó como solución a este problema orientándolo a la emulación de funciones que realizan los experimentos del satélite, además de permitir la visualización del tráfico de información en la red del satélite, con el fin de contar con una herramienta que pueda validar el hardware y el software desarrollado por cada institución.

El software de operaciones del microsatélite SATEX fundamentalmente se enfoca a dos procesos: el control de operaciones del satélite y las comunicaciones con el Software de Estación Terrena, por ello resulta indispensable, para la sobrevivencia y el éxito de la misión, contar con un software que permitan detectar señales tangibles de la operación del hardware del satélite. Es decir, los procesos que realiza el satélite tienen una manifestación física que permiten determinar si su funcionamiento es correcto o incorrecto. Sin embargo existen procesos, considerando al satélite como una caja negra, que no tienen manifestación física y que son de gran importancia para la misión espacial como: el contacto entre la Estación Terrena y el satélite, la finalización de la transmisión de información, o la correcta recepción de un comando de misión, éstos sólo son perceptibles dentro del canal de comunicación asignado para la transmisión de datos entre equipos del satélite, de ahí la necesidad de contar con un software que permita esta función.

SOFDEVO debe cumplir con requerimientos muy especiales: su interfaz debe ser suficientemente amigable para que cualquier persona pueda utilizarla, su información debe estar organizada de tal forma que sea fácil de localizar e interpretar, tiene que ser transparente para todos los equipos del satélite en vista de que su función es vigilar el trafico de la información que viaja a través de la red de comunicaciones interna del SATEX y, debe ser lo suficientemente robusto para permitir la validación de las operaciones del SATEX.

Las principales funciones y usos de SOFDEVO son las siguientes:

- ✓ Interceptar el tráfico de comunicaciones en la red interna del satélite para mostrar con textos de baja velocidad los procesos internos que se desarrollan de forma automática en el satélite.
- ✓ Facilitar los procesos de depuración y validación de los subsistemas del satélite que se han desarrollado en diversos sitios de México, en el sentido de que no sea necesario contar con la presencia de todos los equipos para validar los sistemas de alguna institución en particular.

El software tiene la capacidad de emular a cualquiera de las computadoras conectadas en red dentro del satélite, exceptuando a la CV. Cuando se emula a una computadora en particular, el software responde de forma automática a todas las peticiones que se le



solicitan vía red interna y también ofrece ayudas al usuario para dar respuestas manuales a eventos aislados.

La utilidad de este software radica en la alta capacidad que ofrece para visualizar sucesos que ocurren en el Microsatélite Experimental SATEX (MES) que de otra forma serían invisibles e imperceptibles, tal como el tráfico de comunicaciones en la red del satélite. De esta forma la operación de la instrumentación del satélite se torna observable, facilitando con ello la depuración del hardware y del software tanto del satélite como de la Estación Terrena.

Adicionalmente, SOFDEVO ha permitido la depuración paulatina del software asociado a los procesos de diagnóstico, detección y corrección de fallas de la arquitectura de cómputo semivirtual que se desarrolló como quinto experimento para el satélite. La arquitectura resuelve autónomamente los procesos de diagnóstico, detección y de corrección automatizada de fallas en la CV contando con el apoyo periódico de los procesadores de experimentos del satélite.

### 2.2.1 Estructura general de SOFDEVO

SOFDEVO está estructurado de tal forma que sea fácil y rápida la identificación de los procesos que éste emula y de las respuestas generadas, La Figura 2.1 muestra su pantalla principal, además de los elementos que la componen

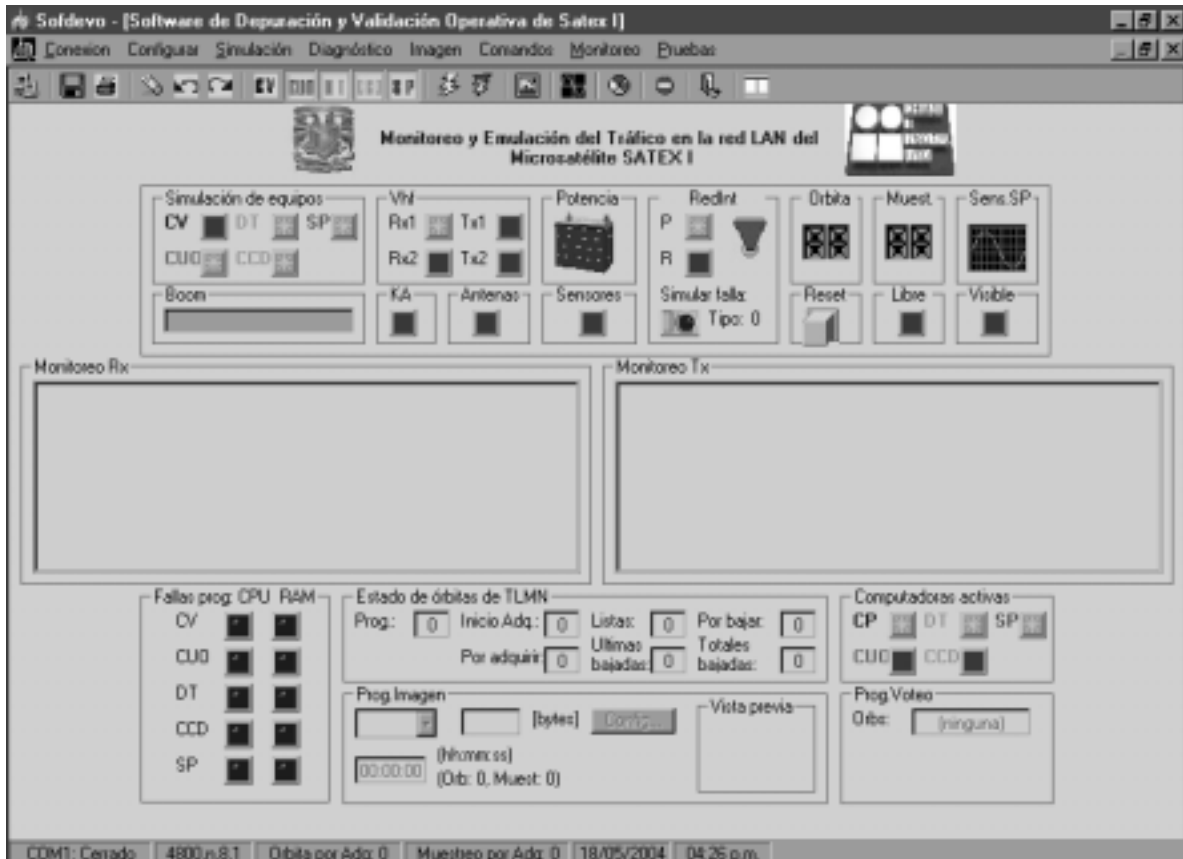


Figura 2.1 Pantalla principal de SOFDEVO



**Bloque de botones de acceso rápido.** Este bloque contiene elementos de acceso rápido a las principales funciones que realiza SOFDEVO asociados con un icono representativo de cada función, orden de izquierda a derecha se encuentran:

- Conectar. Abre el canal de comunicación por medio del puerto serial.
- Guardar. Almacena en un archivo de texto la información desplegada en las pantallas de monitoreo.
- Imprimir. Envía la información de las pantallas de monitoreo al dispositivo de impresión predeterminado de Windows.
- Borrar. Hace un respaldo de la información almacenada en las pantallas de monitoreo y después elimina su contenido de las pantallas de monitoreo.
- Deshacer. Recupera el texto eliminado.
- Rehacer. Elimina texto recuperado.
- Simulación CUO. Al presionar se emula o deja de emular el experimento satelital CUO.
- Simulación SIMS. Al presionar se emula o deja de emular al SIMS.
- Simulación CCD. Al presionar se emula o deja de emular a CDD.
- Simulación SP. Al presionar se emula o deja de emular al SP.
- Falla en diagnóstico de computadoras. Permite modificar el diagnóstico reportado por la computadora emulada (inserción de fallas).
- Falla en red interna. Permite simular fallas en la red interna, ya sea en el protocolo de comunicación o en la respuesta a comandos.
- Configurar imagen. Despliega una pantalla en la que se puede seleccionar la imagen que se utilizará para la emulación del experimento CDD.
- Señal. Permite configurar el tipo de señales que se generarán al emular los sensores del subsistema de potencia.
- Leyenda. Muestra el color asociado con alguno de los botones virtuales mostrados en SOFDEVO.
- Ayuda. Muestra información de la versión de SOFDEVO.
- Salir. Salida de la aplicación principal.
- Expandir. Permite modificar el tamaño de las pantallas de monitoreo para mejorar la claridad de la información mostrada.



Bloque de simulación de equipos

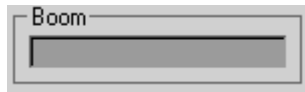
Está compuesto por cinco botones virtuales que representan a los equipos emulados por SOFDEVO. El color verde en el botón indica que el equipo está siendo emulado.



Bloque de VHF

Agrupada cuatro botones virtuales que señalan el estado (activo o inactivo) de los equipos de comunicaciones del SATEX. Un botón de color verde indica el equipo activo.

## CAPÍTULO 2. DESCRIPCIÓN DE EQUIPOS Y SUBSISTEMAS



**Bloque Boom**

Barra de progreso que representa visualmente el despliegue del gradiente gravitacional.



**Bloque Ka**

Un botón virtual que indica si se está energizado o no el experimento Ka.



**Bloque Sensores**

Botón virtual que indica si los sensores del sistema de potencia son emulados por SOFDEVO.



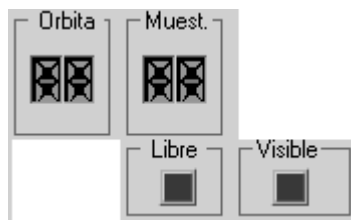
**Bloque Potencia**

Imagen que muestra la potencia emulada de las baterías. Permite modificar este valor para impedir el encendido de algún procesador durante un proceso de emulación.



**Bloque Red Interna**

Muestra qué red está actualmente en uso (principal o redundante) y permite la conmutación entre estas, además indica si se está simulando o no alguna falla en la red y de que tipo de falla se trata.



**Bloque Muestreo**

Cuenta con dos indicadores digitales para la orbita y el muestro actual, los cuales se obtienen de los datos de la ultima misión enviada. Adicionalmente cuenta con botones virtuales que indican si el satélite se encuentra visible para la Estación Terrena.



**Bloque Reset**

Permite reinicializar las variables que maneja SOFDEVO para le emulación de equipos, simulación de fallas, etc. Cada vez que se presiona este botón Visual Basic permite almacenar estos valores en el archivo \*.ini, que se genera con los valores predefinidos en el código de SOFDEVO.

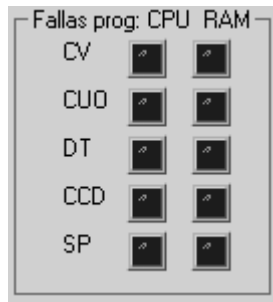


**Bloque Sensores de Potencia**

Este bloque permite configurar las señales asociadas con el sistema de potencia que son simulados por SOFDEVO; permite enviar señales triangulares, senoidales y cuadradas, definiendo su amplitud y frecuencia.

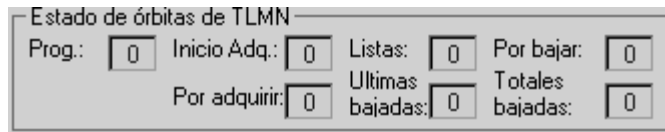


**Bloque de Monitoreo.** Contiene dos pantallas de monitoreo, una correspondiente a la información recibida por red interna y otra que muestra la información que se envía como respuesta a comandos.

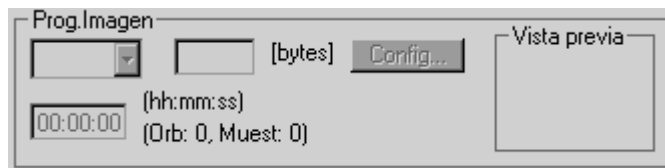


**Bloque Fallas programadas**

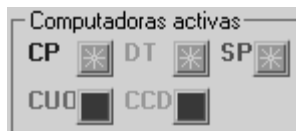
Está compuesto por 10 botones virtuales, a cada computadora de SATEX se le asocian dos botones, el primero al estado del CPU y el siguiente al estado de la memoria RAM. Permiten simular fallas en los resultados enviados como respuesta por SOFDEVO durante el proceso de MACV. El color rojo indica que se está simulando una falla en el equipo correspondiente a ese botón.



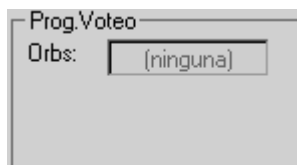
**Bloque de Estado de TLMN.** Contiene información sobre el número de órbitas programadas de telemetría normal, órbitas por adquirir, órbitas por bajar, órbitas listas para transferir a la ET y el numero total de orbitas descargadas. Esta información se actualiza automáticamente.



**Bloque de Programación de Imagen.** Permite configurar la imagen que se enviará a la CV al emularse el experimento de la CDD y muestra la información de la misión que solicitó la ejecución de este experimento.



**Bloque de Computadoras Activas.** Indica qué computadoras se encuentran encendidas con base en la información que proporciona el sistema de potencia del satélite.



**Bloque Programación de Votéo.** Indica el número de orbitas satelitales a las cuales se les programó votéo.



**Barra de Estado.** Contiene información sobre el estado del puerto de comunicaciones, configuración de comunicaciones (baudaje, puerto, paridad), número de orbitas y muestreos por adquirir, además de la hora y fecha del sistema

En el capítulo 3 de esta tesis se describe con mayor detalle el Software de Emulación, Depuración y Validación de Operaciones del SATEX.

### 2.3 Software de Estación Terrena del Proyecto SATEX

El software del satélite que realiza la comunicación con Tierra constituye el medio de control remoto a través del cual se le indican al satélite las tareas a realizar y se lleva a cabo su monitoreo. Su complemento, en Tierra, se encarga de enviar comandos específicos para la ejecución de tareas en el satélite como: la captura de imágenes, la adquisición de telemetría, el envío de la solicitud de transmisión de telemetría, etc. Además, permite el envío de parámetros a cualquiera de las cargas útiles mediante comandos.

La UNAM desarrolló el Software de Estación Terrena (SET) con una interfaz muy amigable y con una gran calidad de presentación que permite realizar la supervisión y el control del satélite; por lo cual todas las ayudas que pueda proveer, las alternativas operativas, la flexibilidad que tenga para interactuar con el satélite, así como su facilidad de uso y de despliegue de información, serán aspectos clave para aprovechar al máximo el vehículo espacial.

Cabe señalar que el desarrollo del software de ET se basó en el análisis de una gran cantidad de artículos presentados en congresos espaciales y de páginas web de proyectos microsatelitales y minisatelitales en todo el mundo. Con atención especial en la información reportada tanto por AMSAT —en particular su proyecto AMSAT fase 3D—, como por la Universidad de Surrey, a quienes públicamente se considera como los mejores desarrolladores de microsatelites en el mundo. Del análisis se observó que una gran cantidad de proyectos utilizan interfaces de software limitadas —en cuanto al despliegue de información— y que sólo algunos como los pertenecientes a las instituciones citadas anteriormente presentan interfaces atractivas, avanzadas y en ambiente Windows. Sin embargo, también se detectó que no utilizan técnicas virtuales para visualizar físicamente reportes de telemetría y que tampoco explotan la presentación de telemetría relacionada con el tiempo.

De este análisis y búsqueda de necesidades y oportunidades se decidió realizar innovaciones que aún no utilizan los sistemas que se consultaron, entre ellos la incorporación de instrumentación virtual para el despliegue de datos de telemetría mediante el uso de diagramas elaborados con “Autocad” y animados con “Flash”. En este sentido el SET ofrece información de telemetría mediante diagramas y animaciones de equipos — generados virtualmente— que facilitan la interpretación del estado operativo de los equipos del satélite. Adicionalmente, exhibe grupos de gráficas bidimensionales —o ampliaciones de cualquiera de ellas— para mostrar el comportamiento de los sensores respecto al tiempo.

Con estas ayudas, se persigue simplificar el proceso de diagnóstico y la interpretación del comportamiento del satélite.

También se consideraron algunos aspectos como la calidad de presentación, fomento a la cultura tecnológica y el acercamiento de este tipo de tecnologías a la gente joven de México, los cuales redundarán en el impacto del proyecto MES cuando se encuentre en operación.

Se debe subrayar que una parte muy importante del Software de Estación Terrena lo constituyen los protocolos y sus tramas de comunicaciones, los cuales son completamente compatibles con los utilizados por la CV del SATEX. Esos procedimientos y reglas se encuentran alojados en la etapas de software relacionadas tanto con la adquisición de telemetría e imagen, como en las secciones de envío de comando y misiones.

El SET admite el envío de nuevo software operativo para el satélite y también permite solicitar y capturar diversos tipos de telemetría (normal, especial, de magnetómetros, etc). También ofrece los resultados del diagnóstico de equipos del satélite, alertando al usuario, en casos de fallas, con alarmas auditivas y visuales, y permite programar también el tiempo de adquisición para capturar imágenes desde el satélite, así como su captura en la Estación Terrena.

Admite, de igual forma, la definición de valores límite para establecer los umbrales de alarma para cada sensor del satélite así como su almacenamiento en una base de datos. De este modo, durante la recepción de telemetría satelital y con ayuda de los datos de umbrales el Software de Estación Terrena hace posible una alerta visual dirigida al personal de Estación Terrena cuando alguno de los sensores sobrepasa sus límites operativos. Con ello se realiza una detección de fallas parcial durante la operación del sistema satelital.

Otro de los aspectos de interés del software radica en su capacidad para detectar automáticamente la presencia del satélite cuando éste aparece en el horizonte de la Estación Terrena. Una vez detectado permite elegir la forma en que se enviará una nueva misión al satélite, la cual puede enviarse de forma manual o de forma automática. Este último aspecto operativo será de gran utilidad cuando el satélite sobrevuele la Estación Terrena de madrugada, con lo cual será posible transferir una nueva misión de trabajo y la recepción de telemetría sin la presencia de personal de control satelital.

Adicionalmente permite interactuar con las cargas útiles del satélite ya sea para transmitirles parámetros o bien, para capturar información (datos operativos, imágenes, etcétera).

El personal que controlará el microsatélite utilizará el SET para enviar una serie de comandos para especificar las tareas que el microsatélite deba realizar, estos comandos se catalogan como comandos de línea y comandos de misión. Cada categoría tiene características propias en lo que se refiere a las actividades que se generan cuando éstos llegan al microsatélite.

La información que SATEX envía a Tierra se cataloga en:

- ✓ Información de Telemetría Normal.
- ✓ Información de Telemetría Especial.
- ✓ Información del Estado de los equipos.
- ✓ Información del Estado de los 5 microprocesadores instalados en el SATEX.
- ✓ Imágenes.
- ✓ Información del experimento de Mantenimiento Automatizado de la CV.

Para cada una de estas funciones, se tiene diseñado e implantado un protocolo de comunicación dedicado. Esto se verá más a detalle en el capítulo 4.

### 2.3.1 Estructura general del software de Estación Terrena

El SET se ha diseñado para cubrir las necesidades principales de las comunicaciones Tierra-Microsatélite, este software no es una versión final ya que se continúan realizando modificaciones, sin embargo su porcentaje de avance es del orden del 99 por ciento, figura 2.2.

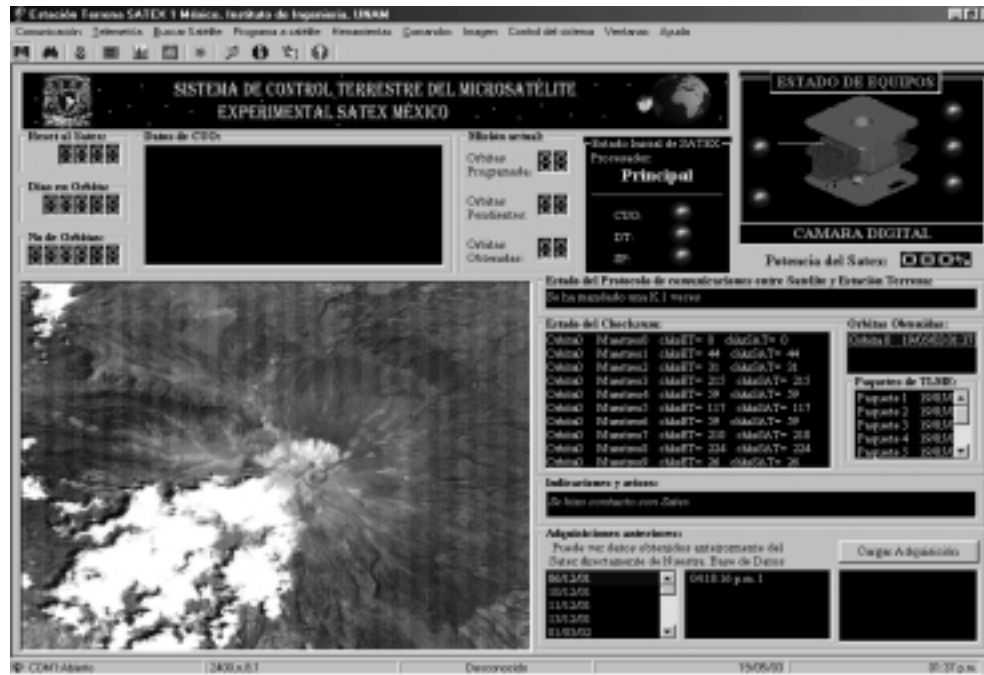
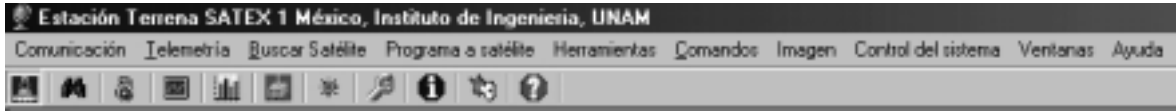


Figura 2.2 Pantalla Principal del Software de Estación Terrena.

A continuación se describen a grandes rasgos el contenido y el propósito de cada una de las secciones de despliegue del SET.



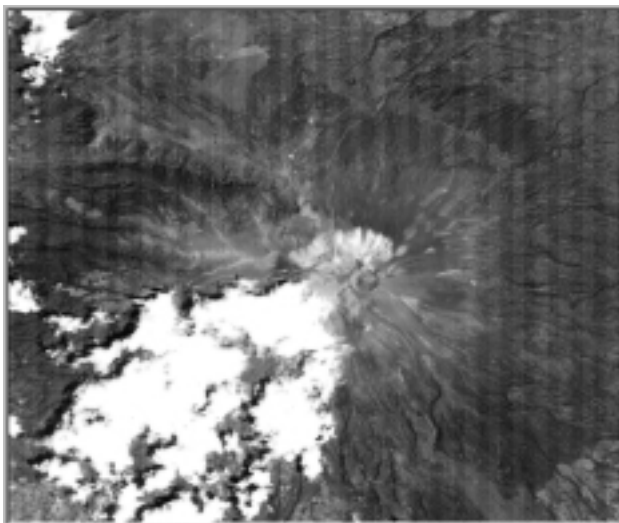
Desde la barra de menús se pueden ejecutar todas las funciones principales del software, debajo de ésta se tiene una barra adicional de botones de acceso rápido a los controles más utilizados para observar parámetros del SATEX.



Presenta el número de reconfiguraciones efectuadas en la computadora de vuelo, el dato se actualiza cada vez que se reciba telemetría del satélite. Adicionalmente muestra el número acumulado de órbitas que ha efectuado el SATEX así como el conteo de los días que lleva en el espacio.



Aquí se observarán resultados de petición de información exclusivamente para el experimento óptico (Carga Útil Óptica).

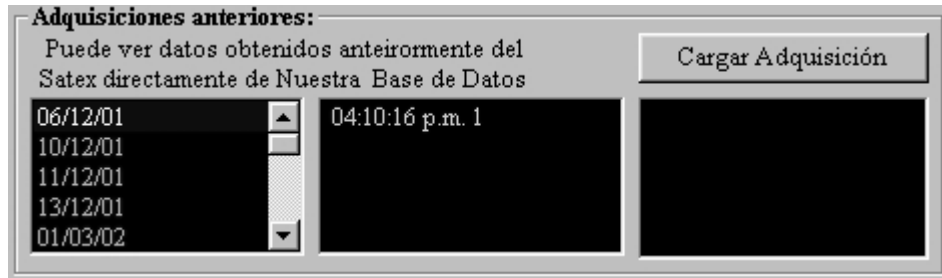


En esta ventana se desplegará una vista de la imagen adquirida por el satélite. La imagen se almacenará en la base de datos para consultas posteriores.

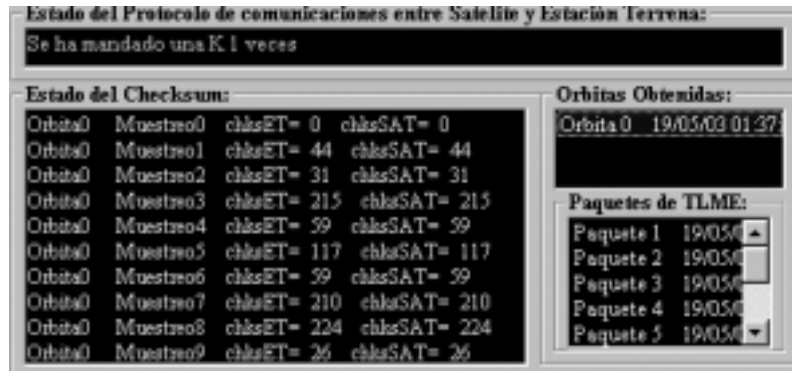




Cada vez que se capture nueva telemetría normal y se hayan producido fallas en el satélite durante algún muestreo, se indicará visualmente la anomalía con esferas parpadeantes. Al presionar con el “ratón” sobre la ventana se obtendrán más detalles sobre la falla en una página adicional.

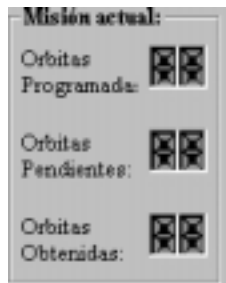


En este lugar pueden recuperar las adquisiciones de Telemetría Normal previamente reportadas por el satélite para revisarla cuidadosamente y evaluar el funcionamiento del SATEX en tal misión. Esta sección permite acceder los datos de las misiones almacenadas en la base de datos con la fecha y la hora.



En esta pantalla se observa el protocolo de comunicaciones durante la adquisición de información del microsatélite (telemetría especial, telemetría normal o imagen). Se pueden verificar los “checksum” que envió el SATEX y compararlos con los calculados por Tierra. En el caso de la adquisición de TLME se puede contabilizar el número de paquetes de información recibida; el total de paquetes es de diez y cada uno contiene 4000 bytes que transportan datos de sensores específicos. Adicionalmente muestra la recepción de la imagen byte por byte indicando el tamaño de la imagen y el progreso de la recepción.

## CAPÍTULO 2. DESCRIPCIÓN DE EQUIPOS Y SUBSISTEMAS



Muestra el estado de las órbitas adquiridas, pendientes y programadas, para tener una idea clara de la misión actual antes de enviar nuevas tareas al satélite.

Cabe mencionar que el SET permitirá enviar nuevos programas a la CV por lo que aún cuando el satélite se encuentre en órbita se le podrá sustituir completamente su software de operaciones. En la figura 2.3 se presenta la pantalla de interfaz para subir programa a la CV. El código de la CV se genera en lenguaje “C”, el cual se compila, se ensambla y liga para generar archivos de diferentes extensiones. De ellos el que tiene la extensión .hex (formato hexadecimal) es el que se transmite a la CV. Para este propósito se utiliza la opción de envío de nuevos programas a la CV que ofrece el SET, desde el cual se carga el archivo .hex para transmitirlo al satélite, [Ortiz, 2003].



Figura 2.3. Ventana para el envío de nuevo software a la CV.

## 2.4 Computadora de Vuelo

La Computadora de Vuelo es uno de los equipos más importantes en el microsátélite ya que comandará todas sus operaciones.

La CV cuenta con tres tarjetas de procesamiento, una principal (CP) y dos redundantes, para respaldo en frío (CR0 y CR1). Cada tarjeta de procesamiento contiene un sensor de efecto “latch-up”, para proteger a cada uno de los microprocesadores, los cuales no son de tipo militar. Adicionalmente, cada una de las tarjetas de procesamiento de la CV cuenta con memoria RAM externa y expandida, destinadas principalmente para la carga de nuevos programas.



Figura 2.4. Módulo de la Computadora de Vuelo

Cada uno de los circuitos impresos de la CV está ensamblado con componentes de tipo militar y de montaje superficial para incrementar las posibilidades de superar las pruebas de certificación, lanzamiento y operación orbital. Cabe mencionar que las pruebas de certificación sólo cubren los aspectos mecánicos, térmicos y de vacío, en cuanto a los equipos, estos quedarán certificados al momento de orbitarlos en el espacio en función del éxito o fracaso de la misión.

Adicionalmente, la CV contiene electrónica para dos redes locales de comunicación entre las computadoras del satélite, esto incrementará sus posibilidades de éxito. Cuenta con 64K bytes de memoria PROM, donde se grabará una versión mínima del software de vuelo de tal forma que permita al satélite iniciar operaciones de captura de telemetría normal, realización de prueba en Kourou y comunicación con el software de Estación Terrena. También cuenta con 256K bytes de memorias RAM para el cargado de nuevos programas desde Tierra, así como 1M Bytes de memoria RAM expandida para el almacenamiento de telemetría e imágenes. Toda la memoria está protegida por un dispositivo EDAC (Error Detection And Correction Unit) para la detección y corrección de errores simples.

La CV es un sistema modular cuya versión final está constituida por 6 tarjetas. El hardware lo conforman tres computadoras (principal y dos redundantes), la electrónica de conmutación de computadoras, el hardware de red, la electrónica de acondicionamiento de señales provenientes de sensores, la electrónica de protección contra la radiación (efecto “latch-up”), manejadores de línea y conectores que agrupan a las señales de entrada y salida. Cada uno de estos módulos se encuentran físicamente ubicados en circuitos impresos separados los cuales se conectan por medio de conectores de costilla, uno en cada extremo.

El software de la CV, fue escrito en lenguaje C lo cual redujo problemas de desarrollo. No obstante, el lenguaje utilizado no cumple con todas las características del ANSI C, pues en realidad constituye un sistema adaptado a las funciones del SAB80166.

La CV de SATEX contiene:

- ✓ Tres tarjetas de procesamiento, cada una con un microcontrolador tipo RISC de 16 bits, con una velocidad de procesamiento de 40 MHz, 1.280M Bytes de memorias RAM y 64K Bytes de ROM. La RAM cuenta con protección EDAC y tiene electrónica de protección de efecto “latch-up” para el microprocesador, que es el único componente que no cumple con la norma militar MIL-STD-883.
- ✓ Un módulo de conmutación de estado sólido que permite transferir toda la instrumentación del satélite (sensores, actuadores, canales de comunicaciones y líneas de entrada/salida) hacia cualquiera de las tres tarjetas de procesamiento.
- ✓ Un módulo de multicanalización que permite adquirir hasta 48 señales eléctricas provenientes de sensores.
- ✓ Un módulo de filtrado y recorte para señales de sensores y la electrónica para la red de área local del satélite, tanto principal como redundante.
- ✓ Manejadores de línea para algunas de las señales digitales de entrada y salida.
- ✓ Contenedor de aluminio espacial que permite el armado y desarmado de la computadora por capas.

- ✓ Adicionalmente, la computadora de vuelo se complementa con un módulo externo, en el cual se aloja la electrónica de acondicionamiento para señales de sensores de temperatura, magnetómetros, corriente y voltajes consumidos por diversos equipos del satélite, figura 2.5.



Figura 2.5. Módulo de Electrónica de sensores

## 2.5 Software de Operación Satelital

El software operativo de la CV es una parte esencial del esquema de tolerancia a fallas propuesto para SATEX. Adicionalmente integra software dedicado para conformar experimentalmente la arquitectura de cómputo semivirtual del satélite, cuyo propósito principal es realizar mantenimiento automatizado a la CV. La arquitectura realiza la ejecución de voto entre los procesadores a bordo, la reconfiguración automática en caso de fallas en tarjetas de procesamiento, la conmutación automática de canales de comunicación ya sea en la red interna o en la comunicación con Tierra en el caso de detectar alguna falla, etc. [VICENTE, 2001].

El Software operativo de vuelo, figura 2.6, fue desarrollado en lenguaje C ANSI y con algunas utilerías propias del microcontrolador SIEMENS SAB80C166. El compilador usado fue el “TASKING Cross C Compiler Ver. 3.5.6” [TASKING, 1993] de la compañía TASKING [www.tasking.com], dicho compilador proporciona las extensiones de lenguaje necesarias para el control de puertos, periféricos y registros del microcontrolador.

```

MINIMO1.C
*****SOFTWARE MINIMO C/ nuevos comandos*****
***** programación para EDAC*****
***** comando CUI*****
PROGRAMA PRINCIPAL
*****
***** en Satellite cmd[0] == cmd[1] en ET and so on *****
#include <stdio.h>
#include <i166.h>
#include <reg166.h>
#include <float.h>
*****
DECLARACION DE VARIABLES GLOBALES
*****
/* LOCALIZACION DE VARIABLES */
#pragma combine nb=a64576
int fran gral[6]; /*Seccion absoluta: 64576 =0xFC40*/
/* direccion FC40 para uso con opcion PECC0 del C A/D*/
/* verificar direccion con rmbesa.map o .lst */
#pragma combine nb=a64588
int fran paso_pec; /*Seccion absoluta: 64588 =0xFC4C*/
/* direccion FC40 para uso con opcion PECC0 del C A/D*/
/* verificar direccion con rmbesa.map o .lst */
*****Arreglos para el muestreo de RAM*****
/*Se especifican arreglos para analizar el contenido de las
100 localidades de la Ram interna del procesador*/
#pragma combine nb=a64590 /*64590 =0xFC4F*/
int fran MuestreoRam[101]; /*[64590-64609] o [0xFC4F-0xFC61]*/
#pragma combine nb=a64610 /*64610 =0xFC62*/
int fran MuestreoRam1[101]; /*[64610-64629] o [0xFC62-0xFC75]*/
#pragma combine nb=a64630 /*64630 =0xFC76*/

```

Figura 2.6. Detalle del software operativo de SATEX en lenguaje C.

No obstante, el lenguaje utilizado no cumple con todas las características del ANSI C, pues ha sido en realidad adaptado a las funciones del 80166.

Las labores que automatiza de la CV del MES son:

- ✓ Tareas que realiza el satélite de forma automática una vez que sea liberado.
- ✓ Recopilación y formateo de telemetría.
- ✓ Transmisión de telemetría a Tierra.
- ✓ Atención a las ordenes que se envíen desde la Estación Terrena en forma de comandos, misiones o programas.
- ✓ Interacción con las cargas útiles para efectuar los experimentos que se ordenen desde la Estación Terrena.
- ✓ La estabilización del satélite.

Para este propósito la UNAM desarrolló en su totalidad la programación requerida para las operaciones del satélite. Debe subrayarse que a su vez cada experimento satelital que contienen su propia computadora requiere de software dedicado, el cual fue desarrollado por la institución responsable del experimento.

El software de vuelo del microsátélite está vinculado fuertemente con el software de la Estación Terrena, pues la mayor parte de las acciones que realiza el satélite son especificadas por el Software de Estación Terrena. Todo el funcionamiento del software operativo de la CV, su interacción con todas las computadoras y equipos que integran al microsátélite se validaron en el laboratorio.

En el capítulo 4 se verá a detalle el desarrollo del Software de Operación Satelital del Microsátélite Experimental SATEX.

### **2.6 Emulador ROM/RAM TRE-200**

El Emulador de memoria ROM/RAM es un sistema basado en una PC que evita la necesidad de quemar y borrar EPROMs para afinar el código por instalar en la CV. El software se puede ejecutar desde un disquete extraíble o del disco duro para la emulación.

El TRE-200 puede emular gran cantidad de memorias, entre ellas:

- ✓ -SRAM: 6264 62256
- ✓ -EPROM: 27 (c) 64 27 (c) 128 27 (c) 256 27 (c) 512  
27 (c) 010 (opcional)  
27 (c) 1024 (opcional)

La emulación se apoya de dos puertos independientes de 8 bits: un puerto para emular todos los bytes pares y el otro puerto para emular los bytes impares de un programa. El emulador tiene un tiempo de acceso estándar de aproximadamente 120 ns. Sustituyendo la unidad RAM con la unidad más rápida (70ns), el tiempo de acceso puede ser reducido a aproximadamente a 90ns.

#### **2.6.1 Modo de empleo del TRE-200**

En esta sección se describirá en forma específica el modo de empleo del Emulador TRE-200 como referencia a los interesados.

Una vez instalado el software TRE-200, se ejecuta desde modo MS-DOS de la PC el archivo EML512.EXE especificando la ruta y el directorio en donde esté ubicado, si el módulo de emulación no es detectado por la PC aparecerá el siguiente mensaje:

*Error Identification on hardware!  
Press <Q> to Quit  
Or press <CR> to continue*

Si el módulo es detectado, aparecerá la ventana de la figura 2.7.

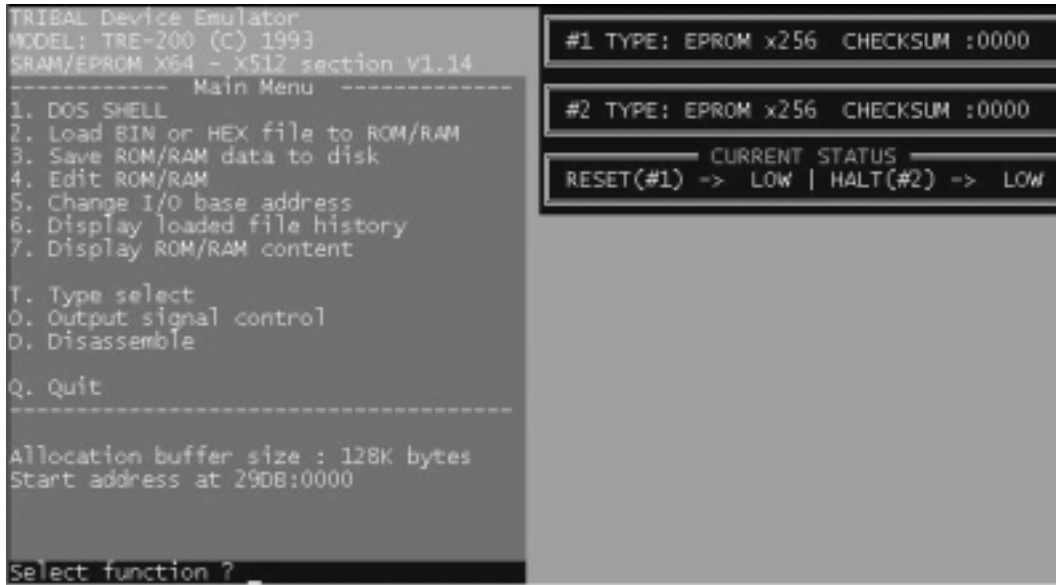


Figura 2.7. Pantalla principal del Emulador ROM/RAM TRE-200

### Paso 1. Seleccionar la memoria por emular

Oprimiendo la tecla **T** se ingresa a la función **Type Select**, donde se seleccionará el tipo de dispositivo a emular en cada puerto.

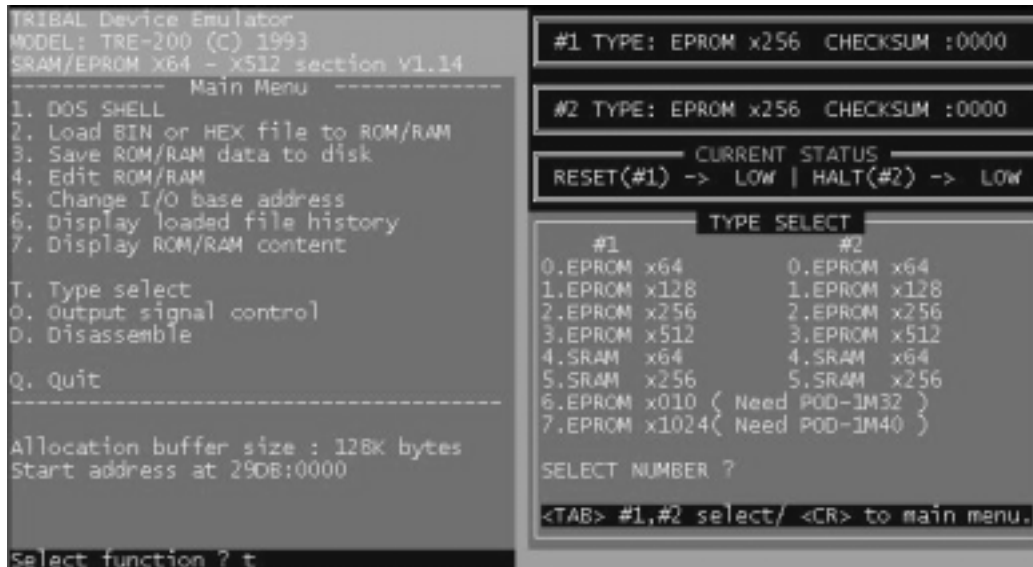


Figura 2.8. Ventana que muestra los tipos de memorias que puede emular

Usando la tecla <TAB> se cambia el cursor entre los puertos #1 y #2 ROM/RAM.

Se selecciona el número al cual pertenece el dispositivo por emular. Las opciones 6 y 7 sólo pueden usarse si se tienen los dispositivos adaptadores para la PC. Para emular un dispositivo 27(C)010 se necesita el adaptador de 32 pines POD-1M32. Para emular un dispositivo 27(C)1024 se requiere del adaptador de 40 pines POD-1M40.

En nuestro caso, se elige el número 2, EPROM x 256, para cada puerto #1 y #2 ROM/RAM y se presiona ESC para regresar al menú principal.

## Paso 2. Cargar el archivo fuente

Oprimiendo la tecla 2 se accesa a la función **Load BIN or HEX file to ROM/RAM**. Esta función se usa para cargar al emulador un archivo binario o un archivo hexadecimal, del disco duro o de un disco flexible. Después de entrar en esta función el cursor se posicionará en el mensaje **Ready to LOAD (1/2/W/CR>)?** Se selecciona '1' para cargar un archivo al puerto #1 ROM/RAM, '2' para cargar un archivo al puerto #2 ROM/RAM, o 'W' para cargar todos los bytes pares del archivo al puerto #1 ROM/RAM y los bytes impares de un archivo al puerto #2 ROM/RAM.



Figura 2.9. Ventana que describe el cargado de los archivos fuente al emulador

Ahora el cursor aparece en una línea para insertar el nombre del archivo con alguno de los siguientes procedimientos:

- A) Insertar el nombre del archivo completo por cargar, si el archivo no está en el mismo directorio donde se encuentra el archivo EML512.EXE, se debe especificar la ruta desde el directorio raíz.
- B) Presionar la tecla <TAB> para cambiar el control a la ventana a la izquierda:
  - ✓ <TAB>: para cambiar la ventana activa a los directorios raíz de la PC.
  - ✓ <UP ARROW> y <DOWN ARROW> para localizar el directorio y subdirectorios en donde se encuentra el archivo.
  - ✓ <CR o Enter>: para seleccionar el archivo por cargar al emulador.
  - ✓ <ESC>: Para salir del submenú y regresar al menú principal.



Una vez elegido el archivo que se trasladará al emulador, el cursor se colocará en la línea **<B>in, <I>ntel HEX, <M>otorola S HEX:**, se deberá seleccionar 'B' si el archivo que se quiere cargar es binario, 'I' si el archivo es un archivo Intel HEX o 'M' si el archivo está en uno de los formatos de Motorola S. Para SATEX se elige la opción <I>ntel HEX, ya que el compilador puede proporcionar este tipo de archivo.

Si se elige el formato <I>ntel HEX el cursor se colocará en la línea **File start sef. (0000):** \_, aquí se indica la dirección del segmento del inicio del archivo. Todos los datos que quedan entre el segmento de la salida y el fin del archivo se convertirán a binario y se cargarán a la memoria del emulador. El rango del segmento es de 0 a F000

Si una de las opciones <I> o <M> fueron seleccionadas el cursor apuntará en la línea **Unused bytes will be <1> 00, <2>, FF <3> Don't care: \_ ?**, donde se especificará al emulador cómo representar los bytes no usados en la emulación. En una aplicación típica cuando se carga un archivo HEX al emulador, sólo una parte de la memoria disponible del EPROM está realmente llena de programa o datos. El resto de las partes no usadas del EPROM puede ser llenada con los datos indicados por el usuario. Para ello se debe elegir las opciones '1' o '2'

Si se desea cargar más de un archivo HEX a un EPROM se debe escoger la opción '3'. Esto significa que las secciones del EPROM no usadas por el primer archivo HEX cargado al emulador puede llenarse con otros datos cualesquiera o bien, con otro archivo HEX.

## **2.7 Programador Universal UPT-400**

El Programador Universal UPT-400 es una herramienta que permite grabar PROMs, EPROMs, EEPROMs, BPROMs y MPUs en diferentes formatos ya sea BIN, HEX o Motorola.

El UPT-400 es muy amigable y se controla mediante un ordenador personal con MS-DOS o PC-DOS.

Para hacer uso del UPT-400 se debe tener lo siguiente:

- ✓ PC/XT/AT/386/486 u ordenador personal compatible.
- ✓ Una velocidad del reloj de sistema máxima de 50 MHz.
- ✓ Un mínimo de 640K byte de memoria.
- ✓ Sistema operativo: MS-DOS O DOS-PC, versión 2.0.
- ✓ Tarjeta de Adaptador de Sistema para PC.
- ✓ Un cable **1-M** (cable con una numeración de pines especial), con terminaciones de 25 pines y un conector tipo **D** al final del cable.
- ✓ Programador UPT-400.

### 2.7.1 Modo de empleo del UPT-400.

En esta sección describiré el modo de empleo del Programador Universal UPT-400. Como referencia a los interesados.

Una vez instalado el software UPT-400, se ejecuta desde modo MS-DOS de la PC, el archivo ACCESS especificando la ruta y el directorio en donde esté ubicado y aparecerá la siguiente ventana de la figura 2.10.



Figura 2.10. Ventana principal del Universal Programmer, Emulator & Tester.

Con ayuda del teclado se puede desplazar por el menú y seleccionar las diferentes funciones. Colocando el cursor en la función de “Programmer”, figura 2.10, se puede ver que el UPT-400 programa EPROMs, EEPROMs, Serial PROMs, BEPROMs, MPUs y PLDs.



Figura 2.11. Tipos de familias de EPROM y compañías que las elaboran.

Entonces se selecciona el tipo de memoria a programar y después el tipo de familia de la memoria por programar para que el UPT-400 tome las especificaciones de la memoria y así determine los pines en los que debe insertar datos y voltaje de programación, así como la capacidad de la memoria.

Una vez seleccionadas las opciones, si el UPT-400 no estuviese energizado, aparecerá el siguiente aviso:

*Error Identification on hardware;  
Press <Q> to Quit  
Or press <CR> to continue*

Una vez que se detecta el módulo de programación, aparece el menú de la figura 2.12 para solicitar direcciones de memoria, tanto en la PC como de la EPROM en donde se almacenará el código fuente.

```

TRIBAL Device Programmer
MODEL: TUP-400/300 (C) 1992
EPROM 512 section V3.33
-----
Main Menu
-----
1. DOS SHELL
2. Load BIN or HEX file to buffer
3. Save buffer to disk
4. Edit buffer
5. Change I/O base address
6. Display loaded file history
7. Display buffer

T. Type select
S. Program speed, algo
Z. Target zone
B. Blank check
P. Program
R. Read
C. Compare & display error
Q. Quit

M. Mfr. select
D. Display
A. Auto(B&P&V)
V. Verify

-----
Allocation buffer size : 256k bytes
Starting address at 5F8A:0000
Select function ?

*Mfr.:NS *BLANK BIT: 1
*TYPE:NM27C256/LC256 *PGMSPEED:NS
*vVPP :12.5V *VCP :6.25V

TARGET ZONE
Buffer start addr.: 00000
end addr.: 07FFF
Check Sum : 0000
Device start addr.: 0000
COUNTER
0000
    
```

Figura 2.12. Menú de funciones del UPT-400

Con el menú de la figura 2.12 se accede a diversas opciones del UPT-400, algunas de uso sencillo como: verificación de que la memoria se encuentre vacía (opción **V**); elección del algoritmo de velocidad para programar la memoria (opción **S**), de las cuales se recomienda usar velocidades bajas para EPROMs que se han utilizado bastante; etcétera. En seguida se describen opciones que merecen explicaciones adicionales.

### 2.7.1.1 **Modificación del Buffer (Target Zone) usado por el UPT-400.**

Presionando la tecla **Z** se selecciona la función para modificar el buffer y aparecerá la ventana de diálogo de la figura 2.13.

La dirección del “Buffer” se refiere al área de memoria en la PC que será usada para cargar cualquier archivo de datos indicado por el usuario. En tanto que la dirección de inicio del “device” se refiere a la dirección de la EPROM a partir de la cual se iniciará la programación de datos con ayuda del UPT-400. Los valores declarados deben ser hexadecimales.



Figura 2.14. Ventana para la modificación del Buffer.

### 2.7.1.2 Cargar el archivo fuente en el buffer

Presionando la tecla 2, se selecciona la función **Load BIN File to Buffer** que permite cargar el archivo deseado al buffer, con ello se despliega la pantalla de la figura 2.15.

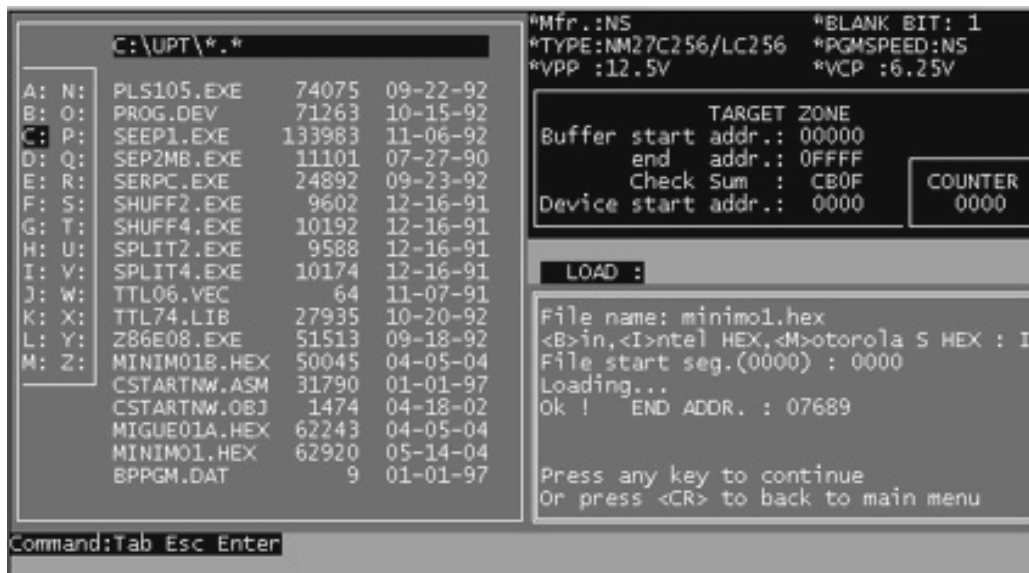


Figura 2.15. Ventana para la carga del archivo fuente en el Buffer

En la línea **File name:** se escribirá el nombre del archivo incluyendo su extensión y además la ruta de dónde se encuentra, luego se presiona <CR>.

Un modo alternativo para indicar el nombre de archivo, es seleccionar el archivo deseado de la ventana de directorio usando los pasos siguientes:

- ✓ Presionando la tecla <TAB> hasta cambiar la ventana activa en directorios o archivos, presionando las teclas <UP> o <DOWN> para moverse en la barra hasta el archivo deseado o subdirectorio.
- ✓ Presionar <ENTER> para seleccionar el archivo deseado o el subdirectorio.

Si se elige un subdirectorio, la ventana de directorio mostrará todos sus archivos para la siguiente selección.

A continuación, el submenú pedirá el formato del archivo fuente que desea cargar en el Buffer el usuario, los tipos de formatos pueden ser: <B>inario, <I>ntel HEX o <M>otorola S HEX. Además, se deberá indicar el lugar en el que inicia el código fuente del archivo deseado. En algunos casos, el inicio del archivo se encuentra en una localidad de memoria distinta que la 0000, es por ello que el UPT-400 pide le indiquen en dónde inicia el archivo cargado.

Para SATEX, el formato de los archivos fuente es Intel HEX, así que la opción es <I>ntel HEX. Además el inicio del programa es desde la localidad 0000, es decir, el programa empieza desde la localidad cero.

### 2.7.1.3 Programación de memoria EPROM

Presionando la tecla P se accederá al submenú mostrado en la figura 2.16.

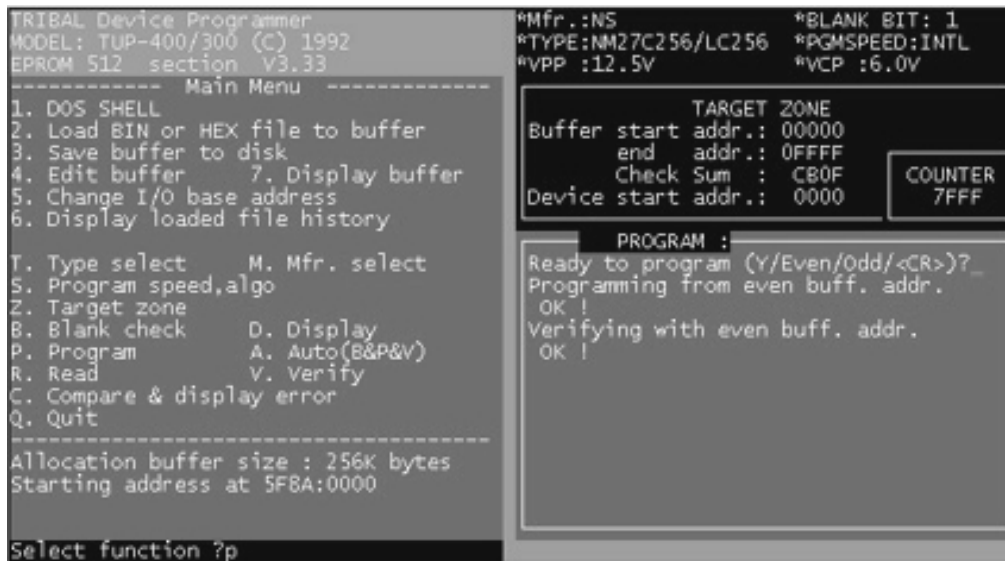


Figura 2.16. Ventana para la programación de los bytes impares.

Se elige la opción **E**, figura 2.16, para comenzar a transferir datos impares del buffer de la PC al EPROM, o se presiona <CR> para volver al menú principal. Cuando finaliza la transferencia, aparece el mensaje *Programa OK!* El mensaje será mostrado sobre la pantalla, además, al final verifica que los datos programados concuerden con los almacenados en el buffer de la PC.



Figura 2.17. Ventana para la programación de los bytes pares.

Se elige la opción **O**, figura 2.17, para comenzar a transferir datos pares del buffer de la PC al EPROM, o se presiona <CR> para volver al menú principal. Cuando finaliza la transferencia, aparece el mensaje *Programa OK!* El mensaje será mostrado sobre la pantalla, además, al final verifica que los datos programados concuerden con los almacenados en el buffer de la PC.

## 2.8 Simulador de satélite (SIMSAT)



Figura 2.18. El simulador de Satélite en el laboratorio.

Cuando el SIMSAT se utiliza en combinación con el software de operaciones del MES, permite visualizar la operación de actuadores y equipos del satélite, entre ellos la liberación de satélite, equipos VHF de comunicaciones, despliegue de antenas, bobinas de torque magnético, gradiente gravitacional, etcétera, y permite además la simulación física

## **CAPÍTULO 2. DESCRIPCIÓN DE EQUIPOS Y SUBSISTEMAS**

de los 48 sensores asociados a la CV. El equipo fue desarrollado para permitir pruebas del software de la CV tanto a nivel tarjeta de procesador, como a nivel de la CV. Para tal propósito, cuenta con conectores para comunicarse ya sea con una tarjeta de procesador por medio de un conector de costilla de 50 contactos, o bien, para comunicarse con la CV por medio de conectores militares tipo DBX de diferentes capacidades.

Para pruebas a nivel de tarjeta de procesamiento contiene el hardware de red, de tal forma que por medio de él se pueden interconectar hasta 4 computadoras personales cada una ejecutando el software SOFDEVO. De esta forma ayudó a validar el software de operaciones del satélite a través de la emulación de las computadoras de experimentos, como si ya estuviese integrado el satélite con todos sus equipos.

Por sus características, el SIMSAT se utilizó para depurar el hardware y el software de operaciones del satélite.

## Capítulo 3

### *Software de Operaciones del Microsatélite*

---

#### **3.1 Introducción**

En este capítulo se describen las características del Software Operativo del microsatélite con base en diagramas de flujo, separando claramente dos versiones operativas. La primera por instalarse en memorias PROM, desde las cuales se ejecutará siempre software de inicio de vida y de operaciones fundamentales del vehículo orbital, al que se refiere como software mínimo (SOFMIN). Y la segunda que incorpora el SOFMIN y el software de operaciones de SATEX.

La programación desarrollada para se puede dividir en tres módulos principales, la primera es la relacionada con las tareas de control y automatización de funciones a bordo, la segunda referente al control de comunicaciones de la red interna de SATEX y la última relacionada con el software de comunicaciones con la Estación Terrena [VICENTE, 1998].

SOFMIN se ejecutará cada vez que se aplique una inicialización, o bien, una reconfiguración de procesadores en la Computadora de Vuelo. Este software contiene funciones para: realizar la adquisición y el empaquetamiento de telemetría normal; las comunicaciones con Tierra, esto es, tareas de adquisición de comando y misión así como de transporte de telemetría; prueba en sitio de lanzamiento y liberación de antenas.

El software de la Computadora de Vuelo coordina las funciones de automatización asignadas al satélite, como el encendido o apagado de los equipos de radiofrecuencia, cargas útiles, liberación del gradiente gravitacional, además de otras funciones como la adquisición de datos de sensores (telemetría), coordinación de procesos de diagnóstico y detección de fallas en las diferentes computadoras del satélite y comunicaciones con Tierra, todo ello a base de comandos y protocolos de red interna diseñados para el SATEX. Por medio de la Estación Terrena se reciben comandos en línea y misiones que especifican al satélite la ejecución de tareas a realizar en tiempos específicos, la adquisición de datos de telemetría normal, telemetría especial y la adquisición de imágenes.

El software es una parte esencial del esquema de tolerancia a fallas propuesto para SATEX. Adicionalmente integra software dedicado para conformar, experimentalmente, la arquitectura de cómputo semivirtual a bordo de SATEX, cuyo propósito principal es realizar el mantenimiento automatizado a la Computadora de Vuelo. La arquitectura realiza la ejecución del voto democrático entre los procesadores a bordo, la reconfiguración automática en caso de errores no corregibles en la memoria RAM, la conmutación automática de canales de comunicación ya sea en la red interna o en la comunicación con Tierra en el caso de detectar alguna falla, etc. [VICENTE, 2001].



### 3.2 Protocolo de comunicación de la Red Interna

El software de comunicación entre las computadoras de SATEX integra un protocolo de red que se encuentra distribuido en cada una de las computadoras. Cada computadora contiene software de recepción y de transmisión de comandos así como de detección automática del canal en uso (principal o redundante).

El hardware de comunicaciones en la red interna constituye el medio de comunicación entre las computadoras del satélite. Esta red es de tipo “difusión” en la que “todos hablan y todos escuchan”. El tipo de comunicación es serial síncrona según el estándar IEEE RS232C ( $\pm 12V$ ), debido a que los microcontroladores utilizados tienen puertos nativos para este protocolo.

Como se comentó, el proceso de comunicación entre computadoras es un proceso síncrono, debido a que la computadora de vuelo coordina el envío y la recepción de los comandos que circulan a través de la red. Por lo tanto, todas las tareas ejecutadas por medio de comandos de red interna se inician con el envío de un comando desde la computadora de vuelo hacia alguna o hacia todas las computadoras.

La trama definida para los comandos de red interna se muestra en la figura 3.1.

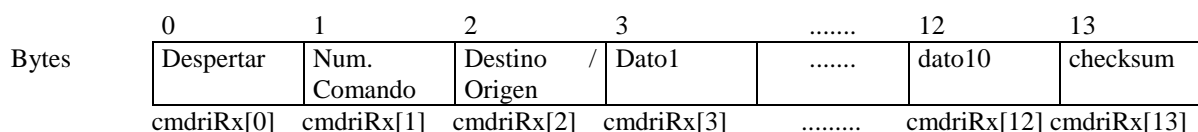


Figura 3.1. Trama de comandos por Red Interna.

La trama consta de 14 bytes numerados de 0 a 13, donde los primeros tres representan el encabezado del comando, los siguientes 10 constituyen la información del comando y el último byte se utiliza para el valor del checksum.

Tabla 3.1. Valores que definen al byte Destino/Origen

Valor Destino/Origen	Computadora que representa
00h	CV
01h	CUO
02h	DT
03h	CCD
04h	SP
05h	SOFDEVO
06h	Todas menos DT
07h	Todas (Broadcasting)

Los valores que definen el comando son los siguientes:

Despertar	2Fh	Indica que el comando fue enviado por la CV
	0Fh	Indica que el comando fue enviado por una carga útil
Num. Comando		Valor hexadecimal que representa al comando enviado
Destino/Origen		Valor hexadecimal dividido a su vez en dos valores de cuatro bits cada uno, los cuatro bits menos significativos indican la computadora que originó el comando y los cuatro bits más significativos indican la computadora a quien va dirigido el comando. Los valores que pueden tomar se muestran en la tabla 3.1.

De esta forma, resulta posible que cada una de las computadoras puedan enviar un comando a una computadora en específico o bien a todas las computadoras pues se tiene una red de difusión, no obstante, al realizar la decodificación del comando, cada computadora detecta si le corresponde o no la ejecución del mismo.

Cuando la computadora receptora identifica el byte destino/origen del comando, procede a calcular el checksum de la trama recibida. Una vez calculado, debe compararlo con el último byte de la trama. En caso de que el checksum calculado sea diferente al checksum de la trama, la computadora debe notificarlo a la computadora origen. En caso contrario también se le informa la recepción correcta del comando. La confirmación de que el comando llegó de manera correcta o incorrecta se realiza enviando por el mismo canal un carácter “K” de reconocimiento (acknowledge) o una “N” (no-acknowledge), respectivamente. En caso de recibir una “N”, o bien de no recibir una respuesta después de 50 ms, la computadora de vuelo retransmite el comando.

Si al transmitir el comando una vez más la respuesta vuelve a ser errónea o no hay respuesta después del tiempo indicado, el nodo transmisor automáticamente conmuta de canal y repite todo el proceso por el canal redundante.

En caso de que después de estos cuatro intentos —dos de ellos por el canal principal y otros dos por el canal redundante— no se transmita el comando. El nodo transmisor realiza nuevamente la conmutación del canal e incrementa un contador que lleva el registro del número de fallas totales en la red interna. Posteriormente, continúa con los procesos pendientes. Si por el contrario, el comando llega exitosamente dentro de esos cuatro intentos, se continúa con el flujo normal del programa.

Cuando una computadora envía un comando en “Broadcasting” (es decir, con destino a todas las computadoras), no espera respuesta acerca de la ejecución correcta del comando. Esto evita la pérdida de la sincronía al recibir el acknowledge de todas las computadoras.

La computadora de vuelo también puede recibir y ejecutar comandos enviados desde cualquiera de las demás computadoras, pero sólo bajo control de la misma computadora de vuelo. Para este propósito, cuando la CV dispone de tiempo libre solicita a las demás computadoras que le envíen los comandos que tienen en su stack. Posteriormente la CV procede a ejecutarlos uno a uno de forma síncrona.

Adicionalmente, la CV puede solicitar parámetros a las demás computadoras y recibirlos mediante diez bytes disponibles que tienen los comandos de red interna. De esta forma, la CV puede solicitar parámetros mediante un comando y recibir los parámetros en un comando de respuesta.

### **3.2.1 Comandos de Red Interna**

La tabla 3.2 muestra una lista que sintetiza y describe los comandos de la red interna. El número de comando se muestra en valor hexadecimal y además se desglosan sus distintas opciones de acuerdo al byte destino/origen. (En el Apéndice A se puede observar la estructura de cada comando).

Tabla 3.2. Lista de los comandos de red interna utilizados por la CV.

<b>Cmd</b>	<b>Función</b>	
<b>01</b>	Solicitud de Diagnóstico a CU's	
	1.1	CV solicita diagnóstico a CUO
	1.2	CV solicita diagnóstico a DT
	1.3	CV solicita diagnóstico a CCD
	1.4	CV solicita diagnóstico a SP
<b>02</b>	Envío de diagnóstico	
	2.1	CV envía su diagnóstico en broadcasting (VOTEO)
	2.2	CUO envía su diagnóstico en broadcasting (VOTEO)
	2.3	DT envía su diagnóstico en broadcasting (VOTEO)
	2.4	CCD envía su diagnóstico en broadcasting (VOTEO)
	2.5	SP envía su diagnóstico en broadcasting (VOTEO)
	2.6	CUO envía su diagnóstico a CV (Diagnóstico simple)
	2.7	DT envía su diagnóstico a CV (Diagnóstico simple)
	2.8	CCD envía su diagnóstico a CV (Diagnóstico simple)
	2.9	SP envía su diagnóstico a CV (Diagnóstico simple)
<b>03</b>	Petición para liberar boom	
	3.1	CV solicita a DT liberar boom
<b>04</b>	Solicitud a CUO de transferencia de datos	
	4.1	CV solicita a CUO que transfiera 10 datos (primeros 10 datos)
	4.2	CV solicita a CUO que transfiera 10 datos (segundos 10 datos)
	4.3	CV solicita a CUO que transfiera 10 datos (últimos 10 datos)
<b>05</b>	Transferencia de CUO de datos a CV	
	5.1	CUO transfiere primeros 10 datos a CV
	5.2	CUO transfiere segundos 10 datos a CV
	5.3	CUO transfiere terceros y últimos 10 datos a CV
<b>06</b>	CV avisa a DT que se va a subir un nuevo programa al satélite	
	6.1	CV avisa a DT que se va a subir un nuevo programa al satélite
<b>08</b>	Solicitud a DT de la conmutación de CV	
	8.1	CV solicita a DT conmutar CV (Demostración)

**CAPÍTULO 3. SOFTWARE DE OPERACIONES**

<b>Cmd</b>	<b>Función</b>	
<b>09</b>	Aviso a CUO de detección de enlace óptico	
	9.1	CV avisa a CUO que ET detectó enlace óptico
<b>0B</b>	Solicitud a CV de retransmisión de programa	
	B.1	CUO solicita a CV la retransmisión del programa
<b>0C</b>	Solicitud a CCD de la adquisición de imagen	
	C.1	CV solicita a CCD la adquisición de una imagen
<b>0D</b>	Solicitud a CV de conmutación de equipo de comunicaciones	
	D.1	DT solicita a CV conmutar equipo de comunicaciones
<b>10</b>	Aviso de liberación de satélite (monitoreo de red interna)	
	10.1	CV avisa a SOFDEVO que el satélite se ha liberado
<b>11</b>	Aviso de envío de TlmN (monitoreo de red interna)	
	11.1	CV avisa a SOFDEVO que envió a ET TlmN de N orbitas
<b>12</b>	Aviso de recepción correcta de nuevo programa (monitoreo de red interna)	
	12.1	CV avisa a SOFDEVO que recibió correctamente un nuevo programa
<b>13</b>	Aviso de ejecución de un comando (monitoreo de red interna)	
	13.1	CV avisa a SOFDEVO que ha ejecutado el comando X
<b>14</b>	Solicitud a CU's de enviar comandos en su stack	
	14.1	CV solicita a CUO envío de comandos en su stack
	14.2	CV solicita a DT envío de comandos en su stack
	14.3	CV solicita a SP envío de comandos en su stack
<b>15</b>	Aviso a CV que no existen más comandos en el stack	
	15.1	CUO avisa a CV que los comandos en su stack se han terminado o que no hay comandos
	15.2	DT avisa a CV que los comandos en su stack se han terminado o que no hay comandos
	15.3	SP avisa a CV que los comandos en su stack se han terminado o que no hay comandos
<b>16</b>	Solicitud a CCD de transferencia de imagen	
	16.1	CV solicita a CCD la transferencia de la imagen
<b>17</b>	Envío de parámetros de ET a CUO y orden para ejecución de nuevo programa	
	17.1	CV envía parámetros de ET a CUO
	17.2	CV da la orden a CUO para que ejecute nuevo programa
<b>19</b>	Aviso de inicio o fin de procesamiento en paralelo (monitoreo por red interna)	
	19.1	CV avisa a SOFDEVO que inicia o finaliza el procesamiento en paralelo
<b>1A</b>	Aviso de inicio o fin de prueba en Kourou (monitoreo por red interna)	
	1ª.1	CV avisa a SOFDEVO que inicia o finaliza prueba en Kourou
<b>1B</b>	Aviso a CU's de inicio de autodiagnóstico	
	1B.1	CV avisa CU's inicio de autodiagnóstico
<b>1C</b>	CV pide diagnósticos acumulados en CU's (VOTEO)	
	1C.1	CV pide a CUO que envíe en broadcast sus diagnósticos acumulados
	1C.2	CV pide a DT que envíe en broadcast sus diagnósticos acumulados
	1C.3	CV pide a CCD que envíe en broadcast sus diagnósticos acumulados
	1C.4	CV pide a SP que envíe en broadcast sus diagnósticos acumulados
<b>1D</b>	CU's regresan a DT sus diagnósticos acumulados (VOTEO)	
	1D.1	CV envía a CU's sus diagnósticos acumulados (en <i>broadcast</i> )
	1D.2	CUO envía a CU's sus diagnósticos acumulados (en <i>broadcast</i> )
	1D.3	DT envía a CU's sus diagnósticos acumulados (en <i>broadcast</i> )
	1D.4	CCD envía a CU's sus diagnósticos acumulados (en <i>broadcast</i> )
	1D.5	SP envía a CU's sus diagnósticos acumulados (en <i>broadcast</i> )

**CAPÍTULO 3. SOFTWARE DE OPERACIONES**

<b>Cmd</b>	<b>Función</b>	
<b>1E</b>	Envío a CV del resultado del voto	
	1E.1	CUO envía a CV el resultado del voto
	1E.2	DT envía a CV el resultado del voto
	1E.3	CCD envía a CV el resultado del voto
	1E.4	SP envía a CV el resultado del voto
<b>20</b>	Solicitud a SP del encendido o apagado de radios	
	20.1	CV solicita a SP el encendido o apagado de radios
<b>22</b>	Solicitud a SP de liberación de Boom	
	22.1	CV solicita a SP liberación de Boom
<b>23</b>	Solicitud a SP del encendido o apagado de KA	
	23.1	ET (por medio de CV) solicita a SP encendido o apagado de KA
<b>24</b>	Solicitud a SP del encendido o apagado de CCD	
	24.1	CV solicita a SP encendido o apagado de CCD
<b>25</b>	Solicitud a SP del encendido o apagado de procesadores	
	25.1	CV solicita a SP el encendido o apagado de procesadores para diagnóstico simple o voto
<b>26</b>	Solicitud a SP de que indique cuáles computadoras puede encender	
	26.1	CV solicita a SP le indique cuáles computadoras se pueden encender para la realización de diagnóstico simple o voto
<b>27</b>	Aviso a CV de cuáles computadoras pueden encenderse	
	27.1	SP le indica a CV cuáles computadoras se pueden encender para la realización de diagnóstico simple o voto
<b>28</b>	<b>Solicitud a SP de que envíe datos de sus 17 sensores</b>	
	28.1	CV solicita a SP le envíe los valores de sus primeros 8 sensores
	28.2	CV solicita a SP le envíe los valores de sus últimos 9 sensores
<b>29</b>	<b>Envío a CV de los valores de los sensores de SP</b>	
	29.1	SP envía a CV los valores de sus primeros 8 sensores
	29.2	SP envía a CV los valores de sus últimos 9 sensores
<b>2A a 2D</b>	Informa sobre variables de la computadora de vuelo	
	2A.1	CV indica a SOFDEVO que no hay órbitas por bajar
	2B.1	CV indica a SOFDEVO que no fue posible bajar ni una órbita por falta de tiempo
	2C.1	CV indica a SOFDEVO que se ha enviado TlmN a ET
	2D.1	CV sólo envía a SOFDEVO el estado de TlmN
<b>2E</b>	Informa sobre la misión de captura de imágenes	
	2E.1	CV informa a SOFDEVO del tiempo de captura de las imágenes programadas en la misión
<b>40</b>	CV solicita a CU's resultado de voto	
	40.1	CV solicita a CUO resultado de voto
	40.2	CV solicita a DT resultado de voto
	40.3	CV solicita a CCD resultado de voto
	40.4	CV solicita a SP resultado de voto
<b>41</b>	CV ordena a CU's generen señales eléctricas de permisos y reconfiguración de CV	
	41.1	CV ordena a CU's la generación de señales eléctricas de permisos para reconfiguración
	41.2	CV ordena a DT la reconfiguración de procesadores
<b>42</b>	CV solicita a DT el estado de los procesadores de CV almacenados en DT	
	42.1	CV solicita a DT el estado de sus procesadores

Cmd	Función	
43	Envío de "status" de computadoras de CV	
	43.1	DT envía el "status" de las computadoras de CV a CV
44	CV envía DT y SP el estado de su procesador activo	
	44.1	CV envía el estado de su procesador activo a DT
	44.2	CV envía el estado de su procesador activo a SP
45	Se actualiza el estado del procesador activo de CV	
	45.1	ET actualiza estado del procesador activo de CV

### 3.3. Protocolo de comunicación de la Red Externa

El software del satélite que realiza la comunicación con Tierra constituye el medio de control remoto del satélite a través del cual se le indican las tareas a realizar y se lleva a cabo su monitoreo. Su complemento, en Tierra, se encarga de enviar comandos específicos para la ejecución de tareas en el satélite como: la captura de imágenes, la adquisición de telemetría, etc. La información que se envía a Tierra se clasifica en tres tipos:

- ✓ Envío de comandos de respuesta a comandos en línea.
- ✓ Envío de Status y Telemetría.
- ✓ Envío de imagen.

#### 3.3.1 Envío de comandos de respuesta a comandos en línea

Los comandos en línea son comandos que se envían desde la Estación Terrena al satélite para ejecutar tareas específicas. Estos comandos se reciben en la Computadora de Vuelo a través de los equipos de comunicaciones.

En la figura 3.2, se muestra el orden que conservan los bytes en un comando de línea de Estación Terrena a Satélite.

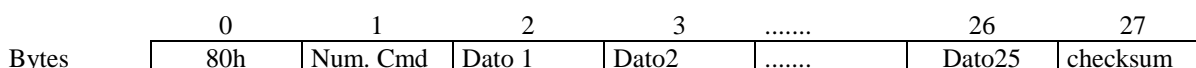


Figura 3.2. Trama de comando en línea de ET a Satélite

Los comandos en línea tienen principalmente tres funciones: ejecutar tareas específicas en el satélite, modificar valores de variables de control del satélite y solicitar información al satélite la cual se captura con el software de ET.

Respecto a la solicitud de información satelital, se divide en: envío de parámetros a Tierra mediante un comando de respuesta, envío de telemetría y envío de imagen capturada.

El envío de parámetros a Tierra mediante un comando de respuesta se utiliza básicamente en dos casos:

- ✓ Durante el envío de variables de estado de SATEX a Tierra
- ✓ Durante el envío de datos del experimento de comunicaciones ópticas (CUO).

Desde la Estación Terrena es posible solicitar al satélite sus variables de control más importantes, como el contador de órbita, el número de imágenes que ha capturado, el tipo de telemetría normal pendiente por bajar, entre otras. Esto se realiza con la finalidad de corroborar que ha ejecutado correctamente sus funciones de tal modo que se puedan corregir posibles problemas antes de enviarle una nueva misión.

El experimento de la CUO, es un sistema de control retroalimentado que envía un “beacon” cónico desde la Estación Terrena con ayuda de un equipo de posicionamiento. La CUO realiza el rastreo para capturar el “beacon” y genera datos que se envían a Tierra como retroalimentación. Estos valores se envían a través de un comando de respuesta.

De esta forma, los comandos enviados a Estación Terrena están formados por 30 bytes (de 0 a 29), como se muestra en la figura 3.3.

	0	1	2	3	.....	28	29
Bytes	55h	Tipo	Dato 1	Dato2	.....	Dato27	checksum

Figura 3.3. Trama de los comandos enviados como respuesta a un comando en línea

El primer byte de estos comandos, contiene el valor hexadecimal 55h que se utiliza como indentificador. El segundo byte, contiene el tipo de comando de respuesta. Por el momento sólo existen dos tipos: envío de variables de estado de SATEX y envío de datos de retroalimentación del experimento de la Carga Útil Óptica. Sin embargo, este comando de respuesta permite el envío de hasta 255 tipos diferentes que pueden utilizarse para otros fines.

Los comandos de respuesta contienen 27 bytes para transportar la información que se envía a Tierra. El último byte de la trama contiene un checksum que permite determinar si la información llegó correctamente.

### 3.3.2 Envío de parámetros de estado y Telemetría

Una función muy importante del satélite es el monitoreo periódico de sus equipos, así como el reporte de éstos a Tierra. En cuanto al proceso de muestreo de datos de estado, se realiza cada diez minutos, por lo tanto, considerando que la órbita de SATEX es de 100 minutos, entonces existen 10 muestreos en una órbita.

Con el software de Estación Terrena se pueden programar misiones de adquisición de telemetría y diagnóstico de computadoras de hasta un máximo de 16 órbitas, cada una compuesta de 10 muestreos. A su vez, cada muestreo genera 20 bytes de “status” de diagnóstico de las computadoras y 130 bytes de telemetría, que corresponden a datos de 65 sensores (48 de la CV y 17 del SP). Además, en estos 130 bytes se encuentra el número de fallas totales generadas en la red interna y el número de pulsos generados por el Gradiente Gravitacional al ser desplegado.

**CAPÍTULO 3. SOFTWARE DE OPERACIONES**

La tabla 3.3 indica cómo es el arreglo de los datos en un muestreo de telemetría detallando cada byte de información.

Tabla 3.3. Estructura de la información de un paquete de muestreo de sensores de telemetría.

BYTE	NUM./CVE. SENSOR	BYTE	NUM./CVE. SENSOR	BYTE	NUM./CVE. SENSOR			
1	0	MAG X1 PB	45	22	SAL15 SENS PB	89	44	SC DT PB
2	0	MAG X1 PA	46	22	SAL15 SENS PA	90	44	SC DT PA
3	1	SFS X1 PB	47	23	SAL4 SENS PB	91	45	SENSOR D PB
4	1	SFS X1 PA	48	23	SAL4 SENS PA	92	45	SENSOR D PA
5	2	CCD PB	49	24	MAG Y2 PB	93	46	SAL19 SENS PB
6	2	CCD PA	50	24	MAG Y2 PA	94	46	SAL19 SENS PA
7	3	SAL20 SENS PB	51	25	SFS X3 PB	95	47	SAL10 SENS PB
8	3	SAL20 SENS PA	52	25	SFS X3 PA	96	47	SAL10 SENS PA
9	4	SAL11 SENS PB	53	26	SC RXVHF2 PB	97	48	SC PANEL 1
10	4	SAL11 SENS PA	54	26	SC RXVHF2 PA	98	49	SC PÁNEL 2
11	5	SAL1 SENS PB	55	27	ST CUO PB	99	50	SC PANEL 3
12	5	SAL1 SENS PA	56	27	ST CUO PA	100	51	SC PANEL 4
13	6	MAG Y1 PB	57	28	SAL16 SENS PB	101	52	SC BAT
14	6	MAG Y1 PA	58	28	SAL16 SENS PA	102	53	ST BAT 1
15	7	SFS Y1 PB	59	29	SAL5 SENS PB	103	54	ST BAT 2
16	7	SFS Y1 PA	60	29	SAL5 SENS PA	104	55	ST DCDC
17	8	SC KA PB	61	30	MAG Z2 PB	105	56	SV BAT 1
18	8	SC KA PA	62	30	MAG Z2 PA	106	57	SV BAT 2
19	9	ST TCV PB	63	31	SFS Y3 PB	107	58	SV BAT 3
20	9	ST TCV PA	64	31	SFS Y3 PA	108	59	SV BAT 4
21	10	SAL13 SENS PB	65	32	SC TXVHF2 PB	109	60	SV BAT 5
22	10	SAL13 SENS PA	66	32	SC TXVHF2 PA	110	61	% DE CARGA
23	11	SAL2 SENS PB	67	33	ST TXVHF1 PB	111	62	LIBRE
24	11	SAL2 SENS PA	68	33	ST TXVHF1 PA	112	63	LIBRE
25	12	MAG Z1 PB	69	34	SAL17 SENS PB	113	64	LIBRE
26	12	MAG Z1 PA	70	34	SAL17 SENS PA	114		DISPONIBLE
27	13	SFS X2 PB	71	35	SAL8 SENS PB	115		DISPONIBLE
28	13	SFS X2 PA	72	35	SAL8 SENS PA	116		DISPONIBLE
29	14	SC RXVHF1 PB	73	36	SC SENS PB	117		DISPONIBLE
30	14	SC RXVHF1 PA	74	36	SC SENS PA	118		DISPONIBLE
31	15	ST DT PB	75	37	SFS X4 PB	119		DISPONIBLE
32	15	ST DT PA	76	37	SFS X4 PA	120		DISPONIBLE
33	16	SAL14 SENS PB	77	38	SC CUO PB	121		DISPONIBLE
34	16	SAL14 SENS PA	78	38	SC CUO PA	122		DISPONIBLE
35	17	SAL3 SENS PB	79	39	ST RXVHF1 PB	123		DISPONIBLE
36	17	SAL3 SENS PA	80	39	ST RXVHF1 PA	124		DISPONIBLE
37	18	MAG X2 PB	81	40	SAL18 SENS PB	125		DISPONIBLE
38	18	MAG X2 PA	82	40	SAL18 SENS PA	126		DISPONIBLE
39	19	SFS Y2 PB	83	41	SAL9 SENS PB	127		DISPONIBLE
40	19	SFS Y2 PA	84	41	SAL9 SENS PA	128		DISPONIBLE
41	20	SC TXCHF1 PB	85	42	SENSOR E PB	129		Num. Fallas Totales en Red Interna
42	20	SC TXVHF1 PA	86	42	SENSOR E PA	130		Pulsos Boom
43	21	ST CCD PB	87	43	SFS Y4 PB			
44	21	ST CCD PA	88	43	SFS Y4 PA			



### CAPÍTULO 3. SOFTWARE DE OPERACIONES

Como se observa, los primeros 48 datos de sensores (bytes 1 a 96) corresponden a los sensores vinculados a la Computadora de Vuelo del microsátélite. Los bytes 97 a 110 corresponden a 14 muestras de un solo byte vinculados al sistema de potencia del vehículo espacial. En este paquete de telemetría también se incluyen algunos bytes que están disponibles para posibles datos que en un futuro se puedan integrar a esta información.

Ahora, en los bytes de diagnóstico de equipos se tiene información que captura la Computadora de Vuelo, cada muestreo realizado genera 20 bytes con la información indicada en la tabla 3.4.

Tabla 3.4. Estructura de la información de los 20 bytes de cada muestreo de SATEX.

BYTE	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
0	Número de órbita				Número de muestreo			
1	1 – CP 0 – CR				1	1	1	0
2	Equipos de comunicaciones							
3	1	Estado puerto serie con ET			Puerto serie de red interna redundante.	Puerto serie red interna		
4	Síndrome de CV							
5	Diagnostico de RAM de CV							
6	Síndrome de CUO							
7	Diagnostico de RAM de CUO							
8	Síndrome de DT							
9	Diagnostico de RAM de DT							
10	Síndrome de CCD							
11	Diagnostico de RAM de CCD							
12	Síndrome de SP							
13	Diagnostico de RAM de SP							
14	Número de correcciones del EDAC							
15	Número de interrupciones del PECC							
16	Resultado de voto de CUO				Resultado de voto de CV			
17	Resultado de voto de CCD				Resultado de voto de DT			
18					Resultado de voto de SP			
19	Se informa si se realizó Diagnostico simple o el experimento de voto.							

Como se aprecia, cada byte contiene información importante del diagnóstico, enseguida se ofrecen detalles de esta información.

**BYTE 0:** en la parte alta del byte se indica el numero de orbita y en la parte baja el número de muestreo de este diagnóstico.

**BYTE 1:** en sus 4 bits menos significativos contiene el valor en hexadecimal “E” o en decimal el valor 14 que identifica al estatus como tal, con el bit 7 indica el procesador que se esta utilizando en la computadora de vuelo, con el valor de “1” nos determina que es el procesador principal y un “0” indica que es el procesador redundante.

**BYTE 2:** indica que equipos de comunicación fueron utilizados por el satélite, el valor hexadecimal de “33” indica que están activos el transmisor 1 y receptor 1, el valor hexadecimal de “C3” indica que están activos el transmisor 2 y receptor

1, el valor hexadecimal de “3C” indica que están activos el transmisor 1 y receptor 2 y por ultimo el valor hexadecimal de “CC” indica que están activos el transmisor 2 y el receptor 2.

**BYTE 3:** este byte indica el estado de los puertos y canales de comunicación, tanto en la red interna como en las comunicaciones con Tierra. El valor hexadecimal de “80” indica que el puerto serie de Estación Terrena y la red interna se encuentra en buen estado, el valor hexadecimal de “F0” indica que el puerto serie de estación terrena se encuentra mal pero la red interna bien, el valor hexadecimal de “87” indica que el puerto serie de Estación Terrena se encuentra bien pero la red interna fallo y por último el valor hexadecimal de “F7” indica que el puerto serie de estación terrena se encuentra mal y que la red interna también fallo.

**BYTES del 4 al 14:** determinan el estado tanto de la memoria RAM como de los procesadores de cada computadora a bordo del microsatélite. El byte reservado para las condiciones de la memoria RAM indica el número de errores que se generaron en el transcurso de 10 minutos.

**BYTE 15:** determina el número de interrupciones del PECC, una interrupción equivale a 255 errores en RAM detectados por el EDAC.

**BYTES 16, 17 y 18:** determinan el resultado del voto obtenido en cada computadora del satélite.

**BYTE 19:** La información en este byte se visualiza en los controles de la estación terrena indicando si se realizó el experimento de voto o sólo diagnóstico simple. Particularmente el byte ofrece los datos indicados en la tabla 3.5:

Tabla 3.5. Estructura de la información del byte 19 en Estación Terrena.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Significado del Diagnostico	Tipo de diagnostico realizado
1	0	0	0	0	0	Computadora Principal en uso	-
0	1	0	0	0	0	No se realiza voto por falla en red interna.	Simple
0	0	1	0	0	0	Se puede hacer voto, 4 computadoras activas.	Voto
0	0	0	1	0	0	No pudo encenderse el DT	Simple
0	0	0	0	1	0	No se programo voto en ninguna orbita	Simple
0	0	0	0	0	1	No llego respuesta a CV durante el envío de cmd.	Simple
0	0	1	1	0	0	No es orbita de voto	Simple
0	1	0	1	0	0	Solo se realiza con SP	Simple

Cuando el satélite adquiere telemetría normal (cada 10 minutos) de todos sus sensores a bordo, estos sensores indican el comportamiento del microsatélite. Estos muestreos se realizan a lo largo de un periodo aproximadamente de 100 minutos el cual corresponde a una orbita de telemetría normal (TLMN). Por ello 16 orbitas de telemetría corresponden a 1600 minutos de vuelo orbital. Este valor corresponde a la mayor autonomía del satélite, es decir, el vehículo puede almacenar telemetría de hasta 16 orbitas sin tener un contacto con su Estación Terrena.

La trama de telemetría normal está formada de la siguiente manera:

<b>1 bytes Despertar</b>	<b>1 byte de orbitas obtenidas</b>	<b>#</b>	<b>220 bytes Diagnostico de equipos</b>	<b>2 bytes # de orbitas</b>	<b>1 bytes FF</b>	<b>1319 bytes Telemetría de sensores</b>
------------------------------	--	----------	---	-------------------------------------	-----------------------	--

El primer byte se utiliza para generar la interrupción por evento del puerto serie con el SET, posteriormente otro byte indica el Número de Órbitas de Telemetría (NOT) capturada por el satélite, por lo cual el tamaño de la trama es variable, siendo su longitud máxima el que se indica en la figura anterior. Posteriormente aparecen (NOT\*22 datos) que corresponden al número de diagnósticos empaquetados en la trama, de acuerdo al siguiente orden:

<b>20 bytes Diagnóstico</b>	<b>2 bytes</b>	<b>20 bytes Diagnóstico</b>	<b>2 bytes</b>	<b>...</b>	<b>2 bytes</b>	<b>20 bytes Diagnóstico</b>	<b>2 bytes</b>
Muestreo 0	"DD"	Muestreo 1	"DD"	...	"DD"	Muestreo 9	"DD"

Después aparecen 2 bytes con la información repetida acerca del número de orbitas de telemetría que capturó el SATEX, seguidos por un byte de separación "FF". Finalmente aparecen un máximo de 1319 bytes con las muestras de telemetría de sensores, con el siguiente formato:

<b>130 bytes</b>	<b>1 byte</b>	<b>1 byte</b>	<b>...</b>	<b>130 bytes</b>	<b>1 byte</b>
TS0	Chk0	"D"	...	TS9	Chk9

TS = Telemetría de sensores.

Chk = Checksum

### **3.3.3 Envío de imagen**

El tercer tipo de información que SATEX envía a tierra es la imagen capturada por la cámara CCD. El proceso de captura de una imagen, comienza con el envío de una misión desde la Estación Terrena.

Esta misión, permite programar la captura de hasta 4 imágenes, especificando la órbita y muestreo de captura, los minutos y segundos adicionales a ese muestreo y la resolución que deberá tener cada una de ellas. Al capturarse una imagen, se almacena en la memoria disponible en la computadora de la cámara CCD, la cual se empaqueta en bloques de 1000 bytes. El objetivo de este empaquetado es proporcionar mayor seguridad en la transferencia de la imagen a Tierra. Debido a que se realiza la verificación de información cada 1000 bytes.

Cada paquete tiene un encabezado de 9 bytes (de 0 a 8) y un byte para almacenar el cálculo del checksum, colocado al final del mismo. Por lo tanto, la información de la imagen reside en 990 bytes, el último paquete puede tener un tamaño que va desde 11 bytes hasta 990 bytes, pues se incluye al encabezado y el byte de checksum.

Los 1000 bytes que forman estos paquetes tienen el formato indicado en la figura 3.4:

0	1	2	3	4	5	6	7	8	.....	999
0xFC	Num. Img.	Num. Total de paquetes	Número de paquete	Num. Bytes último paquete	Tipo	Bytes de imagen	chk			

Figura 3.4. Paquete de 1000 bytes para el envío de imagen a Tierra

El byte cero (FCh), sirve como identificador de los paquetes que contienen información de una imagen. El byte uno contiene el número de la imagen capturada en la misión (de 1 a 4). El segundo y tercer byte contiene el número total de paquetes utilizados para almacenar la imagen. Esto permite almacenar imágenes hasta en un máximo de 65536 paquetes (de 0h a FFFFh), aunque en promedio sólo sean utilizados 60 o en un caso extremo hasta 300. Los bytes 4 y 5 contienen el número del paquete de la imagen. Los bytes 6 y 7 contienen el número total de bytes del último paquete. Este número incluye a los 9 bytes de encabezado y al byte de checksum.

Cuando se definió el formato de los paquetes, aún no se conocían con detalle las características que tendría la imagen, entre ellos el tamaño real y su formato. Por tal razón, se agregó el byte 8 que se refiere al formato de la imagen, 1 para una imagen con formato JPG, 2 para una imagen tipo BMP y 3 para indicar que se trata de una imagen tipo GIF.

El protocolo de transferencia de las imágenes se lleva a cabo una vez que la Estación Terrena solicita su envío. Esta solicitud se hace mediante el comando en línea número 10. Cuando la CV recibe esta solicitud, envía un comando a la CCD, pidiéndole que inicie la transferencia de la imagen. La CCD envía uno a uno a la CV los paquetes que formó previamente. La CV se encarga de hacer “eco” a Tierra de estos paquetes recibidos por red interna.

### 3.2 Estructura general del software de operación satelital

Dadas las características del microcontrolador usado para las tarjetas de procesamiento del SATEX —SIEMENS SAB80C166—, toda la programación de software operativo se desarrolló en lenguaje C con ayuda de algunas utilerías propias del microcontrolador. Así mismo, el control del satélite se efectúa con base en interrupciones para obtener un comportamiento controlado por eventos definidos y autónomos. En la tabla 3.6 se describen las interrupciones y su nivel de prioridad [SIEMENS, 1997].

Tabla 3.6. Niveles de interrupción en el SATEX

Registro SAB80C166 / Función		Grupo	Nivel
CC15IC	Interrupción del pin 15, puerto 2 para lectura del EDAC	3	15
ADCIC	Lectura del convertidor Analógico/Digital	0	14
T5IC	Interrupción del timer 5 (control de estabilización)	3	13
T6IC	Interrupción del timer 6 (control de estabilización)	3	12
S0RIC	Interrupción en la recepción del puerto serie 0 (Red externa)	3	11
S1RIC	Interrupción en la recepción del puerto serie 1 (Red Interna)	3	10
T2IC	Interrupción del timer 2 (Control de tiempos orbitales)	3	9
T4IC	Interrupción del timer 2 (Contabilización de tiempos orbitales)	3	8
T0IC	Interrupción del timer 0 (Retrasos en tiempos de protocolos)	3	5

### CAPÍTULO 3. SOFTWARE DE OPERACIONES

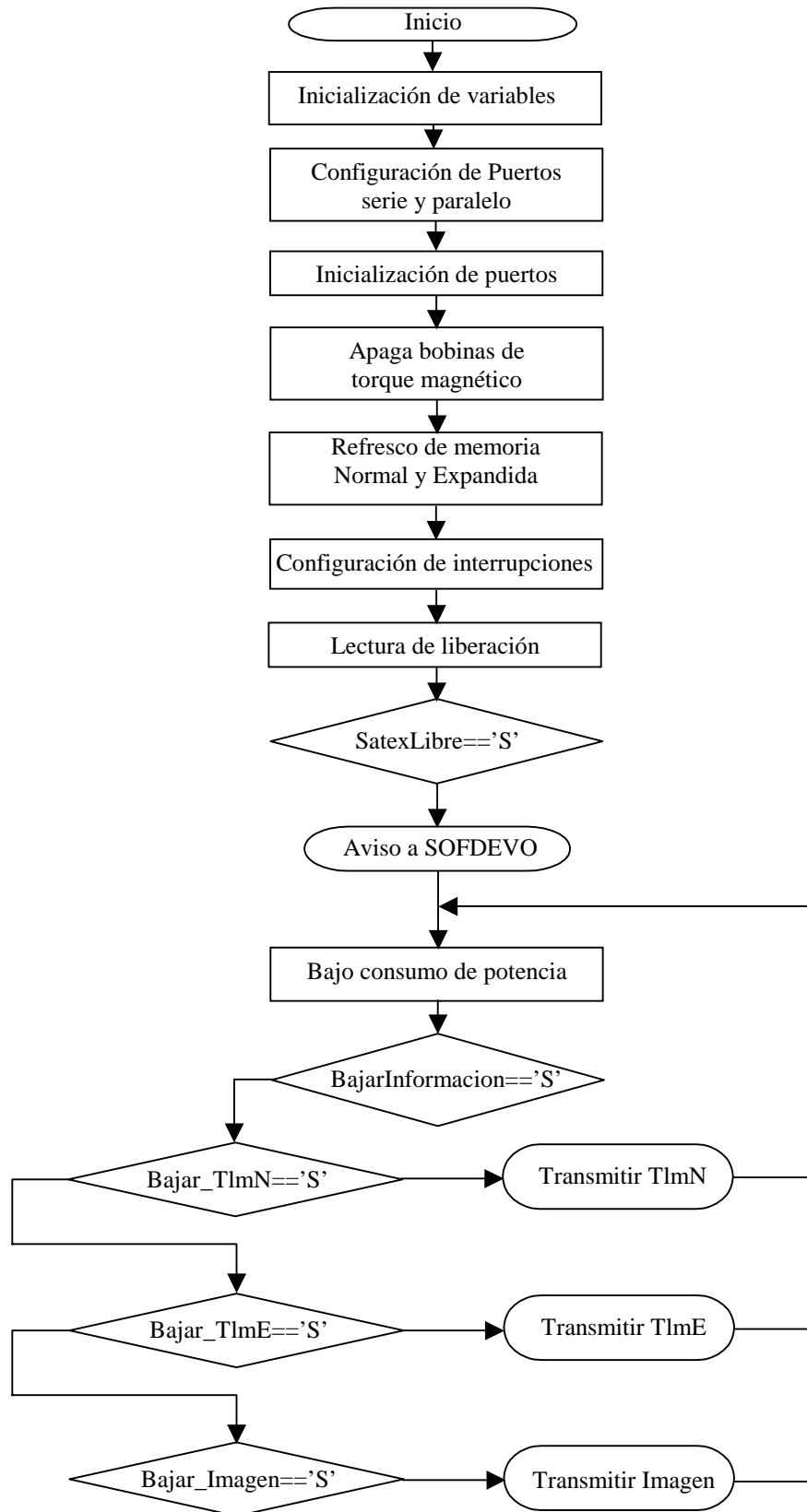


Figura 3.5. Diagrama de Flujo Básico del software operativo de SATEX

Cabe señalar que en la comunicación con la red interna, al recibir el microcontrolador el primer byte de una trama de comando, la recibe por interrupción y los siguientes bytes los recibe por poleo.

El compilador utilizado para desarrollar el software de la computadora de vuelo es el compilador cruzado 80C166 Cross-Compiler BSO TASKING [TASKING, 1993] de la compañía Tasking Boston Systems Office [ www.tasking.com ], el cual utiliza librerías específicas para el microcontrolador. El compilador genera código ejecutable a partir del código fuente en lenguaje ensamblador.

El software de operación satelital, en el inicio de su vida o después de cualquier reconfiguración de la CV debe realizar las siguientes funciones:

- ✓ Inicialización de variables globales.
- ✓ Configurar puertos series y puertos paralelos.
- ✓ Inicializar puertos.
- ✓ Refrescar memoria.
- ✓ Configurar interrupciones.

Después de realizar lo anterior, debe realizar labores de automatización:

- ✓ Las labores que realiza de forma automática una vez que sea liberado.
- ✓ Las tareas de recopilación y formateo de telemetría.
- ✓ La transmisión de telemetría a Tierra.
- ✓ La atención a las ordenes que se envíen desde la estación terrena en forma de comandos, misiones o programas.
- ✓ La interacción con las cargas útiles para efectuar los experimentos que se ordenen desde la estación terrena.
- ✓ La estabilización del satélite.

Ya que el software de la CV se ha escrito en lenguaje C, su programación es de tipo estructurada, entonces podemos representar mediante un diagrama de flujo básico sus principales procesos realizados durante su periodo de vida. Este diagrama se muestra en la figura 3.5, en él se muestran también los procesos que se efectúan a través de interrupciones.

### ***3.3 Descripción de los mapas de memoria de la Computadora de Vuelo***

En la Computadora de Vuelo existen dos mapas de memoria diferentes, uno que admite el uso tanto de memoria ROM como de memoria RAM y el otro que controla exclusivamente memoria RAM.

La electrónica que define el mapa de memoria para acceder memoria ROM, direcciona un programa grabado de forma previa al lanzamiento del satélite, esto sirve como un método de protección en casos en que el programa cargado en memoria RAM presente errores incorregibles por efecto de la radiación. En la figura 3.6 y 3.7 se presentan dos diagramas que ilustran los dos mapas de memoria de la CV.

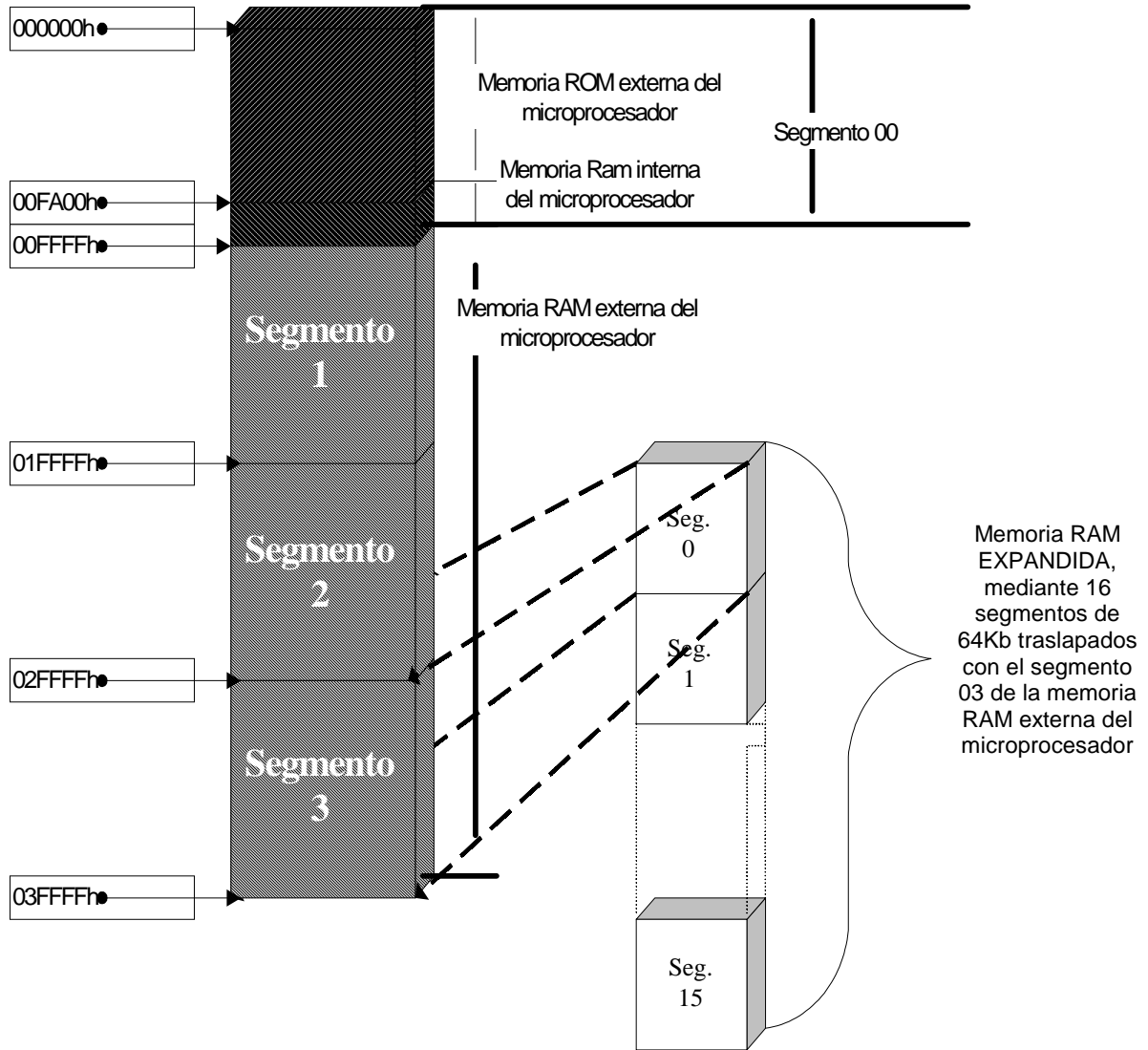


Figura 3.6. Mapa de memoria que usa ROM y RAM

Cuando se utiliza el mapa de memoria de la figura 3.6, la CV cuenta con los siguientes recursos de memoria:

- ✓ 63 Kb de Memoria ROM externa, usada para almacenar el software mínimo de operaciones de SATEX, el que se utilizará siempre como un modo seguro de software.
- ✓ 1 Kb de memoria RAM interna, usada por el microcontrolador para almacenar registros de trabajo y configuración de sistema para la implementación de la pila o Stack. El espacio restante puede ser administrado libremente por el usuario.

- ✓ 192 Kb de memoria RAM externa, usada para almacenar datos y código de programa. El esquema de carga de programa a la CV utiliza espacio de esta área de memoria para almacenar el nuevo programa.
- ✓ 920 Kb de memoria RAM expandida, accedida mediante el traslape del segmento 03 de la memoria externa, este acceso se controla mediante software en la computadora de vuelo, por esto solamente se utiliza para almacenar datos de telemetría o imagen, no de código, de lo contrario ocasionaría resultados inesperados.

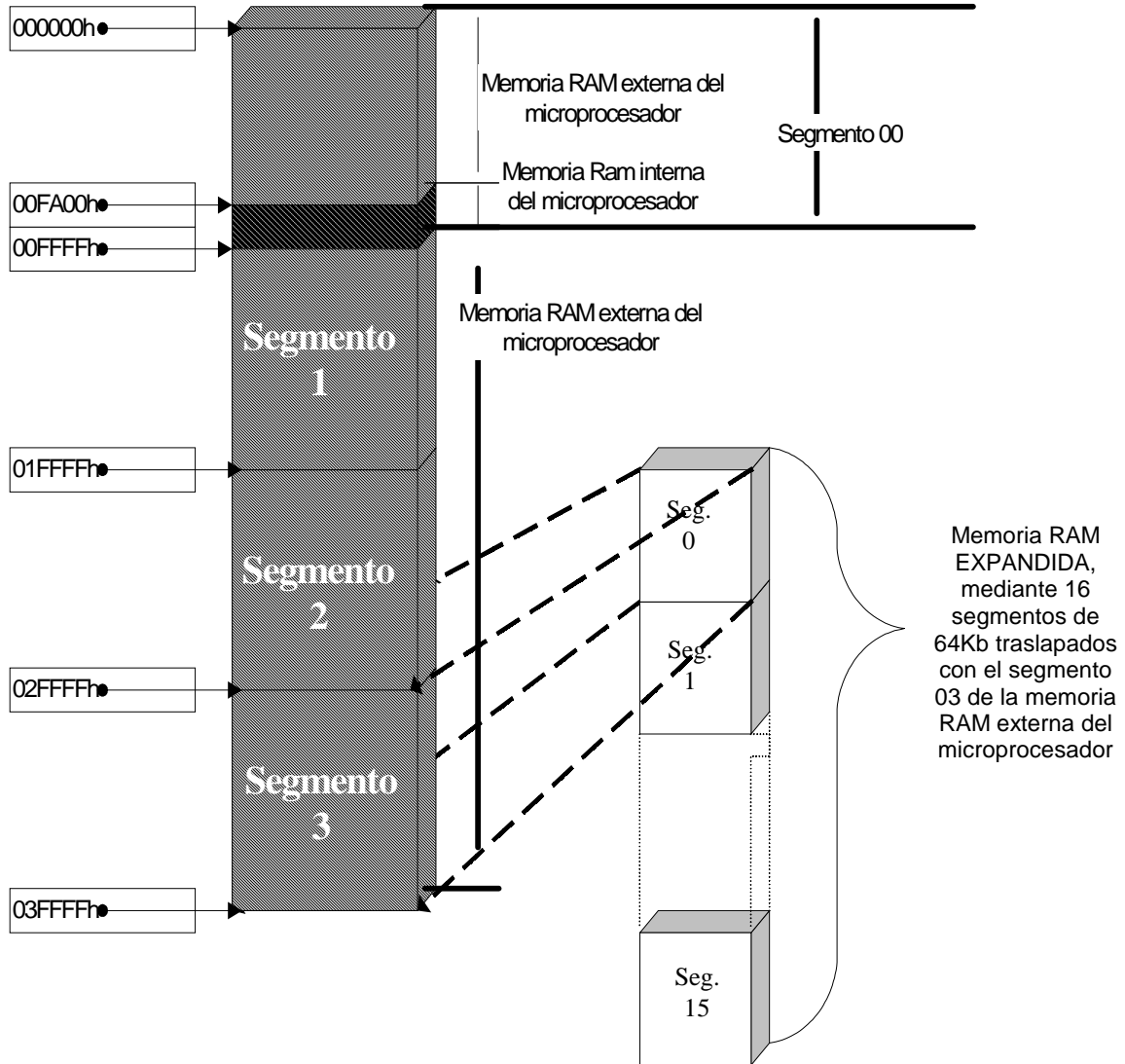


Figura 3.7. Mapa de memoria que utiliza sólo RAM

El mapa de memoria que cuenta exclusivamente con memoria tipo RAM se utiliza para cargar nuevos programas mediante el modo BOOTSTRAP LOADER (BTL). Cuando se utiliza este mapa de memoria la computadora de vuelo cuenta con los siguientes recursos de memoria:

- ✓ 255 K bytes de memoria RAM externa que puede ser usada para el almacenamiento de código o datos de programa.



- ✓ 1 KB de Memoria RAM interna del microprocesador, usada para el almacenamiento de registros de trabajo y configuración de sistema, y para la implementación de la pila o “Stack”. El espacio restante puede ser administrado libremente por el usuario [SIEMENS, 1997].
- ✓ 960 KB de memoria RAM expandida, accedida mediante el traslape del segmento 03 de la memoria externa. El acceso se controla mediante software en la computadora de vuelo, por esto solamente se utiliza para almacenar datos de telemetría o imagen, no de código, de lo contrario ocasionaría resultados inesperados.

Como se observa, los mapas de memoria difieren solamente en el segmento 00 de la memoria externa, el cual está compuesto por ROM en el primer caso y por RAM en el segundo. Este último mapa de memoria es el que debe ser usado con el esquema de carga y ejecución de programas mediante el BTL, lo que permite usar más de 63Kbytes de memoria para el programa de control a diferencia del primer mapa de memoria en el cual el software de control está limitado a dicha cantidad de memoria.

### **3.4 Software mínimo de operación satelital**

Como se mencionó anteriormente, la Computadora de Vuelo de SATEX puede manejar dos tipos de mapas de memoria. Es por ello que el software de operación satelital tiene dos tipos de software, uno de un tamaño menor a 64 Kb —para el mapa de memoria que utiliza ROM y RAM— y otro que puede tener un tamaño mayor a 64 Kb. Gracias a esto, se puede implementar un Software Mínimo (SOFMIN) de operación satelital, el cual deberá usar el mapa de memoria ROM/RAM.

Cuando el mapa de memoria ROM/RAM se encuentra activado, la unidad de procesamiento cuenta con 64 Kb de Memoria ROM externa, en estas localidades deberá ir el SOFMIN. Por sus características, en cuanto a tamaño, SOFMIN debe contener las rutinas más importantes para que SATEX pueda operar en órbita.

Las operaciones básicas de SOFMIN son la captura de telemetría Normal y la comunicación continua con Estación Terrena. Esto para que ET conozca detalladamente el estado operativo del satélite, además hacer factible la subida de un nuevo software de operación satelital que permita que SATEX realice todas sus funciones con éxito.

SOFMIN también es controlado por interrupciones, en menor número que las utilizadas en el Software General. Las interrupciones que se descartan en SOFMIN son las de Estabilización (T5IC y T6IC).

En el diagrama de flujo de la figura 3.8 se observa con detalle el funcionamiento de SOFMIN y se indican cuales son las rutinas que se necesitan para la sobrevivencia mínima del satélite.

### CAPÍTULO 3. SOFTWARE DE OPERACIONES

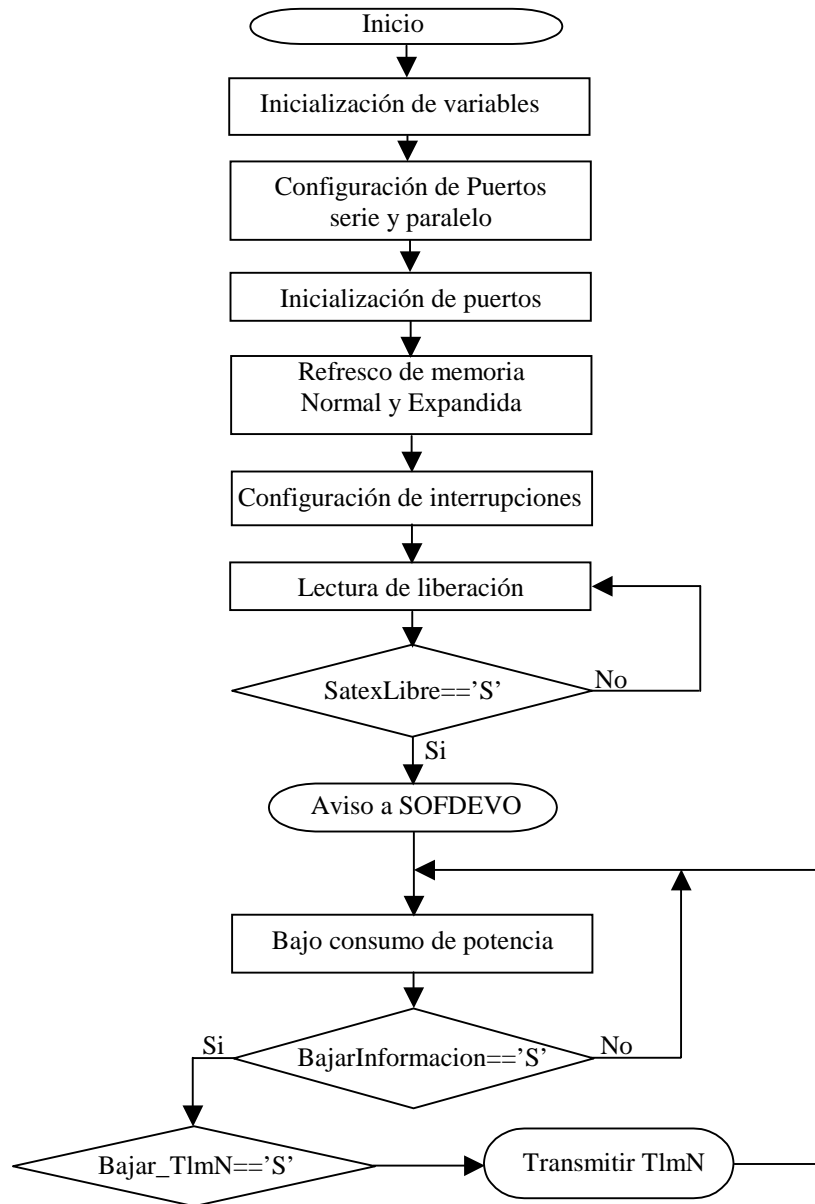


Figura 3.8. Diagrama de Flujo Básico del Software Mínimo (SOFMIN) operativo de SATEX.

## Capítulo 4

### ***Software de Depuración, Emulación y Validación Operativa Para Experimentos del Satélite***

---

#### ***4.1 Introducción***

En este capítulo se presentan los detalles operativos del software SOFDEVO que emular digitalmente los procesos de comunicaciones de cualquiera de los experimentos satelitales. Se describen los detalles del lenguaje de programación, las tramas y protocolos de la red en que basa su operación. El software evita la necesidad de contar físicamente con los experimentos del satélite para realizar pruebas de depuración o de validación operativa entre el software de operaciones del microsatélite y experimentos. Los detalles se exponen con diagramas de flujo para reducir el texto de tesis y con base en ellos en el capítulo siguiente se describen las actualizaciones que se desarrollaron para este software.

SOFDEVO debe de cumplir con requerimientos muy especiales: su interfaz debe ser lo suficientemente amigable para que cualquier persona pueda utilizarlo con entrenamiento reducido, su información debe estar organizada de tal forma que sea fácil de localizar e interpretar, tiene que ser transparente para todos los equipos del SATEX en vista de que su función es vigilar el tráfico de la información que viaja a través de la red de comunicaciones interna del SATEX. Debe dar la apariencia funcional de uno o varios equipos cuando emule cargas útiles y debe ser lo suficientemente robusto para permitir la validación de las operaciones del SATEX.

#### ***4.2. Lenguaje de programación utilizado para el desarrollo de SOFDEVO***

Las interfaces gráficas de usuario o GUI (del Inglés, Graphical User Interfaces) han revolucionado la industria de las microcomputadoras. Debido a que una imagen vale más que mil palabras por lo que estas aplicaciones tienen que ser consistentes, es decir, los usuarios pueden disponer de menos tiempo para dominar la aplicación y requerir menos restricciones acerca de las teclas por pulsar dentro de los menús, por ello se necesita utilizar la herramienta GUI para desarrollar aplicaciones eficientes.

Cuando no existían los lenguajes visuales como Visual Basic el desarrollo de las aplicaciones requería programadores expertos en C y centenares de líneas para realizar una tarea sencilla, debido a esto Visual Basic consideró la herramienta más potente y sencilla para el desarrollo de aplicaciones. Visual Basic utiliza programación manejada por eventos en donde el código no sigue una ruta predeterminada, lo que hace es ejecutar diferentes partes del código en respuesta a los eventos. Los eventos se activan por acciones, por mensajes del sistema u otras aplicaciones, o también por la aplicación misma. La secuencia

#### CAPÍTULO 4. SOFDEVO

de estos eventos determina el camino del código ejecutado, de esta forma, la ruta a través de la codificación del código difiere cada vez que el programa se ejecuta.

SOFDEVO se desarrolló con el lenguaje de programación Visual Basic 6.0 de Microsoft. Se eligió este lenguaje por que permite crear aplicaciones visuales en ambiente Windows de Microsoft, el cual es muy amigable para el usuario, otro punto importante para la elección de este lenguaje es la facilidad con que controla la comunicación serial, a través de un control "ActiveX", nativo de VB el cual se encarga de esta función. El control "ActiveX" utilizado es el "Mscomm32.ocx" el cual se instala junto con la versión profesional de desarrollo de VB 6.0.

El control "Mscomm32.ocx" permite utilizar cualquier puerto de comunicaciones seriales de una PC con una velocidad de 110 a 256K baudios. Puede atender el puerto ya sea por interrupción o por poleo, en ambos casos utiliza un proceso de interrupción almacenando su lectura en un "buffer" de recepción, evitando de esta forma la pérdida de datos. Para el envío de datos cuenta con un "buffer" de transmisión. El tamaño de ambos "buffers" se puede definir durante el diseño y/o durante la ejecución, en tanto que su valor depende de las capacidades de la PC.

Por otra parte la utilización del puerto paralelo fue importante para hacer pruebas de validación para el experimento de Mantenimiento Automatizado de la Computadora de Vuelo (MACV), así como para realizar la reconfiguración de la CV desde el SIMS por medio del envío de señales lógicas (0 Volts nivel bajo, 5 Volts nivel alto). Para utilizar esta función es necesario el archivo de control de periféricos, llamado "win32io.dll", este archivo es nativo de Microsoft Windows y sirve para el control (lectura y escritura), del puerto paralelo.

SOFDEVO solo puede ejecutarse en PC's que tengan instalado el sistema operativo Windows Millennium o anterior, además se recomienda una resolución mínima del monitor de 1024x768 píxeles.

Durante el desarrollo del proyecto SATEX ha sido necesaria la implantación de nuevos comandos para satisfacer o complementar sus necesidades, ya sea de la Computadora de Vuelo o en la operación de sus cargas útiles, además la depuración del software del sistema satelital ha generado ciertas modificaciones a comandos existentes. Adicionalmente se ha requerido verificar el correcto funcionamiento operativo del software después de efectuar dichas modificaciones, para ello SOFDEVO representa una herramienta de vital importancia que permite monitorear la información que circula a través del canal de comunicaciones de la red interna en uso (principal o redundante). Esto permite verificar y certificar que la información fluya en el orden adecuado y se generen las respuestas requeridas en cada operación.

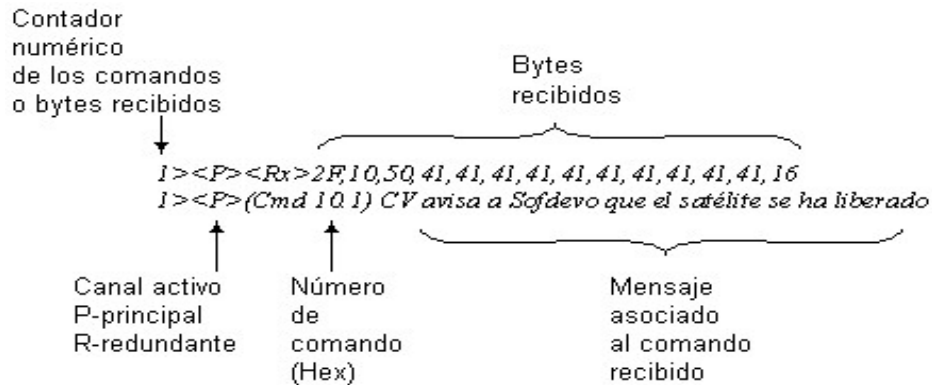
### 4.3 Software asociado con operaciones satelitales que son detectadas exclusivamente con SOFDEVO

El software de operación satelital que registra SOFDEVO está relacionado con todas las solicitudes informativas que realiza la CV a cada uno de los experimentos, SOFDEVO responde por medio de un comando de red interna ,si es necesario hará eco de esta respuesta en las pantallas de monitoreo, en caso contrario solo hará eco de la solicitud recibida.

Particularmente SOFDEVO puede generar dos tipos de respuestas a los comandos que circulen por la red interna:

- ✓ **Automáticas.** Entiéndase por automáticas todas aquellas respuestas cuya información está previamente definida dentro del código de SOFDEVO.
- ✓ **Programables.** Aquellas cuya información pueda ser modificada por el usuario.

La información que genera SOFDEVO se presenta, ya sea por medio de mensajes desplegados en las pantallas de monitoreo, o bien con el encendido o apagado de algún botón virtual de la pantalla principal. El formato de la información desplegada en las pantallas de monitoreo Rx y Tx de SOFDEVO, es el siguiente:



#### 4.3.1 Avisos generados al detectar comandos asociados con operaciones satelitales

Los avisos generados al detectar comandos asociados con operaciones satelitales se presentan cuando se envía nueva misión desde la estación terrena al SATEX. Estos presentan información sobre acciones que difícilmente se podrían apreciar sin SOFDEVO, como el inicio o fin de procesos y la correcta ejecución de los comandos. Estos son:

- ✓ **Liberación de satélite.** Se presenta cada vez que se da reset a la computadora de vuelo e indica que el software de operaciones ha comenzado a trabajar.
- ✓ **Contacto del Satélite.** Paso previo al envío de misión, asegura que Tierra y el satélite han iniciado comunicaciones.

- ✓ **Programación de nueva Misión.** Confirma que se le ha enviado a SATEX una nueva misión.
- ✓ **Envío de telemetría.** Indica que el satélite ha reportado a Tierra el estado operativo de computadoras y valores de sensores que tiene el microsatélite.
- ✓ **Aviso de inicio o fin de prueba en Kourou.** Prueba que reporta datos de telemetría del satélite cuando éste se encuentra en línea de vista con la Estación Terrena.

### 4.3.2 Respuestas programables para comandos de red interna asociados con operaciones satelitales.

Las respuestas programables permiten al usuario cambiar los valores de respuesta a solicitudes informativas por parte de la CV; SOFDEVO cuenta con respuestas programables para:

- Emitir reporte de fallas durante la simulación de diagnóstico de procesadores
- Simular fallas de red interna
- Simular señales de reconfiguración para la CV
- Simulación de señales de sensores de SP

La figura 4.1 presenta un ejemplo de este tipo de respuestas.

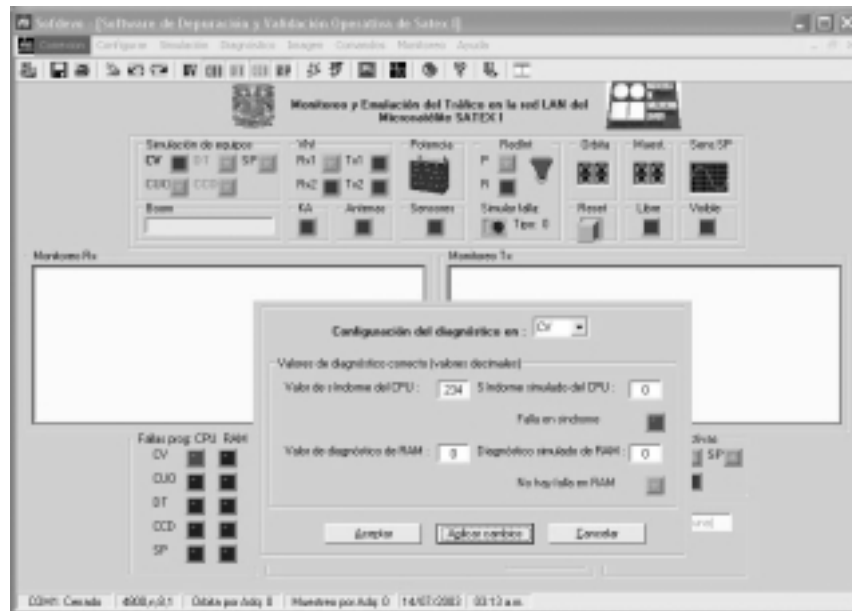


Figura 4.1. Respuesta programable de SOFDEVO asociada con el experimento de MACV.

### 4.3.3 Respuestas automáticas para comandos de red interna asociados con operaciones satelitales

Las respuestas automáticas simulan la entrega de datos por parte de las cargas útiles como respuesta a una solicitud hecha por CV, en este caso el usuario de SOFDEVO no tiene control de los valores de respuesta a cada solicitud. La figura 4.2 presenta un ejemplo de este tipo de respuestas.

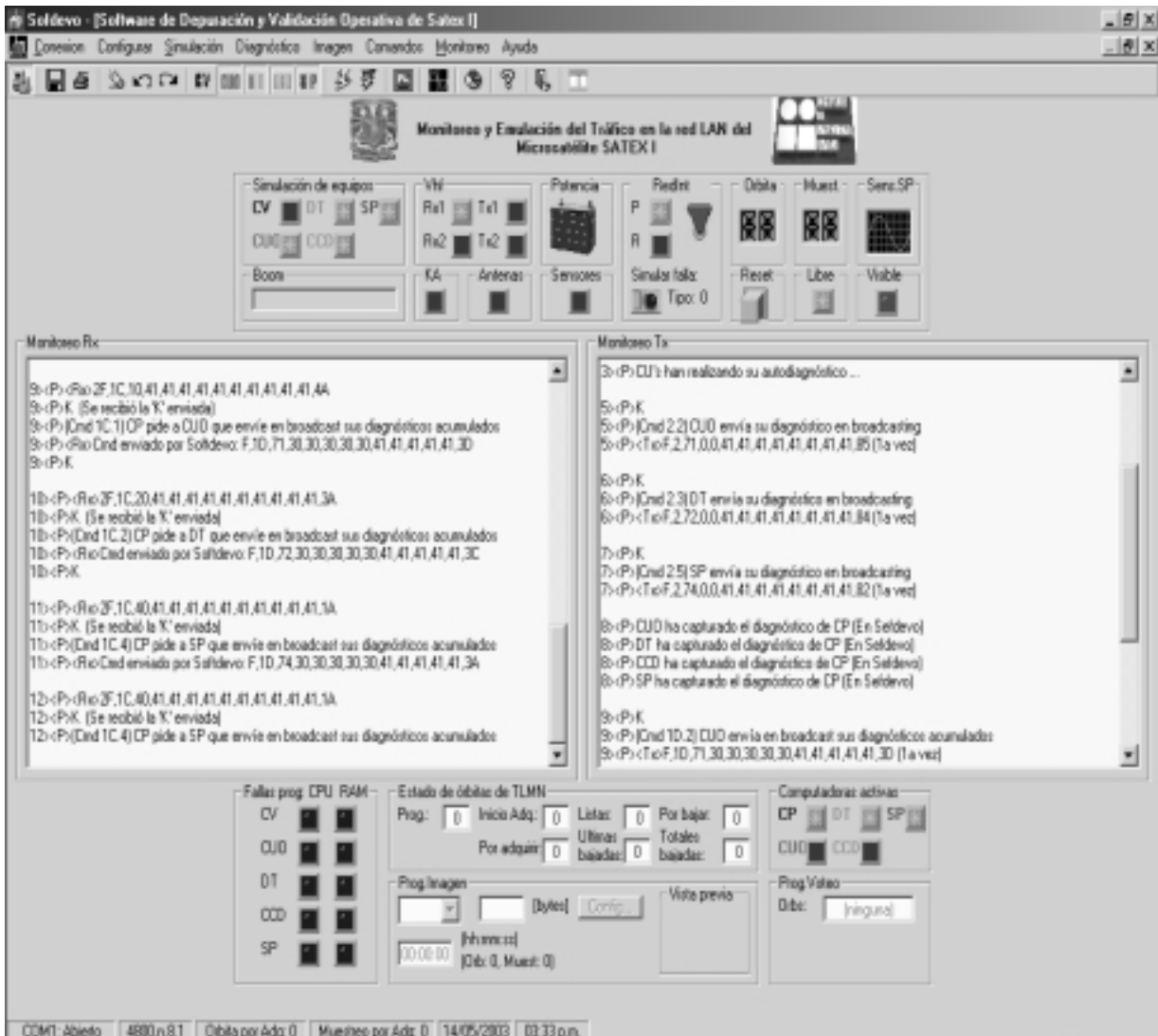


Figura 4.2. Respuestas automáticas de SOFDEVO asociados con el experimento de MACV.

#### 4.4. Tramas y protocolo de comunicaciones

SOFDEVO interactúa siempre con la Computadora de Vuelo, para ello emplea tramas y protocolos de comunicaciones especialmente desarrolladas para el proyecto microsatelital.

El proceso de comunicación entre las computadoras del SATEX es un proceso síncrono, debido a que la computadora de vuelo coordina el envío y la recepción de los comandos que circulan a través de la red. Por lo tanto todas las tareas ejecutadas por medio de comandos de red interna se inician con el envío de un comando desde la computadora de vuelo hacia alguna o hacia todas las computadoras. Para que este proceso se realice de manera exitosa y se garantice la ejecución de los comandos, estos deben de cumplir con una estructura definida para que su decodificación sea correcta.

**CAPÍTULO 4. SOFDEVO**

La trama utilizada se presenta en la figura 4.3 (para más detalle, ver el capítulo 3).

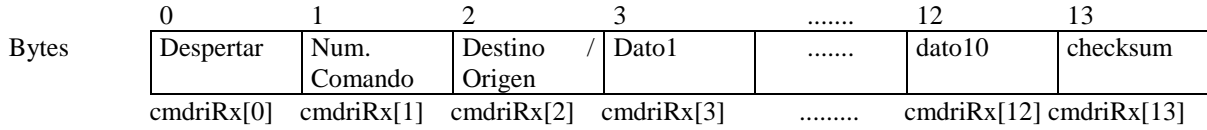


Figura 4.3. Trama de comandos por Red Interna.

El byte de checksum es calculado por SOFDEVO mediante la siguiente rutina:

```

Dim Check1 As Double
Dim Check2 As Double
Dim Index As Integer
    Check1 = 0
    Check2 = 0
    'Inicio de checksum
    For Index = 2 To 13
        Check1 = Check1 + Asc(Mid(texto_tx, Index, 1))
    Next Index
    Check1 = Check1 And &HFF
    Check2 = 0 - Check1
    Check2 = Check2 And &HFF

```

Al recibir un comando SOFDEVO calcula el checksum de acuerdo con la función anterior y lo compara con el valor recibido en el comando, este valor se utiliza para verificar que el comando se haya recibido adecuadamente.

Una vez definida la trama requerida para transmitir comandos, se utiliza un protocolo que permite la comunicación entre computadoras. Las figura 4.4 y 4.5 presentan diagramas de flujo que describen el proceso utilizado para el envío de información por red interna de SATEX, el proceso de recepción de información se muestra en la figura 4.6.

En estos diagramas se observa que al no recibir respuesta al comando enviado, este se retransmite una vez más antes de cambiar de canal de comunicaciones, este proceso se repite hasta alcanzar cuatro intentos de envío de comando (dos en cada canal), y de no recibir respuesta se reporta fallo total en la red interna.

Al recibir la respuesta por parte del equipo al que se envió el comando, es necesario identificar si el comando requiere, a su vez, de otro comando como respuesta. De ser necesario se espera la transmisión del comando de respuesta; si no el proceso termina. En el caso específico de comandos enviados por “broadcasting” no se espera respuesta al comando con el fin de evitar errores de sincronización.



**CAPÍTULO 4. SOFDEVO**

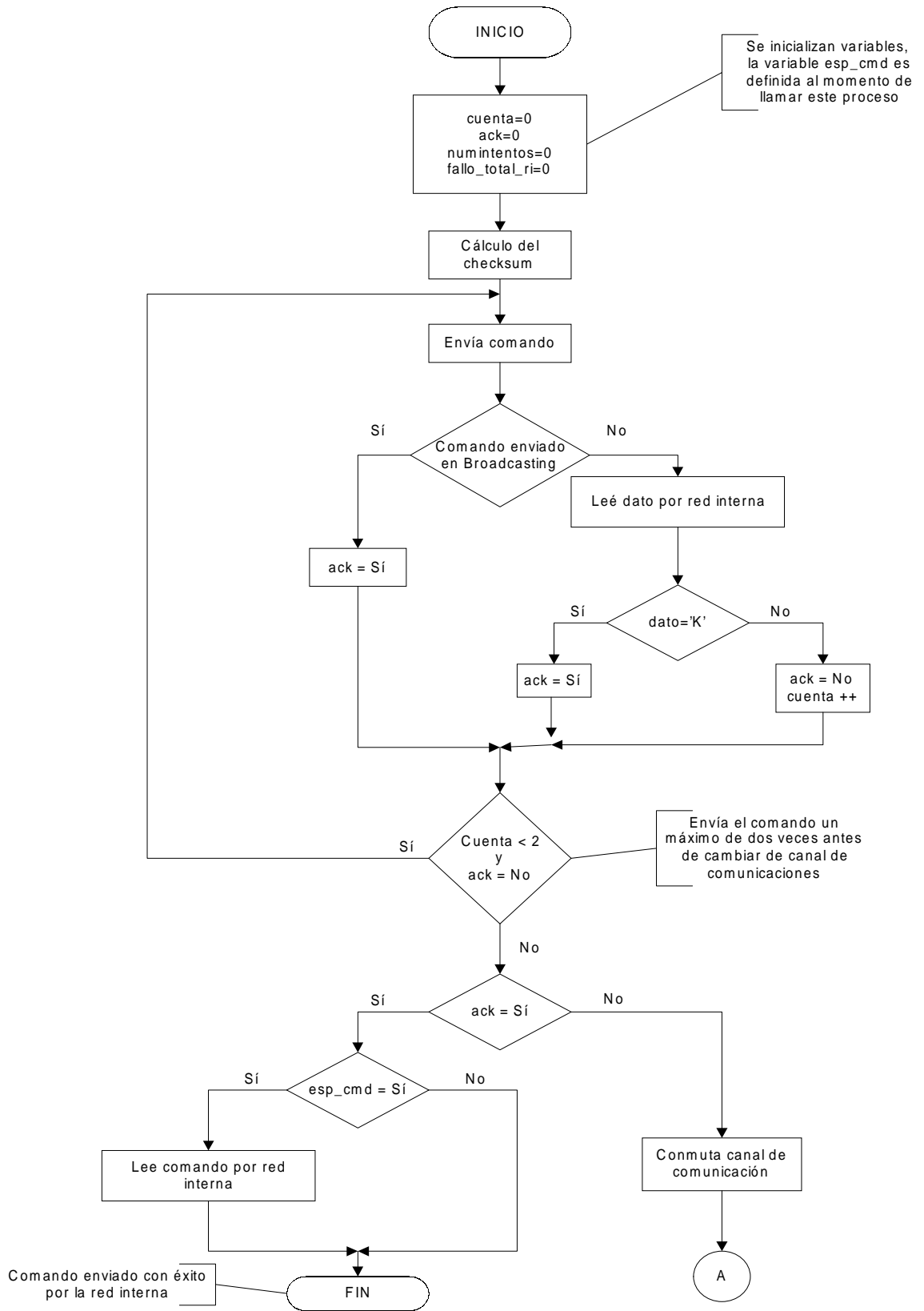


Figura 4.4. Protocolo de envío de información por red interna.

CAPÍTULO 4. SOFDEVO

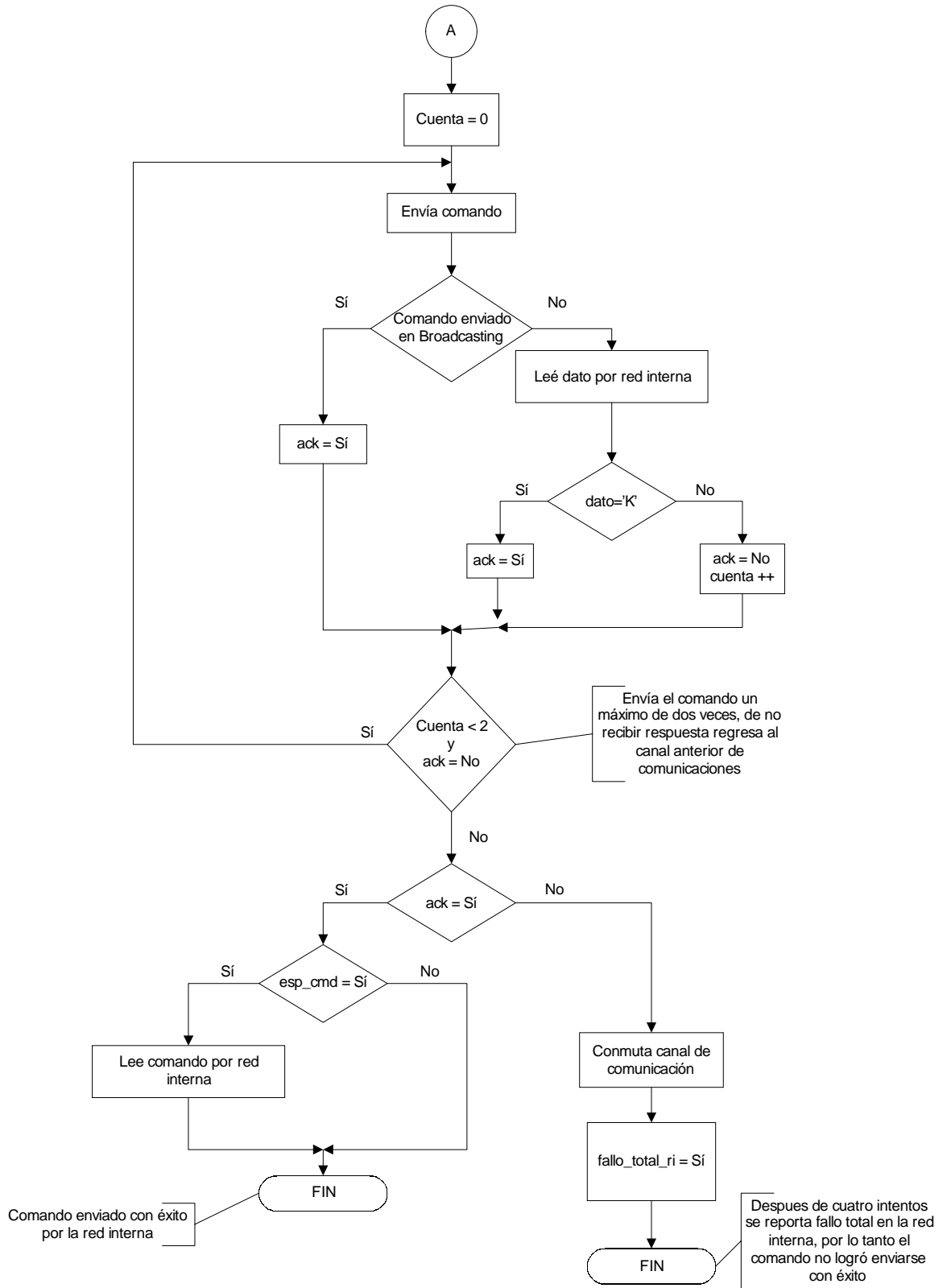


Figura 4.5. Protocolo de envío de información por red interna

CAPÍTULO 4. SOFDEVO

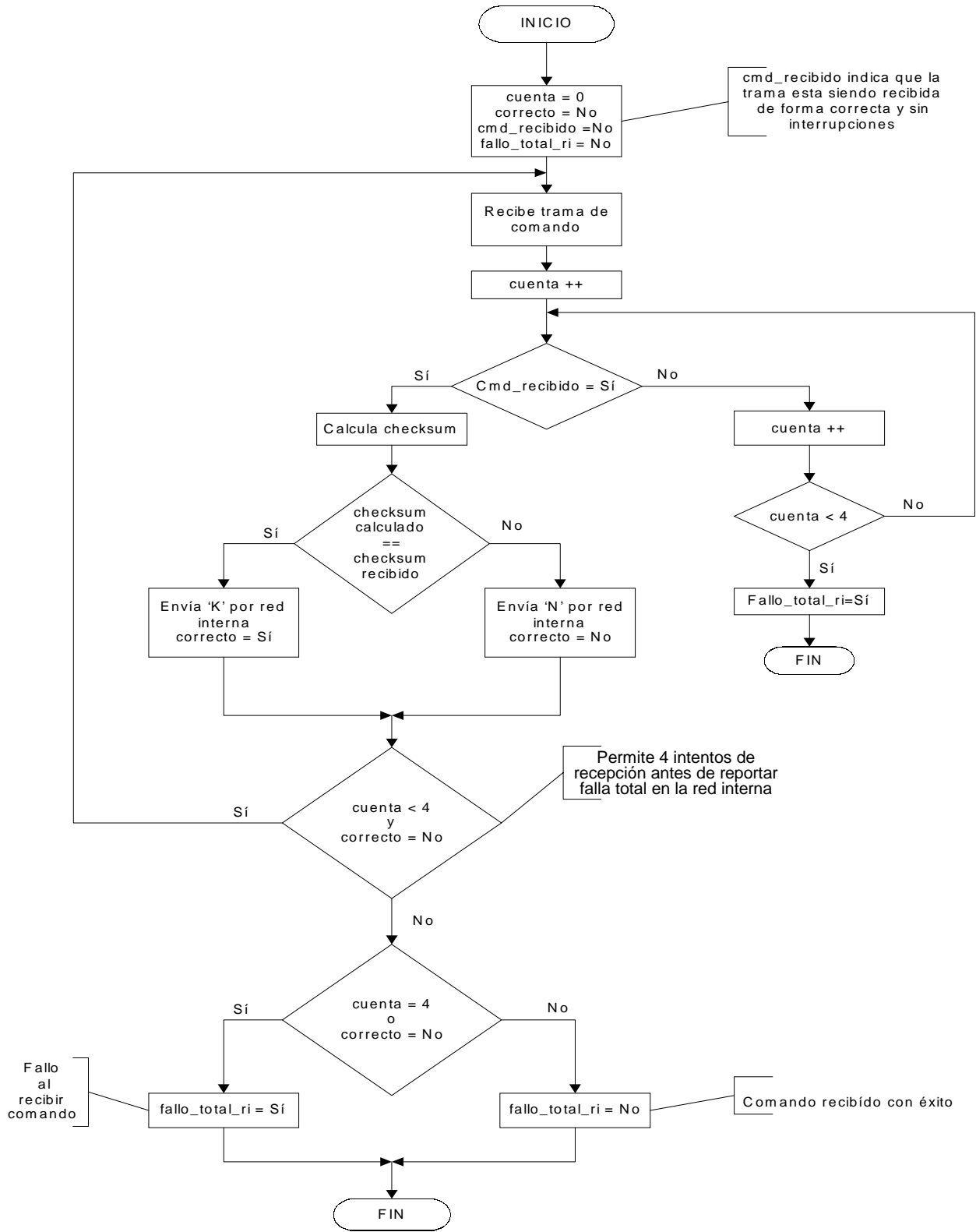


Figura 4.6. Protocolo de recepción de información

#### 4.5 *Software asociado con el experimento de comunicaciones ópticas*

El experimento de comunicaciones ópticas o Carga Útil Óptica (CUO) pretende validar un sistema de comunicaciones ópticas por medio de un láser que se envía desde el satélite hacia Tierra. Inicialmente la CUO se encarga de rastrear un “beacon” enviado desde Tierra, de tal forma que al detectarlo inicia el proceso de comunicación óptica. Para efectos de esta tesis sólo se pretende verificar el tráfico de información que envía y recibe este experimento.

El software asociado con este experimento da respuesta a los comandos recibidos por red interna que requieren respuesta por parte de este experimento, a continuación se listan los comandos a los cuales se responde automáticamente.

Número de Comando	Solicitud que realiza
Comando 1	Diagnóstico
Comando 4	Transferencia de datos
Comando 14	Envío de comandos en Stack
Comando 17	Ejecución de nuevo programa
Comando 1C	Envío de diagnósticos acumulados
Comando 40	Resultado de Votéo
Comando 41	Generación de señales de permiso y reconfiguración de CV

#### 4.6 *Software asociado con el experimento de la cámara digital de percepción remota*

El software asociado con este experimento da respuesta a los comandos recibidos por red interna que requieren de respuesta por parte de este experimento, a continuación se listan los comandos a los cuales responde automáticamente.

Número de Comando	Solicitud que realiza
Comando 1	Diagnóstico
Comando C	Adquisición de imagen
Comando 16	Transferencia de imagen
Comando 1C	Envío de diagnósticos acumulados
Comando 40	Resultado de Votéo
Comando 41	Generación de señales de permiso y reconfiguración de CV

#### 4.7 *Software asociado con el experimento del sistema mínimo de sobrevivencia*

El software asociado con este experimento da respuesta a los comandos recibidos por red interna que requieren de una respuesta por parte de este experimento, a continuación se enlistan los comandos a los cuales responde.

CAPÍTULO 4. SOFDEVO

Número de Comando	Solicitud que realiza
Comando 1	Diagnóstico
Comando 3	Liberación del Boom
Comando 8	Conmutación de CV
Comando 14	Envío de comandos en stack
Comando 1C	Envío de diagnósticos acumulados
Comando 40	Resultado de Votéo
Comando 41	Generación de señales de permiso y reconfiguración de CV
Comando 42	Envío de estado de procesadores de CV almacenados en DT

En las figuras 4.7, 4.8, 4.9 y 4.10 se presenta el diagrama de flujo de votéo.

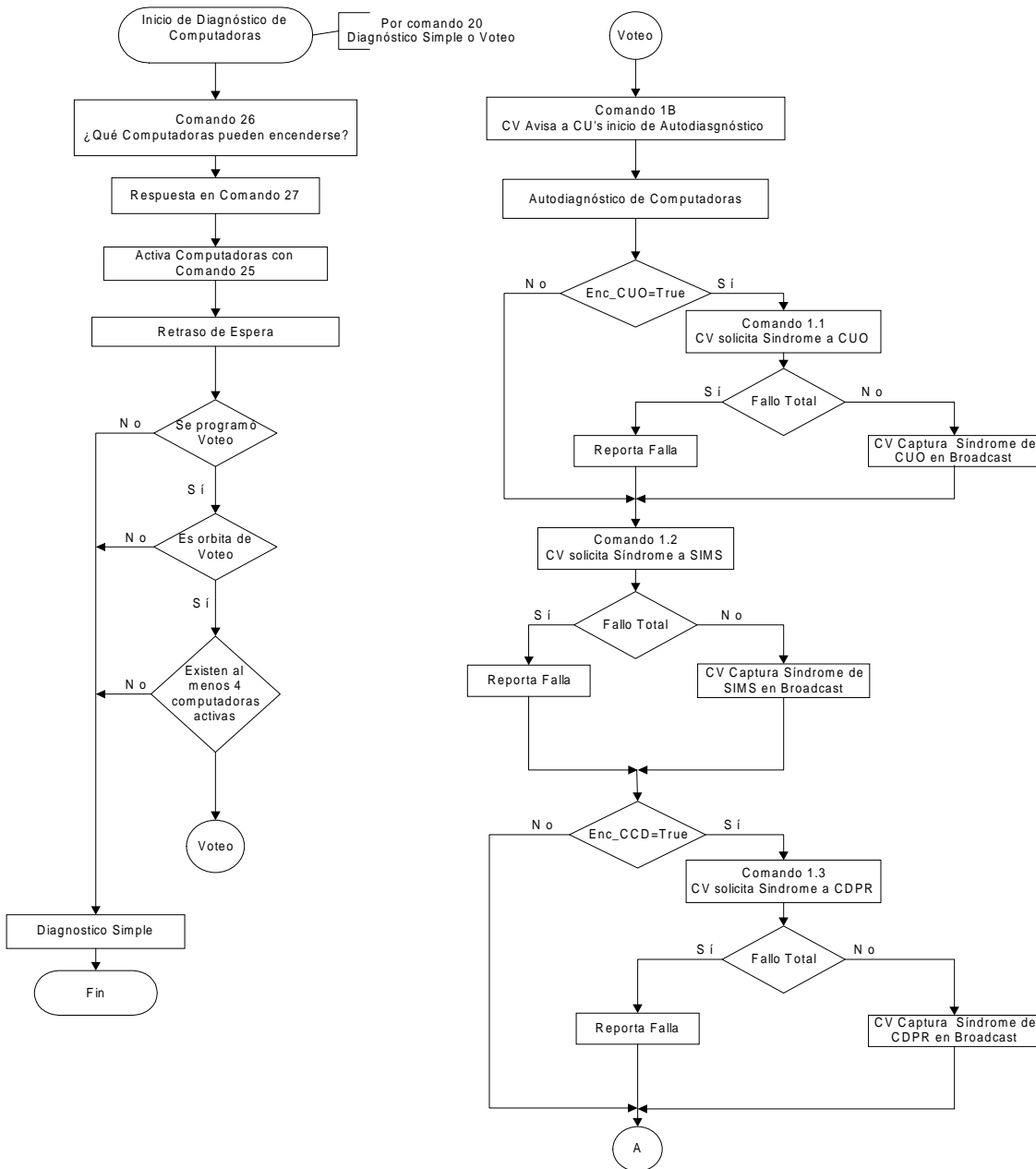


Figura 4.7. Diagrama de flujo del experimento de MACV

**CAPÍTULO 4. SOFDEVO**

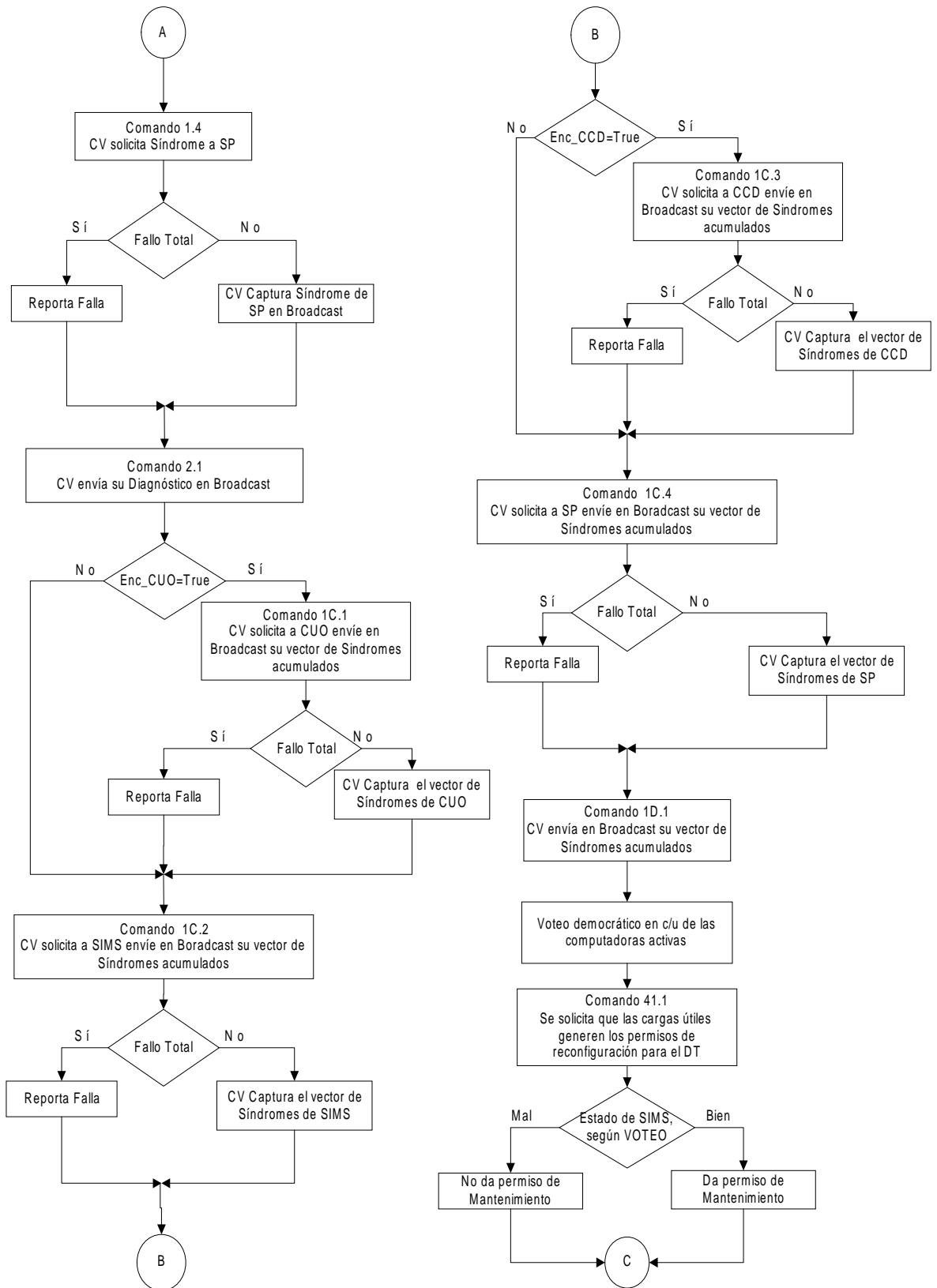


Figura 4.8. Diagrama de flujo del experimento de MACV

**CAPÍTULO 4. SOFDEVO**

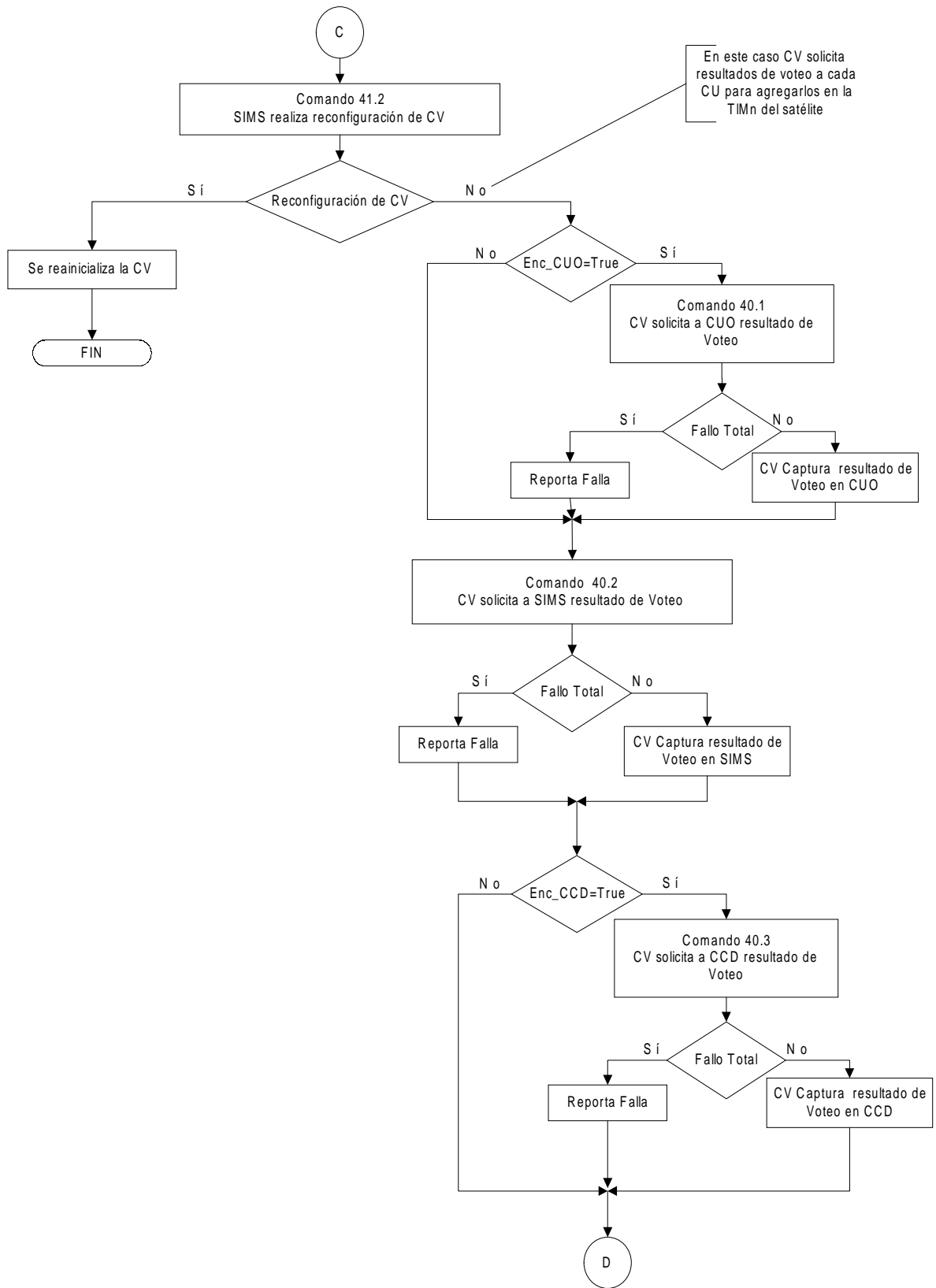


Figura 4.9. Diagrama de flujo del experimento de MACV

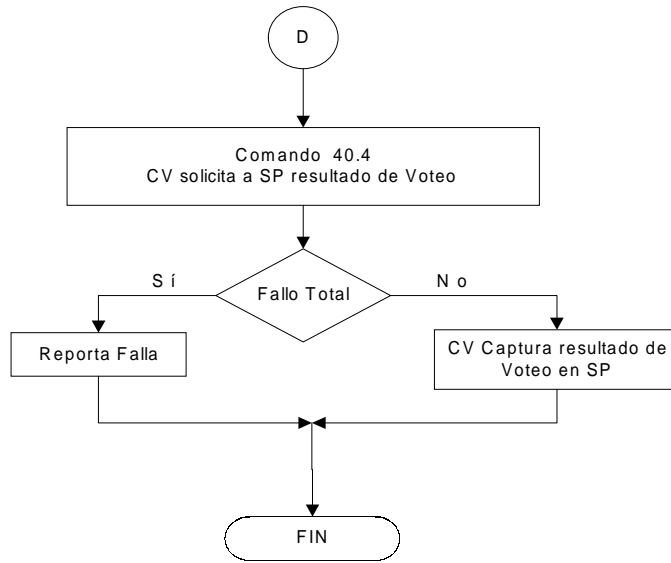


Figura 4.10. Diagrama de flujo del experimento de MACV

#### 4.8 Software asociado con el experimento de MACV

Este experimento tiene como objetivo la conmutación automática de las tarjetas de procesamiento de la computadora de vuelo, para esto se desarrolló en el Instituto de Ingeniería de la UNAM (IINGEN) un algoritmo que permite realizar el autodiagnóstico, detección de fallas y reconfiguración de las tarjetas de procesamiento.

Para la automatización de este proceso se implantó una arquitectura de cómputo semivirtual tolerante a fallas, la organización topológica del hardware se controla desde la estación terrena por medio de software y está conformada por una parte física (la computadora de vuelo) y una virtual (las computadoras de las cargas útiles), el proceso de detección de fallas se realiza por medio de voto mayoritario en cada uno de los nodos de esta estructura semivirtual. La figura 4.11 muestra la forma en que esta estructurada la arquitectura de cómputo semivirtual.

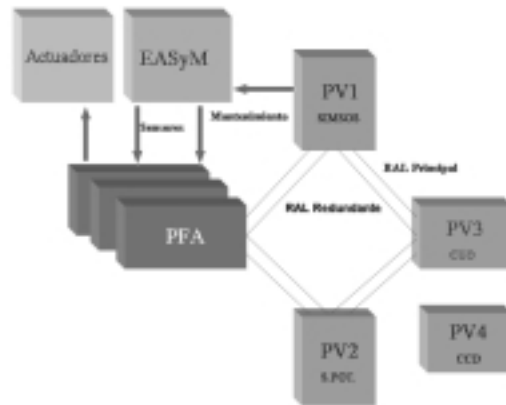


Figura 4.11. Arquitectura de cómputo semivirtual tolerante a fallas



De acuerdo a los resultados de voto, las cargas útiles emiten su opinión acerca del estado de la tarjeta de procesamiento de la computadora de vuelo y generan señales para que el sistema mínimo de sobrevivencia pueda conmutar de tarjeta de procesamiento en caso de reportarse una falla en la tarjeta activa de la CV al momento de realizar este proceso.

La figura 4.12 presenta los elementos requeridos para realizar este experimento sin la necesidad de contar con la presencia de las cargas útiles. Este proceso se describe en los siguientes capítulos.



Figura 4.12. Equipo requerido para realizar el experimento de MACV

Al ser MACV un experimento que demanda la interacción y participación de todas las computadoras del satélite, es necesario conocer detalladamente su proceso operativo así como sus comandos asociados, los cuales fueron validados con el apoyo de SOFDEVO. El objetivo de este experimento es obtener información sobre el estado del procesador activo de la CV, para decidir si se encuentra en buenas condiciones. Si se determina que el procesador activo de la CV presenta una falla, de forma automática se conmutará al procesador de reserva.

#### 4.9 Software asociado con la operación del subsistema de potencia

El software asociado con este experimento emula a los sensores del subsistema de potencia. Para esto, SOFDEVO genera valores de telemetría que corresponden a señales triangulares, cuadradas y senoidales, a las que el software puede modificar su amplitud pico a pico. La forma en que se programan estas señales se muestra en la figura 4.13.

## CAPÍTULO 4. SOFDEVO

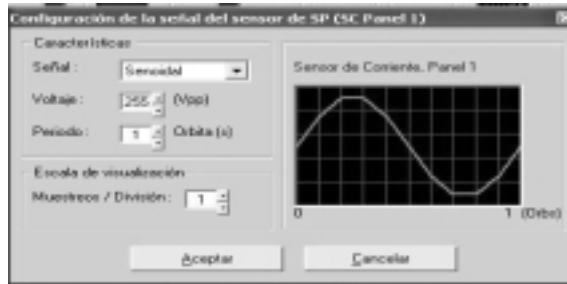


Figura 4.13. Ventana de SOFDEVO que permite definir el tipo de señal a emular.

Las muestras de las señales referidas son enviadas por SOFDEVO a la CV cada vez que ésta se los solicita por medio del comando 29h.

Otra parte del software permite la programación de la potencia disponible en el SP para encender los equipos del satélite. La CV solicita esta información al SP por medio del comando 26h, y la figura 4.14 muestra la programación de la potencia disponible para encender cargas útiles.



Figura 4.14. Programación de la potencia disponible en el SP para encender cargas útiles

Además de atender a estas solicitudes, el software asociado al subsistema de potencia admite y ejecuta los siguientes comandos:

Número de Comando	Solicitud que realiza
Comando 1	Diagnóstico
Comando 14	Envío de comandos en Stack
Comando 1C	Envío de diagnósticos acumulados
Comando 20	Encendido o apagado de radios
Comando 22	Liberar el Boom
Comando 24	Encendido o apagado de la CDPR
Comando 25	Encendido o apagado de procesadores para diagnóstico simple
Comando 26	Información sobre energía disponible para encender equipos Para diagnóstico simple o experimento de MACV
Comando 28	Envío de información de telemetría de sus sensores
Comando 40	Resultado de voto

## Capítulo 5

### *Actualización del Software de Operaciones del Satélite y del software SOFDEVO*

---

#### **5.1 Introducción**

En este capítulo se describen las actualizaciones al Software de Operación Satelital, contemplando la actualización de niveles de interrupción para la red interna de SATEX, la depuración del software mínimo que permite la sobrevivencia del satélite en el espacio, las técnicas de compilación tanto para el Software de Operación Satelital como para el Software Mínimo de Operación Satelital (SOFMIN), la depuración y actualización del experimento MACV, la actualización de comandos de red interna para el experimento MACV, la incorporación de los eventos del experimento MACV en SOFDEVO, la adición de mensajes para las cargas útiles emuladas por SOFDEVO, el software de control para el dispositivo EDAC y el software de control de tiempo orbital.

#### **5.2 Interrupción de Red Interna de SATEX**

Un procedimiento de prueba que es de gran utilidad para conocer el funcionamiento instantáneo de todos los equipos que integran el microsátélite es la rutina llamada Prueba en Kourou. Esta prueba realiza el diagnóstico de todas las computadoras de SATEX durante tres muestreos rápidos, todos realizados en línea de vista con la Estación Terrena lo cual indica que es un comando en línea de la ET.

La rutina de Prueba en Kourou está diseñada para que el personal en Tierra pueda saber con rapidez el estado operativo de las computadoras que integran al SATEX. El diagrama de flujo de la estructura de esta rutina se presenta en la figura 5.1 en donde se observa que la configuración de las interrupciones que inicialmente se plantearon es errónea para que la rutina de Prueba en Kourou se realice a la perfección.

La interrupción cuyo nivel se asignó erróneamente es la interrupción de red interna **S1RIC**, la cual tenía un nivel de interrupción 10, que es inferior al nivel de interrupción de red externa que tiene un nivel de 11. Esto quiere decir que la interrupción de red interna no tenía un nivel de prioridad alto, entonces, los comandos de red interna no se ejecutaban.

En la tabla 5.1 se observa la actualización en el nivel de interrupción y se destaca el nivel de interrupción de red interna que ahora es la de mayor nivel, la número 12.

CAPÍTULO 5. ACTUALIZACIONES.

Tabla 5.1. Actualización de interrupciones de SATEX

Registro SAB80C166 / Función		Grupo	Nivel
CC15IC	Interrupción del pin 15, puerto 2 para lectura del EDAC	3	15
ADCIC	Lectura del convertidor Analógico/Digital	0	14
T5IC	Interrupción del <i>timer</i> 5 (control de estabilización)	3	13
T6IC	Interrupción del <i>timer</i> 6 (control de estabilización)	3	12
S0RIC	Interrupción en la recepción del puerto serie 0 (Red externa)	3	11
S1RIC	Interrupción en la recepción del puerto serie 1 (Red Interna)	3	12
T2IC	Interrupción del <i>timer</i> 2 (Control de tiempos orbitales)	3	9
T4IC	Interrupción del <i>timer</i> 2 (Contabilización de tiempos orbitales)	3	8
T0IC	Interrupción del <i>timer</i> 0 (Retrasos en tiempos de protocolos)	3	5

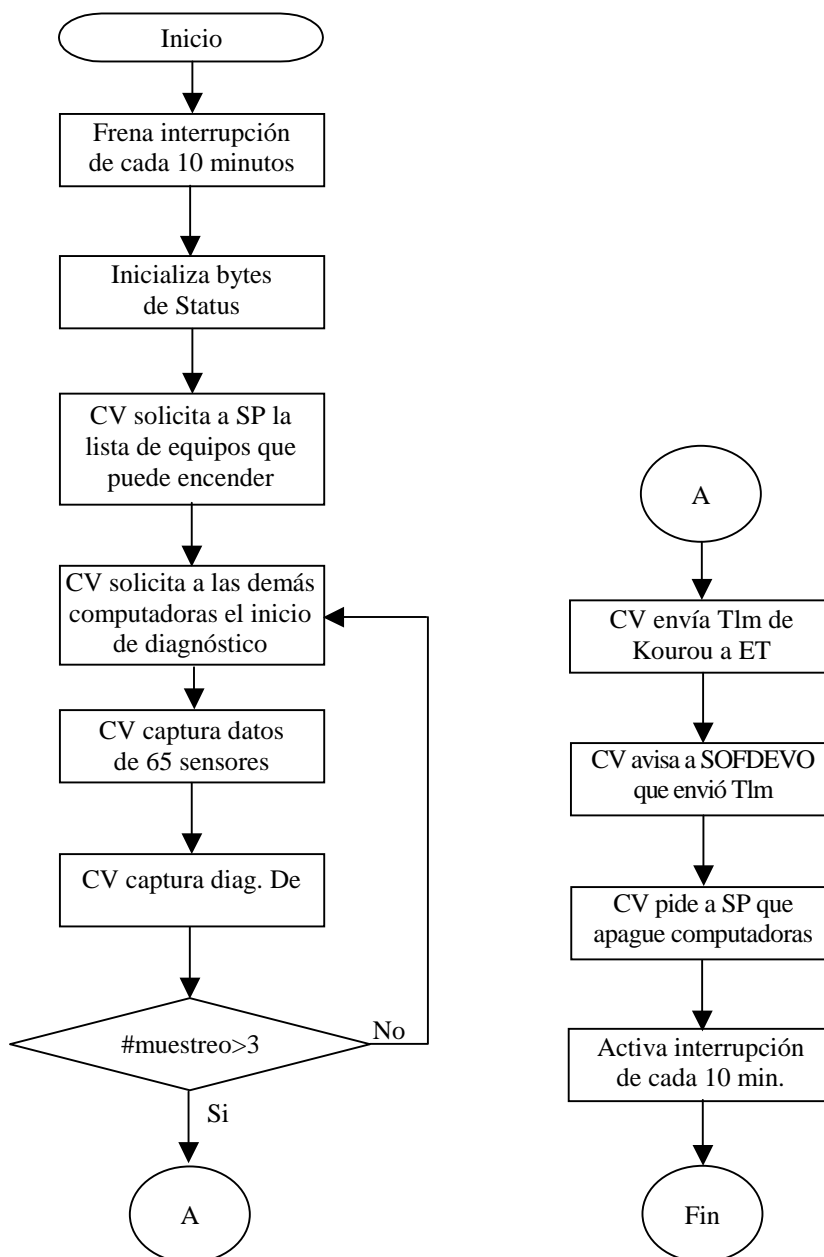


Figura 5.1. Diagrama de Flujo del Comando 16 (Prueba en Kourou)

Durante pruebas preliminares del software del satélite se detectó que la CV no realizaba la rutina de Prueba en Kourou, es decir, la ET contactaba al satélite y le ordenaba que hiciera Prueba en Kourou, pero la CV no ejecutaba la rutina y seguía con sus labores de misión. Ahora después de asignar el nuevo nivel de interrupción de la red interna, la CV tiene prioridad con los comandos de red interna y así realiza todas las pruebas relacionadas con la ET.

### **5.3. Dispositivo electrónico EDAC**

Los sistemas espaciales están expuestos a condiciones de radiación que afectan la operación de sus componentes electrónicos, un efecto adverso que se presenta con memorias se le conoce como “Single Event Upsets” (SEU).

Debido a las dimensiones tan pequeñas de los componentes electrónicos modernos, una partícula cargada de alta energía puede depositar a su paso suficiente carga para cambiar un dígito binario. Cuando esto sucede, un bit guardado en RAM puede convertir, por ejemplo, una instrucción de algún programa almacenado en otra instrucción que podría generar caos en todo el sistema.

Debido a que es prácticamente imposible predecir el tiempo o lugar en el que un SEU ocurrirá, el diseño del sistema debe asumir que este puede ocurrir en cualquier parte y momento. Los SEU's pueden ser detectados y corregidos usando códigos de detección y corrección de errores (EDC), vigías de tiempo (“Watchdog Timers”), retornos en falla (Fult Rollback) y otros métodos.

El caso de SATEX no es la excepción, por ello se propuso la utilización de un dispositivo para la detección y corrección de errores en RAM, EDAC por sus siglas en inglés “Error Detection And Correction unit”, que se basa en la utilización de una palabra de síndrome para la detección de errores.

La función básica de un dispositivo EDAC es analizar la integridad de los datos leídos del sistema de memoria, señalar errores ocurridos y corregirlos si es posible. La mayoría de los dispositivos EDAC implementan esta función usando los principios generales similares, con algunas variaciones entre cada dispositivo.

La operación de un dispositivo EDAC puede dividirse en:

- ✓ Generación de una palabra codificada basándose en la palabra de datos que se escribe a memoria. Esta palabra codificada se llama “Check-bit Word” o palabra de chequeo. A esta operación se le llama Generación.
- ✓ Detección de errores en la palabra leída de memoria, mediante la comparación de la palabra de chequeo con una nueva palabra de chequeo (generada a partir de la lectura de la palabra de datos). Si es posible se efectúa la corrección del error, la comparación de las dos palabras de chequeo (una función lógica OR exclusiva) produce una palabra de síndrome, ésta operación se llama Detección/Corrección.

El esquema de codificación empleado por la mayoría de los EDAC es un código Hamming (desarrollado en los laboratorios AT&T Bell) con alguna variación. Por cada palabra de datos escrita a memoria, se genera una palabra de chequeo. La nueva palabra (palabra de datos + palabra de chequeo) se establece como un código válido. La distancia Hamming entre cada uno de los códigos válidos varía de dispositivo a dispositivo.

La distancia Hamming entre dos palabras es el número de bits en que difieren una de la otra, por ejemplo:

10001110	11100101
00111000	11110111
D=5	D=2

Dos datos son más fáciles de distinguir cuanto mayor sea su distancia Hamming, ya que si la distancia es  $d$  será necesario que se produzcan  $d$  errores para que una palabra se convierta en la otra. De este análisis se desprende que la eficacia de un código será función de su distancia Hamming, que se define como la mínima distancia que puede encontrarse entre dos palabras que pertenezcan a ese código.

En general:

- ✓ Un código de distancia mínima de Hamming  $d$  será capaz de detectar  $d-1$  errores.
- ✓ Un código de distancia mínima de Hamming  $d$  deberá ser capaz de corregir  $(d-1)/2$  errores.
- ✓ Un código que corrija  $t$  errores y detecte  $d$  ( $d > t$ ) errores debe tener una distancia mínima igual a  $d = t + 1$ .

Las principales reglas relativas al control de paridad en los códigos Hamming son:

- ✓ Dos bits de paridad no pueden controlar la paridad de un mismo conjunto de bits de información.
- ✓ No se puede incluir en el conjunto de bits controlado por un bit de paridad, otros bits de paridad.
- ✓ Un error en un bit de información debe afectar a dos o más bits de paridad.

### 5.3.1 EDAC 29C516E

El dispositivo EDAC 29C516E de TEMIC es una unidad de detección y corrección de errores de muy baja potencia, con dos buses de datos de usuario. Durante un ciclo de escritura de procesador, añade una palabra de chequeo (6 ó 8 Bit de longitud) por cada localidad de memoria (16 Bit de longitud). Cuando realiza una operación de lectura, el 29C516E verifica la combinación entera de la palabra de chequeo y de datos. Puede detectar y corregir el 100% de los errores en un solo bit y detecta todos los errores en dos bits. Todos los errores son señalizados al sistema maestro (mediante 2 banderas de error) para permitir que el procesador efectúe la acción necesaria. Para mayor información consulte [ORTIZ, 2003].

**CAPÍTULO 5. ACTUALIZACIONES.**

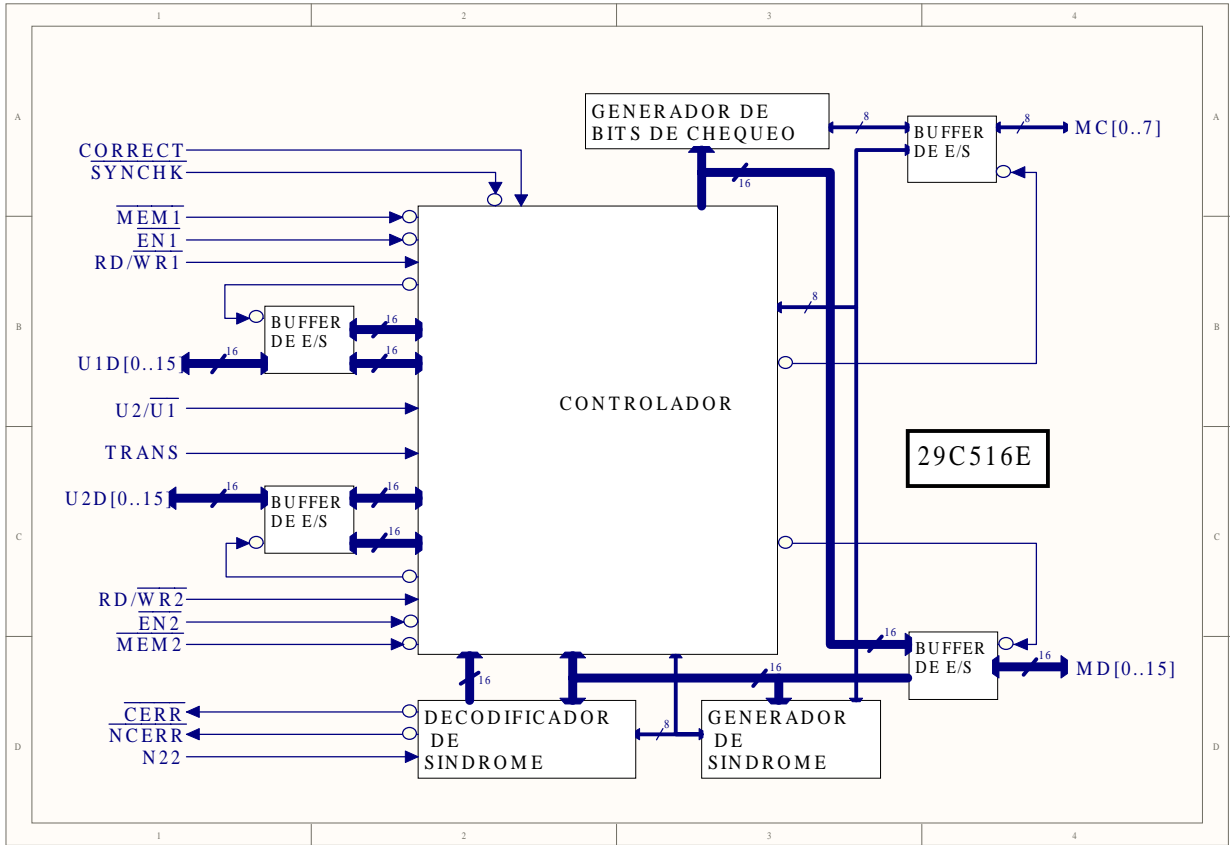


Figura 5.2. Diagrama funcional del 29C516E

El 29C516E opera en dos modos posibles: Modo Corrección y Modo Detección. En el modo Corrección, los errores de 1 bit se corrigen, entonces la palabra corregida de datos se envía al puerto de salida y se activa la bandera “Correctable Error Flag” (bandera de Error Corregido). En el caso de errores de doble bit (o más), los datos corrompidos se envían al puerto de salida y se activa la bandera “Uncorrectable Error Flag”. Hay que notar que algunos patrones de múltiples bits, pueden aparecer como posibles errores corregibles [TEMIC, 1997-2]. En nuestro caso la probabilidad de ocurrencia de este tipo de errores es despreciable [PISACANE, 1994]. La figura 5.2 muestra el diagrama funcional del dispositivo 29C516E [TEMIC, 1997-2].

El dispositivo 29C516E proporciona las señales de control, datos y polarización [TEMIC, 1997-2] que se indican en la tabla 5.2.

Tabla 5.2. Descripción de los alfileres del 29C516E

Nombre	Num. de alfiler	I/O	ACTIVO	DESCRIPCIÓN
<b>BUSES</b>				
U1D[0..15]	53,49..47,45..42,40..37,35..33,28	I/O*	ALTO	Datos del usr 1
U2D[0..15]	23..28,18..15,13..10,8..5	I/O*	ALTO	Datos del usr 2
MD[0..15]	59..62,64..67,69..72,74..77	I/O*	ALTO	Datos de memoria
MC[0..7]	83..86,88..91	I/O*	ALTO	Memoria de chequeo
<b>BANDERAS DE ERROR</b>				
CERR#	26	O	BAJO	Error corregible
NCERR#	25	O	BAJO	Error no corregible

**CAPÍTULO 5. ACTUALIZACIONES.**

SEÑALES DE CONTROL GENERAL				
CORRECT	98	I*	ALTO	Cuando esta activa, el EDADC esta en modo Corrección, si no, el EDAC esta en modo Detección
SYNCHK#	97	I*	BAJO	Selecciona si la palabra de síndrome y la palabra de chequeo son enviados por el bus de datos de usuario (byte alto y byte bajo respectivamente)
N22	27	I*	ALTO	Cuando esta activa, el EDAC usa 6 bit de chequeo, si no, usa 8 bit
TRANS	96	I*	A/ B	Selecciona la ruta de datos que se usara. Si es alto, el EDAC accede a memoria si es bajo accede al buffer de transferencia.
U2/U1#	95	I*	A/B	Selecciona quien es el maestro Usr1 o Usr2, el maestro es responsable de aplicar las señales: RD/WR#, MEMx# y ENx de forma correcta
SEÑALES DE CONTROL DEL USUARIO 1				
RD/WR1#	55	I*	A/B	Señal de lectura/escritura del usuario 1
EN1#	56	I*	BAJO	Habilitación de salida de usuario 1
MEM1#	57	I*	BAJO	Selector de memoria de usuario 1
SEÑALES DE CONTROL DEL USUARIO 2				
RD/WR2#	99	I*	A/B	Señal de lectura/escritura del usuario 2
EN1#	94	I*	BAJO	Habilitación de salida de usuario 2
MEM2#	3	I*	BAJO	Selector de memoria de usuario 2
BUFFERS DE POTENCIA				
VCC <sub>B</sub>	9,19,32,41,54,63,73,87	I	---	Alimentación de buffers (5Volts Nominales)
GND <sub>B</sub>	4,14,24,36,46,58,68,78,92	I	---	Referencia nominal de 0Volts de buffers
POLARIZACIÓN (NÚCLEO)				
VCC <sub>C</sub>	100	I	---	Alimentación del núcleo (5Volts nominales)
GND <sub>C</sub>	93	I	---	Referencia 0Volts del núcleo.
* <b>BUFFERS PULL-UP</b>				

En el caso de la computadora de vuelo de SATEX, el dispositivo EDAC no utiliza su capacidad para servir a 2 usuarios, por lo que las líneas de control: **U2/U1#**, **TRANS**, se encuentran fijas a 5Volts (se eligió al usuario 2 por cuestiones de localización de los buses de datos), además la línea **SYNCHK#**, también se encuentra fijada en 5Volts. Además se optó por apagar el procesador cuando se detecte un error no corregible ya que el aumento en el número de errores en la memoria RAM implica un aumento en la dosis de radiación recibida, lo que podría ocasionar el efecto Latch-up (técnica alterna para detectar excesos de radiación en el satélite) en el procesador.

La línea **N22**, se encuentra fija a 0Volts, para permitir la detección de todos los errores de doble bit en la palabra compuesta (Datos + Chequeo).

El esquema general de operación que debe cumplir el dispositivo EDAC, en la aplicación es el siguiente:

Al comenzar operaciones, el EDAC se encontrará funcionando en el modo deshabilitado (no corrección), por lo cual durante este tiempo no podrá activar el circuito de eliminación de efecto Latch-up, esto es por que inicialmente el contenido de la memoria de chequeo no concordaría con los síndromes de la memoria de datos,



## CAPÍTULO 5. ACTUALIZACIONES.

Para activar el EDAC, es necesario realizar un ciclo de escritura en todas las localidades de memoria externa, ya sea expandida o normal. La activación del EDAC se efectúa por medio de la línea **ENEDAC** (alfiler 72 del microprocesador, denominado **P2.10**)

Al activarse el EDAC deberá corregir todos los errores de un bit encontrados en la memoria RAM (Normal y Expandida), no en la ROM. Al corregir los errores, éstos serán notificados al procesador mediante el cambio de estado Alto->Bajo de la línea **CERRFLAG#** (alfiler 77 del procesador denominado P2.15).

Si el EDAC es incapaz de corregir el error en la memoria RAM (errores múltiples), activará el circuito de eliminación de efecto Latch-up, si y solo si la localidad con error no pertenece a la memoria expandida, mediante una transición alto->bajo en la línea **NCERRFLAG#**, la cual activará también el sensor de efecto Latch-up mediante una compuerta lógica AND. Debe resaltarse que en la memoria expandida no se alojarán datos críticos para el buen funcionamiento de la computadora de vuelo.

De todo lo descrito anteriormente y analizando todas las posibles combinaciones se obtiene la siguiente tabla de verdad [ORTIZ, 2003]:

Tabla5.3. Tabla de Verdad para el control del EDAC

Señales generadas por:	<b>Decodificador de memoria</b>	<b>ROMSEL#</b>	X	0	1	1	1	1	1
		<b>XMEMSEL#</b>	X	1	0	0	1	1	1
		<b>RAMSEL#</b>	X	1	1	1	0	0	0
	<b>MCU</b>	<b>ENEDAC</b>	0	1	1	1	1	1	
	<b>EDAC</b>	<b>CERR#</b>	X	X	0	1	X	0	1
		<b>NCERR#</b>	X	X	X	X	0	1	1
Señales de salida para:	<b>EDAC</b>	<b>CORRECT</b>	0	0	1	1	1	1	1
	<b>MPU</b>	<b>CERRFLAG#</b>	1	1	0	1	X	0	1
	<b>Sensor de efecto Latch-up</b>	<b>NCERRFLAG#</b>	1	1	1	1	0	1	1

Las líneas **ROMSEL#**, **XMEMSEL#** y **RAMSEL#** son generadas por el circuito de decodificación de memoria de la computadora e indican si el acceso es a ROM, memoria expandida o a RAM respectivamente, para más información consultar [ORTIZ, 2003].

De la tabla de verdad anterior y del hardware asociado al control del EDAC se tiene que se tienen las siguientes funciones para el control del EDAC:

**CORRECT** = **ENEDAC AND ROMSEL#**

**CERRFLAG#** = ( NOT **CORRECT** ) OR **CERR#**

**NCERRFLAG#** = ( NOT **CORRECT** ) OR **NCERR#** OR ( NOT **XMEMSEL#** )

### 5.3.2 Software de control de EDAC

El software de control del EDAC para SATEX realiza los procesos que se indican en la figura 5.3:

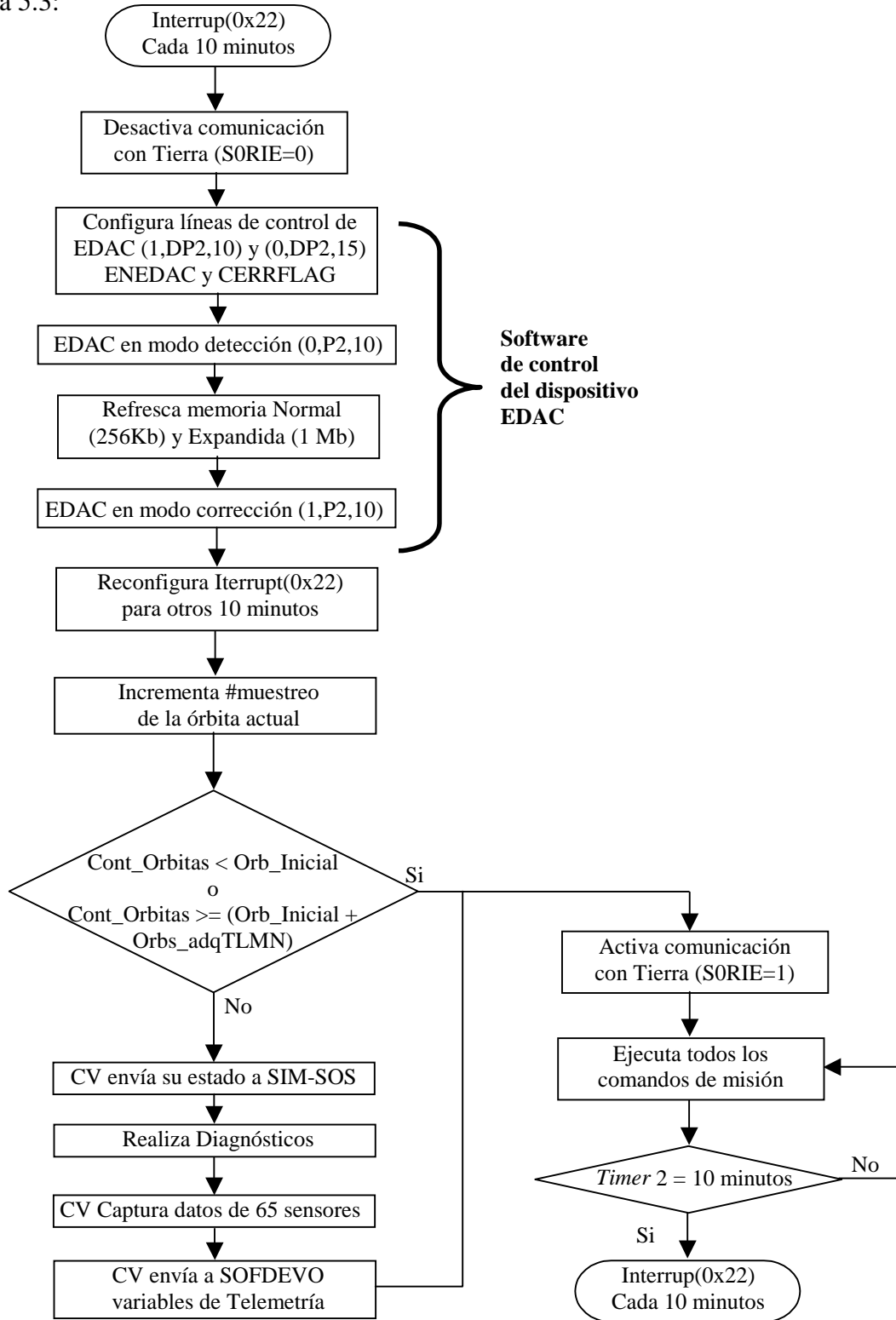


Figura 5.3. Diagrama de Flujo de la interrupción de cada 10 minutos Interrupt(0x22).

Cada interrupción de 10 minutos realiza un muestreo de los sistemas que controla la Computadora de Vuelo de SATEX y cada 10 muestreos representan una órbita del satélite. Entonces, la computadora de vuelo empaqueta datos de telemetría y los almacena en memoria cada 10 minutos.

La rutina que activa la forma de trabajar del EDAC se ejecuta al refrescar la memoria de la Computadora de Vuelo. En el diagrama de la figura 5.4 se ejemplifica la forma de refrescar toda la memoria de la Computadora de Vuelo. Se debe comprender que el EDAC siempre que se inicien las acciones de vida del satélite, estará en Modo Detección y pasa al Modo Corrección cuando comienza a capturar telemetría, es decir, cada 10 minutos.

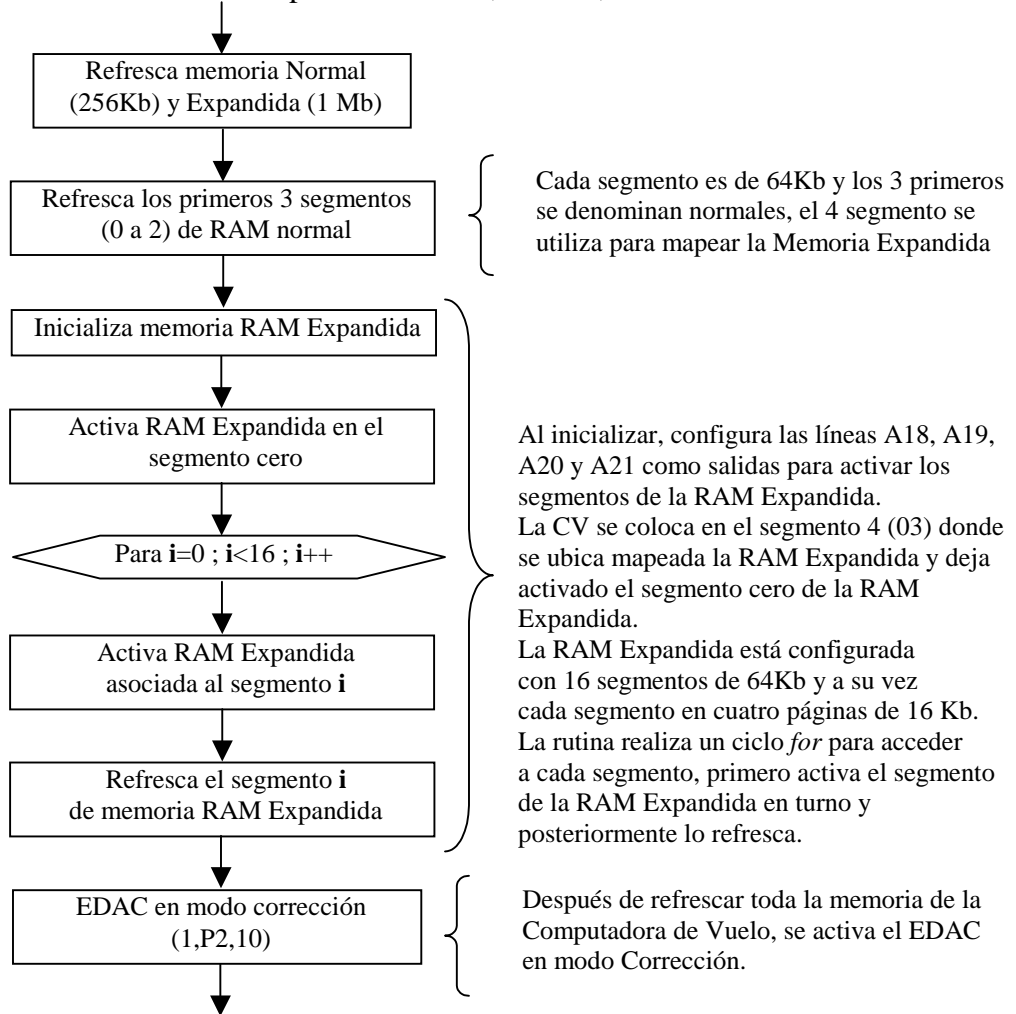


Figura 5.4. Detalle del diagrama que ejemplifica la forma de refrescar la memoria de la CV del satélite.

Al refrescarse la memoria Normal y Expandida —leer cada localidad de memoria y volver a escribirla con el mismo dato leído— el dispositivo EDAC genera y compara el síndrome nuevo con el síndrome viejo asociado a la localidad en turno, si encuentra errores corregibles los corrige y continúa. Al terminar el proceso de refrescamiento el EDAC tiene todas las palabras de síndrome actualizadas para que cuando el microprocesador requiera escribir o leer en memoria, el EDAC revise la información, detecte errores y los corrija. En el Apéndice B se puede observar el código fuente respectivo en lenguaje C.

## **5.4 Mantenimiento automatizado de la CV**

La CV interactúa con la mayor parte de los subsistemas del microsatélite para realizar diversas funciones de automatización. Entre ellas destacan las tareas de inicio de vida, la captura de telemetría, la comunicación con la Estación Terrena, las primeras tareas de control orbital para llevar el vehículo a una dinámica de vuelo segura para liberar el gradiente gravitacional, la liberación del gradiente gravitacional, las comunicaciones con los experimentos, etcétera. Ante esto, la CV constituye sin duda uno de los eslabones más críticos del satélite. Cuando sea orbitado el MES, éste permanecerá inaccesible para efectuarle operaciones de mantenimiento manual, por esta razón es importante que la CV tolere y resuelva diversos tipos de fallas en sus procesadores, memorias, líneas I/O y periféricos, debido a que de ellos dependen importantes funciones del vehículo espacial.

En cuanto a arquitecturas de computadoras tolerantes a fallas existen esquemas conocidos, [Siewiorek, 1982] y [Johnson, 1989], cuyas implantaciones conllevan fuertes penalizaciones de peso, espacio y consumo de potencia, sobre todo aquellas arquitecturas activas diseñadas con un fuerte acoplamiento por hardware. En aplicaciones críticas algunas de las técnicas que se utilizan más son las de redundancia modular activa por hardware (que realizan detección, ubicación y la recuperación ante fallas) como la duplicación con comparación y la de par y repuesto, las cuales conviven con resultados erróneos mientras el sistema se reconfigura para alcanzar nuevamente su estado operativo [Johnson, 1989], en otras palabras, estas arquitecturas no tienen la capacidad de mascarar fallas. La técnica de duplicación con comparación permite detectar fallas, pero no ubicarlas, por lo cual se le utiliza como técnica fundamental de detección de fallas en arquitecturas activas. Entre sus problemas principales se encuentran las fallas del módulo de comparación y las fallas genéricas si se utilizan procesadores iguales. Una solución para el módulo de comparación lo constituye la comparación por software con intercambio de parámetros para detectar fallas operativas.

Con el objeto de reducir las redundancias y su hardware de control, a la CV del MES se le desarrolló como una computadora con redundancias en frío, con un máximo de dos procesadores de refacción. La arquitectura de procesador principal con refacciones en frío permite además equilibrar la disponibilidad con el consumo de potencia, proyectando así un módulo con un consumo de potencia muy parecido al de un sistema símplex.

Asimismo, para contar con una CV eficiente, su arquitectura permite conmutar completamente los instrumentos (actuadores, sensores, protección de efecto “latch-up” para partes comerciales) y señales de control de todo el satélite hacia cualquiera de los procesadores de refacción.

Hasta el momento se ha descrito globalmente la arquitectura de la CV, sin embargo se persigue desarrollar y validar una arquitectura de cómputo satelital que además de conferir alta disponibilidad operativa, permita su mantenimiento automatizado, es decir, que sea tolerante a fallas en sentido estricto, lo cual implica que realice de forma automática los procesos de diagnóstico, detección de fallas y de recuperación (mantenimiento).

## CAPÍTULO 5. ACTUALIZACIONES.

Por estas razones la arquitectura de cómputo semivirtual del MES aglutina las microcomputadoras de carga útil para conformar una arquitectura de redundancia modular múltiple, de operación concurrente y acoplada por software. Para no afectar las operaciones de control y el rendimiento de los procesadores de experimentos, se integra durante tiempos cortos la arquitectura, de forma periódica y controlada por software, utilizando a la CV como núcleo (parte física) y a los procesadores de experimentos y subsistemas inteligentes como parte virtual. De acuerdo con la cantidad de procesadores disponibles en el satélite se puede armar una arquitectura de redundancia modular cuádruple y contar aun con un procesador de refacción para la parte virtual y dos refacciones en frío para la parte física, figura 5.5.

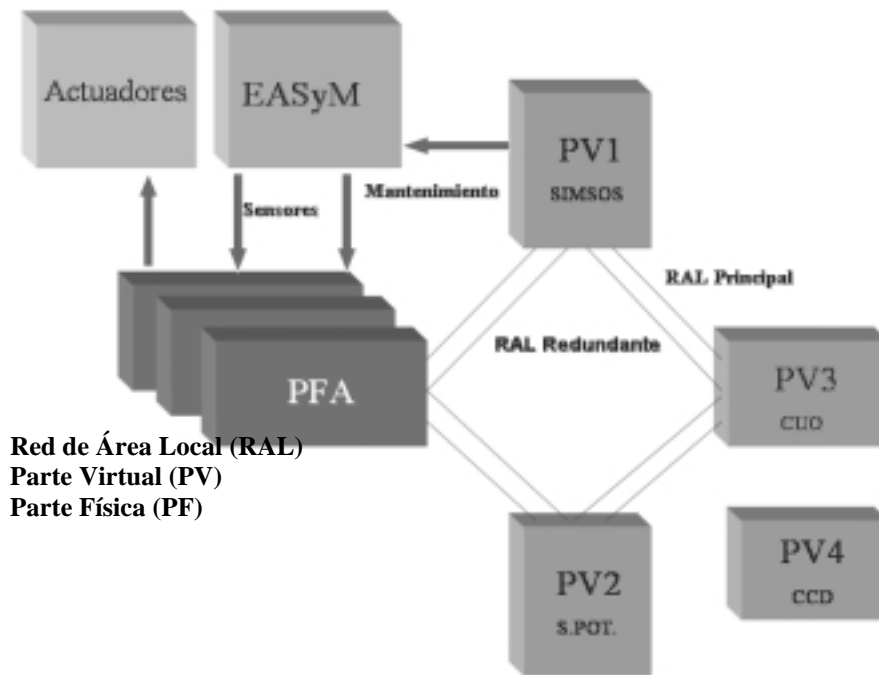


Figura 5.5. Arquitectura de cómputo semivirtual desarrollada para el MES.

Se diseñaron algoritmos dedicados para el diagnóstico de procesadores y para el Voto Mayoritario (VM) que realiza la detección de fallas. Después de realizar el VM cualquiera de los nodos participantes tiene información suficiente del estado operativo de todos los nodos de la arquitectura y el potencial para reconfigurar la CV, en caso necesario, por medio de hardware mínimo y confiable. A este respecto, el mejor proceso de reconfiguración sería aquel en el que participaran todos los nodos, sin embargo, ello implicaría incrementar el hardware de control de procesadores y el consecuente decremento en su confiabilidad operativa, que afectaría a la confiabilidad de la arquitectura semivirtual en conjunto. Por estas razones el SIM-SOS (Sistema Mínimo de Supervivencia) realiza la reconfiguración y reinicio de la CV y a su vez los nodos restantes le otorgan un permiso por voto mayoritario para acceder al hardware de reconfiguración. De esta forma, el permiso total para reconfigurar a la CV se otorga siempre que el SIM-SOS esté libre de fallas. Por su parte, el módulo de voto mayoritario está constituido por hardware mínimo de calificación militar.

**CAPÍTULO 5. ACTUALIZACIONES.**

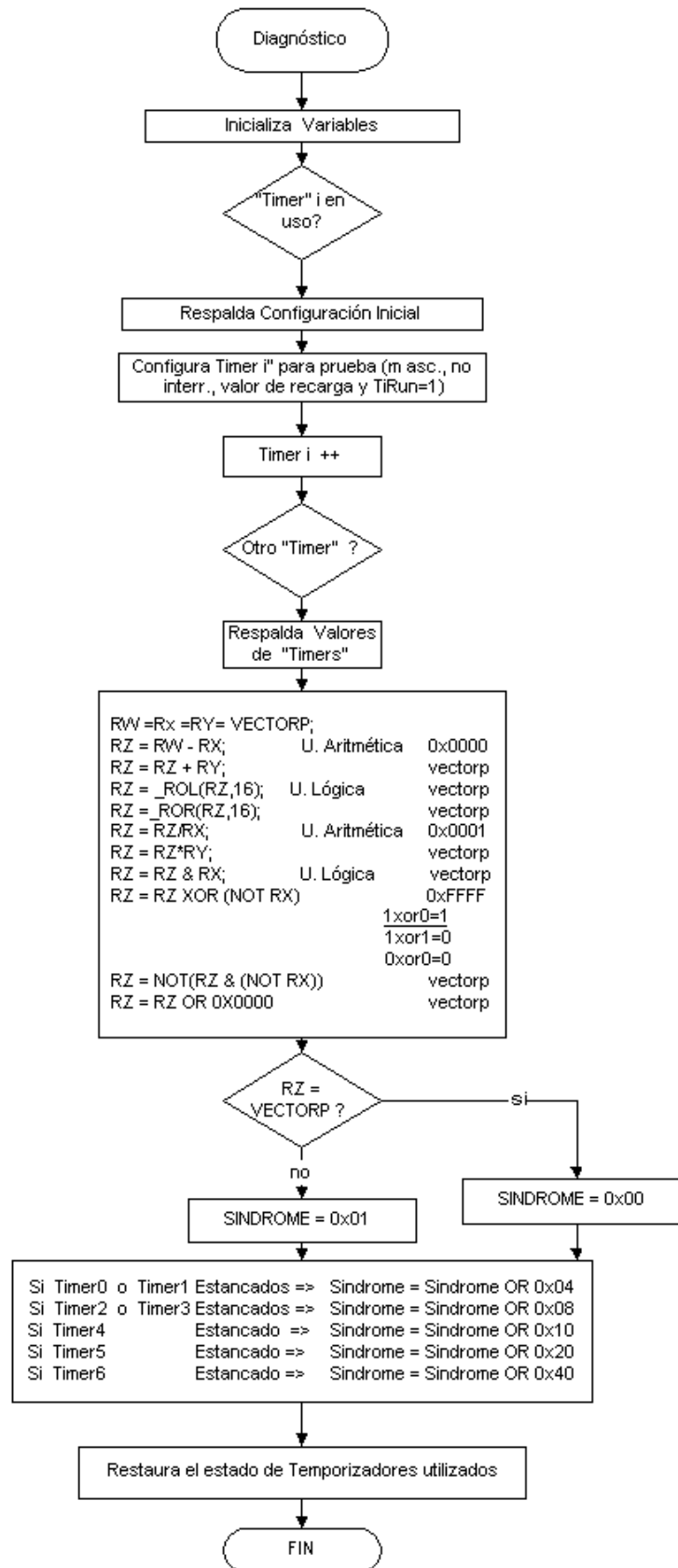



Figura 5.6. Función de diagnóstico utilizada para evaluar la operatividad de las microcomputadoras

**CAPÍTULO 5. ACTUALIZACIONES.**

En caso de determinar algún tipo de falla en el SIM-SOS, el hardware lleva automáticamente a la CV a un estado de operación seguro debido a que el SIM-SOS se tornaría inseguro para toda acción de control o de procesamiento, por lo cual, si se encontraba controlando y especificando el estado operativo de la CV, tal control se vuelve inseguro y por tanto demanda la adopción de una estrategia de seguridad operativa. Esta estrategia consiste en llevar a la arquitectura de cómputo semivirtual tolerante a fallas (ACSTF) a un estado de alta probabilidad de operación correcta, en este caso operando a la CV con el segundo procesador de refacción. Esta refacción se planea utilizar sólo por tiempos reducidos, en tanto que el procesador principal y la primera refacción serán los procesadores que más se utilicen para controlar al microsatélite.

De esta forma, al armar la arquitectura se genera virtualmente una arquitectura de cómputo de redundancia modular cuádruple con refacciones, con cierta centralización de operaciones en la CV. La centralización se utiliza para conducir procesos de diagnóstico y detección de fallas entre procesadores (figura 5.6), así como para guiar el proceso de voto de diagnósticos en los nodos de la computadora. De igual forma la RAL permite realizar el “broadcasting” de resultados de voto y de avisos. Con ayuda de esta infraestructura el SIM-SOS puede reconfigurar a la computadora de vuelo de forma totalmente automatizada para aislar el procesador con falla y sustituirlo por otro en buen estado. Previendo el caso de fallas en la CV que impidan conformar la ACSTF el software del SIM-SOS utiliza un proceso de “watch-dog” que se activa cada vez que se implanta la ACSTF, el cual se inicializa cada 10 minutos durante la conformación de la arquitectura. En casos de falla en la CV no se realiza la inicialización del “watch-dog” del SIM-SOS y en consecuencia éste interpreta el suceso como una falla de la CV y procede a reconfigurarla automáticamente. Cabe resaltar que en este caso se pierde la misión corriente del satélite, sin embargo hace factible la comunicación continua entre satélite y estación terrena.

El resultado del diagnóstico realizado en cada una de las computadoras, comprende dos valores numéricos de un byte, en donde el primer valor es un “síndrome” de la computadora evaluada, figura 5.7, y el segundo es el resultado del diagnóstico de la memoria RAM de la computadora en estudio.



bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
-------	-------	-------	-------	-------	-------	-------	-------

bit	Síndrome
0	Falla en la ALU del microcontrolador de la CV
1	Errores en RAM
2	Falla en <i>timers</i> T0 ó T1 del microcontrolador de la CV
3	Falla en <i>timers</i> T2 ó T3 del microcontrolador de la CV
4	Falla en <i>timer</i> T4 del microcontrolador de la CV
5	Falla en <i>timer</i> T5 del microcontrolador de la CV
6	Falla en <i>timer</i> T6 del microcontrolador de la CV
7	Falla en el <i>WatchDog Timer</i> del microcontrolador de la CV

Figura 5.7. Bits que conforman el byte del síndrome de la CV.

Un valor “1”, en cada uno de estos bits, indican una falla, mientras que el valor “0” indica que se encuentra funcionando correctamente.

En el proceso de diagnóstico simple, el síndrome y el diagnóstico de la RAM de cada una de las computadoras se envían a la CV con la finalidad de bajarlos a Tierra posteriormente como parte de la telemetría.

En el proceso de voto democrático, los datos de diagnóstico de la RAM se almacenan directamente en la telemetría, mientras que el síndrome de cada una de las computadoras se utiliza para realizar del voto, el cual también se envía a Tierra.

Cuando se programa desde Tierra la realización del diagnóstico por medio de voto democrático, su ejecución está sujeta a la energía disponible en el satélite, pues es necesario encender por lo menos cuatro computadoras. Además, dentro de estas cuatro, debe de estar incluida la computadora del SIM-SOS, pues es ésta quien se encarga de reconfigurar a la CV.

Si no existe suficiente energía para encender el mínimo de computadoras, se lleva a cabo entonces el diagnóstico simple entre las computadoras que se hayan podido encender. Del mismo modo, cuando se programa el diagnóstico simple, se verifica previamente la energía disponible para realizar el encendido de las computadoras y posteriormente se diagnostica a aquellas que se encuentren energizadas.

#### **5.4.1 Diagnóstico de computadoras del SATEX**

La rutina de diagnóstico de las computadoras del SATEX tiene utilidad para dos procesos. El primero de ellos se denomina Diagnóstico Simple, figura 5.9, esta rutina hará que cada procesador que se pueda encender, por medio del Sistema de Potencia, realice su autodiagnóstico y después de realizarlo, lo envíe a la CV. Esta rutina se ejecutará cada 10 minutos, durante el muestreo de sensores que realiza el SATEX; la CV almacenará la información en sus arreglos de telemetría y luego enviarla a la Estación Terrena para su análisis. El segundo proceso es el denominado Voto Democrático, figura 5.10, en el cual se realiza el diagnóstico de cuatro computadoras de SATEX, la información de los diagnósticos no sólo se envía a la CV, sino que se trasmite a todas las computadoras que participan en el Voto Democrático, de tal modo que todas las computadoras tienen la información de las demás y de ella misma. Ya que tienen la información, realizan un voto, es decir, determinan cual de las computadoras que participaron se encuentra mal, con falla, y envían sus resultados a cada computadora. En este proceso es donde acontece el Mantenimiento Automatizado de la CV (MACV).

En la figura 5.8 se observa que el tipo de diagnóstico que se realiza finalmente es determinado por la energía disponible en el satélite. El valor de las banderas *Enc\_CUO*, *Enc\_DT* y *Enc\_CCD* se define al recibir la respuesta de la computadora del subsistema de potencia (SP), pues es quien realiza el control de la energía del satélite. Si hay algún error al intentar conocer cuáles computadoras pueden encenderse o bien durante el proceso de encendido, se les asigna el valor “N” indicando que no fue posible encenderlas. Entonces, en este caso, se realiza el diagnóstico simple únicamente entre la CV y el SP.



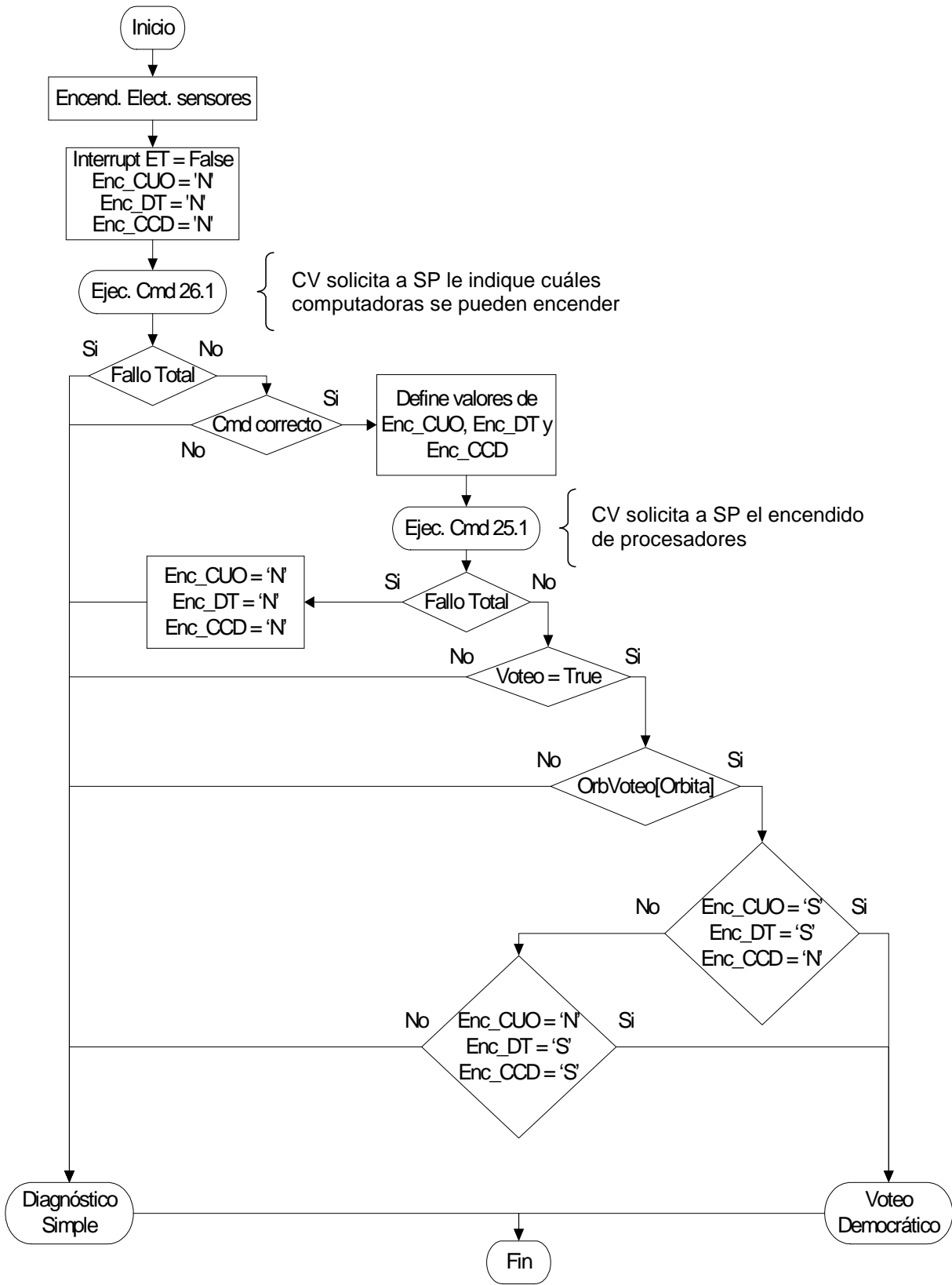


Figura 5.8. Diagrama de flujo del proceso de decisión del tipo de diagnóstico a realizarse.

CAPÍTULO 5. ACTUALIZACIONES.

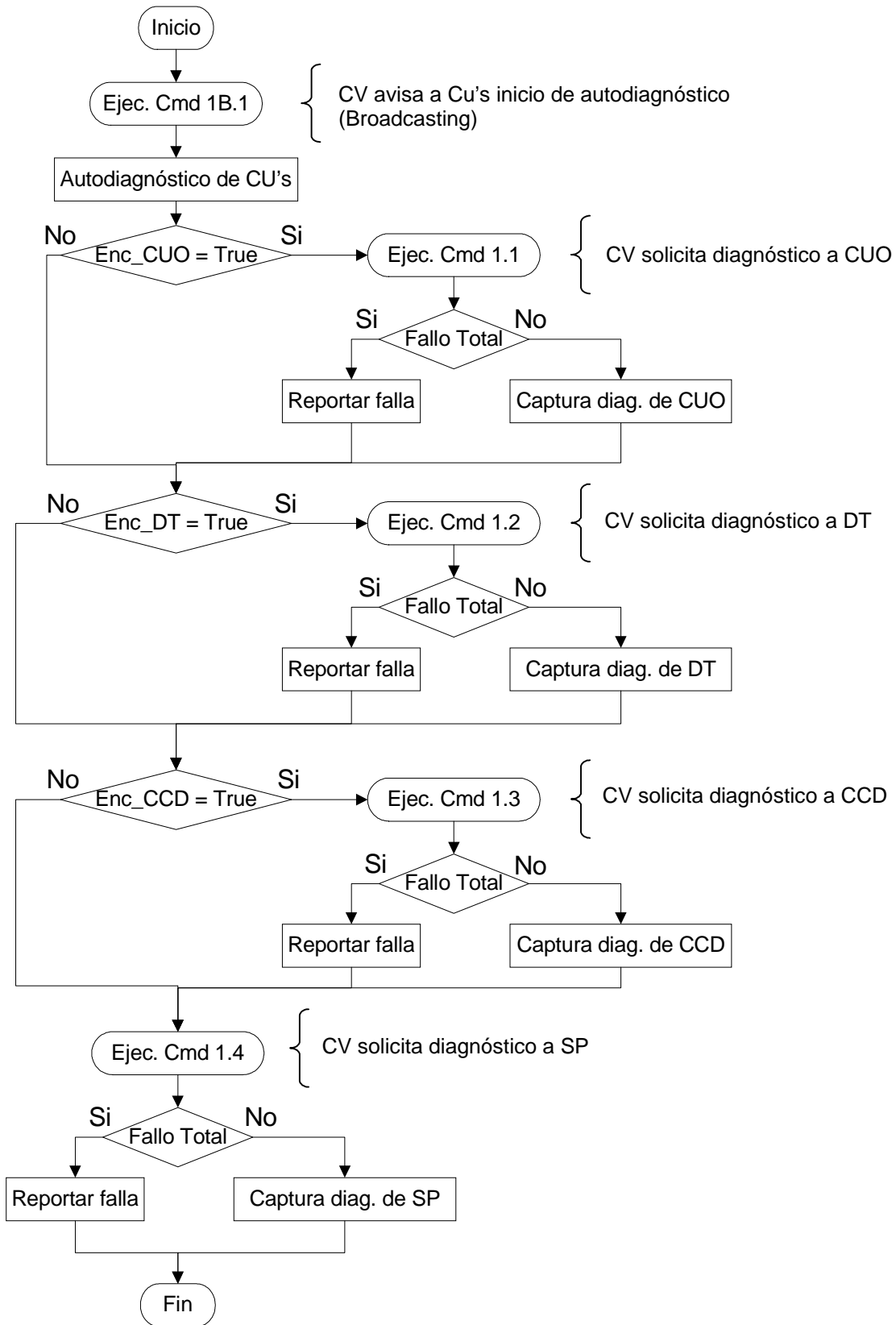


Figura 5.9. Diagrama de flujo del proceso de diagnóstico por el método simple.

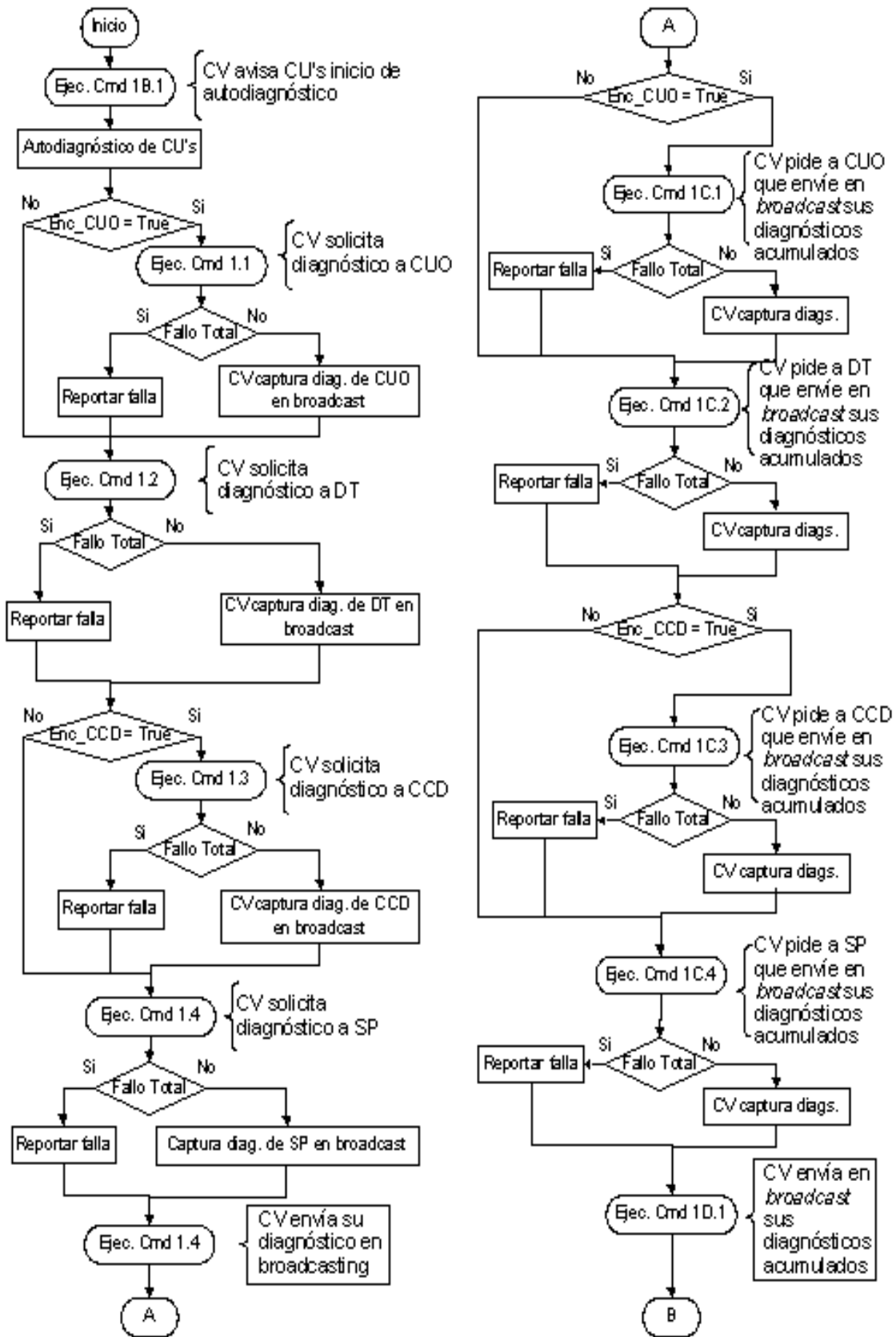


Figura 5.10. Proceso detallado de voto mayoritario (a)

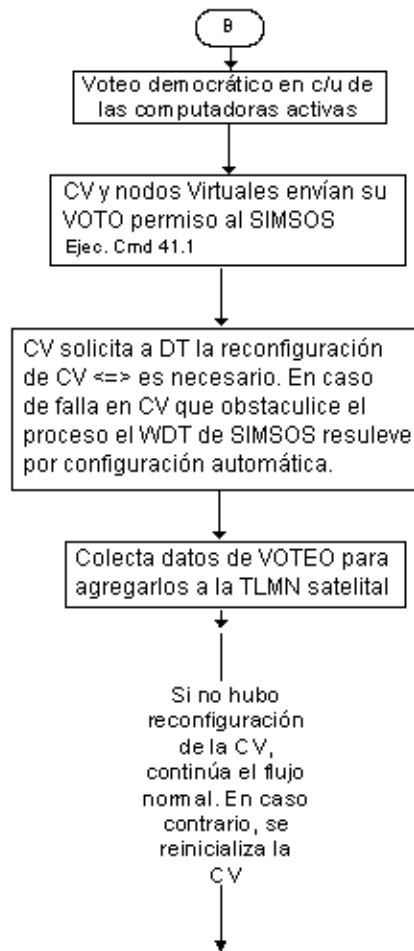


Figura 5.10. Continuación del proceso detallado de voto mayoritario (b)

### 5.4.2 Reconfiguración y reinicio

Cuando se determina una falla en la CV, el SIM-SOS procede a reconfigurar la CV para tolerar su falla, sin embargo al existir la duda de que el SIM-SOS llegase a fallar inmediatamente después de ocurrida la falla en la parte física, los nodos que no participan directamente en la reconfiguración generan un permiso mayoritario para asegurar que el SIM-SOS se encuentre libre de fallas.

En la figura 5.11 se muestra el hardware de reconfiguración de la CV. Inicialmente, el SIM-SOS genera las señales eléctricas: **ON#/OFF\_CP**, **ONCR#** y **SELCR0-1**, de modo que se utilice la Tarjeta de Procesamiento (TP) principal como procesador vigente. Por tanto, de registrarse una falla durante el voto, el SIM-SOS genera las señales requeridas para inicializar ya sea a la primera o a la segunda redundancia, en ese orden y de acuerdo con posibles fallas previas. Como se observa en la figura, los procesadores complementarios al procesador de reconfiguración generan la señal de permiso para que el SIM-SOS pueda reconfigurar a la CV. Sin embargo, sólo cuando la señal de permiso

presenta nivel alto se permite al SIM-SOS realizar la reconfiguración, en tanto que un nivel bajo (generado al detectar fallas en el PM) fuerza al demulticanalizador de mantenimiento a generar señales para llevar a la CV a un estado seguro, en este caso obligando a que la CV utilice a la segunda redundancia de procesador, la que sólo se utilizará en caso de fallas del PM o cuando dos tarjetas de procesamiento hallan fallado previamente. Con este arreglo se logra que el núcleo ACSTF pueda seguir operando aun ante fallas del procesador de mantenimiento.

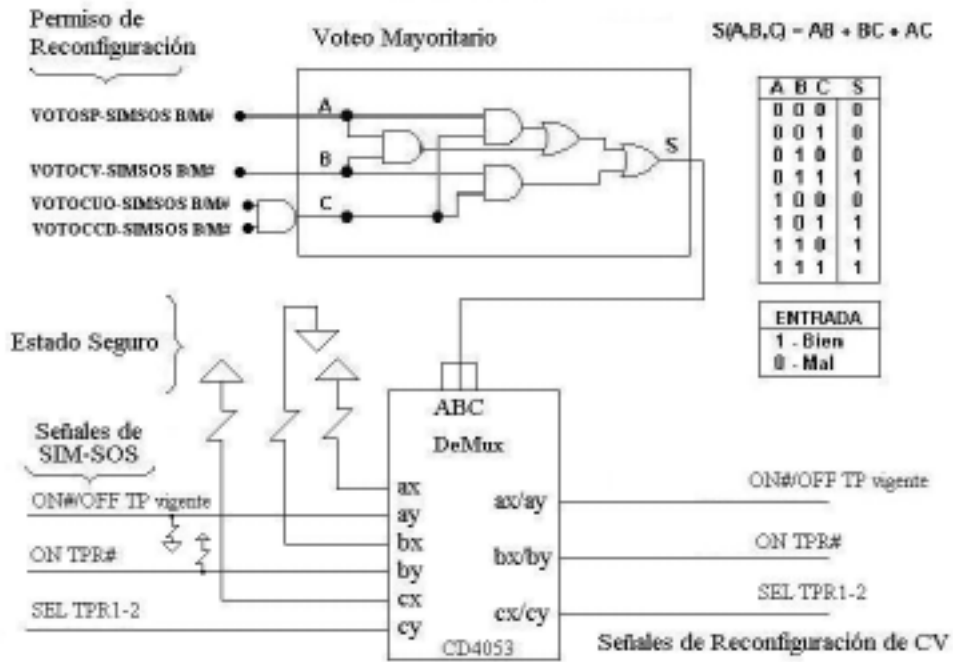


Figura5.11. Hardware de reconfiguración de la CV y hardware de permiso mayoritario para el procesador que realiza el mantenimiento.

Del hardware indicado en la figura anterior resaltan tres aspectos:

- El hardware de voto mayoritario determina el estado del SIM-SOS y genera el permiso para que pueda reconfigurar al procesador redundante.
- De presentarse anomalías operativas en el SIM-SOS el demulticanalizador de reconfiguración asigna valores programados para las señales de reconfiguración. En este caso se obliga el uso de la segunda redundancia de procesador como procesador vigente, con el objeto de utilizar la tarjeta de procesamiento menos utilizada en la arquitectura y por tanto la que presenta mayores posibilidades de encontrarse en buen estado.
- En caso de que el SIM-SOS se encuentre en buen estado, el demulticanalizador de mantenimiento le permite definir las señales eléctricas que realizan el mantenimiento del procesador vigente.

Como también se aprecia en la figura, el hardware de reconfiguración es mínimo (de calificación militar) y constituye a su vez un módulo de voto mayoritario.

El procedimiento que utiliza el SIM-SOS para reconfigurar es el siguiente:

- Desenergiza a todos los procesadores de la CV para confinar y aislar al procesador físico con falla.
- Consulta sus variables para determinar el estado de los procesadores de la CV, identificando fallas previas en las Tarjetas de procesamiento y estableciendo cuál o cuáles procesadores se encuentran libres de fallas para elegir a uno y darlo de alta. Este aspecto es muy importante debido a que en un momento dado el procesador vigente puede ser el procesador principal o el procesador redundante 1 si existió previamente una falla en el procesador principal. Cabe aclarar que cuando existen dos tarjetas de procesamiento en buen estado siempre se elige en el orden TP principal, TP redundante 1 y TP redundante 2.
- Genera las señales eléctricas apropiadas para controlar la electrónica de encendido y apagado de los procesadores físicos de la computadora de vuelo. Al ordenar el encendido de una TP libre de falla ésta inicia con todas las variables y salidas eléctricas en estados seguros de operación.

Una vez finalizado el proceso de reconfiguración el software de la CV procede a adquirir telemetría normal y el estado de equipos (incluyendo a la TP vigente) durante una órbita compuesta por 10 muestreos, los cuales se reportan a Tierra en el siguiente avistamiento. No obstante que el proceso de reconfiguración aborta la misión corriente se obtiene la continuidad funcional del microsatélite, de tal forma que durante la siguiente visita a la Estación Terrena, el software de la CV responde a los llamados de la Estación Terrena con la trama de comunicaciones "*no existe misión previa*", además de ofrecer telemetría instantánea de sensores y de diagnóstico de equipos, con lo cual Tierra obtiene información acerca del procesador vigente de SATEX, deduciendo de forma sencilla la existencia de una falla en el procesador anterior. En ese momento el personal de control satelital tiene la posibilidad de planear una misión de diagnóstico de procesadores para evaluar si se presentó una falla transitoria o permanente.

Previendo el caso de fallas en la CV que impidan conformar la ACSTF el software del SIM-SOS utiliza un proceso de "watch-dog" que se activa cada vez que se genera la ACSTF, el cual se inicializa durante la conformación de la arquitectura. En casos de falla en la CV no se realiza la inicialización del "watch-dog" del SIM-SOS y en consecuencia éste interpreta el suceso como una falla de la CV y procede a reconfigurarla automáticamente. Cabe resaltar que en este caso se pierde la misión corriente del satélite, sin embargo hace factible la comunicación continua entre satélite y estación terrena.

### **5.4.3 Restitución de procesadores físicos con falla**

Una vez que se etiqueta con falla a una Tarjeta de Procesamiento la única forma de restituirla es por medio del diagnóstico y supervisión directa desde la Estación Terrena, seguida por un comando de cambio de estado operativo para la TP analizada. Debido a que

este proceso se realiza en línea, es decir, cuando satélite y Estación Terrena se encuentran en contacto, se elige el momento en que el MES realice una de las visitas más prolongadas. Este tiempo se planea con ayuda de software de dinámica orbital y de seguimiento satelital, el cual puede ser “shareware” disponible en Internet, o bien, el software que ha realizado el CIMAT para el MES.

A este procedimiento se le denominó Comando de Diagnóstico Iterado (CDI). Una vez que se conoce la fecha y la hora en que se diagnosticará a un procesador de la CV, se conoce también la órbita y la hora de contacto MES-ET que antecede a la prueba de diagnóstico. Durante ésta, se programa y envía la misión más simple que consiste en adquirir telemetría normal de sensores. Una vez que se ejecuta esta misión y una vez que el satélite contacta nuevamente a la Estación Terrena se realiza el siguiente procedimiento de diagnóstico en línea de una tarjeta de procesador con falla:

- ✓ Se detecta con el SET la llegada del MES con ayuda del comando “contacto satelital”.
- ✓ Se procede a bajar la telemetría normal de la misión previa.
- ✓ Se envía un comando en línea dirigido al SIM-SOS indicando que se ejecutará el Comando de Diagnóstico Iterado para la TP indicada en el comando. Posteriormente Tierra recibe el reconocimiento de la llegada del comando.
- ✓ Al recibir el CDI el SIM-SOS reconoce que debe realizar las siguientes tareas inmediatas.
  - Registra y respalda el número del procesador vigente, el cual constituye un procesador libre de fallas.
  - Desactiva el procesador vigente.
  - Activa al procesador que se diagnosticará (PD). En vista de que se realizarán pruebas de diagnóstico con un procesador que potencialmente puede tener problemas operativos serios, programa la contabilización de un tiempo máximo de prueba de 15 minutos.
  - Si el PD llegara a estar sano, se espera que antes de 15 minutos regrese el control al SIM-SOS para que reconfigure a la computadora de vuelo y la deje operando con el procesador que tenía como vigente.
- ✓ Una vez que termina de inicializarse el PD, ET lo captura con el comando “contacto satelital”.
- ✓ ET envía el comando “Inicia prueba en sitio de lanzamiento” también denominado “Prueba en Kourou” el cual realiza tres muestreos completos (sensores, estado de equipos y diagnóstico simple entre procesadores del MES) por lo cual obliga el diagnóstico del PD y el envío de resultados a Tierra. Una vez recibidos los resultados, el usuario puede reenviar el comando para realizar iteradamente la prueba bajo control del personal que supervisa el proceso.

**CAPÍTULO 5. ACTUALIZACIONES.**

- ✓ Tierra analiza los resultados de telemetría, de tal forma que después de varias pruebas llega a una conclusión acerca del estado operativo del PD.

En este paso, o en el paso anterior, puede darse el caso de que no se registren actividades, o bien, de que se pierda sincronización en las comunicaciones debido a fallas del PD. En estos casos se procede a ejecutar el último punto.

- ✓ ET envía un comando al SIM-SOS para que retorne a la CV a su procesador vigente (sin falla).
- ✓ Una vez inicializado el procesador vigente y una vez detectado el MES con el comando “contacto satelital”, Tierra envía el comando “actualiza estado de procesador” al SIM-SOS y al SP para actualizar el estado del procesador diagnosticado, quedando ya sea como “libre de falla” o “con falla” de acuerdo con los datos registrados por el personal de la estación terrena. Aquí finaliza el diagnóstico del PD.
- ✓ En caso de que se presenten problemas de comunicación entre ET y el PD, se conmutan las comunicaciones MES-Tierra a modo analógico y se ordena al SIM-SOS que reconfigure a la CV para retornar el control al procesador respaldado como vigente. Cuando Tierra recibe el reconocimiento del comando nuevamente conmuta las comunicaciones a modo digital. Si hay tiempo para esperar la respuesta de la CV se puede incluso enviar una misión nueva al MES. De otra forma, el software del vuelo recaba telemetría normal y en la siguiente visita aceptará una nueva misión.

El proceso descrito anteriormente, constituye el único medio para restituir el estado de cualquier procesador físico en el que se haya detectado algún tipo de falla.

La trama del comando CDI enviado de ET a la CV es la siguiente:

	0	1	2	3	4	5	6	.....	27
Bytes	Dec (128)	Num. Cmd	Inicio o Fin	TP	Edo. TP PP	Edo. TP PR0	Edo. TP PR1	41	Cheksum

Donde en caso de :

Si es Inicio o Fin de CDI, TP indica el Procesador por diagnosticar:

Valores permitidos para TP:

- 0x00 = ‘Procesador Principal
- 0x55 = ‘Procesador R0’
- 0xAA = ‘Procesador R1’

Si es Fin de CDI, el byte **Edo.** indica el Estado del procesador.

Valores permitidos:

- 0x0B = ‘Bueno’
- 0x0F = ‘Falla’
- 41 = No altera



CV recibe la trama y la reenvía al SIM-SOS de la siguiente forma:

	0	1	2	3	4	5	6	7	.....	.	13
Bytes	2F	47	20	Inicio o Fin	TP	Edo. TP PP	Edo. TP PR0	Edo. TP PR1		41.	Cheksum

Trama de comando CDI de ET a CV.

Donde para el byte 3:

'I' = 'Inicia CDI'

'F' = 'Fin CDI'

41 = 'A' = No altera proceso

Valores permitidos para TP:

0x00 = 'Procesador Principal'

0x55 = 'Procesador R0'

0xAA = 'Procesador R1'

Valores permitidos para Edo. de procesador:

0x0B = 'Bueno'

0x0F = 'Falla'

41 = No altera

## 5.5 Software Mínimo de SATEX (SOFMIN)

Como ya se mencionó en el capítulo 3, la CV de SATEX fue diseñada para que tenga un software pregrabado en PROM con la finalidad de que al momento de iniciar su vida de operación siempre tenga forma de responder a comandos enviados por la ET para incrementar las posibilidades de comunicación de SATEX con su ET.

Para este propósito, la CV cuenta con dos mapas de memoria, uno para utilizar sólo memoria RAM y otro que admite memoria RAM y ROM.

En el caso de que utilice sólo memoria RAM, el software de operación satelital puede tener el tamaño que se desee, siempre y cuando no exceda los 256 Kb de memoria RAM normal de la CV, este mapa se utiliza cuando a la CV se le envía un nuevo software de operaciones desde ET y no permite el uso de software pregrabado. Pero en el caso contrario —que la CV utilice el mapa con memoria RAM y ROM—, la CV sólo puede tener un software pregrabado que no exceda los 64 Kb de memoria ROM externa.

El tamaño del software fue una de las restricciones para el grabado de las memorias EPROM que tendrán el SOFMIN para su funcionamiento en el espacio. El software de operación satelital de SATEX tiene un tamaño actual de 90 Kb, una capacidad mucho mayor de 64 Kb permitidos para grabar el software en PROM.

### 5.5.1 Depuración de SOFMIN

El software mínimo de operación satelital no sólo se depuró para lograr un tamaño aceptable en las EPROM sino que se le agregó software dedicado para controlar el EDAC, así como la integración de un comando nuevo muy importante para MACV, es decir, el

Comando de Diagnóstico Iterado. Para ello, varias de las rutinas y comandos realizados por la CV se tuvieron que eliminar sólo para SOFMIN. A continuación se explican esas rutinas:

- **ADQUISICIÓN DE TELEMETRÍA ESPECIAL (TLME)**

Es un comando de misión de gran importancia en el sistema satelital que a diferencia de la telemetría normal no reporta el estado de los equipos de SATEX, sólo contiene las muestras obtenidas de los sensores programados desde el SET, los cuales pueden ser hasta 20 (en la telemetría normal se obtienen muestras de 62 sensores, 10 veces por orbita).

Durante la TLME la computadora de vuelo almacena 20,000 muestras las cuales son enviadas posteriormente a ET. Debido a que es un comando de misión se le tienen que indicar ciertos parámetros para que la CV ejecute el comando programado.

- ✓ La orbita y el muestreo en que se iniciará la captura de datos.
- ✓ La frecuencia con que se realizará el muestreo.
- ✓ El número de sensores por muestrear.

En el momento que se envía el comando desde el SET la CV activa banderas y utiliza un temporizador para generar la frecuencia de muestreo indicada desde Tierra. Las frecuencias pueden ser :16, 10, 8, 6.4, 4, 1, [kHz]; además: 500, 250, 100 y 40 [Hz]

Para la captura de esta información se tiene reservado un arreglo de 20,000 bytes que se llena completamente con las muestras obtenidas sin importar qué sensores se hayan especificado. Cuando se llena el arreglo asignado para la telemetría especial queda lista la información para enviarla en el siguiente avistamiento de ET con el satélite.

- **EXPERIMENTO DE LA CÁMARA DE PERCEPCIÓN REMOTA**

La captura de las imágenes es otro de los experimentos que realizará el microsátélite. El experimento implica la captura de imágenes con una cámara comercial; tal petición se realiza con el protocolo de la cámara y los datos que necesita para realizar la captura. El proceso comienza desde el envío de un comando de misión de adquisición de imagen.

Al capturarse una imagen, se almacena en la memoria disponible de la CV, la cual se empaqueta en bloques de 1000 bytes. El objetivo de este empaquetado es proporcionar mayor seguridad en la transferencia de la imagen a Tierra. Debido a que se realiza la verificación de información cada 1000 bytes.

Cada paquete tiene un encabezado de 9 bytes (de 0 a 8) y un byte para almacenar el cálculo del checksum, colocado al final del mismo. Por lo tanto, la información de la imagen reside en 990 bytes, el último paquete puede tener un

## CAPÍTULO 5. ACTUALIZACIONES.

tamaño que va desde 11 bytes hasta 990 bytes, pues se incluye al encabezado y el byte de checksum.

Los 1000 bytes que forman estos paquetes tienen el siguiente formato:

0	1	2	3	4	5	6	7	8	.....	999
0xFC	Num. Img.	Num. Total de paquetes	Número de paquete	Num. Bytes último paquete	Tipo	Bytes de imagen	chk			

### Paquete de 1000 bytes para el envío de imagen a Tierra

La información de adquisición que envía la trama desde el segundo byte hasta el sexto es la siguiente: órbita, muestreo, minuto, segundo y resolución, la importancia de la resolución es el tamaño de la imagen que se formará, si se tiene una resolución muy alta el tamaño del archivo puede ser demasiado grande para el tiempo de transmisión que se tenga disponible en el momento del avistamiento del satélite.

- **CARGA ÚTIL ÓPTICA**

Se clausuraron muchas de las líneas de código para este experimento que validará un sistema de comunicaciones ópticas por medio de un láser que se envía desde el satélite hacia Tierra. Sólo se conservó el código relacionado con los comandos del diagnóstico simple y voto.

- **EXPERIMENTO KA**

Este experimento quedó totalmente clausurado en SOFMIN. Su rutina solicitaba al SP que energizara el receptor de banda Ka, diseñado para trabajar en una frecuencia de 23 [G Hz].

- **CONTROL DE ESTABILIZACIÓN**

El control de estabilización también quedó totalmente clausurado para SOFMIN. Es una rutina que utiliza dos etapas fundamentales de estabilización, la primera usa un algoritmo que sensa el campo magnético y aplica corriente a las bobinas de tal forma que se obtenga un efecto favorable para reducir el movimiento rotativo del microsatélite, la segunda es una etapa de estabilización más fina.

- **EXPERIMENTO DE MANTENIMIENTO AUTOMATIZADO DE LA CV**

Este experimento, propuesto por la UNAM, no fue clausurado completamente, sino que se condicionó a la aprobación de la Estación Terrena. Es decir, que mediante un comando de misión enviado por ET se le puede indicar a SATEX que realice el experimento MACV.

## CAPÍTULO 5. ACTUALIZACIONES.

Anteriormente la CV realizaba el experimento en cada muestreo, pero las líneas de código para que realizara la rutina MACV eran demasiadas y para reducir espacio se llegó a la conclusión de que se realizara MACV sólo si ET lo solicitaba.

Todas estas modificaciones se realizaron paulatinamente, ya que muchas de ellas dependen una de otra, así que a cada rutina de código clausurada se le realizó una prueba de funcionamiento con el fin de que no se alteraran bloques importantes para su buen funcionamiento.

El diagrama diagrama de bloques SOFMIN se muestra eb la figura 5.12:

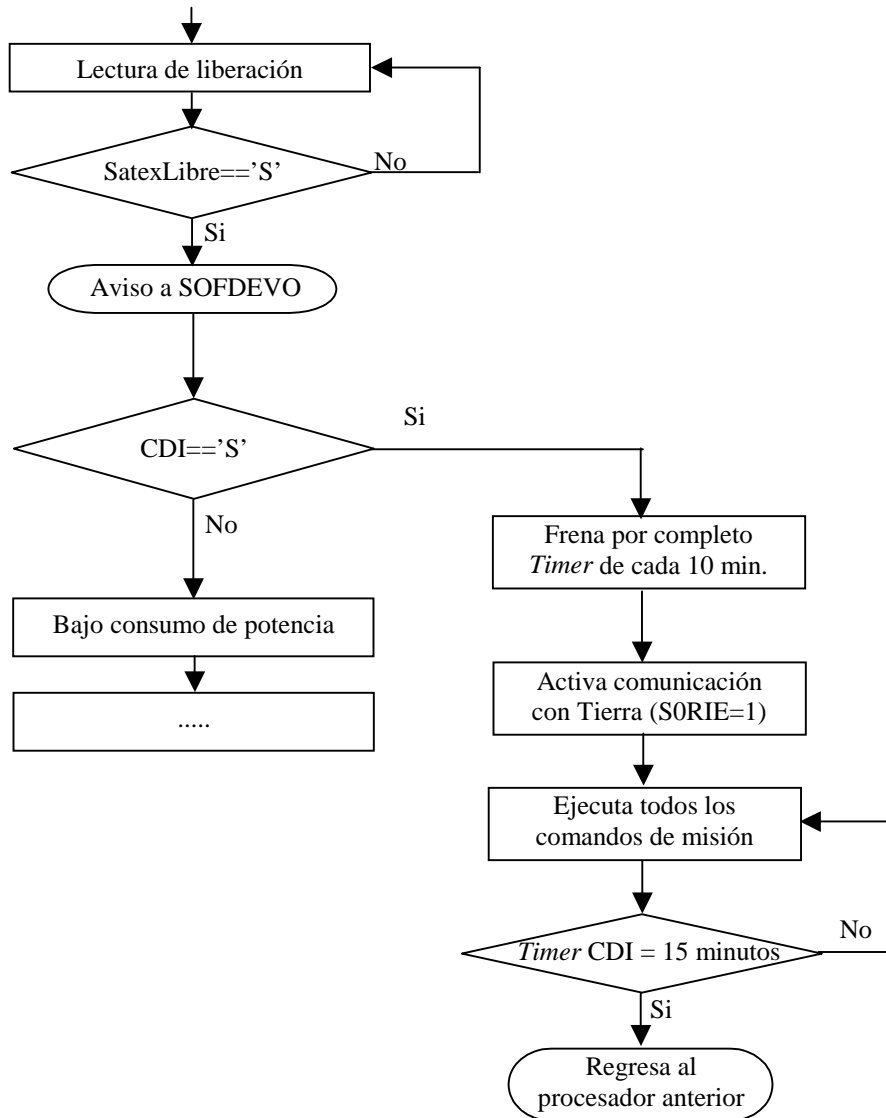


Figura 5.12. Modificación en el diagrama de bloques del funcionamiento de SOFMIN.

SOFMIN tiene un tamaño actual de 62 Kb, una longitud aceptable para su grabado en EPROM.

### 5.6 Compilación del Software de Operación Satelital

El compilador utilizado para desarrollar el software de la computadora de vuelo es el compilador cruzado 80C166 Cross-Compiler BSO TASKING [TASKING, 1993] de la compañía Tasking Boston Systems Office [ www.tasting.com ], el cual utiliza librerías específicas para el microcontrolador. El compilador genera código ejecutable a partir de código fuente en lenguaje ensamblador, figura 5.13.

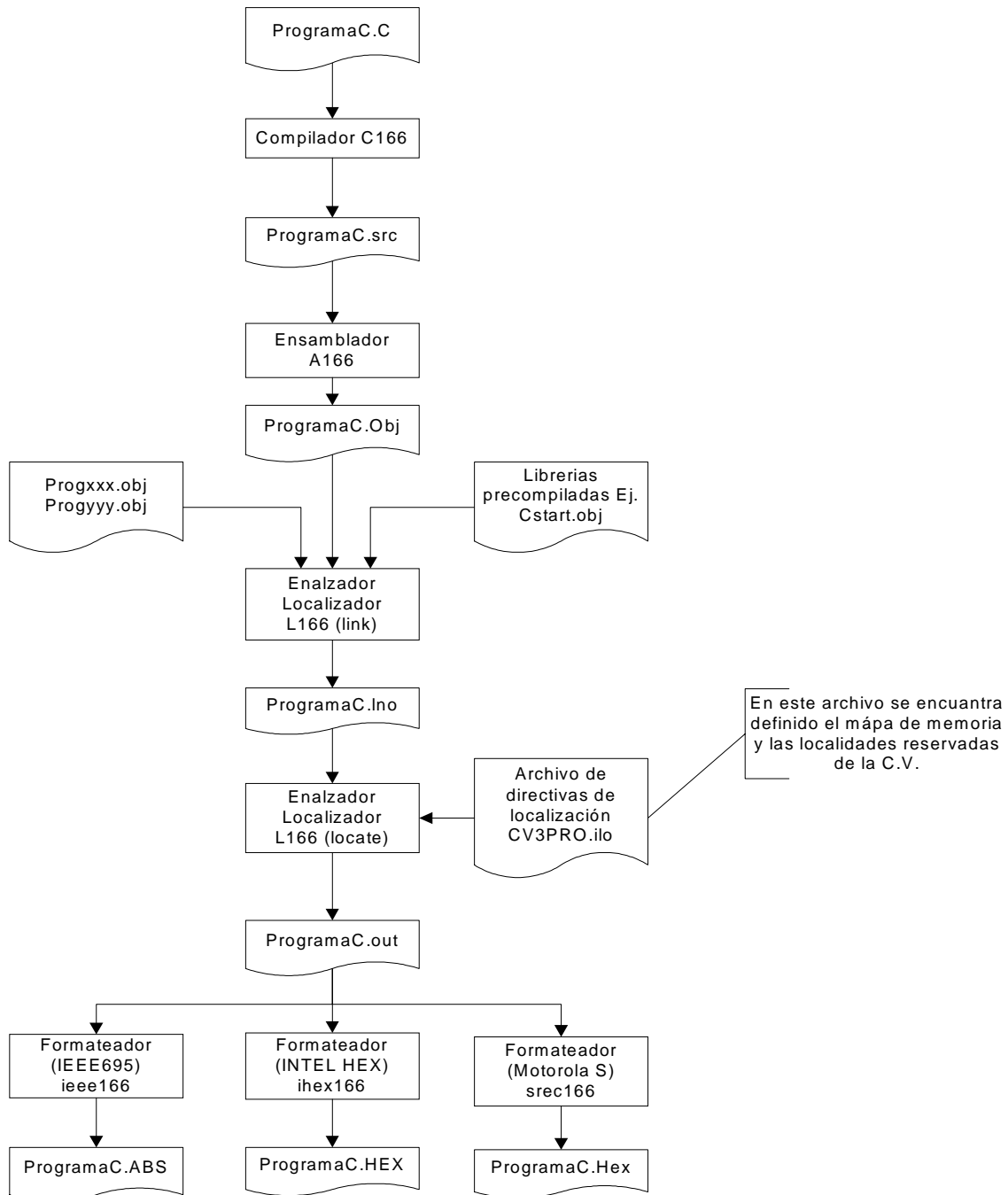


Figura 5.13. Diagrama de flujo para el proceso de compilación del software operativo del microsátélite

Cuando se efectúa una compilación a partir de lenguaje C, es necesario enlazar el código objeto con la librería precompilada llamada "CSTART", en esta librería se encuentran definidas las opciones de arranque del microprocesador, opciones como: configuración inicial del registro SYSCON, código para la inicialización de variables, configuraciones de memoria externa, habilitación del Watchdog, etc. Asimismo, con el archivo.ILO, en el cual se especifica el mapa de memoria del microprocesador.

Para cambiar cualquier opción de inicio para los programas que se compilan a partir de C, es necesario modificar el código fuente de la librería "cstart", para esto el compilador incluye un directorio con el archivo fuente "CSTART.C" y un archivo de control de compilación para cada uno de los modelos de compilación (Small, Medium, large, etc.).

Conforme al hardware que existe entre el microprocesador y la memoria RAM y ROM, se determinan los ciclos de espera del microprocesador, para tal condición, se determinaron cuando menos 5 ciclos de espera para la lectura y escritura. Básicamente por los tiempos que tarda el EDAC en lectura y escritura de las localidades de memoria se determinaron esos 5 ciclos de espera, para mayor detalle sobre estos, leer [ORTIZ, 2003].

La compilación de los dos tipos de software que pueden controlar al SATEX difiere en un sólo aspecto, el mapa de memoria que se especifica en el archivo ligado a la compilación \*.ILO.

El procedimiento para la compilación de SOFMIN es el siguiente

```
PATH C:\C166\BIN\  
set LINK166=LIBPATH(C:\C166\lib\NPI) F166L.LIB, C166L.LIB CASE  
c166 -Ml-s -g -Ot -OZ %1.c -o %1.src  
pause  
a166 %1.src TO %1.obj  
pause  
l166 link %1.obj,c:\c166\bin\CSTART.obj to c:\c166\bin\%1.lno  
pause  
l166 LOCATE @mikeepro.ILO TASK %1 TO %1.out  
pause  
ieee166 c:\c166\bin\%1.out c:\c166\bin\%1.abs  
ihex166 -i16 -l200 -c %1.out -o %1.hex
```

- ✓ Primero se determina la ruta donde se encuentran las librerías de C y los archivos que se necesitan ligar a la compilación.
- ✓ El comando C166 genera el pre-código fuente para el ensamblador.
- ✓ El comando a166 ensambla el archivo scr y lo convierte en un archivo objeto.
- ✓ El comando l166, liga el código objeto con el archivo de inicialización CSTART.OBJ y genera un archivo de salida \*lno
- ✓ Después se vuelve a llamar el comando l166 para localizar y colocar todas las rutinas de C en el sitio adecuado, RAM, IRAM, ROM, esto gracias al archivo *mikeepro.ILO* y genera una archivo absoluto.

## CAPÍTULO 5. ACTUALIZACIONES.

- ✓ Después se le asignan las normas de la IEEE para C166.
- ✓ Ya que está en las normas de la IEEE para C166, genera el código deseado para su grabado en EPROM, en un archivo en código hexadecimal \*.hex.

El archivo mikeram.ILO generado para SOFMIN es el siguiente:

```
classes( CPROGRAM (00000h TO 0FA00h)) ;
                area que asignamos para ubicar;al CSTART.OBJ
MEMORY
(
ROM (00000h to 0F9FFh) ;reserva mem para viejo progr que proteja
                ;areas delicadas del uP Siemens.
IRAM (0FA00h)      ;RAM interna + SFR's del P Siemens
RAM (10000h to 3FFFFh) ;reserva parte segmto0+segmto1+algo segmto2
)
```

El procedimiento para la compilación de Software General de SATEX es el siguiente

```
PATH C:\C166\BIN\
set LINK166=LIBPATH(C:\C166\lib\WP) F166L.LIB, C166L.LIB CASE
c166 -Ml-s -g -Ot -OZ %1.c -o %1.src
pause
a166 %1.src TO %1.obj
pause
l166 link %1.obj,c:\c166\bin\CSTART.obj to c:\c166\bin\%1.lno
pause
l166 LOCATE @mikeram.ILO TASK %1 TO %1.out
pause
ieee166 c:\c166\bin\%1.out c:\c166\bin\%1.abs
ihex166 -i16 -l200 -c %1.out -o %1.hex
```

- ✓ Primero se determina la ruta donde se encuentran las librerías de C y los archivos que se necesitan ligara a la compilación.
- ✓ El comando C166 genera el pre-código fuente para el ensamblador.
- ✓ El comando a166 ensambla el archivo scr y lo convierte en un archivo objeto.
- ✓ El comando l166, liga el código objeto con el archivo de inicialización CSTART.OBJ y genera un archivo de salida \*.lno
- ✓ Después se vuelve a llamar el comando l166 para localizar y colocar todas las rutinas de C en el sitio adecuado, RAM, IRAM, ROM, esto gracias al archivo *mikeram.ILO* y genera una archivo absoluto.
- ✓ Después se le asignan las normas de la IEEE para C166.
- ✓ Ya que está en las normas de la IEEE para C166, genera el código deseado para su grabado en EPROM, en un archivo en código hexadecimal \*.hex.

El archivo mikeram.ILO generado es el siguiente:

```
classes( CPROGRAM (00000h TO 0FA00h)) ;
          area que asignamos para ubicar;al CSTART.OBJ
MEMORY
(
RAM (00000h to 0F9FFh) ; RAM normal
IRAM (0FA00h) ;RAM interna + SFR's del P Siemens
RAM (10000h to 3FFFFh) ;;reserva parte segmto0+segmto1+algo segmto2
)
```

Todos los comandos de línea usados para la compilación de ambos softwares — Software General y SOFMIN— son en muchos casos recomendaciones que sugiere el compilador. Muchas de las recomendaciones persiguen que el código compilado no sea demasiado grande, las variables no sean usadas o declaradas más de una ocasión, etc. Para más detalle consulte 80C166 Cross-Compiler BSO TASKING [TASKING, 1993].

### **5.7 Actualización de SOFDEVO**

SOFDEVO debe emular a las computadoras a bordo del SATEX: SIM-SOS, CUO, CCD y SP, es por ello que debe realizar todas las funciones de las computadoras que constituyen la arquitectura de cómputo semivirtual tolerante a fallas (con excepción de la CV). La actualización para SOFDEVO comprende todo lo relacionado con el Mantenimiento Automatizado de la Computadora de Vuelo, además de la prueba del Comando de Diagnóstico Iterado, el cual permite restituir o afirmar el estado operativo de un procesador de SATEX.

SOFDEVO captura comandos que son dirigidos hacia las cargas útiles que emula, así que los mismos comandos que son actualizados en la CV de SATEX, son igualmente implementados en SOFDEVO.

Las actualizaciones para SOFDEVO son el Mantenimiento Automatizado de la Computadora de Vuelo (Apartado 5.4 de este capítulo). Propiamente realiza el diagnóstico de fallas, realiza el Voto Democrático, genera permisos de reconfiguración y realiza el mantenimiento de la CV.

También debe ejecutar el Comando de Diagnóstico Iterado, recibir la trama que reenvía la CV y realizar lo concerniente al apartado 5.4.3 del presente capítulo.

Entonces, será mucho mejor ver las funciones concernientes a SOFDEVO ya que las líneas de código son muchas para que se presenten por escrito y con ello se reduce la extensión de la presente tesis.



## Capítulo 6

### ***Validación del Software de Operación Satelital con Ayuda de SOFDEVO y el Software de Estación Terrena***

---

#### **6.1 Introducción**

En este capítulo se expone el funcionamiento del Software de Operación Satelital, tanto el General como SOFMIN. Para ello se incorporan en la CV memorias EPROM para efectuar las pruebas de laboratorio y se describen los aspectos de compilación del software, emulación y grabado.

Se explica la elección de las EPROM y de su tiempo de acceso, ya que no cualquier EPROM puede ser utilizada en las tarjetas de procesamiento, debido a que entre el microcontrolador y las memorias existe el dispositivo EDAC, el cual retrasa la lectura y escritura de datos en EPROM.

También, se describe el proceso de Mantenimiento Automatizado de la Computadora de Vuelo de SATEX ya que con ello se valida la arquitectura de Cómputo Semivirtual Tolerante a Fallas diseñada para SATEX. Esta validación se realizó con las herramientas de software de SOFDEVO, SIMSAT, EASyM y el Software de Estación Terrena.

#### **6.2 Instalación de SOFMIN y EPROM en las Tarjeta de Procesamiento**

Para la instalación de SOFMIN fue necesario que el software funcionara bien, para ello se utilizó el Emulador ROM/RAM TRE-200, figura 6.1, que, como ya se explicó en el capítulo 2 de la presente tesis, permite la emulación de memoria tanto ROM como RAM.

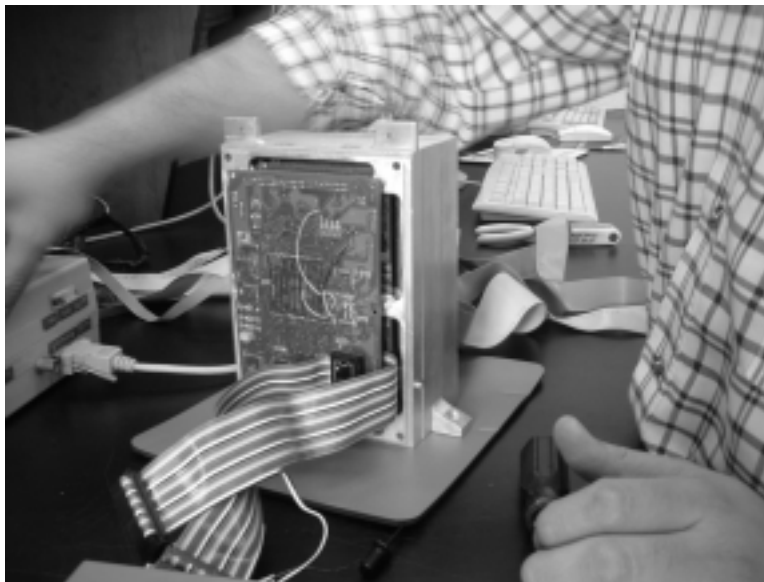


Figura 6.1. Pruebas de SOFMIN con ayuda del Emulador ROM/RAM

El tipo de memoria ROM que utiliza la tarjeta de procesamiento es del tipo 256 Kb (32 K x 8-Bit) en su configuración Dual In Line Package (DIP), figuras 6.2 y 6.5.

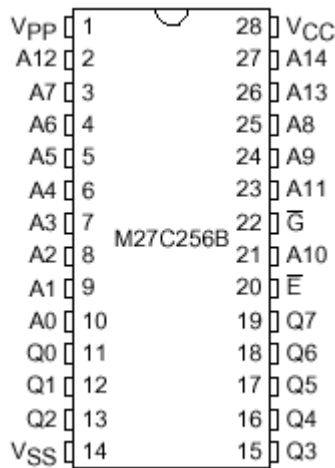


Figura 6.2. Patigrama de la memoria de 256 Kb (32 K x 8 Bit)

Pero no sólo esas especificaciones son necesarias para la EPROM que necesita la tarjeta de procesamiento de la Computadora de Vuelo sino que además deben tomarse en cuenta los tiempos de acceso para escritura y lectura de datos.

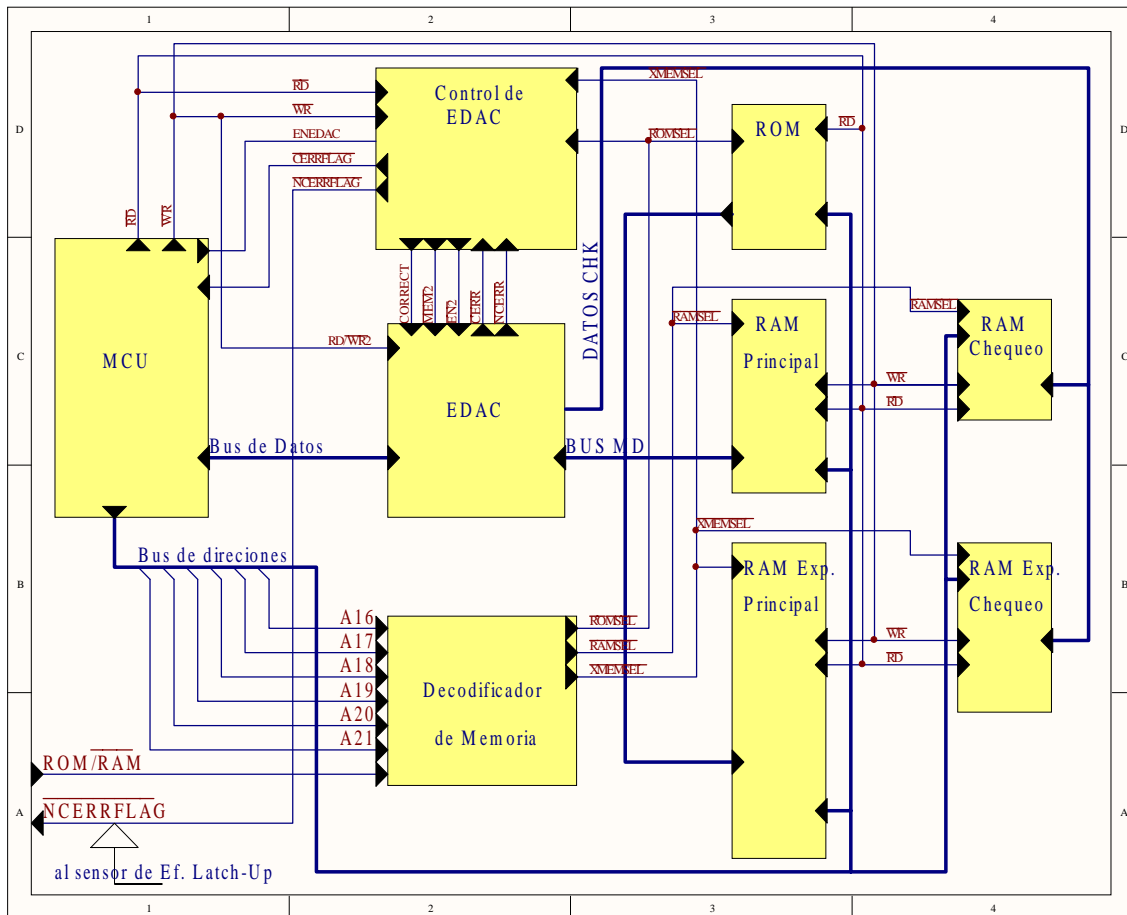


Figura 6.3. Diagrama de bloques del conjunto MCU-EDAC-MEMORIA.

De la figura 6.3 y considerando que los circuitos RAM tienen un tiempo de acceso 25ns se obtiene el diagrama de tiempos de acceso a memoria mostrado en la figura 6.4, en el cual se observa que el tiempo de acceso máximo a memoria es de aproximadamente 135ns. De acuerdo con las tablas de programación para los ciclos de acceso del microprocesador [SIEMENS, 1997], para garantizar un acceso correcto a la memoria de la CV es necesario extender el tiempo de acceso del microprocesador en 3 ciclos de espera.

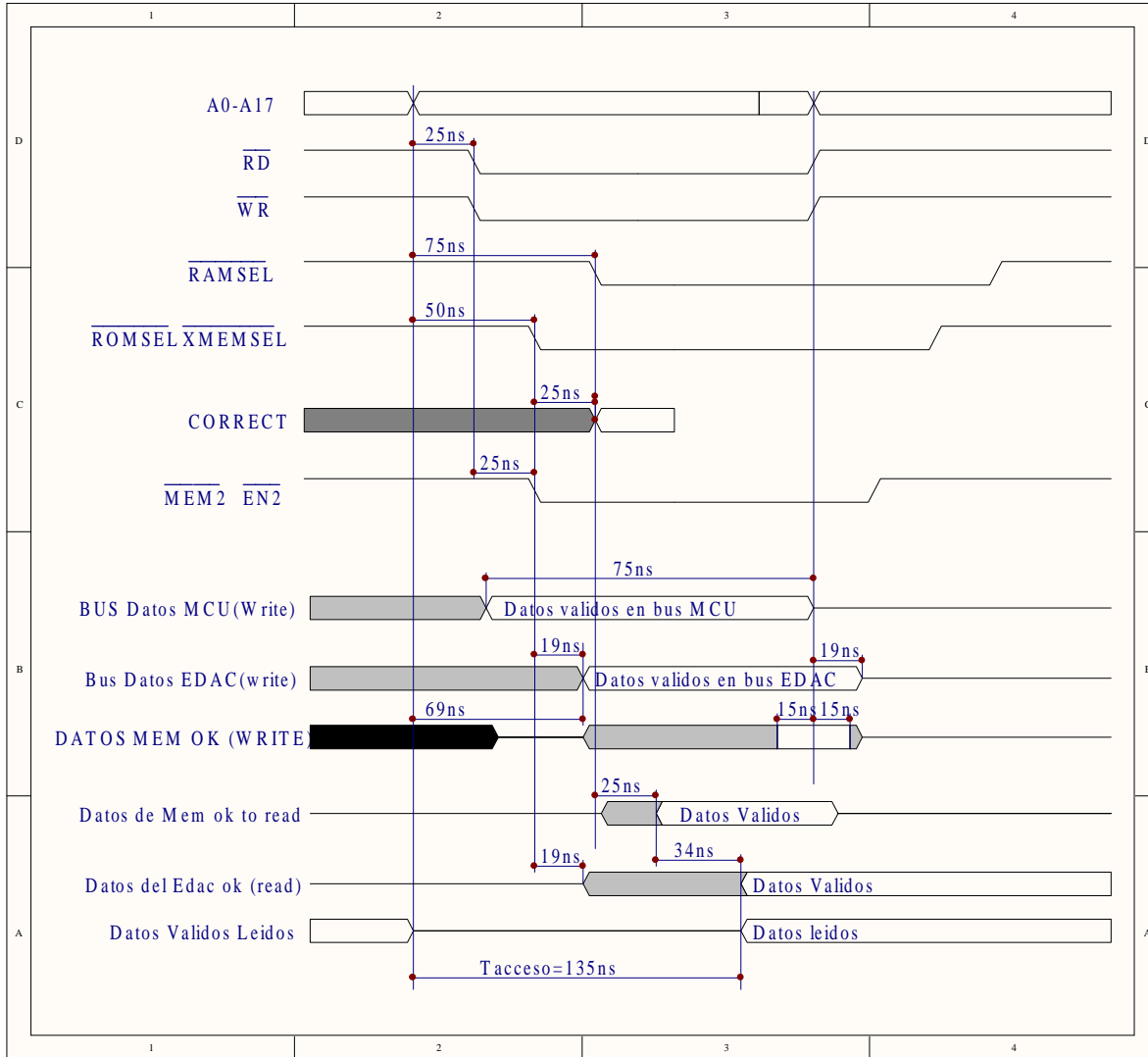


Figura 6.4. Diagrama de tiempo de acceso a la memoria de la CV

Este tiempo de acceso se logra gracias a que las RAM utilizadas en la tarjeta de procesamiento tienen una velocidad de 25 ns, lo que implica que las EPROM también deben tener una velocidad igual o similar a las RAM.

Para extender el tiempo de acceso del procesador Siemens es necesario programar el registro MCTC y MTC del SAB80C166 [SIEMENS, 1997] con el valor deseado. La programación de estos registros debe efectuarse en el código de arranque de programas, ubicado en el archivo "CSTART.ASM".

Por el momento, sólo se cuenta con EPROMs con velocidad de 150 ns, es decir que son por lo menos 6 veces más lentas que las RAM de SATEX, lo que implicaría que el procesador debe tener programados más de 15 ciclos de espera. Esta cantidad de ciclos no son programables en el microcontrolador usado para la Computadora de Vuelo. La CV necesita la instalación de unas memorias EPROM con tiempos de acceso por lo menos de entre 25 ns y 100 ns, mientras más rápidas, su comportamiento será mejor.

Este inconveniente se puede salvar ya que, como la CV de SATEX estaba diseñada para 2 redundancias en frío, una de esas redundancias no tiene instalado el dispositivo EDAC, que es la causa del retraso de la información pedida por el microprocesador. Entonces el flujo de datos se puede hacer directamente entre las EPROM y el microprocesador, además la instalación de las EPROM que se tienen (M27C256B -12F1) se realizó en esa tarjeta de procesamiento.



Figura 6.5. Memoria M27C256B -12F1 utilizada para almacenar SOFMIN

Este tipo de memoria sólo puede ser usada en el laboratorio ya que es comercial. Para poder usarla en el espacio, debe tener calificación militar, es decir, que su temperatura de funcionamiento sea de  $-40$  a  $125$  °C, además que sea inmune al “Efecto Latch-Up” que es ocasionado por la radiación espacial y que provoca corrientes tipo avalancha de suficiente magnitud causando que fluyan altas corrientes provenientes de la fuente de polarización a través del dispositivo, estas altas corrientes provocan usualmente la destrucción del dispositivo en pocas decenas de milisegundos.

Cualquiera de las memorias M27C256B, con sus diferentes velocidades de acceso puede servir para la validación del software de operaciones satelital de SATEX en laboratorio. Además, las pruebas se realizaron con la tarjeta de procesamiento que no tiene la instalación del EDAC.

Entonces, ya que se probó el funcionamiento de SOFMIN con ayuda del Emulator ROM/RAM TRE-200, se grabó el software SOFMIN en las EPROM, este procedimiento se explicó en el capítulo 2 de la presente tesis.

Ya que se tienen las EPROM con el software, se colocan en las tarjetas de procesamiento, en la figura 6.6 se puede observar la instalación de las EPROM.

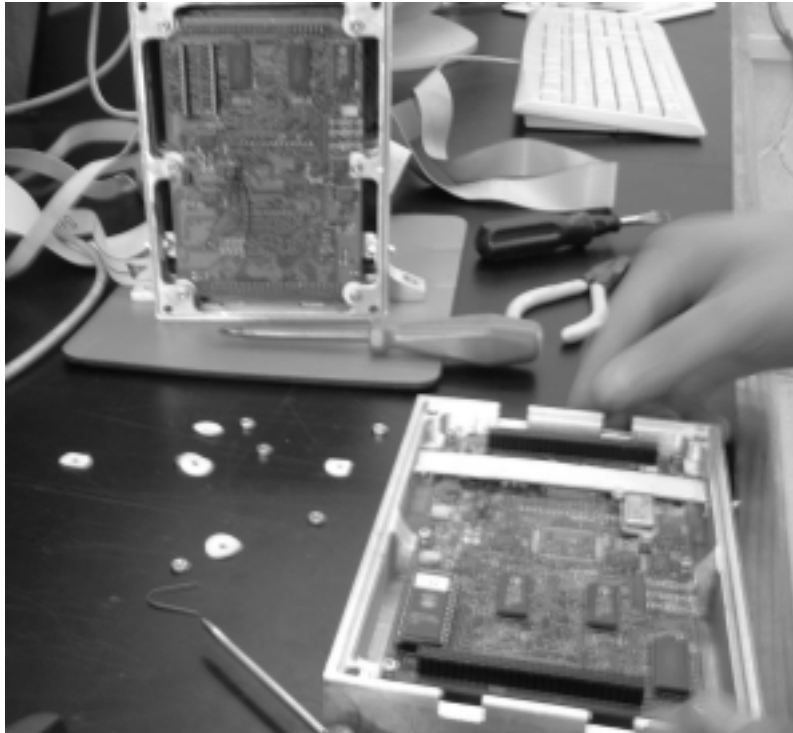


Figura 6.6. Instalación de la memorias EPROM con SOFMIN.

Como el microprocesador es de 16 bit, el grabado del software se realiza en dos EPROM, en una de ellas contienen todos los bytes pares y en otra todos los bytes impares.

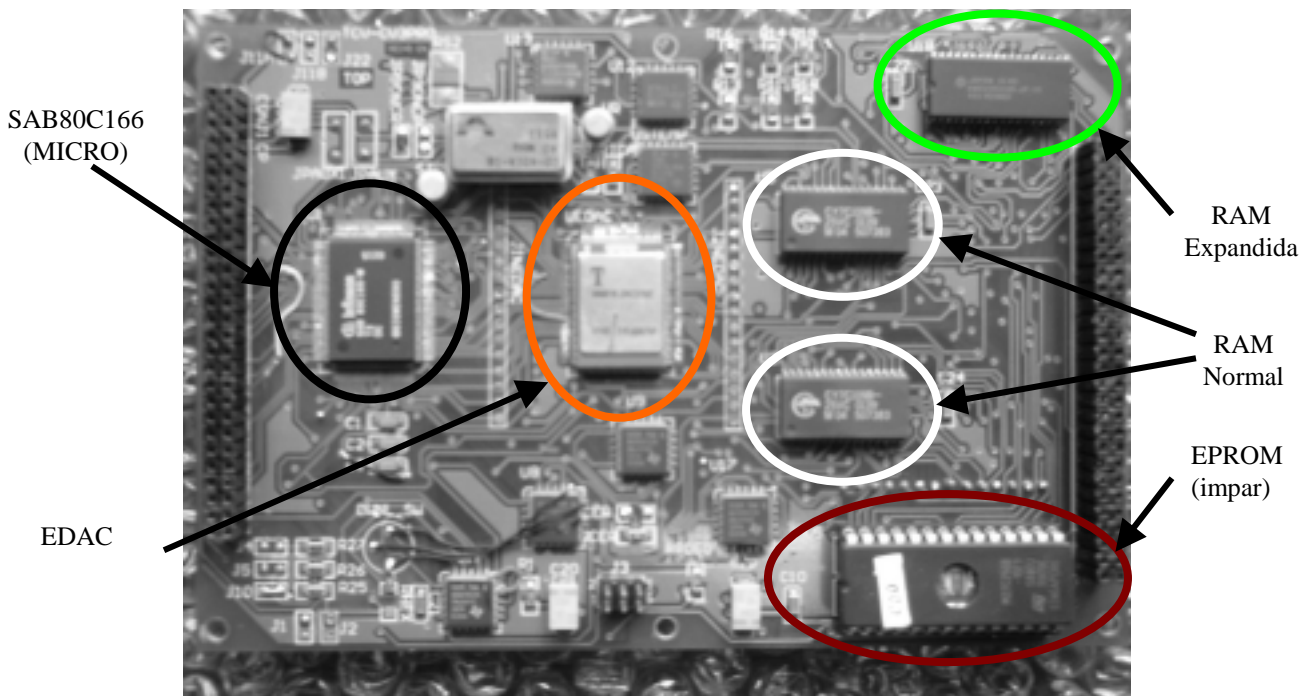


Figura6.7. Vista frontal de la tarjeta de procesamiento

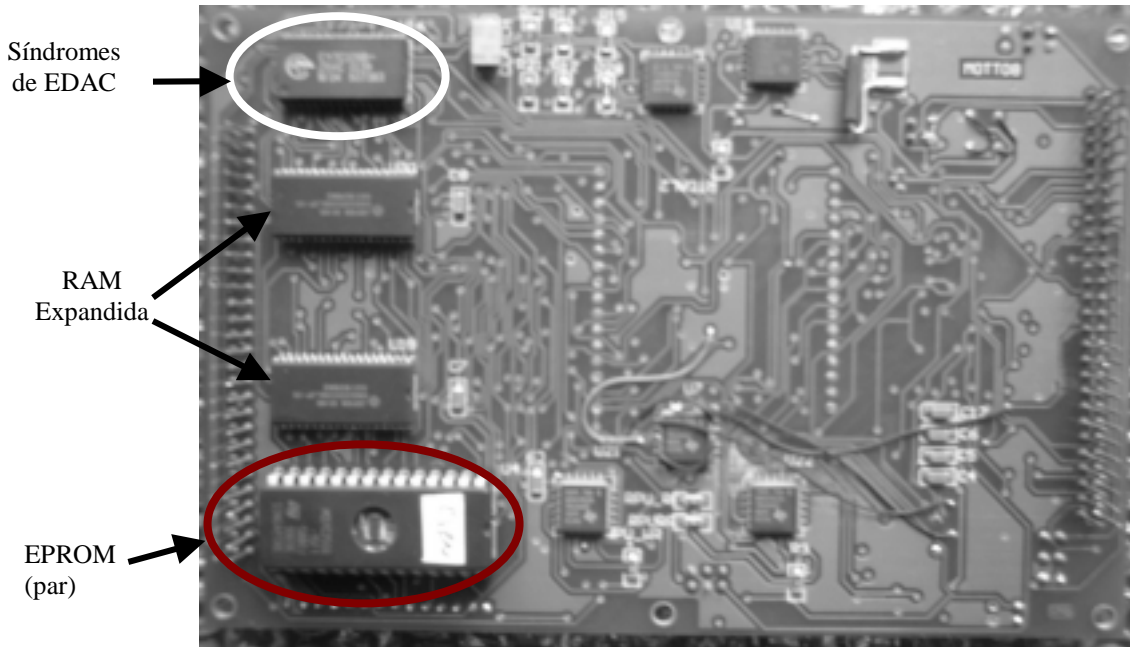


Figura 6.8. Vista trasera de la tarjeta de procesamiento de SATEX.

Cabe resaltar que en las tarjetas de procesamiento se instalaron bases para la colocación de las EPROM con el fin de no dañar tanto el impreso. Las bases sirven para que se puedan retirar las EPROM sin ningún problema y sin desoldar los circuitos. Ya instaladas las EPROM, se procede a realizar las pruebas de validación del Software de Operaciones Satelital.

### **6.3 Preparación de equipos para la validación del Software de Operación Satelital**

Los equipos que son utilizados para las pruebas de validación en laboratorio, como se mencionó en el capítulo 2, son:

- ✓ Software de Depuración y Emulación y Validación Operativa (SOFDEVO), que permite la emulación de las cargas útiles de SATEX como son: Carga Útil Óptica, Cámara de Percepción Remota, Sistema de Potencia, Sistema Mínimo de Supervivencia.
- ✓ Software de Estación Terrena (SET), que permite la observación de los datos capturados por la Computadora de Vuelo.
- ✓ Simulador de Satélite (SIMSAT), que permite la simulación de 48 sensores, red interna, visualización de la operación de magnetómetros y bobinas de toque magnético.
- ✓ Módulo de Electrónica de Acondicionamiento de Sensores y Votación (EASyV), que acondiciona todas las señales de los sensores y votación.

En la figura 6.9 pueden observarse los equipos utilizados en el laboratorio.

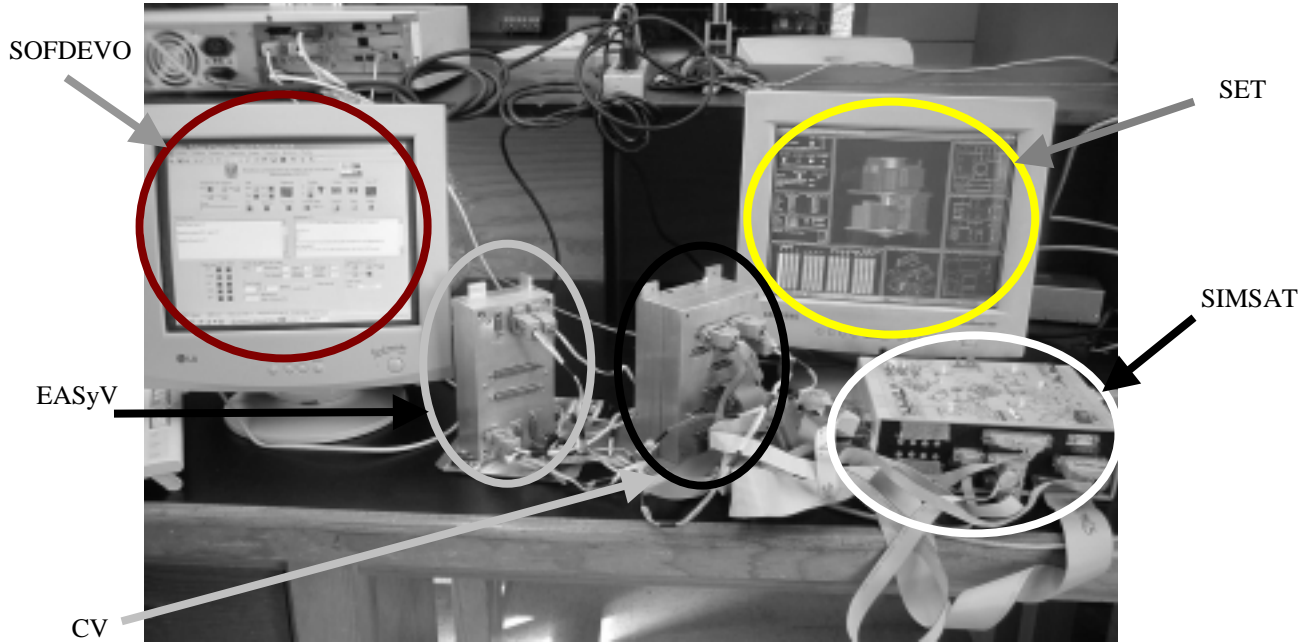


Figura 6.9. Imagen de todos los equipos utilizados para la validación del Software Satelital

La conexión de los equipos se efectúa por medio del SIM-SAT, como se aprecia en la figura 6.9, todos los conectores DBX son conectados al simulador. El puerto serie de la PC que tiene a SOFDEVO y el de la PC que aloja a SET también son conectados al simulador. Además, SOFDEVO realiza el mantenimiento de la CV por medio de señales que son enviadas por el puerto paralelo el cual, tiene la configuración de pines que se presenta en la tabla 6.1.

Tabla 6.1. Configuración de cada pin del Puerto Paralelo

Numero de Pin	Entrada/ Salida	Numero de Pin	Entrada o Salida
1	Salida	14	Salida
2	Salida	15	Entrada
3	Salida	16	Salida
4	Salida	17	Salida
5	Salida	18	Tierra
6	Salida	19	Tierra
7	Salida	20	Tierra
8	Salida	21	Tierra
9	Salida	22	Tierra
10	Entrada	23	Tierra
11	Entrada	24	Tierra
12	Entrada	25	Tierra
13	Entrada		

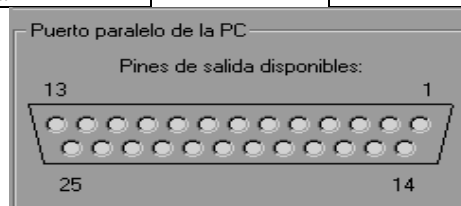


Figura 6.10. Orden de los pines del conector DB25 del puerto paralelo de una PC

En SOFDEVO se usan los pines del 2 al 9, configurados como salidas y las señales emuladas son las que se muestran en la tabla 6.2.

Tabla 6.2. Señales emuladas por SOFDEVO mediante el Puerto Paralelo

Señal	Nombre de la Variable	PIN de Salida
Voto SP-DT B/M#	PinSalidaSenal_SPEDT	2
Voto CUO-DT B/M#	PinSalidaSenal_CUOEDT	3
DT ON#/OFF CP	PinSalidaSenal_ONOFFCP	4
DT SEL CR0-1	PinSalidaSenal_SELCR01	5
DT ON CR#	PinSalidaSenal_ONCR	6
RESET CV#	PinSalidaSenal_RESETCV	7
DWNLSW TTL#	PinSalidaSenal_DWNLWSW_TTL	8
ROM/RAM#	PinSalidaSenal_ROMRAM	9

Las señales de SOFDEVO están conectadas de a siguiente manera:

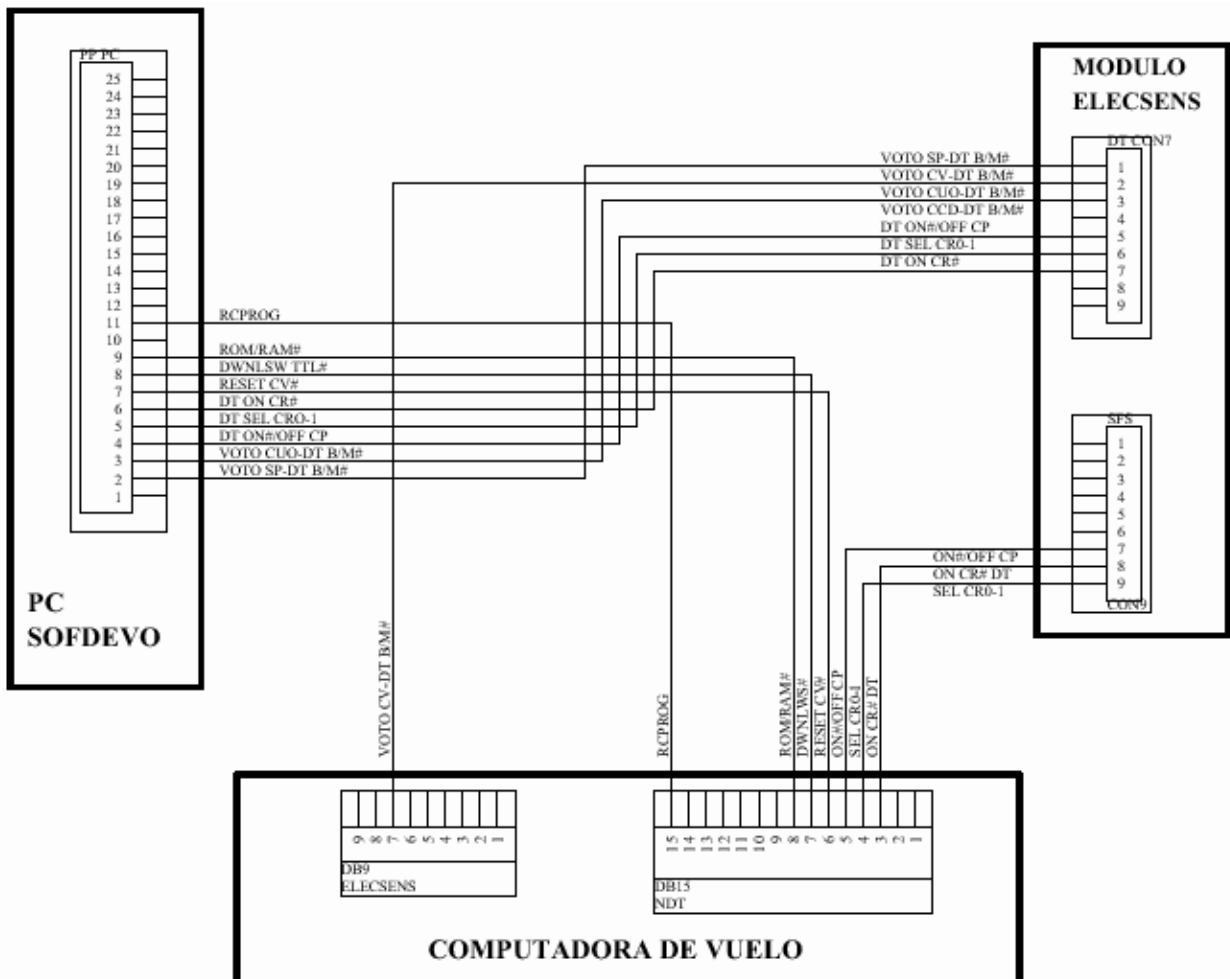


Figura 6.11. Diagrama de conexiones de SOFDEVO por medio del Puerto Paralelo



Con ayuda de las señales enviadas por SOFDEVO, figura 6.11, es posible sustituir muchas de las funciones básicas del SIMSAT, las cuales son:

- ✓ Elegir la tarjeta de procesamiento (Principal, Redundante 0 y Redundante 1)
- ✓ Liberación del Satélite
- ✓ Activar modo RAM o ROM
- ✓ Activar o desactivar modo BTL
- ✓ Reinicio de CV

Ahora SOFDEVO realiza la activación de estas señales vía puerto paralelo.

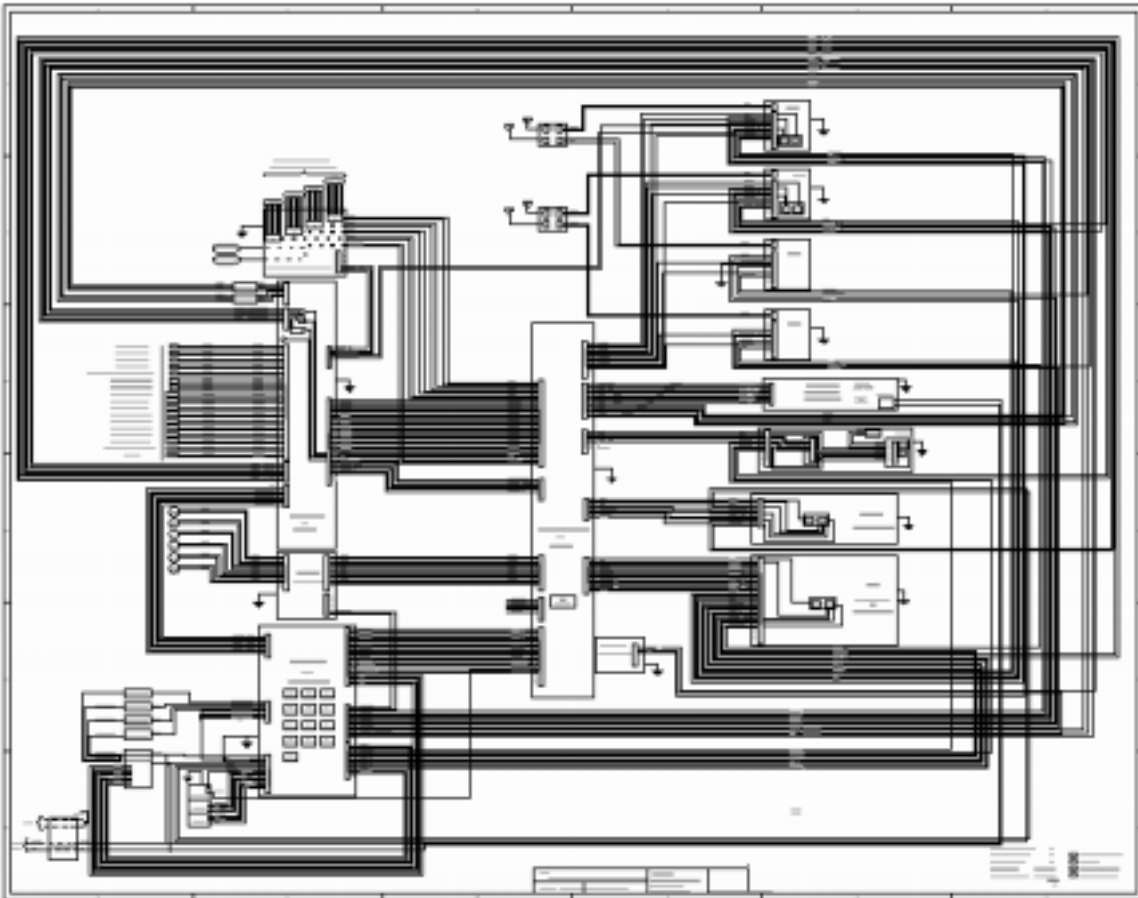


Figura 6.12. Diagrama general de las computadoras de SATEX.

En la figura 6.12 se aprecia el diagrama general de conexiones para la instrumentación de SATEX, buena parte de estas líneas pasan por el SIMSAT y son centralizadas hacia la CV.

Ya que se conectaron todos los equipos, se procede a energizar toda la instrumentación, con ayuda de una fuente de poder que suministra todos los voltajes necesarios para la CV, entre los que destacan +12, -12, +5, -5 [Volts].

### 6.4 Inicio de vida de SATEX

SATEX inicia su vida con ayuda de SOFMIN, ya que las localidades de la memoria RAM de las tarjetas de procesamiento no tienen ninguna información. Entonces, el SIM-SOS envía las señales pertinentes para el encendido de el procesador principal en modo ROM, es decir, accionando el mapa de memoria que permite la utilización de memoria ROM y RAM.

Las señales que debe enviar el SIM-SOS, son emuladas por SOFDEVO, quien las envía por el puerto paralelo y son la siguientes:

PIN de Salida	Señal	Nivel
4	ON#/OFF CP	5 [V]
5	SEL CR0-1	0 [V]
6	DT ON CR#	0 [V]
9	ROM/RAM#	5 [V]

Después, envía un reset a la CV y desactiva el modo BTL:

PIN de Salida	Señal	Nivel
7	RESET CV#	5 [V]
8	DWNLSW TTL#	0 [V]

Espera tres segundos y quita el reset:

PIN de Salida	Señal	Nivel
7	RESET CV#	0 [V]

SOFMIN inicia la adquisición de TLMN durante una órbita, SOFMIN está programado para que realice una órbita de adquisición de telemetría, además de la comunicación con Tierra en los diferentes momentos en que se activa la comunicación de red externa, propiamente cuando se activa la interrupción SORIE. SOFDEVO monitorea todas las acciones de red interna y contesta a los comandos de las computadoras que emula.

Entonces, es tiempo que el usuario conecte a SOFDEVO a la red interna de SATEX por medio del puerto serial de la PC y con ayuda del menú de SOFDEVO, figura 6.13.



Figura 6.13. Conexión de SOFDEVO a la red interna de SATEX

Cuando la CV inicia sus operaciones, envía dos comandos por red interna a SOFDEVO que son la liberación y el estado de sus variables de telemetría. Estos mensajes de la red interna de SATEX se pueden observar con SOFDEVO gracias a su programación que permite el monitoreo de la red, figura 6.14.

Solicitud de Comando	Respuesta de Comando
1><P><Rx>2F,10,50,41,41,41,41,41,41,41,41,41,16 1><P>(Cmd 10.1) CP avisa a Sefdevo que el satélite se ha liberado  2><P><Rx>2F,2D,50,0,0,0,0,0,0,0,0,0,83 2><P>(Cmd 45.1) CP envía a Sofdevo el estado de TLMN	

Figura 6.14. Pantallas de monitoreo de SOFDEVO donde se muestra el tráfico de la red interna del satélite.

Ya que SOFDEVO recibió la información de las variables de telemetría de la CV, todos los displays digitales son inicializados en los valores enviados por la CV, figura 6.15.

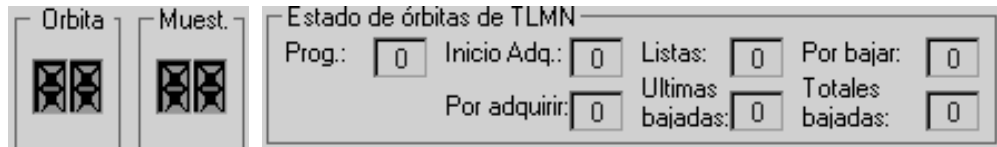


Figura 6.15. Displays de SOFDEVO para la contabilización de muestreos, órbitas y estados de telemetría.

El software de operación satelital está programado para que realice siempre una órbita de telemetría normal, es decir, siempre que inicie su vida, la CV estará capturando información de telemetría durante una órbita, siempre y cuando no sea enviada una misión a la CV.

Cada muestreo lo realiza mediante la interrupción que se genera cada 10 minutos, y a su vez las interrupciones son registradas en el contador de muestreos. Al llegar a 10 muestreos, indica que la CV realizó una órbita completa y está lista para bajar la información de esa órbita cuando la ET se lo indique. En la figura 6.11 se presenta el monitoreo de un muestreo de la adquisición de telemetría de la CV realizado con SOFMIN.

Solicitud de Comando	Respuesta de Comando
1><R><Rx>2F,48,20,41,41,41,41,41,41,41,41,41,E 1><R>K (Se recibió la 'K' enviada) 1><R>(Cmd 48) CV solicita a SP le indique si está haciendo CDI 1><R><Rx>Cmd enviado por Softdevo: F,49,2,4E,41,41,41,41,41,41,41,41,1E 1><R>K  2><R><Rx>2F,44,20,55,41,41,41,41,41,41,41,41,FE 2><R>K (Se recibió la 'K' enviada) 2><R>(Cmd 44.1) CV envía a DT el status de su procesador activo 2><R>Procesador <b>Redundante</b> en uso  3><R><Rx>2F,26,40,41,41,41,41,41,41,41,41,41,10 3><R>K (Se recibió la 'K' enviada) 3><R>(Cmd 26.1) CP solicita a SP le indique cuáles computadoras se pueden encender	1><R>K 49 1E 1><R><Tx>F,49,2,4E,41,41,41,41,41,41,41,41,41,1E (1a vez)  2><R>K  3><R>K 27 10 3><R>(Cmd 27.1) SP le indica a CP cuáles

**CAPÍTULO 6. VALIDACIÓN DEL SOFTWARE**

<p>3&gt;&lt;R&gt;&lt;Rx&gt;Cmd enviado por Softdevo: F,27,4,AA,AA,AA,41,41,41,41,41,41,10 3&gt;&lt;R&gt;K</p> <p>4&gt;&lt;R&gt;&lt;Rx&gt;2F,25,40,AA,AA,41,41,41,41,41,41,41,3F 4&gt;&lt;R&gt;K (Se recibió la 'K' enviada) 4&gt;&lt;R&gt;(Cmd 25.1) CP solicita a SP el encendido o apagado de procesadores para diagnóstico simple o votero 4&gt;&lt;R&gt;CP solicita a SP encienda la CUO 4&gt;&lt;R&gt;CP solicita a SP encienda el DT 4&gt;&lt;R&gt;No se ha definido correctamente el cmd 25 (dec) para CCD. (Cmd no ejecutado)</p> <p>5&gt;&lt;R&gt;&lt;Rx&gt;2F,1B,70,55,41,41,41,41,41,41,41,41,D7 5&gt;&lt;R&gt;(Cmd 1B.1) CP avisa CU's inicio de autodiagnóstico 5&gt;&lt;R&gt;Se realizara el diagnostico SIMPLE</p> <p>6&gt;&lt;R&gt;0 6&gt;&lt;R&gt;CP ha probado el puerto Serie</p> <p>7&gt;&lt;R&gt;&lt;Rx&gt;2F,1,10,41,41,41,41,41,41,41,41,41,65 7&gt;&lt;R&gt;K (Se recibió la 'K' enviada) 7&gt;&lt;R&gt;(Cmd 1.1) CP solicita diagnóstico a CUO 7&gt;&lt;R&gt;&lt;Rx&gt;Cmd enviado por Softdevo: F,2,1,0,0,41,41,41,41,41,41,41,41,F5 7&gt;&lt;R&gt;K</p> <p>8&gt;&lt;R&gt;&lt;Rx&gt;2F,1,20,41,41,41,41,41,41,41,41,41,55 8&gt;&lt;R&gt;K (Se recibió la 'K' enviada) 8&gt;&lt;R&gt;(Cmd 1.2) CP solicita diagnóstico a DT 8&gt;&lt;R&gt;&lt;Rx&gt;Cmd enviado por Softdevo: F,2,2,0,0,41,41,41,41,41,41,41,41,F4 8&gt;&lt;R&gt;K</p> <p>9&gt;&lt;R&gt;&lt;Rx&gt;2F,1,40,41,41,41,41,41,41,41,41,41,35 9&gt;&lt;R&gt;K (Se recibió la 'K' enviada) 9&gt;&lt;R&gt;(Cmd 1.4) CP solicita diagnóstico a SP 9&gt;&lt;R&gt;&lt;Rx&gt;Cmd enviado por Softdevo: F,2,4,2,5,41,41,41,41,41,41,41,41,EB 9&gt;&lt;R&gt;K</p> <p>10&gt;&lt;R&gt;&lt;Rx&gt;2F,14,10,41,41,41,41,41,41,41,41,41,52 10&gt;&lt;R&gt;K (Se recibió la 'K' enviada) 10&gt;&lt;R&gt;(Cmd 14.1) CP solicita a CUO envío de comandos en su stack 10&gt;&lt;R&gt;&lt;Rx&gt;Cmd enviado por Softdevo: F,15,1,78,78,78,78,78,78,78,78,78,78,3A 10&gt;&lt;R&gt;K</p> <p>11&gt;&lt;R&gt;&lt;Rx&gt;2F,14,20,41,41,41,41,41,41,41,41,41,42 11&gt;&lt;R&gt;K (Se recibió la 'K' enviada) 11&gt;&lt;R&gt;(Cmd 14.2) CP solicita a DT envío de comandos en su stack 11&gt;&lt;R&gt;&lt;Rx&gt;Cmd enviado por Softdevo: F,15,2,78,78,78,78,78,78,78,78,78,78,39 11&gt;&lt;R&gt;K</p> <p>12&gt;&lt;R&gt;&lt;Rx&gt;2F,14,40,41,41,41,41,41,41,41,41,41,22 12&gt;&lt;R&gt;K (Se recibió la 'K' enviada) 12&gt;&lt;R&gt;(Cmd 14.3) CP solicita a SP envío de comandos en su stack</p>	<p>computadoras se pueden encender 3&gt;&lt;R&gt;&lt;Tx&gt;F,27,4,AA,AA,AA,41,41,41,41,41,41,10 (1a vez)</p> <p>4&gt;&lt;R&gt;K 4&gt;&lt;R&gt;SP ha encendido la CUO 4&gt;&lt;R&gt;SP ha encendido el DT</p> <p>5&gt;&lt;R&gt;CU's han realizando su autodiagnóstico ...</p> <p>7&gt;&lt;R&gt;K 2 F5 7&gt;&lt;R&gt;(Cmd 2.6) CUO envía su diagnóstico a CP 7&gt;&lt;R&gt;&lt;Tx&gt;F,2,1,0,0,41,41,41,41,41,41,41,41,F5 (1a vez)</p> <p>8&gt;&lt;R&gt;K 2 F4 8&gt;&lt;R&gt;(Cmd 2.7) DT envía su diagnóstico a CP 8&gt;&lt;R&gt;&lt;Tx&gt;F,2,2,0,0,41,41,41,41,41,41,41,41,F4 (1a vez)</p> <p>9&gt;&lt;R&gt;K 2 EB 9&gt;&lt;R&gt;(Cmd 2.9) SP envía su diagnóstico a CP 9&gt;&lt;R&gt;&lt;Tx&gt;F,2,4,2,5,41,41,41,41,41,41,41,41,EB (1a vez)</p> <p>10&gt;&lt;R&gt;K 15 3A 10&gt;&lt;R&gt;(Cmd 15.1) CUO avisa a CP que los comandos en su stack se han terminado o que no hay comandos 10&gt;&lt;R&gt;&lt;Tx&gt;F,15,1,78,78,78,78,78,78,78,78,78,78,3A (1a vez)</p> <p>11&gt;&lt;R&gt;K 15 39 11&gt;&lt;R&gt;(Cmd 15.2) DT avisa a CP que los comandos en su stack se han terminado o que no hay comandos 11&gt;&lt;R&gt;&lt;Tx&gt;F,15,2,78,78,78,78,78,78,78,78,78,78,39 (1a vez)</p> <p>12&gt;&lt;R&gt;K 15 37 12&gt;&lt;R&gt;(Cmd 15.3) SP avisa a CP que los comandos en</p>
--	---

<p>12&gt;&lt;R&gt;&lt;Rx&gt;Cmd enviado por Softdevo: F,15,4,78,78,78,78,78,78,78,78,37 12&gt;&lt;R&gt;K</p> <p>13&gt;&lt;R&gt;&lt;Rx&gt;2F,25,40,55,55,55,41,41,41,41,41,41,41,D5 13&gt;&lt;R&gt;K (Se recibió la 'K' enviada) 13&gt;&lt;R&gt;(Cmd 25.1) CP solicita a SP el encendido o apagado de procesadores para diagnóstico simple o voto 13&gt;&lt;R&gt;CP solicita a SP apague la CUO 13&gt;&lt;R&gt;CP solicita a SP apague el DT 13&gt;&lt;R&gt;CP solicita a SP apague la CCD, pero ésta ya se encontraba apagada</p> <p>14&gt;&lt;R&gt;&lt;Rx&gt;2F,28,40,11,41,41,41,41,41,41,41,41,3E 14&gt;&lt;R&gt;K (Se recibió la 'K' enviada) 14&gt;&lt;R&gt;(Cmd 28.1) CP solicita a SP le envíe los valores de sus primeros 8 sensores 14&gt;&lt;R&gt;&lt;Rx&gt;Cmd enviado por Softdevo: F,29,4,11,80,80,0,80,80,0,80,80,41,81 14&gt;&lt;R&gt;K</p> <p>15&gt;&lt;R&gt;&lt;Rx&gt;2F,28,40,22,41,41,41,41,41,41,41,41,2D 15&gt;&lt;R&gt;K (Se recibió la 'K' enviada) 15&gt;&lt;R&gt;(Cmd 28.2) CP solicita a SP le envíe los valores de sus últimos 9 sensores 15&gt;&lt;R&gt;&lt;Rx&gt;Cmd enviado por Softdevo: F,29,4,22,80,80,80,80,80,0,80,80,80,B1 15&gt;&lt;R&gt;K</p> <p>16&gt;&lt;R&gt;&lt;Rx&gt;2F,2D,50,0,1,0,1,1,0,0,1,1,4E,30 16&gt;&lt;R&gt;(Cmd 45.1) CP envía a Sefdevo el estado de TLMN</p>	<p>su stack se han terminado o que no hay comandos 12&gt;&lt;R&gt;&lt;Tx&gt;F,15,4,78,78,78,78,78,78,78,78,78,37 (1a vez)</p> <p>13&gt;&lt;R&gt;K 13&gt;&lt;R&gt;SP ha apagado la CUO 13&gt;&lt;R&gt;SP ha apagado el DT</p> <p>14&gt;&lt;R&gt;K 29 81 14&gt;&lt;R&gt;(Cmd 29.1) SP envía a CP los valores de sus primeros 8 sensores 14&gt;&lt;R&gt;&lt;Tx&gt;F,29,4,11,80,80,0,80,80,0,80,80,41,81 (1a vez)</p> <p>15&gt;&lt;R&gt;K 29 B1 15&gt;&lt;R&gt;(Cmd 29.2) SP envía a CP los valores de sus últimos 9 sensores 15&gt;&lt;R&gt;&lt;Tx&gt;F,29,4,22,80,80,80,80,80,0,80,80,80,B1 (1a vez)</p>
---	--

Figura 6.16. Monitoreo de todo un muestreo utilizando el software SOFMIN.

Durante el monitoreo del primer muestreo de la CV utilizando SOFMIN, figura 6.16, se puede observar:

- ✓ La incorporación del Comando de Diagnóstico Iterado (CDI) como primer instrucción de CV a SIM-SOS. Esto lo realizará siempre en el primer muestreo de adquisición de telemetría, en los siguientes no se realizará. Esto con el fin de que no altere los tiempos orbitales del satélite.
- ✓ El procesador activo es el redundante. Como ya se comentó, por ahora la tarjeta que tiene a SOFMIN instalado es la tarjeta redundante. Esto porque las EPROM con las que se cuenta son lentas (150 ns) y además porque esa tarjeta no tiene el dispositivo EDAC. La utilización de las EPROM son de gran importancia para la validación de SOFMIN.
- ✓ No realiza la prueba de VOTEO, es decir, el diagnóstico de las computadoras será simple y no se hará el experimento de MACV, esto para reducir las líneas de código fuente y que su tamaño sea menor a 64 Kb. Además, el experimento MACV puede realizarse si la ET se lo indica en una misión nueva.

### 6.5 Validación de fallas simuladas

SOFDEVO puede simular fallas en los diagnósticos de las computadoras que simula, además de diversas fallas en red interna. En particular, SOFDEVO puede simular las fallas que se muestran en la figura 6.17.

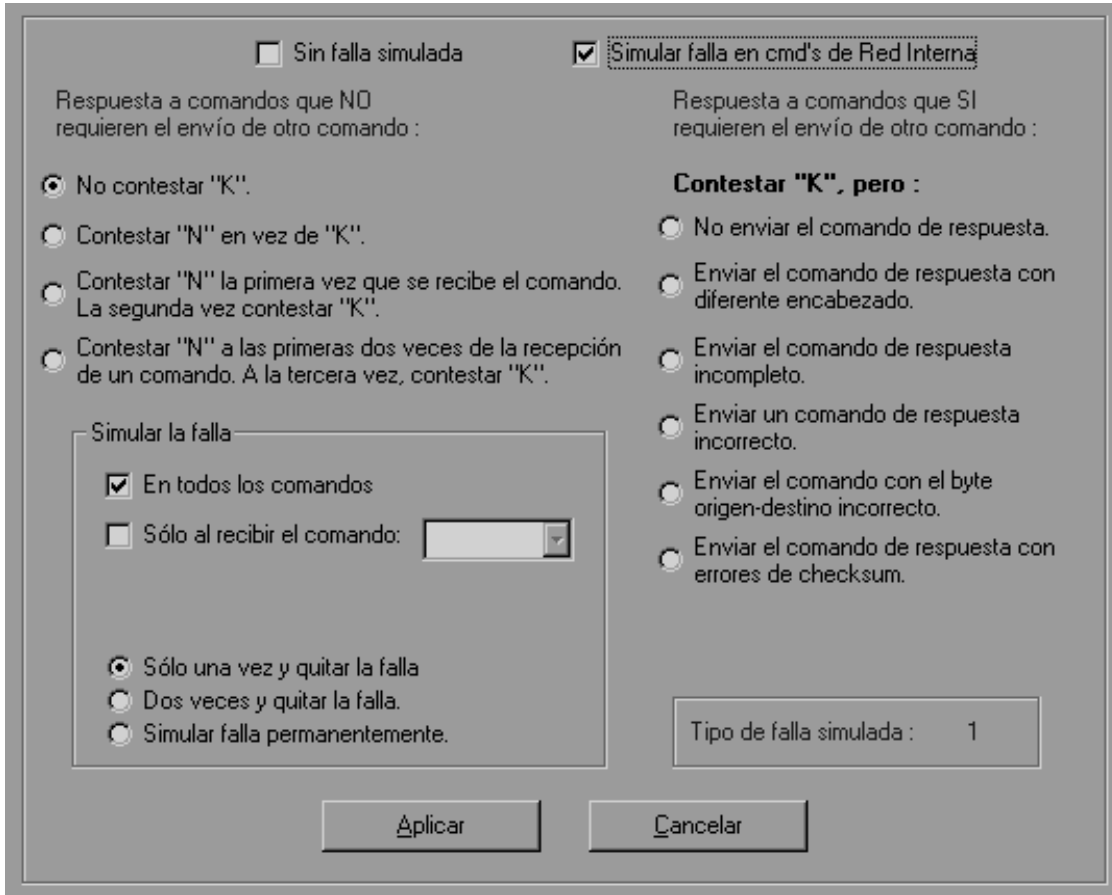


Figura 6.17. Ventana que utiliza SOFDEVO para la simulación de cualquier falla en Red Interna.

Todas las fallas que se pueden elegir en la ventana anterior son registradas y monitoreadas en las pantallas de Red Interna. SOFDEVO enciende un foco virtual alertando que se está simulando una falla en red interna, se pueden simular 56 tipos de fallas en la red interna, dependiendo de las configuraciones de las diferentes opciones, figura 6.18.



Figura 6.18. Diferentes imágenes que muestran el tipo de falla simulada en red interna.

La simulación de fallas en los diagnósticos se realiza con SOFDEVO, tal como lo muestra la figura 6.19.

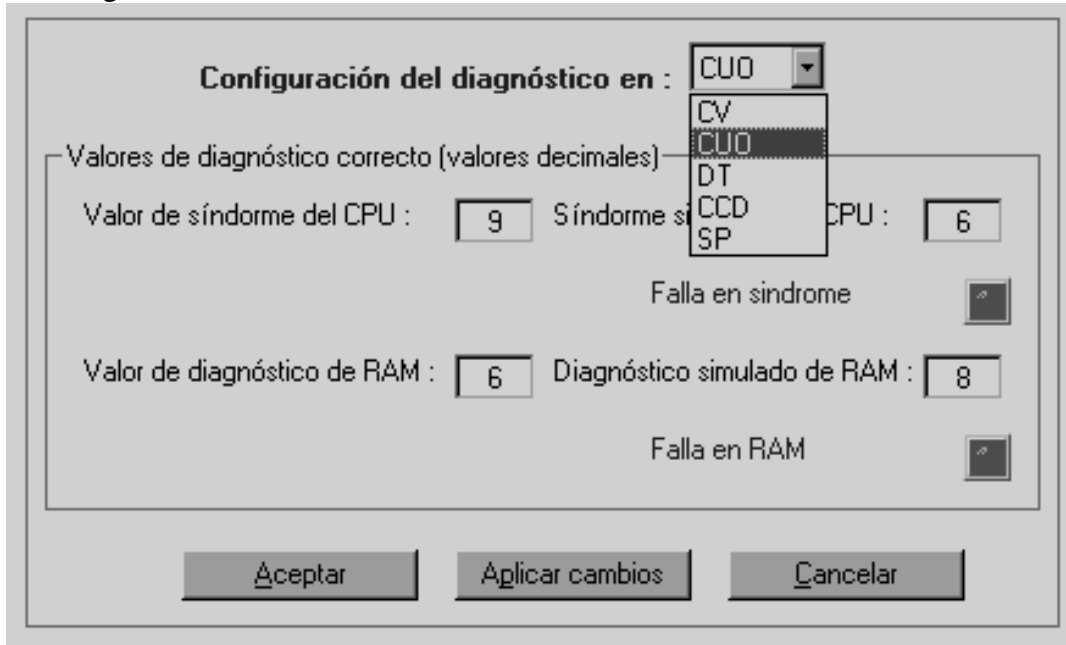


Figura 6.19. Ventana que utiliza SOFDEVO para la simulación de cualquier falla en los equipos.

Un diagnóstico simulado o el síndrome de cualquier computadora simulada, debe ser un número diferente de 0 y hasta 256. Las fallas simuladas se visualizan en el panel de control de SOFDEVO, como se observa en la figura 6.20.

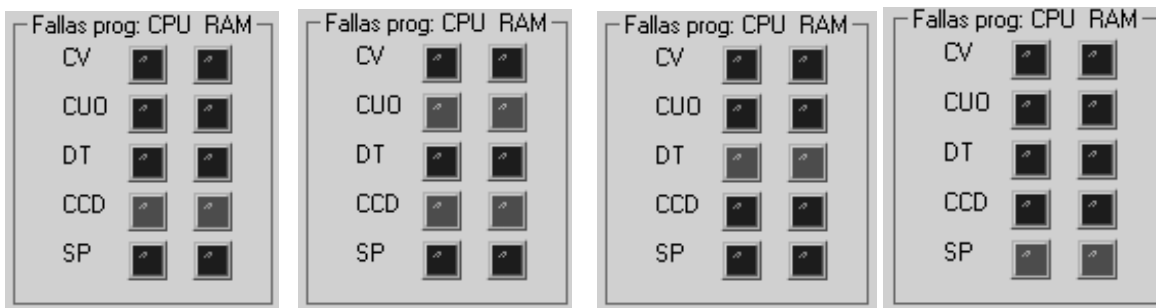


Figura 6.20. Diferentes casos de simulación de fallas en la computadoras emuladas por SOFVEDO.

Dependiendo de la falla programada en cualquiera de los diagnósticos de la CV, repercutirá en la matriz de voto mayoritario y en el diagnóstico que se almacena en la telemetría capturada por la CV. Cada falla programada se almacenará en la telemetría de la CV y además se visualizará en SOFDEVO.

La CV almacena toda la información de los muestreos adquiridos y cuando completa una órbita, la Estación Terrena puede hacer la petición de bajar la telemetría de una órbita. Para ello La ET abre su puerto de comunicaciones y contacta a SATEX mediante el comando Búsqueda de SATEX, este comando permite la comunicación con el satélite a tres tasas de baudaje: 9600, 4800 y 2400 baudios, figura 6.21.



Figura 6.21. Ventana que despliega el SET para la búsqueda de SATEX.

Entonces la ET pide a SATEX bajar información de telemetría, ya sea normal, figura 6.22, o Especial.

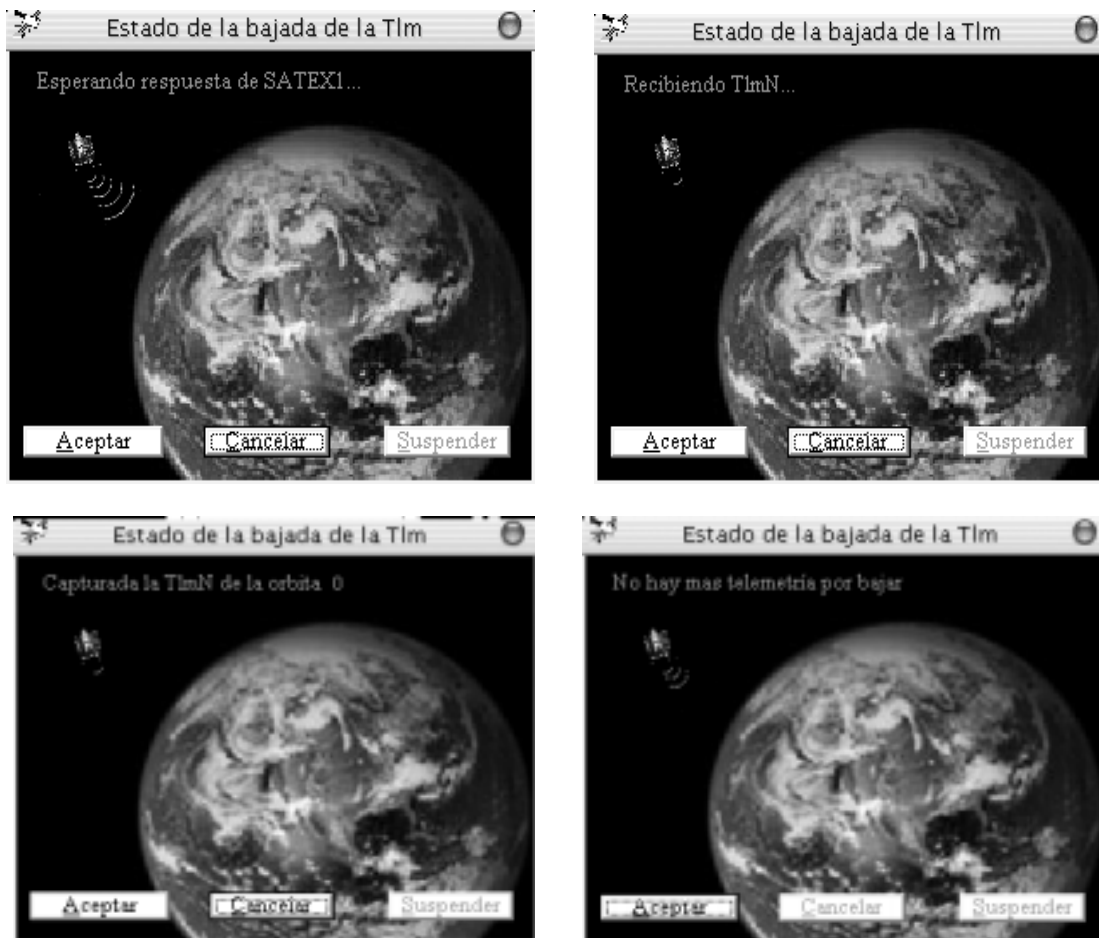


Figura 6.22. Imágenes que muestran el proceso de recepción de TLMN en ET.



Ya que se bajó la telemetría, SET la corrobora y la almacena en su base de datos y muestra que se bajó una órbita de telemetría o los paquetes de telemetría Especial, figura 6.23.



Figura 6.18. Objeto que indica la recepción de TLMN.

Cuando el SET decodifica la información enviada por la CV, el personal en Tierra puede analizar la información almacenada y observar el estado de los equipos de SATEX, figura 6.24.

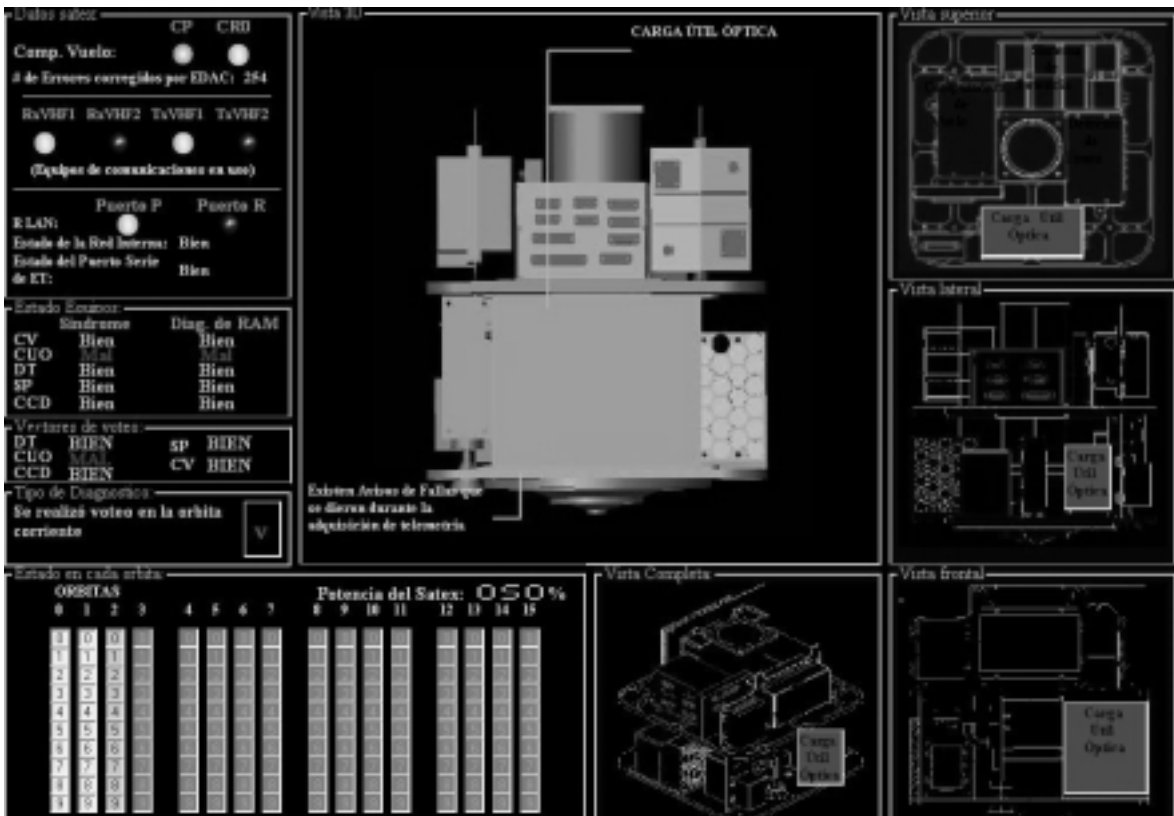


Figura 6.24. Ventana del SET donde se analiza el estado de los equipos de SATEX.

Con esta pantalla que despliega el SET es posible observar cuáles de las computadoras de SATEX tienen falla, además de la red interna que está utilizando, el transmisor y el número de computadoras que fue posible anergizar.

En esta ventana se pueden observar las fallas que se pueden presentar en el funcionamiento de SATEX; en SOFDEVO se simula cualquier tipo de falla y el SET alerta automáticamente sobre la falla emulada.

## 6.6 Subir programa

Una vez que SATEX inicia su vida, el software que está rigiendo los controles de SATEX es SOFMIN y como ya se mencionó, SOFMIN contiene funciones básicas para que el satélite pueda ser contactado por la ET, además de la adquisición de una órbita de telemetría normal. Pero SATEX puede realizar muchas más tareas y experimentos, por ello tiene la capacidad de que a la CV se le pueda subir un nuevo software desde la ET, que permita un uso más provechoso del satélite. Para tales condiciones, el software de operaciones del satélite tiene una rutina que realiza y valida el proceso de carga de nuevos programas de acuerdo con el siguiente procedimiento:

- 1) Mediante la ET, se contacta a SATEX cuando se encuentra en línea de vista, luego se le envía el comando “*Nuevo programa*” a la CV, figura 6.25, indicando el procesador destino, ésta lo retransmite al SIM-SOS y espera el reconocimiento del mismo en Tierra. De esta forma SIM-SOS sabe cual procesador es el que estaba activado y en caso de que falle el intento de subir programa, deberá ordenar (al finalizar el intento) que quede como activo tal procesador en la CV.

Solicitud de Comando	Respuesta de Comando
1><P><Rx>2F,13,50,17,1,41,41,41,41,41,41,7D 1><P>(Cmd 13.1) CP avisa a sefdevo que ha ejecutado el comando 23[Línea] 1><P>ET ha contactado a Satex 2><P><Rx>2F,6,20,52,41,41,41,41,41,41,41,3F 2><P>K (Se recibió la 'K' enviada) 2><P>(Cmd 6.1) CP avisa a DT que se va a subir un nuevo programa al satélite	

Figura 6.25. Monitoreo de la red interna donde se aprecia la solicitud de subir nuevo programa.

- 2) SIM-SOS apaga a todos los procesadores mediante el las líneas: ON\_CR\_DTL=1, ON#/OFF CP=1, SELCR0-1=1, figura 6.26. Activa el mapa de memoria que utiliza sólo RAM, mediante la línea ROM/RAM#=0. Enciende la CV de acuerdo con las opciones recibidas en el comando de *Subir programa* (Principal o Redundante). Espera a que la CV envíe un cambio de estado en la línea RCPROG de “1” a “0” lógico.

Solicitud de Comando	Respuesta de Comando
Se apagan todos los procesadores  La TP destino es la Redundante  Esperando la señal continúa RCPROG=0	

Figura 6.26. Monitoreo donde SOFDEVO avisa de las acciones que realiza.

- 3) Cuando SIM-SOS recibe la señal continúa, activa el hardware de arranque en modo BTL mediante la línea DWNLSW\_TTL#=0. Envía un pulso por la línea RESET CV#, espera 3 segundos y desactiva el hardware de arranque en modo BTL mediante el establecimiento de la línea DWNLSW TTL# a un nivel alto, con ello avisa que puede enviar el primero de cinco intentos para subir programa.

- 4) El Software de Estación Terrena ejecuta la función de “Carga de Programa”, figura 6.27.



Figura 6.27. Ventana que despliega el SET para la el envío de nuevo programa.

- 5) SIM-SOS espera por un tiempo  $T_{0max} = 50$  seg. la respuesta que dará la CV mediante una transición de un 0 lógico a un 1 lógico en la línea **RCPROG**, en caso de que no se obtenga una respuesta de la CV, SIM-SOS debe proceder al paso 3 con un máximo de 5 reintentos, si no se recibe la respuesta “éxito” después de 3 intentos se procede al paso 8.
- 6) SIM-SOS envía nuevamente un pulso por la línea **RESET CV#** y termina la carga de programa con éxito.
- 7) SIM-SOS debe activar el mapa de memoria que utiliza ROM/RAM y encender la CV con el procesador indicado en el paso 0 mediante el uso de las líneas pertinentes.
- 8) Finalmente SIM-SOS, envía un pulso por la línea **RESET CV#**, en este caso se suspende el intento de transferencia de nuevo programa al satélite.

Con la técnica de cargar programas en modo “BOOTSTRAP LOADER” (BTL), que es un cargador de arranque que está alojado en una memoria ROM de arranque especial, es posible cargar un programa de 32 Kb en la memoria del SAB80C166 vía puerto serial. El pequeño programa de arranque del SAB puede utilizarse para cargar software de usuario o aplicaciones más amplias a memoria RAM interna o externa, se puede consultar en [ORTIZ, 2003].

### 6.7 Validación del Experimento MACV

SOFDEVO permite la depuración del software asociado a los procesos de diagnóstico, detección y corrección de fallas de la arquitectura de cómputo semivirtual que se desarrolló como quinto experimento para el microsatélite. La arquitectura resuelve autónomamente los procesos de diagnóstico, detección y de corrección automatizada de fallas en los procesadores de la computadora de vuelo contando con el apoyo periódico de las 4 computadoras del SATEX (emulados en este caso con SOFDEVO).

Para que el SIM-SOS pueda dar mantenimiento a la CV de SATEX, como ya se explicó en el capítulo anterior, se debe presentar una falla en la CV o una falla en el SIM-SOS, pero con la condición de que se esté realizando el Voto Mayoritario.

Para validar el funcionamiento del Experimento MACV, SOFDEVO permite emular fallas en las dos computadoras, CV y SIM-SOS, para poder observar mediante el monitoreo de la red interna de SATEX, el mantenimiento de la CV.

Para ello se necesita:

- ✓ Conectar toda la instrumentación de SATEX la CV, EAS y M, SOFDEVO, SET, SIMSAT y la fuente de poder.
- ✓ Que las tarjetas de procesamiento tengan instalados el SOFMIN.
- ✓ Subir el software General de Operación Satelital.
- ✓ Que SATEX esté realizando sus rutinas de adquisición de telemetría.

Ya que se tienen los pasos anteriores, la ET envía una misión a SATEX, indicándole que realice el Voto Mayoritario, figura 6.28.

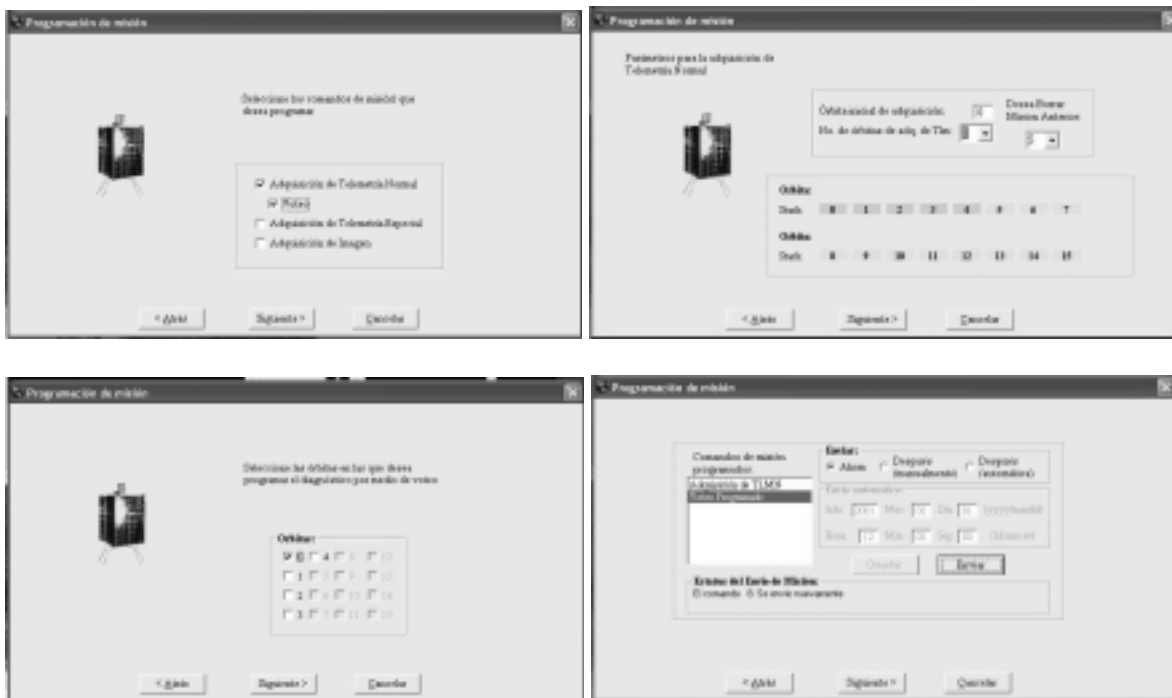


Figura 6.28. Ventanas del SET donde se va formando el comando de misión a SATEX

En la figura 6.28 se puede observa la formación y el envío de un comando de misión en el cual se programa una misión de adquisición de TLMN y VOTEO. En este caso, se programó una adquisición de TLMN en cinco órbitas y en la primera se realizará la prueba de VOTEO.

En las pantallas de monitoreo de SOFDEVO se puede apreciar la ejecución y los avisos de los comandos enviados por el SET, figura 6.29.

Solicitud de Comando	Respuesta de Comando
1><R><Rx>2F,13,50,17,1,41,41,41,41,41,41,7D 1><R>(Cmd 13.1) CP avisa a sefdevo que ha ejecutado el comando 23[Línea] 1><R>ET ha contactado a Satex	
2><R><Rx>2F,13,50,8,1,41,41,41,41,41,41,8C 2><R>(Cmd 13.1) CP avisa a sefdevo que ha ejecutado el comando 8[Línea] 2><R>Programación de diagnóstico por Voteco	
3><R><Rx>2F,13,50,7,1,41,41,41,41,41,41,8D 3><R>(Cmd 13.1) CP avisa a sefdevo que ha ejecutado el comando 7[Línea] 3><R>Programación de adquisición de TlmN	

Figura 6.29. Monitoreo donde se aprecia el envío de una misión al satélite.

Posteriormente entra en acción la interrupción de cada 10 minutos y el Software de Operación Satelital comienza la adquisición de telemetría. Para los primeros 5 muestreos, no se le simula ninguna falla en los diagnósticos de las computadoras de SATEX, por lo que el monitoreo de los muestreos no presenta ninguna falla y la matriz de resultados de VOTEO no presenta ninguna falla en las computadoras participantes.

Pero en el quinto muestreo, con ayuda de SOFDEVO, se le simula una falla en la CV, propiamente en su diagnóstico y su Síndrome de RAM, figura 6.30.



Figura 6.30. Ventanas donde se aprecia la simulación de una falla en la CV.

Ya que se simula la falla con SOFDEVO, en las pantallas de monitoreo se puede observar paso a paso la ejecución del el experimento MACV, figura 6.31.

Solicitud de Comando	Respuesta de Comando
<p>1&gt;&lt;P&gt;&lt;Rx&gt;2F,44,20,10,41,41,41,41,41,41,41,0,84                      1&gt;&lt;P&gt;K (Se recibió la 'K' enviada)                      1&gt;&lt;P&gt;(Cmd 44.1) CV envía a DT el status de su procesador activo                      1&gt;&lt;P&gt;Procesador Principal en uso</p>	<p>1&gt;&lt;P&gt;K</p>
<p>2&gt;&lt;P&gt;&lt;Rx&gt;2F,26,40,41,41,41,41,41,41,41,41,10                      2&gt;&lt;P&gt;K (Se recibió la 'K' enviada)                      2&gt;&lt;P&gt;(Cmd 26.1) CP solicita a SP le indique cuáles computadoras se pueden encender                      2&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo:                      F,27,4,AA,AA,AA,41,41,41,41,41,41,10                      2&gt;&lt;P&gt;K</p>	<p>2&gt;&lt;P&gt;K                      27                      10                      2&gt;&lt;P&gt;(Cmd 27.1) SP le indica a CP cuáles computadoras se pueden encender                      2&gt;&lt;P&gt;&lt;Tx&gt;F,27,4,AA,AA,AA,41,41,41,41,41,41,10 (1a vez)</p>
<p>3&gt;&lt;P&gt;&lt;Rx&gt;2F,25,40,AA,AA,41,41,41,41,41,41,41,3F                      3&gt;&lt;P&gt;K (Se recibió la 'K' enviada)                      3&gt;&lt;P&gt;(Cmd 25.1) CP solicita a SP el encendido o apagado de procesadores para diagnóstico simple o voteo                      3&gt;&lt;P&gt;CP solicita a SP encienda la CUO                      3&gt;&lt;P&gt;CP solicita a SP encienda el DT, pero éste ya se encontraba encendido                      3&gt;&lt;P&gt;No se ha definido correctamente el cmd 25 (dec) para CCD. (Cmd no ejecutado)</p>	<p>3&gt;&lt;P&gt;K                      3&gt;&lt;P&gt;SP ha encendido la CUO</p>
<p>4&gt;&lt;P&gt;&lt;Rx&gt;2F,1B,70,AA,41,41,41,41,41,41,41,82                      4&gt;&lt;P&gt;(Cmd 1B.1) CP avisa CU's inicio de autodiagnóstico                      4&gt;&lt;P&gt;Se realizara el diagnostico por VOTEO</p>	<p>4&gt;&lt;P&gt;CU's han realizando su autodiagnóstico ...</p>
<p>5&gt;&lt;P&gt;0                      5&gt;&lt;P&gt;CP ha probado el puerto Serie</p>	
<p>6&gt;&lt;P&gt;&lt;Rx&gt;2F,1,10,41,41,41,41,41,41,41,41,65                      6&gt;&lt;P&gt;K (Se recibió la 'K' enviada)                      6&gt;&lt;P&gt;(Cmd 1.1) CP solicita diagnóstico a CUO                      6&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo:                      F,2,71,0,0,41,41,41,41,41,41,41,85                      6&gt;&lt;P&gt;K</p>	<p>6&gt;&lt;P&gt;K                      2                      85                      6&gt;&lt;P&gt;(Cmd 2.2) CUO envía su diagnóstico en broadcasting                      6&gt;&lt;P&gt;&lt;Tx&gt;F,2,71,0,0,41,41,41,41,41,41,41,85 (1a vez)</p>
<p>7&gt;&lt;P&gt;&lt;Rx&gt;2F,1,20,41,41,41,41,41,41,41,41,55                      7&gt;&lt;P&gt;K (Se recibió la 'K' enviada)                      7&gt;&lt;P&gt;(Cmd 1.2) CP solicita diagnóstico a DT                      7&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo:                      F,2,72,0,0,41,41,41,41,41,41,41,84                      7&gt;&lt;P&gt;K</p>	<p>7&gt;&lt;P&gt;K                      2                      84                      7&gt;&lt;P&gt;(Cmd 2.3) DT envía su diagnóstico en broadcasting                      7&gt;&lt;P&gt;&lt;Tx&gt;F,2,72,0,0,41,41,41,41,41,41,41,84 (1a vez)</p>
<p>8&gt;&lt;P&gt;&lt;Rx&gt;2F,1,40,41,41,41,41,41,41,41,41,35                      8&gt;&lt;P&gt;K (Se recibió la 'K' enviada)                      8&gt;&lt;P&gt;(Cmd 1.4) CP solicita diagnóstico a SP                      8&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo:                      F,2,74,0,0,41,41,41,41,41,41,41,82                      8&gt;&lt;P&gt;K</p>	<p>8&gt;&lt;P&gt;K                      2                      82                      8&gt;&lt;P&gt;(Cmd 2.5) SP envía su diagnóstico en broadcasting                      8&gt;&lt;P&gt;&lt;Tx&gt;F,2,74,0,0,41,41,41,41,41,41,41,82 (1a vez)</p>
<p>9&gt;&lt;P&gt;&lt;Rx&gt;2F,2,70,0,0,41,41,41,41,41,41,41,86                      9&gt;&lt;P&gt;(Cmd 2.1) CP envía su diagnóstico en broadcasting</p>	<p>9&gt;&lt;P&gt;CUO ha capturado el diagnóstico de CP (Softdevo)                      9&gt;&lt;P&gt;DT ha capturado el diagnóstico de CP (Softdevo)                      9&gt;&lt;P&gt;CCD ha capturado el diagnóstico de CP (Softdevo)                      9&gt;&lt;P&gt;SP ha capturado el diagnóstico de CP (Softdevo)</p>
<p>10&gt;&lt;P&gt;&lt;Rx&gt;2F,1C,10,41,41,41,41,41,41,41,41,4A                      10&gt;&lt;P&gt;K (Se recibió la 'K' enviada)                      10&gt;&lt;P&gt;(Cmd 1C.1) CP pide a CUO que envíe en broadcast sus diagnósticos acumulados</p>	<p>10&gt;&lt;P&gt;K                      1D                      28</p>

**CAPÍTULO 6. VALIDACIÓN DEL SOFTWARE**

<p>10&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo: F,1D,71,5,0,0,0,0,41,41,41,41,28 10&gt;&lt;P&gt;K</p> <p>11&gt;&lt;P&gt;&lt;Rx&gt;2F,1C,20,41,41,41,41,41,41,41,41,3A 11&gt;&lt;P&gt;K (Se recibió la 'K' enviada) 11&gt;&lt;P&gt;(Cmd 1C.2) CP pide a DT que envíe en broadcast sus diagnósticos acumulados 11&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo: F,1D,72,5,0,0,0,0,41,41,41,41,27 11&gt;&lt;P&gt;K</p> <p>12&gt;&lt;P&gt;&lt;Rx&gt;2F,1C,40,41,41,41,41,41,41,41,41,1A 12&gt;&lt;P&gt;K (Se recibió la 'K' enviada) 12&gt;&lt;P&gt;(Cmd 1C.4) CP pide a SP que envíe en broadcast sus diagnósticos acumulados 12&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo: F,1D,74,5,0,0,0,0,41,41,41,41,25 12&gt;&lt;P&gt;K</p> <p>13&gt;&lt;P&gt;&lt;Rx&gt;2F,1D,70,0,0,0,0,0,41,41,41,41,2E 13&gt;&lt;P&gt;(Cmd 1D.1) CV envía en broadcast sus diagnósticos acumulados 13&gt;&lt;P&gt;CUO ha capturado los diagnosticos acumulados de CV 13&gt;&lt;P&gt;CCD ha capturado los diagnosticos acumulados de CV 13&gt;&lt;P&gt;CCD ha capturado los diagnosticos acumulados de CV 13&gt;&lt;P&gt;CV ha capturado los diagnosticos acumulados de CV 13&gt;&lt;P&gt;CUO está realizando el votoe... 13&gt;&lt;P&gt;CUO concluyó el votoe! 13&gt;&lt;P&gt;DT está realizando el votoe... 13&gt;&lt;P&gt;DT concluyó el votoe! 13&gt;&lt;P&gt;SP está realizando el votoe... 13&gt;&lt;P&gt;SP concluyó el votoe!</p> <p>14&gt;&lt;P&gt;&lt;Rx&gt;2F,41,60,50,41,41,41,41,41,41,41,41,C6 14&gt;&lt;P&gt;(Cmd 41.1) CV ordena a CU's la generación de señales eléctricas de permisos y de reconfiguración</p> <p>15&gt;&lt;P&gt;&lt;Rx&gt;2F,41,20,44,41,41,41,41,41,41,41,12 15&gt;&lt;P&gt;K (Se recibió la 'K' enviada) 15&gt;&lt;P&gt;(Cmd 41.1) DT realiza votoe y reconfiguración de CV 15&gt;&lt;P&gt;(Cmd 41.1) Se apagan procesadores 15&gt;&lt;P&gt;Se apagan todos los procesadores 15&gt;&lt;P&gt;(Cmd 41.1) Conmuta a procesador redundante Se da reset a la CV  15&gt;&lt;P&gt;Quita Reset de la CV  Desactiva modo BTL de la CV  15&gt;&lt;P&gt;Termina Reset de CV 16&gt;&lt;P&gt;&lt;Rx&gt;2F,10,50,41,41,41,41,41,41,41,41,16 16&gt;&lt;P&gt;(Cmd 10.1) CP avisa a Sefdevo que el satélite se ha liberado  17&gt;&lt;P&gt;&lt;Rx&gt;2F,2D,50,0,0,0,0,0,0,0,0,83 17&gt;&lt;P&gt;(Cmd 45.1) CP envía a Sefdevo el estado de TLMN</p>	<p>10&gt;&lt;P&gt;(Cmd 1D.2) CUO envía en broadcast sus diagnósticos acumulados 10&gt;&lt;P&gt;&lt;Tx&gt;F,1D,71,5,0,0,0,0,41,41,41,41,28 (1a vez)</p> <p>11&gt;&lt;P&gt;K 1D 27 11&gt;&lt;P&gt;(Cmd 1D.3) DT envía en broadcast sus diagnósticos acumulados 11&gt;&lt;P&gt;&lt;Tx&gt;F,1D,72,5,0,0,0,0,41,41,41,41,27 (1a vez)</p> <p>12&gt;&lt;P&gt;K 1D 25 12&gt;&lt;P&gt;(Cmd 1D.5) SP envía en broadcast sus diagnósticos acumulados 12&gt;&lt;P&gt;&lt;Tx&gt;F,1D,74,5,0,0,0,0,41,41,41,41,25 (1a vez)</p> <p>13&gt;&lt;P&gt;****RESULTADO DE VOTEO EN CUO**** 13&gt;&lt;P&gt;***** CV ***** CUO ***** DT ***** CCD ***** SP *** 13&gt;&lt;P&gt;*****MAL***** BIEN*****BIEN*****No P*****BIEN 13&gt;&lt;P&gt;****RESULTADO DE VOTEO EN DT**** 13&gt;&lt;P&gt;***** CV ***** CUO ***** DT ***** CCD ***** SP *** 13&gt;&lt;P&gt;*****MAL***** BIEN*****BIEN*****No P*****BIEN 13&gt;&lt;P&gt;****RESULTADO DE VOTEO EN SP**** 13&gt;&lt;P&gt;***** CV ***** CUO ***** DT ***** CCD ***** SP *** 13&gt;&lt;P&gt;*****MAL***** BIEN*****BIEN*****No P*****BIEN</p> <p>15&gt;&lt;P&gt;K</p>
---	---

**CAPÍTULO 6. VALIDACIÓN DEL SOFTWARE**

<p>1&gt;&lt;P&gt;&lt;Rx&gt;2F,48,20,41,41,41,41,41,41,41,41,E  1&gt;&lt;P&gt;K (Se recibió la 'K' enviada)  1&gt;&lt;P&gt;(Cmd 48) CV solicita a SP le indique si está haciendo CDI  1&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo:  F,49,2,4E,41,41,41,41,41,41,41,1E  1&gt;&lt;P&gt;K</p> <p>2&gt;&lt;P&gt;&lt;Rx&gt;2F,44,20,55,41,41,41,41,41,41,41,FE  2&gt;&lt;P&gt;K (Se recibió la 'K' enviada)  2&gt;&lt;P&gt;(Cmd 44.1) CV envía a DT el status de su procesador activo  2&gt;&lt;P&gt;Procesador Redundante en uso</p> <p>3&gt;&lt;P&gt;&lt;Rx&gt;2F,26,40,41,41,41,41,41,41,41,41,10  3&gt;&lt;P&gt;K (Se recibió la 'K' enviada)  3&gt;&lt;P&gt;(Cmd 26.1) CP solicita a SP le indique cuáles computadoras se pueden encender  3&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo:  F,27,4,AA,AA,AA,41,41,41,41,41,41,10  3&gt;&lt;P&gt;K</p> <p>4&gt;&lt;P&gt;&lt;Rx&gt;2F,25,40,AA,AA,41,41,41,41,41,41,41,3F  4&gt;&lt;P&gt;K (Se recibió la 'K' enviada)  4&gt;&lt;P&gt;(Cmd 25.1) CP solicita a SP el encendido o apagado de procesadores para diagnóstico simple o voteo  4&gt;&lt;P&gt;CP solicita a SP encienda la CUO, pero ésta ya se encontraba encendida  4&gt;&lt;P&gt;CP solicita a SP encienda el DT, pero éste ya se encontraba encendido  4&gt;&lt;P&gt;No se ha definido correctamente el cmd 25 (dec) para CCD. (Cmd no ejecutado)</p> <p>5&gt;&lt;P&gt;&lt;Rx&gt;2F,1B,70,55,41,41,41,41,41,41,41,D7  5&gt;&lt;P&gt;(Cmd 1B.1) CP avisa CU's inicio de autodiagnóstico  5&gt;&lt;P&gt;Se realizara el diagnostico SIMPLE</p> <p>6&gt;&lt;P&gt;0  6&gt;&lt;P&gt;CP ha probado el puerto Serie</p> <p>7&gt;&lt;P&gt;&lt;Rx&gt;2F,1,10,41,41,41,41,41,41,41,41,65  7&gt;&lt;P&gt;K (Se recibió la 'K' enviada)  7&gt;&lt;P&gt;(Cmd 1.1) CP solicita diagnóstico a CUO  7&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo:  F,2,1,0,0,41,41,41,41,41,41,41,F5  7&gt;&lt;P&gt;K</p> <p>8&gt;&lt;P&gt;&lt;Rx&gt;2F,1,20,41,41,41,41,41,41,41,41,55  8&gt;&lt;P&gt;K (Se recibió la 'K' enviada)  8&gt;&lt;P&gt;(Cmd 1.2) CP solicita diagnóstico a DT  8&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo:  F,2,2,0,0,41,41,41,41,41,41,41,F4  8&gt;&lt;P&gt;K</p> <p>9&gt;&lt;P&gt;&lt;Rx&gt;2F,1,40,41,41,41,41,41,41,41,41,35  9&gt;&lt;P&gt;K (Se recibió la 'K' enviada)  9&gt;&lt;P&gt;(Cmd 1.4) CP solicita diagnóstico a SP  9&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo:  F,2,4,0,0,41,41,41,41,41,41,41,F2  9&gt;&lt;P&gt;K</p>	<p>1&gt;&lt;P&gt;K  49  1E  1&gt;&lt;P&gt;&lt;Tx&gt;F,49,2,4E,41,41,41,41,41,41,41,41,1E (1a vez)</p> <p>2&gt;&lt;P&gt;K</p> <p>3&gt;&lt;P&gt;K  27  10  3&gt;&lt;P&gt;(Cmd 27.1) SP le indica a CP cuáles computadoras se pueden encender  3&gt;&lt;P&gt;&lt;Tx&gt;F,27,4,AA,AA,AA,41,41,41,41,41,41,10 (1a vez)</p> <p>4&gt;&lt;P&gt;K</p> <p>5&gt;&lt;P&gt;CU's han realizando su autodiagnóstico ...</p> <p>7&gt;&lt;P&gt;K  2  F5</p> <p>7&gt;&lt;P&gt;(Cmd 2.6) CUO envía su diagnóstico a CP  7&gt;&lt;P&gt;&lt;Tx&gt;F,2,1,0,0,41,41,41,41,41,41,41,F5 (1a vez)</p> <p>8&gt;&lt;P&gt;K  2  F4  8&gt;&lt;P&gt;(Cmd 2.7) DT envía su diagnóstico a CP  8&gt;&lt;P&gt;&lt;Tx&gt;F,2,2,0,0,41,41,41,41,41,41,41,F4 (1a vez)</p> <p>9&gt;&lt;P&gt;K  2  F2  9&gt;&lt;P&gt;(Cmd 2.9) SP envía su diagnóstico a CP  9&gt;&lt;P&gt;&lt;Tx&gt;F,2,4,0,0,41,41,41,41,41,41,41,F2 (1a vez)</p>
---	---



**CAPÍTULO 6. VALIDACIÓN DEL SOFTWARE**

<p>10&gt;&lt;P&gt;&lt;Rx&gt;2F,14,10,41,41,41,41,41,41,41,41,41,52  10&gt;&lt;P&gt;K (Se recibió la 'K' enviada)  10&gt;&lt;P&gt;(Cmd 14.1) CP solicita a CUO envío de comandos en su stack  10&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo:  F,15,1,78,78,78,78,78,78,78,78,78,78,3A  10&gt;&lt;P&gt;K</p> <p>11&gt;&lt;P&gt;&lt;Rx&gt;2F,14,20,41,41,41,41,41,41,41,41,41,42  11&gt;&lt;P&gt;K (Se recibió la 'K' enviada)  11&gt;&lt;P&gt;(Cmd 14.2) CP solicita a DT envío de comandos en su stack  11&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo:  F,15,2,78,78,78,78,78,78,78,78,78,78,39  11&gt;&lt;P&gt;K</p> <p>12&gt;&lt;P&gt;&lt;Rx&gt;2F,14,40,41,41,41,41,41,41,41,41,41,22  12&gt;&lt;P&gt;K (Se recibió la 'K' enviada)  12&gt;&lt;P&gt;(Cmd 14.3) CP solicita a SP envío de comandos en su stack  12&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo:  F,15,4,78,78,78,78,78,78,78,78,78,78,37  12&gt;&lt;P&gt;K</p> <p>13&gt;&lt;P&gt;&lt;Rx&gt;2F,25,40,55,55,55,41,41,41,41,41,41,41,D5  13&gt;&lt;P&gt;K (Se recibió la 'K' enviada)  13&gt;&lt;P&gt;(Cmd 25.1) CP solicita a SP el encendido o apagado de procesadores para diagnóstico simple o voteo  13&gt;&lt;P&gt;CP solicita a SP apague la CUO  13&gt;&lt;P&gt;CP solicita a SP apague el DT  13&gt;&lt;P&gt;CP solicita a SP apague la CCD, pero ésta ya se encontraba apagada</p> <p>14&gt;&lt;P&gt;&lt;Rx&gt;2F,28,40,11,41,41,41,41,41,41,41,41,3E  14&gt;&lt;P&gt;K (Se recibió la 'K' enviada)  14&gt;&lt;P&gt;(Cmd 28.1) CP solicita a SP le envíe los valores de sus primeros 8 sensores  14&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo:  F,29,4,11,80,80,0,80,80,0,80,80,41,81  14&gt;&lt;P&gt;K</p> <p>15&gt;&lt;P&gt;&lt;Rx&gt;2F,28,40,22,41,41,41,41,41,41,41,41,2D  15&gt;&lt;P&gt;K (Se recibió la 'K' enviada)  15&gt;&lt;P&gt;(Cmd 28.2) CP solicita a SP le envíe los valores de sus últimos 9 sensores  15&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo:  F,29,4,22,80,80,80,80,80,1,80,80,80,B0  15&gt;&lt;P&gt;K</p> <p>16&gt;&lt;P&gt;&lt;Rx&gt;2F,2D,50,0,1,0,1,1,0,0,1,1,4E,30  16&gt;&lt;P&gt;(Cmd 45.1) CP envía a Sefdevo el estado de TLMN</p>	<p>10&gt;&lt;P&gt;K  15  3A  10&gt;&lt;P&gt;(Cmd 15.1) CUO avisa a CP que los comandos en su stack se han terminado o que no hay comandos  10&gt;&lt;P&gt;&lt;Tx&gt;F,15,1,78,78,78,78,78,78,78,78,78,78,3A (1a vez)</p> <p>11&gt;&lt;P&gt;K  15  39  11&gt;&lt;P&gt;(Cmd 15.2) DT avisa a CP que los comandos en su stack se han terminado o que no hay comandos  11&gt;&lt;P&gt;&lt;Tx&gt;F,15,2,78,78,78,78,78,78,78,78,78,78,39 (1a vez)</p> <p>12&gt;&lt;P&gt;K  15  37  12&gt;&lt;P&gt;(Cmd 15.3) SP avisa a CP que los comandos en su stack se han terminado o que no hay comandos  12&gt;&lt;P&gt;&lt;Tx&gt;F,15,4,78,78,78,78,78,78,78,78,78,78,37 (1a vez)</p> <p>13&gt;&lt;P&gt;K  13&gt;&lt;P&gt;SP ha apagado la CUO  13&gt;&lt;P&gt;SP ha apagado el DT</p> <p>14&gt;&lt;P&gt;K  29  81  14&gt;&lt;P&gt;(Cmd 29.1) SP envía a CP los valores de sus primeros 8 sensores  14&gt;&lt;P&gt;&lt;Tx&gt;F,29,4,11,80,80,0,80,80,0,80,80,41,81 (1a vez)</p> <p>15&gt;&lt;P&gt;K  29  B0  15&gt;&lt;P&gt;(Cmd 29.2) SP envía a CP los valores de sus últimos 9 sensores  15&gt;&lt;P&gt;&lt;Tx&gt;F,29,4,22,80,80,80,80,80,1,80,80,80,B0 (1a vez)</p>
--	---

Figura 6.31. Monitoreo donde se aprecia un muestreo del satélite donde se realizó el experimento MACV.

En estas pantallas de monitoreo de SOFDEVO se puede apreciar la rutina que permite validar el Experimento de Mantenimiento Automatizado de la Computadora de Vuelo, además de la Arquitectura de Cómputo Semivirtual Tolerante a Fallas propuesta por el IINGEN. Las pantallas muestran todo lo concerniente a los comandos de reconfiguración de la CV y la tolerancia de su falla.

De las pantallas de monitoreo mostradas se puede destacar lo siguiente:

- ✓ El procesador activo es el principal.
- ✓ Se realizó la prueba de VOTEO.
- ✓ Las computadoras enviaron sus diagnósticos en “Broadcast”.
- ✓ Las computadoras enviaron sus diagnósticos acumulados.
- ✓ Las computadoras realizaron el VOTEO.
- ✓ Se generó la matriz de VOTEO.
- ✓ SIM-SOS realizó el VOTEO
- ✓ Las computadoras enviaron sus señales de permiso para la reconfiguración
- ✓ SIM-SOS realiza la reconfiguración.
  - Se apagan todos los procesadores.
  - Conmuta al procesador redundante.
  - Activa la señal de reset.
  - Desactiva el modo BTL de la CV.
  - Desactiva la señal de reset.

Entonces, SIM-SOS realizó el mantenimiento de la CV, gracias a que las demás computadoras vieron al SIM-SOS sin falla y le otorgaron sus permisos para la reconfiguración. Después de ello:

- ✓ CV solicita a SIM-SOS le indique si se está ejecutando el comando CDI.
- ✓ CV envía el estatus de su procesador activo.
- ✓ El procesador activo es el redundante.
- ✓ Se realizará diagnóstico SIMPLE.

Con esto, el experimento de Mantenimiento Automatizado de la Computadora de Vuelo queda validado y operando a la perfección. En otras palabras, cuando se le simuló una falla al SIM-SOS, se realizó automáticamente la reconfiguración de la CV.

```

13><P****RESULTADO DE VOTEO EN CUO****
13><P***** CV ***** CUO ***** DT ***** CCD ***** SP *****
13><P*****BIEN***** BIEN*****MAL*****No P*****BIEN
13><P****RESULTADO DE VOTEO EN DT****
13><P***** CV ***** CUO ***** DT ***** CCD ***** SP *****
13><P*****BIEN***** BIEN*****MAL*****No P*****BIEN
13><P****RESULTADO DE VOTEO EN SP****
13><P***** CV ***** CUO ***** DT ***** CCD ***** SP *****
13><P*****BIEN***** BIEN*****MAL*****No P*****BIEN
    
```

Figura 6.32. Matriz de voto donde se presenta una falla en el SIM-SOS.

Cuando en la matriz de VOTEO se registra que el SIM-SOS tiene falla, figura 6.32, la reconfiguración de la CV es automática, es decir, siempre que se presente una falla en el SIM-SOS de la CV de SATEX, se reconfigurará la CV al procesador confiable, en este caso a la redundancia.

### 6.8 Validación del Comando de Diagnóstico Iterado (CDI)

Como se mencionó en el capítulo anterior, este comando constituye la única forma de corroborar o cambiar el etiquetado de un procesador con falla. Es decir si un procesador de la CV de SATEX tiene la etiqueta de falla, por medio de este comando se puede saber si la falla del procesador es permanente o no, además, se puede cambiar la bandera que lo cataloga como procesador con falla.

Para la validación de este comando se necesita lo siguiente:

- ✓ Conectar toda la instrumentación de SATEX la CV, EAS y M, SOFDEVO, SET, SIMSAT y la fuente de poder.
- ✓ Que las tarjetas de procesamiento tengan instalados el SOFMIN.
- ✓ Subir el software General de Operación Satelital.
- ✓ Que SATEX esté realizando sus rutinas de adquisición de telemetría.

Ya que se tienen los pasos anteriores, la ET contacta a SATEX y le envía el comando CDI. Como se mencionó en el capítulo anterior, el CDI se indica qué procesador se debe diagnosticar, en este caso se hará el CDI al procesador redundante, figura 6.33.



Figura 6.33. Ventana de comandos en línea, se envía el comando en línea CDI.

De este modo, SIM-SOS (SOFDEVO) automáticamente, al recibir la indicación de CDI, apaga todos los procesadores y cambia toda la instrumentación de SATEX al procesador por diagnosticar, figura 6.34. Guarda en sus variables el procesador que era vigente y levanta una bandera para indicar que está ejecutando el CDI.

Solicitud de Comando	Respuesta de Comando
1><P><Rx>2F,13,50,17,1,41,41,41,41,41,41,41,7D 1><P>(Cmd 13.1) CP avisa a sefdevo que ha ejecutado el comando 23[Línea] 1><P>ET ha contactado a Satex	

**CAPÍTULO 6. VALIDACIÓN DEL SOFTWARE**

2><P><Rx>2F,47,20,49,52,41,41,41,41,41,41,41,F6 2><P>K (Se recibió la 'K' enviada) 2><P>(Cmd 47.1) CV envía a DT Inicio o Fin de CDI  Se apagan todos los procesadores  Se hará CDI a Procesador Redundante	2><P>K
---	--------

Figura 6.34. Monitoreo donde se recibe el comando en línea CDI y los mensajes de SOFDEVO.

Entonces SATEX inicia sus rutinas con el procesador por diagnosticar e inicia preguntando por la bandera de CDI, figura 6.35.

Solicitud de Comando	Respuesta de Comando
3><P><Rx>2F,10,50,41,41,41,41,41,41,41,41,16 3><P>(Cmd 10.1) CP avisa a Sefdevo que el satélite se ha liberado  4><P><Rx>2F,2D,50,0,0,0,0,0,0,0,0,83 4><P>(Cmd 45.1) CP envía a Sefdevo el estado de TLMN  <hr/> 1><P><Rx>2F,48,20,41,41,41,41,41,41,41,41,E 1><P>K (Se recibió la 'K' enviada) 1><P>(Cmd 48) CV solicita a SP le indique si está haciendo CDI 1><P><Rx>Cmd enviado por Softdevo: F,49,2,53,41,41,41,41,41,41,41,19 1><P>K	1><P>K 49 19 1><P><Tx>F,49,2,53,41,41,41,41,41,41,41,19 (1a vez)

Figura 6.35. Monitoreo donde la CV pide que se le indique si se realiza CDI.

Como la respuesta de SIM-SOS para la CV es afirmativa, es decir, que está realizando el CDI, la CV automáticamente frena sus interrupciones de adquisición de telemetría y sólo contestará y hará las rutinas indicadas por Estación Terrena.

En este caso, ET contacta a SATEX y le indica que realice la Prueba en Kourou, que es una prueba rápida de tres muestreos de diagnóstico SIMPLE de las computadoras de SATEX, figura 6.36.

Solicitud de Comando	Respuesta de Comando
2><P><Rx>2F,13,50,17,1,41,41,41,41,41,41,41,7D 2><P>(Cmd 13.1) CP avisa a sefdevo que ha ejecutado el comando 23[Línea] 2><P>ET ha contactado a Satex  3><P><Rx>2F,13,50,10,1,41,41,41,41,41,41,41,84 3><P>(Cmd 13.1) CP avisa a sefdevo que ha ejecutado el comando 16[Línea] 3><P>Inicio o fin de prueba en Kourou  4><P><Rx>2F,26,40,41,41,41,41,41,41,41,41,10 4><P>K (Se recibió la 'K' enviada) 4><P>(Cmd 26.1) CP solicita a SP le indique cuáles computadoras se pueden encender 4><P><Rx>Cmd enviado por Softdevo: F,27,4,AA,AA,AA,41,41,41,41,41,41,10 4><P>N	4><P>K 27 10 4><P>(Cmd 27.1) SP le indica a CP cuáles computadoras se pueden encender 4><P><Tx>F,27,4,AA,AA,AA,41,41,41,41,41,41,10 (1a vez) 27

**CAPÍTULO 6. VALIDACIÓN DEL SOFTWARE**

<p>4&gt;&lt;P&gt;&lt;Rx&gt;Cmd enviado por Softdevo: F,27,4,AA,AA,AA,41,41,41,41,41,41,10</p> <p>5&gt;&lt;P&gt;&lt;Rx&gt;2F,25,40,AA,AA,AA,41,41,41,41,41,41,D6</p> <p>5&gt;&lt;P&gt;K (Se recibió la 'K' enviada)</p> <p>5&gt;&lt;P&gt;(Cmd 25.1) CP solicita a SP el encendido o apagado de procesadores para diagnóstico simple o voteo</p> <p>...</p>	<p>10</p> <p>4&gt;&lt;P&gt;(Cmd 27.1) SP le indica a CP cuáles computadoras se pueden encender</p> <p>4&gt;&lt;P&gt;&lt;Tx&gt;F,27,4,AA,AA,AA,41,41,41,41,41,41,10 (2a vez)</p> <p>5&gt;&lt;P&gt;K</p> <p>5&gt;&lt;P&gt;SP ha encendido la CUO</p> <p>5&gt;&lt;P&gt;SP ha encendido el DT</p> <p>5&gt;&lt;P&gt;SP ha encendido la CCD</p> <p>...</p>
---	--

Figura 6.36. Monitoreo donde se recibe el comando Prueba en Kourou y un detalle de su ejecución.

Después de realizar los tres muestreos rápidos, la CV envía su telemetría de Prueba en Kourou a Tierra, figura 6.37.

Solicitud de Comando	Respuesta de Comando
<p>30&gt;&lt;P&gt;&lt;Rx&gt;2F,11,50,41,41,41,41,41,41,41,41,15</p> <p>30&gt;&lt;P&gt;(Cmd 11.1) CP avisa a Sefdevo que envió TLMN de 1 órbita a ET</p> <p>31&gt;&lt;P&gt;&lt;Rx&gt;2F,25,40,55,55,55,41,41,41,41,41,41,D5</p> <p>31&gt;&lt;P&gt;K (Se recibió la 'K' enviada)</p> <p>31&gt;&lt;P&gt;(Cmd 25.1) CP solicita a SP el encendido o apagado de procesadores para diagnóstico simple o voteo</p> <p>31&gt;&lt;P&gt;CP solicita a SP apague la CUO</p> <p>31&gt;&lt;P&gt;CP solicita a SP apague el DT</p> <p>31&gt;&lt;P&gt;CP solicita a SP apague la CCD</p>	<p>31&gt;&lt;P&gt;K</p> <p>31&gt;&lt;P&gt;SP ha apagado la CUO</p> <p>31&gt;&lt;P&gt;SP ha apagado el DT</p> <p>31&gt;&lt;P&gt;SP ha apagado la CCD</p>

Figura 6.37. Monitoreo donde se recibe el envío de telemetría a ET.

Posteriormente la ET recibe la telemetría de Prueba en Kourou, figura 6.38.

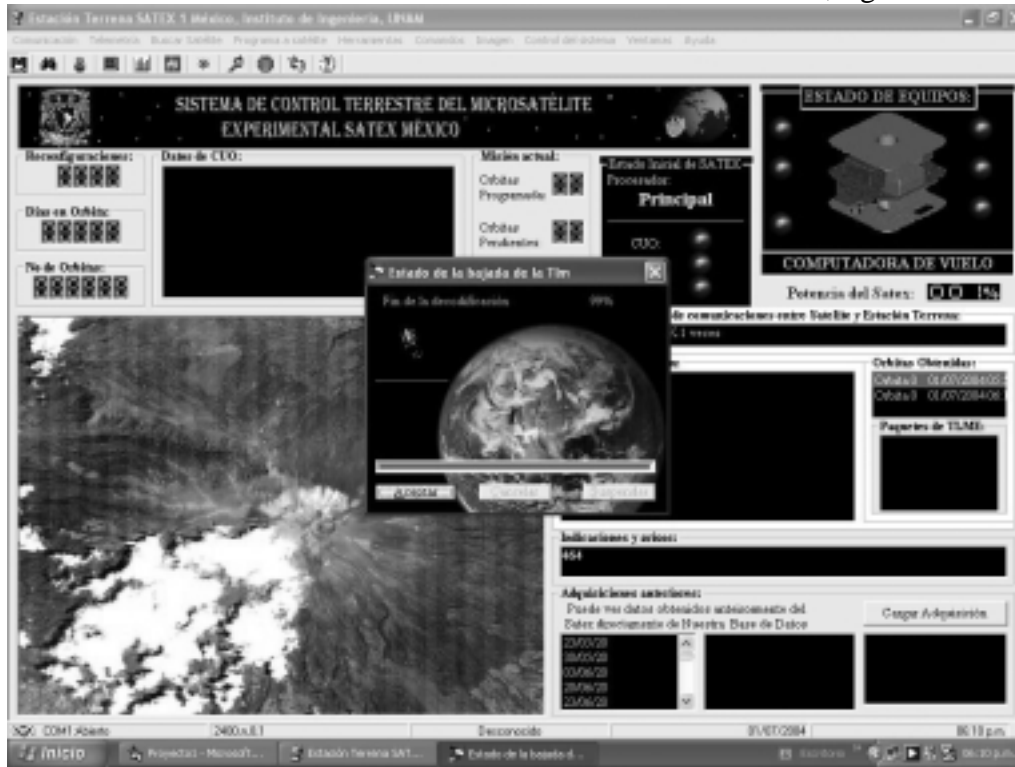


Figura 6.38. Ventana que indica la recepción de la telemetría en prueba en Kourou.

Ya que recibió la telemetría, el personal en Tierra debe analizar los datos obtenidos y después determinar si el procesador diagnosticado tiene o no una falla permanente, figura 6.39.

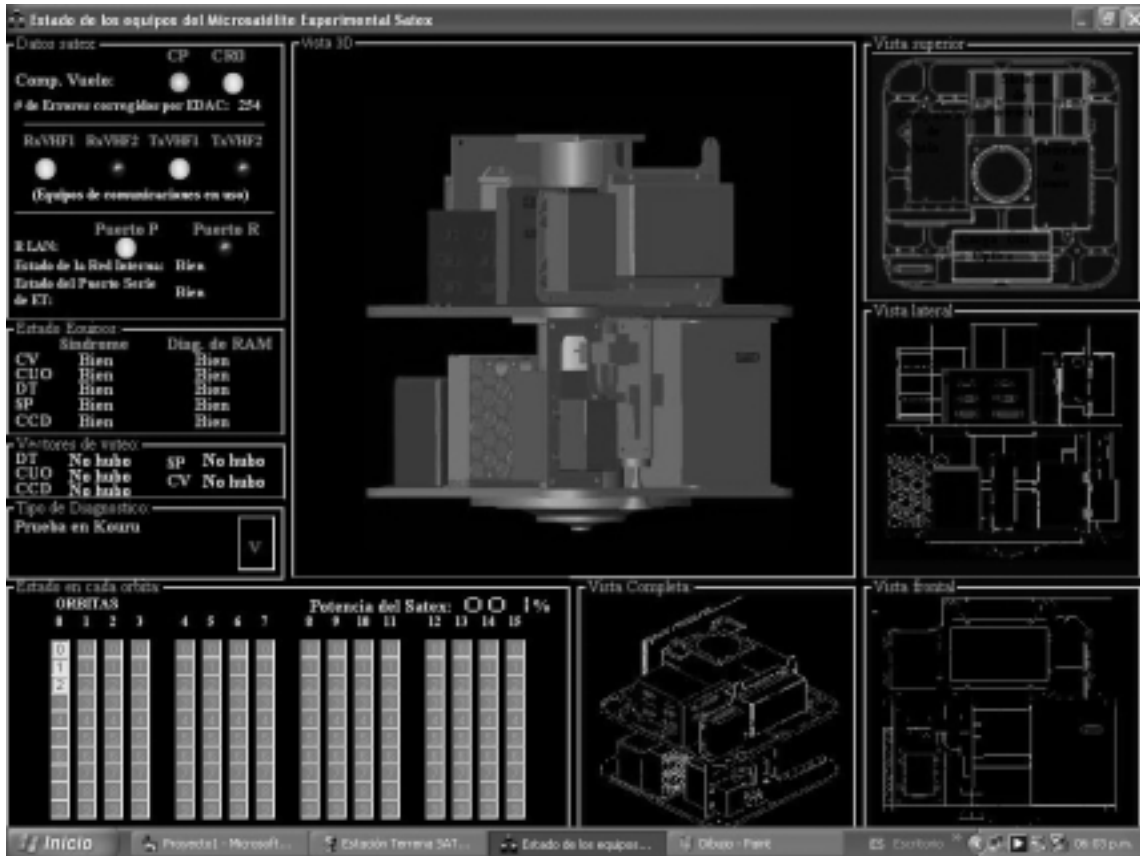


Figura 6.39. En esta ventana, el personal en Tierra puede observar el estado de las computadoras de SATEX.

Ya que se determinó el resultado del CDI aplicado al procesador, ET vuelve a contactar a SATEX y le envía el Fin del CDI, figura 6.40. Con ello se actualizará la bandera del procesador diagnosticado, ya sea que quede etiquetado con falla o como bueno.

Solicitud de Comando	Respuesta de Comando
46><P><Rx>2F,13,50,17,1,41,41,41,41,41,41,7D 46><P>(Cmd 13.1) CP avisa a sefdevo que ha ejecutado el comando 23[Línea] 46><P>ET ha contactado a Satex	
47><P><Rx>2F,47,20,46,52,41,42,41,41,41,41,41,F8 47><P>K (Se recibió la 'K' enviada) 47><P>(Cmd 47.1) CV envía a DT Inicio o Fin de CDI	47><P>K

Figura 6.40. Monitoreo donde se observa el envío del comando en línea CDI.

Una vez terminado el CDI, SIM-SOS vuelve a reconfigurar la CV con el fin de que toda la instrumentación de SATEX regrese al procesador que anteriormente estaba en uso y además, actualiza la bandera de CDI, figura 6.41. Al finalizar la prueba y para que en acciones posteriores no se interfieran las rutinas de adquisición de telemetría de SATEX, la CV vuelve a su inicio de vida con ayuda de SOFMIN.

Solicitud de Comando	Respuesta de Comando
Se apagan todos los procesadores  Se regresa al procesador Principal Se da reset a la CV  Quita Reset de la CV  Desactiva modo BTL de la CV  Termina Reset de CV  1><P><Rx>2F,10,50,41,41,41,41,41,41,41,41,16 1><P>(Cmd 10.1) CP avisa a Sefdevo que el satélite se ha liberado  2><P><Rx>2F,2D,50,0,0,0,0,0,0,0,0,83 2><P>(Cmd 45.1) CP envía a Sefdevo el estado de TLMN  <hr/> 1><P><Rx>2F,48,20,41,41,41,41,41,41,41,41,41,E 1><P>K (Se recibió la 'K' enviada) 1><P>(Cmd 48) CV solicita a SP le indique si está haciendo CDI 1><P><Rx>Cmd enviado por Softdevo: F,49,2,4E,41,41,41,41,41,41,41,41,1E 1><P>K  2><P><Rx>2F,44,20,10,41,41,41,41,41,41,41,41,43 2><P>K (Se recibió la 'K' enviada) 2><P>(Cmd 44.1) CV envía a DT el status de su procesador activo 2><P>Procesador Principal en uso ... ... ...	1><P>K 49 1E 1><P><Tx>F,49,2,4E,41,41,41,41,41,41,41,41,1E (1a vez)  2><P>K  3><P>K 27 10 ... ... ...

Figura 6.41. Monitoreo de mensajes enviados por SOFDEVO y su reconfiguración al procesador vigente.

Con esto se da por concluida la validación del comando CDI.

## Capítulo 7

### *Conclusiones y recomendaciones*

---

#### **7.1 Conclusiones**

Con la terminación de esta tesis se puede concluir lo siguiente:

1. La validación del Mantenimiento Automatizado de la Computadora de Vuelo, cuyo propósito es tolerar las fallas de las tarjetas de procesamiento principal y redundante, así como la tolerancia de la falla en la computadora de la carga útil del Sistema Mínimo de Supervivencia (SIM-SOS). Esta rutina está incorporada en los dos Softwares de Operación Satelital del microsátélite y funciona correctamente.
2. La incorporación y validación del Comando de Diagnóstico Iterado (CDI) que permite verificar si la falla en una tarjeta de procesamiento del microsátélite es permanente, para tal caso, se puede reetiquetar su estado como falla o en su defecto como bueno. Este comando ya está integrado en los dos Softwares de Operación Satelital del microsátélite y operando correctamente.
3. La incorporación del software de control del dispositivo EDAC donde su importancia radican en corroborar que los datos leídos y guardados en memoria RAM sean los correctos, este software de control ya está integrado en los dos Softwares de Operación Satelital del microsátélite.
4. La validación e instalación del software mínimo de operación satelital (SOFMIN) que permite la supervivencia del microsátélite en el espacio, durante su fase de inicio de vida o apagado de la Computadora de Vuelo por razones de protección. Cabe mencionar que será pregrabado de forma previa al ensamble y lanzamiento del microsátélite.
5. Se validó la técnica automatizada del comando en línea “Subir nuevo programa”, con la que Estación Terrena puede subir el Software General de Operación Satelital, el cual puede tener un tamaño menor a 256 Kb. Esta técnica.
6. Se depuró y actualizó el Software de Depuración, Emulación y Validación Operativa (SOFDEVO) que ahora es capaz de emular los nuevos comandos incorporados al Software de Operaciones del microsátélite y también cuenta los eventos relacionados a cualquiera de los comandos que en esta tesis se incorporaron y validaron.



Con todo lo anterior y afirmando que todos los comandos están incorporados, puedo inferir que se tiene por completo la versión de vuelo oficial del Software de Operación Satelital del Microsatélite Experimental SATEX, asimismo una versión de vuelo oficial del software mínimo, el cual tiene un tamaño menor a 64K, tamaño permitido para ser grabado en las EPROM que servirán de contenedor de SOFMIN.

## 7.2 Recomendaciones

Algunas de las recomendaciones que se hacen después de esta tesis son:

### Software de Operación Satelital

Ya que se encuentre la Computadora de Vuelo en la fase de ensamble, se deberán ajustar las interrupciones que controlan las operaciones de los tiempos orbitales determinados por las demás computadoras de las cargas útiles del microsatélite. Por mencionar un ejemplo, la interrupción de cada 10 minutos está escalada para que se realice en un tiempo de 2 minutos, esto para realizar las pruebas en el laboratorio un poco más rápidas y no retrasar el avance del proyecto.

Es necesario habilitar las líneas que están comentarizadas en el código fuente que permiten el uso del **Watchdog**, que es un contador de 16 bits que se alimenta con una señal de reloj, la frecuencia del microcontrolador SAB80C166. Después de cualquier tipo de **reset** al microprocesador, el Watchdog se habilita y empieza a contar desde 0 con una frecuencia de oscilación  $f_{osc}/4$ . Cuando el Watchdog no se deshabilita, continuará contando aún en modo **IDLE** (Ocio), si este no recibe un servicio por parte de la instrucción **SRVWDT**, dentro del tiempo que éste alcance la cuenta de 0xFFFFh, el Watchdog se desbordará y causará un **reset** interno y se activará la bandera **WDTR** y permanecerá así hasta que ocurra otro **reset** por hardware o hasta que se ejecuta la instrucción **SRVWDT**. Es por lo anterior, que se deben habilitar dichas líneas del código fuente. Para que el funcionamiento del microsatélite sea el correcto y no se presente alguna anomalía.

También, como actualmente no se cuenta con las EPROM de vuelo que deben ser rápidas, es necesario que, cuando se tengan los dispositivos, se actualicen los tiempos de acceso del microcontrolador hacia las EPROM. Esto se realiza en el archivo CSTART.ASM, donde se debe programar cuando menos 3 ciclos de espera, cada ciclo es de 50 ns. El archivo \*.DAT para compilar al cstart.asm se llama de igual forma cstart.dat, en el Apéndice aparece el contenido del archivo de compilación.

## SOFDEVO

Como SOFDEVO es un software desarrollado en Visual Basic, es un software que responde a eventos, es por ello que, al recibir un dato por el puerto serie de la PC, realiza la emulación de las computadoras que integran al microsatélite. Entonces es recomendable que cualquier adición de comandos o de eventos sea realizada en torno a los eventos etiquetados en el software como **RecibirCom**, **recibe\_01**, **recibe\_02** y **recibe\_03**. Estas rutinas son las principales para el funcionamiento de SOFDEVO, ya que estas rutinas componen un **SELECT CASE**, donde todos los eventos están programados.

## Bibliografía

---

- [GUTIÉRREZ, 2003] Gutiérrez Medina Luis Ramón  
**Depuración, actualización y certificación operativa del software de recepción de telemetría y control de operaciones del microsatélite SATEX.**  
 Tesis de licenciatura, Facultad de Ingeniería.  
 UNAM, 2003
- [JOHNSON, 1989] Johnson Barry W.  
**Design and Analysis of Fault Tolerant Digital Systems.**  
 Addison-Wesley 1989.
- [MEJÍA, 2002] Mejía Sosa Iris A.  
**Análisis de confiabilidad de la instrumentación de vuelo y sensores para un microsatélite experimental.**  
 Tesis de licenciatura, Facultad de Estudios Superiores Cuautitlán, Enero 2002.
- [MELO, 1996] Melo Serrano Víctor.  
**Teleadquisidor de datos para aplicaciones espaciales.**  
 Tesis de licenciatura, Facultad de Ingeniería UNAM, 1996.
- [ORTIZ, 2003] Ortiz Martínez Hugo Andrés  
**Implantación de técnicas de tolerancia a fallas, ensamble y validación operativa de la computadora de vuelo del microsatélite SATEX.**  
 Tesis de licenciatura, Facultad de Ingeniería.  
 UNAM, 2003
- [PISACANE, 1994] Pisacane L.Vincent and Moore C. Robert..  
**Fundamentals of space systems**  
 Oxford University Press, 1994.
- [SIEMENS, 1997] Users Manual 06.90/08.97  
**Microcomputer Components SAB 80C166/83C166 16-Bit CMOS Single-Chip Microcontrollers for Embedded Control Applications.**  
 Simens 1997.
- [TASKING, 1993] 80166 C Compiler Manual  
**C 80C166 Cross-Compiler.**  
 BSO/Tasking 1993.
- [TORRES, 2002] Torres Fuentes Juan Ramón  
**Software de operaciones, tolerancia a fallas y de telecomunicaciones para un microsatélite experimental.**  
 Tesis de licenciatura, Facultad de Ingeniería.  
 UNAM, 2002.

## BIBLIOGRAFÍA

- [*TEMIC, 1997-2*] Data Sheet 29C516E.  
**16-Bit Flor-Through EDAC  
Error Detection And Correction unit**  
TEMIC Semiconductors, Rev D (Dec-1997).
- [*VICENTE, 1998*] Vicente Vivas Esaú et al.  
**Validación de la instrumentación para el  
microsatélite SATEX.**  
Informe técnico del proyecto 6143, patrocinado por el  
IMC, Instituto de Ingeniería UNAM, Junio 1998.
- [*VICENTE, 2001*] Vicente Vivas Esaú et al.  
**Instrumentación de vuelo espacial para microsatélites.**  
Informe técnico, Instituto de Ingeniería UNAM, Octubre 2001.

## Apéndice A

## Comandos de red interna

### COMANDOS EN DONDE INTERACTÚAN CV- CUO

1.1) CV solicita diagnóstico a CUO

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	01	10	41	41	41	41	41	41	41	41	41	41	Ck

2.2) CUO envía su diagnóstico en broadcasting (VOTEO)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	02	71	Syndrome CUO	REP. RAM DT	41	41	41	41	41	41	41	41	ck

2.6) CUO envía su diagnóstico a CV (Diagnóstico Simple)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	02	01	Syndrome CUO	REP. RAM DT	41	41	41	41	41	41	41	41	ck

4.1) CV solicita a CUO que transfiera 10 datos (primeros 10 datos)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	04	10	01	41	41	41	41	41	41	41	41	41	ck

4.2) CV solicita a CUO que transfiera 10 datos (segundos 10 datos)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	04	10	02	41	41	41	41	41	41	41	41	41	ck

4.3) CV solicita a CUO que transfiera 10 datos (terceros y últimos 10 datos)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	04	10	03	41	41	41	41	41	41	41	41	41	ck

5.1) CUO transfiere primeros 10 datos a CV

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	05	01	00	03	vmod	vmod	ilas	ilas	tlas0	tlas0	tlas1	tlas1	ck

5.2) CUO transfiere segundos 10 datos a CV

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	05	01	01	03	plas	plas	apdc	apdc	apdb	apdb	apda	Apda	ck

5.3) CUO transfiere terceros y últimos 10 datos a CV

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	05	01	02	03	apdd	apdd	ccuo con	ccuo con	treal	treal	treal	treal	ck

9.1) CV avisa a CUO que ET detectó enlace óptico

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	09	10	41	41	41	41	41	41	41	41	41	41	ck

0A.1) CV envía nuevo programa a CUO

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	0A	10	41	41	41	41	41	41	41	41	41	41	ck

**APÉNDICE A. COMANDOS DE RED INTERNA**

0B.1) CUO solicita a CV la retransmisión del programa

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	0B	01	41	41	41	41	41	41	41	41	41	41	ck

0E.1) CV solicita silencio a CUO

1	2	3	4	5	6	7	8	9	10	11	12	13	14
2F	0E	10	Num. Min	41	41	41	41	41	41	41	41	41	ck

Num. de minutos: indica el numero de minutos en que deberán dejar de atender a su puerto serie

14.1) CV solicita a CUO envío de comandos en su stack

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	14	10	41	41	41	41	41	41	41	41	41	41	ck

15.1) CUO avisa a CV que los comandos en su stack se han terminado o que no hay comandos

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	15	01	78	78	78	78	78	78	78	78	78	78	ck

17.1) CV envía parámetros de ET a CUO

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	17	10	S/G	E	TR	TR	TR	TR	T3	T3	T4	T4	Ck

Si es el caso de que S/G = 'G', el comando se utiliza para dar parámetros a CUO y por lo tanto, el resto de los bytes contendrán información que deberá de decodificar CUO como sigue:

En el casillero E, indica la elevación de la visión de ET con SATEX.

Los casilleros marcados con TR, entre los cuatro forman una variable sin signo de formato largo (unsigned long) que define el tiempo base del reloj de tiempo real de CUO.

Los dos bytes nombrados T3, forman otra variable, esta vez sin signo y de formato de entero (unsigned int), con la que se da el valor de inicio para el temporizador T3 de CUO.

Los últimos dos bytes llamados T4, forman otra variable unsigned int, que define en este caso el valor inicial del temporizador T4.

17.2) CV da la orden a CUO para que ejecute nuevo programa

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	17	10	S/G	41	41	41	41	41	41	41	41	41	ck

Si el este byte S/G tiene el valor del carácter 'S', indica que el comando da la orden para que CUO ejecute el nuevo programa, y el resto de los bytes del comando se llenarán con 'A'=41xh, como se muestra en la trama.

1C.1) CV pide a CUO que envíe en *broadcast* sus diagnósticos acumulados (VOTEO)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	1C	10	41	41	41	41	41	41	41	41	41	41	ck

1E.1) CUO envía a CV el resultado del voteo

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	1E	01	v.CV	Sindrome CV2	v.CUO	Sindrome CUO2	v.DT	Sindrome DT2	v.CCD	Sindrome CCD2	v.SP	Sindrome SP2	Ck

**APÉNDICE A. COMANDOS DE RED INTERNA**

V.CV, V.CUO, V.DT, V.CCD, V.SP son el resultado del voto obtenido en la CUO, de las computadoras CV, CUO, DT, CCD y SP respectivamente y está dado por los valores:

- 0xAA – Resultado correcto en el voto.
- 0x55 – Resultado incorrecto en el voto.

Síndrome CV2, Síndrome CUO2, Síndrome DT2, Síndrome CCD2, y Síndrome SP2 son los síndromes de las computadoras Principal, CUO, DT, CCD y SP vistos desde la CUO. Éstos se obtienen realizando el voto a nivel de bits de cada uno de los síndromes utilizados para la obtención del voto.

40.1) CV solicita a CUO resultado del voto (VOTEO)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	40	10	41	41	41	41	41	41	41	41	41	41	ck

**COMANDOS EN DONDE INTERACTÚAN CV- DT**

0.2) CV solicita a DT que se reinicialice

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	00	20	41	41	41	41	41	41	41	41	41	41	ck

1.2) CV solicita diagnóstico a DT

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	01	20	41	41	41	41	41	41	41	41	41	41	ck

2.3) DT envía su diagnóstico en broadcasting (VOTEO)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	02	72	Syndrome DT	REP. RAM DT	41	41	41	41	41	41	41	41	ck

2.7) DT envía su diagnóstico a CV (Diagnóstico Simple)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	02	02	Syndrome DT	REP. RAM DT	41	41	41	41	41	41	41	41	ck

3.1) CV solicita a DT liberar boom

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	03	20	41	41	41	41	41	41	41	41	41	41	ck

6.1) CV avisa a DT que se va a subir un nuevo programa al satélite

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	06	20	CV	CR1	CR2	CV	CR1	CR2	41	41	41	41	ck

- Los bytes 3, 4 y 5 indican la computadora destino del nuevo programa.
- Los bytes 6, 7 y 8 definen la computadora de seguridad en caso de que el envío del nuevo programa sea fallido; entonces el DT sabrá cual computadora activar.
- Los valores que pueden tomar los bytes CV, CR1 y CR2 son:
  - 0xAA : activa a la computadora, ya sea para destino de programa o para seguridad.
  - 0x55 : desactiva a la computadora, ya sea para destino de programa o para seguridad.

8.2) CV solicita a DT conmutar CV (Demostración)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	08	20	41	41	41	41	41	41	41	41	41	41	ck

**APÉNDICE A. COMANDOS DE RED INTERNA**

0D.1) DT solicita a CV conmutar equipo de comunicaciones

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	0D	02	Tx	Rx	41	41	41	41	41	41	41	41	ck

Si Tx : 0F Transmisor T1 activado  
 FF Transmisor T2 activado  
 Rx: 0F Receptor R1 activado  
 FF Receptor R2 activado

0E.2) CV solicita silencio a DT

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	0E	20	Num. Min	41	41	41	41	41	41	41	41	41	ck

Num. de minutos: indica el numero de minutos en que deberán dejar de atender a su puerto serie

14.2) CV solicita a DT envío de comandos en su stack

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	14	20	41	41	41	41	41	41	41	41	41	41	ck

15.2) DT avisa a CV que los comandos en su stack se han terminado o que no hay comandos

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	15	02	78	78	78	78	78	78	78	78	78	78	ck

1C.2) CV pide a DT que envíe en *broadcast* sus diagnósticos acumulados (VOTE0)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	1C	20	41	41	41	41	41	41	41	41	41	41	ck

1D.1) CV envía a CU's sus diagnósticos acumulados (en *broadcast*) (VOTE0)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	1D	70	Syndro me CV1	Syndro me CUO1	Syndro me DT1	Syndro me CCD1	Syndro me SP1	41	41	41	41	41	Ck

1E.2) DT envía a CV el resultado del voto

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	1E	02	V.CV	Sindro me CV3	V.CUO	Sindro me CUO3	V.DT	Sindro me DT3	V.CCD	Sindro me CCD3	V.SP	Sindro me SP3	Ck

V.CV, V.CUO, V.DT, V.CCD, V.SP son el resultado del voto obtenido en el DT, de las computadoras CV, CUO, DT, CCD y SP respectivamente y está dado por los valores:

0xAA – Resultado correcto en el voto.  
 0x55 – Resultado incorrecto en el voto.

Sindrome CV3, Sindrome CUO3, Sindrome DT3, Sindrome CCD3, y Sindrome SP3 son los síndromes de las computadoras Principal, CUO, DT, CCD y SP vistos desde el DT. Éstos se obtienen realizando el voto a nivel de bits de cada uno de los síndromes utilizados para la obtención del voto.

40.2) CV solicita a DT resultado del voto (VOTE0)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	40	20	41	41	41	41	41	41	41	41	41	41	ck

42.1) CV solicita a DT el "status" de sus computadoras

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	42	20	41	41	41	41	41	41	41	41	41	41	ck



**APÉNDICE A. COMANDOS DE RED INTERNA**

43.1) DT envía el “status” de las computadoras de CV a CV

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	43	02	CP	CR1	CR2	41	41	41	41	41	41	41	ck

Donde CP, CR1, CR2 se refieren a las computadoras principal, redundante 1 y redundante 2, respectivamente y pueden tener los siguientes valores:

- 0x00 – No activa, sin falla previa
- 0x0F – Activa, sin falla previa
- 0xFF – Con falla previa

47.1) Comando de Diagnóstico Iterado. Indica a DT el inicio/fin de Diagnóstico Iterado a nueva TP de CV (TP con falla previa). Actualizar bandera de estado operativo de TP.

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	47	20	Inicio o Fin.	TP por Diag.	Edo. TP PP	Edo. TP PR0	Edo. TP PR1	41	41	41	41	41	ck

Donde:

- 'I'= 'Inicia CDI' (0x49)
- 'F'= 'Fin CDI' (0x46)
- 41= 'A' = No altera proceso

- Valores permitidos para TP:
- 0x50 = “P” ‘Procesador Principal
  - 0x52 = “R” ‘Procesador R0”
  - 0x4E = “N” ‘Procesador R1”

- Valores permitidos para Edo.:
- 0x42 = “B” ‘Bueno’
  - 0x4D = “M” ‘Malo’
  - 0x4E = “N” No altera

**COMANDOS EN DONDE INTERACTÚAN CV- CCD**

0.3) CV solicita a CCD que se reinicialice

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	00	30	41	41	41	41	41	41	41	41	41	41	ck

1.3) CV solicita diagnóstico a CCD

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	01	30	41	41	41	41	41	41	41	41	41	41	ck

2.4) CCD envía su diagnóstico en broadcasting (VOTEO)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	02	73	Syndrome CCD	REP. RAM CCD	41	41	41	41	41	41	41	41	ck

2.8) CCD envía su diagnóstico a CV (Diagnóstico Simple)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	02	03	Syndrome CCD	REP. RAM CCD	41	41	41	41	41	41	41	41	ck

**APÉNDICE A. COMANDOS DE RED INTERNA**

0C.1) CV solicita a CCD la adquisición de una imagen

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	0C	30	Num. imagen	MINS [0-9]	SEGS [0-59]	41	41	41	41	41	41	41	Ck

Num. Imagen : Indica la imagen a capturar (0-3).

MINS (0-9)

SEGS(0-9) La CCD deberá adquirir la imagen a partir del momento que decodifique el comando más lo que se indique en MINS+SEGS. Es un offset de tiempo para cumplir exactamente con el tiempo indicado desde estación terrena, el cual puede ser de hasta 16(110mins).

16.1) CV solicita a CCD la transferencia de la imagen

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	16	30	41	41	41	41	41	41	41	41	41	41	Ck

1C.3) CV pide a CCD que envíe en *broadcast* sus diagnósticos acumulados (VOTEO)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	1C	30	41	41	41	41	41	41	41	41	41	41	ck

1E.3) CCD envía a CV el resultado del voto

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	1E	03	V.CV	Síndrome CV4	V.CUO	Síndrome CUO4	V.DT	Síndrome DT4	V.CCD	Síndrome CCD4	V.SP	Síndrome SP4	Ck

V.CV, V.CUO, V.DT, V.CCD, V.SP son el resultado del voto obtenido en la CCD, de las computadoras CV, CUO, DT, CCD y SP respectivamente y está dado por los valores:

0xAA – Resultado correcto en el voto.

0x55 – Resultado incorrecto en el voto.

Síndrome CV4, Síndrome CUO4, Síndrome DT4, Síndrome CCD4, y Síndrome SP4 son los síndromes de las computadoras Principal, CUO, DT, CCD y SP vistos desde la CCD. Éstos se obtienen realizando el voto a nivel de bits de cada uno de los síndromes utilizados para la obtención del voto.

40.3) CV solicita a CCD resultado del voto (VOTEO)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	40	30	41	41	41	41	41	41	41	41	41	41	ck

**COMANDOS EN DONDE INTERACTÚAN CV- SP**

1.4) CV solicita diagnóstico a SP

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	01	40	41	41	41	41	41	41	41	41	41	41	ck

2.5) SP envía su diagnóstico en broadcasting (VOTEO)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	02	74	Síndrome SP	REP. RAM SP	41	41	41	41	41	41	41	41	ck

2.9) SP envía su diagnóstico a CV (Diagnóstico Simple)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	02	04	Síndrome SP	REP. RAM SP	41	41	41	41	41	41	41	41	ck

**APÉNDICE A. COMANDOS DE RED INTERNA**

0E.3) CV solicita silencio a SP

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	0E	40	Num. Min	41	41	41	41	41	41	41	41	41	ck

Num. de minutos: indica el numero de minutos en que deberán dejar de atender a su puerto serie

14.3) CV solicita a SP envío de comandos en su stack

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	14	40	41	41	41	41	41	41	41	41	41	41	Ck

15.3) SP avisa a CV que los comandos en su stack se han terminado o que no hay comandos

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	15	04	78	78	78	78	78	78	78	78	78	78	ck

1C.4) CV pide a SP que envíe en *broadcast* sus diagnósticos acumulados (VOTE0)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	1C	40	41	41	41	41	41	41	41	41	41	41	ck

1E.4) SP envía a CV el resultado del voto

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	1E	04	V.CV	Sindrome CV5	V.CUO	Sindrome CUO5	V.DT	Sindrome DT5	V.CCD	Sindrome CCD5	V.SP	Sindrome SP5	Ck

V.CV, V.CUO, V.DT, V.CCD, V.SP son el resultado del voto obtenido en el SP, de las computadoras CV, CUO, DT, CCD y SP respectivamente y está dado por los valores:  
 0xAA – Resultado correcto en el voto.  
 0x55 – Resultado incorrecto en el voto.

Sindrome CV5, Sindrome CUO5, Sindrome DT5, Sindrome CCD5, y Sindrome SP5 son los síndromes de las computadoras Principal, CUO, DT, CCD y SP vistos desde el SP. Éstos se obtienen realizando el voto a nivel de bits de cada uno de los síndromes utilizados para la obtención del voto.

20.1) CV solicita a SP el encendido o apagado de radios

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	20	40	Tx1	Rx1	Tx2	Rx2	E/A	41	41	41	41	41	Ck

Donde Tx1, Rx1, Tx2, Rx2 pueden tener los siguientes valores:

- 0x55 : Apagado
- 0xAA : Encendido
- E/A; 'E': Encendido; 'A': Apagado (propiamente un mensaje para Sofdevo.

22.1) CV solicita a SP liberación de Boom

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	22	40	41	41	41	41	41	41	41	41	41	41	Ck

23.1) ET (por medio de CV) solicita a SP encendido o apagado de KA

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	23	40	E/A	41	41	41	41	41	41	41	41	41	Ck

Si E/A : 0x55 Indica apagar experimento KA  
 0xAA Indica encender experimento KA

24.1) CV solicita a SP encendido o apagado de CCD

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	24	40	E/A	41	41	41	41	41	41	41	41	41	Ck

**APÉNDICE A. COMANDOS DE RED INTERNA**

Si E/A : 0x55 Indica apagar CCD  
 0xAA Indica encender CCD

25.1) CV solicita a SP el encendido o apagado de procesadores para diagnóstico simple o voto.

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	25	40	CUO	DT	CCD	41	41	41	41	41	41	41	Ck

Donde CUO, DT, CCD pueden tener los siguientes valores:

0x55 : Indica apagar la CU indicada

0xAA : Indica encender la CU indicada

26.1) CV solicita a SP le indique cuáles computadoras se pueden encender para la realización de diagnóstico simple o voto.

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	26	40	41	41	41	41	41	41	41	41	41	41	Ck

27.1) SP le indica a CV cuáles computadoras se pueden encender para la realización de diagnóstico simple o voto.

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	27	04	CUO	DT	CCD	41	41	41	41	41	41	41	Ck

Donde CUO, DT, CCD pueden tener los siguientes valores:

0x55 : Indica que NO hay energía disponible para encender la CU indicada

0xAA : Indica que hay energía disponible para encender la CU indicada

28.1) CV solicita a SP le envíe los valores de sus primeros 8 sensores.

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	28	04	11	41	41	41	41	41	41	41	41	41	ck

28.2) CV solicita a SP le envíe los valores de sus últimos 9 sensores.

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	28	04	22	41	41	41	41	41	41	41	41	41	ck

29.1) SP envía a CV los valores de sus primeros 8 sensores

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	29	40	11	S1	S2	S3	S4	S5	S6	S7	S8	41	ck

Donde

S1 : Sensor de Corriente, Panel1

S5 : Sensor de Corriente, Batería 1

S2 : Sensor de Corriente, Panel2

S6 : Sensor de Corriente, Batería 2

S3 : Sensor de Corriente, Panel3

S7 : Sensor de Temperatura, DCDC

S4 : Sensor de Corriente, Panel4

S8 : Sensor de Temperatura, Batería 1

29.2) SP envía a CV los valores de sus últimos 9 sensores

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	29	40	22	S9	S10	S11	S12	S13	S14	S15	S16	S17	ck

Donde

S9 : Sensor de Temperatura, Batería 2

S14 : Voltaje de Batería 5

S10 : Voltaje de Batería 1

S15 : Voltaje de Batería 6

S11 : Voltaje de Batería 2

S16 : Voltaje de Batería 7

S12 : Voltaje de Batería 3

S17 : Carga de Batería

S13 : Voltaje de Batería 4

## APÉNDICE A. COMANDOS DE RED INTERNA

40.4) CV solicita a SP resultado del voto (VOTEO)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	40	40	41	41	41	41	41	41	41	41	41	41	ck

43.2) CV envía el "status" de sus computadoras a DT

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	43	20	CP	CR1	CR2	41	41	41	41	41	41	41	ck

Donde CP, CR1, CR2 se refieren a las computadoras principal, redundante 1 y redundante 2, respectivamente y pueden tener los siguientes valores:

0x00 – No activa, sin falla previa

0x0F – Activa, sin falla previa

0xFF – Con falla previa

### **COMANDOS EN DONDE INTERACTÚAN (CUO, DT, CCD, SP) – CU's**

1D.2) CUO envía a CU's sus diagnósticos acumulados (en *broadcast*) (VOTEO)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	1D	71	Syndrome CV2	Syndrome CUO2	Syndrome DT2	Syndrome CCD2	Syndrome SP2	41	41	41	41	41	ck

1D.3) DT envía a CU's sus diagnósticos acumulados (en *broadcast*) (VOTEO)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	1D	72	Syndrome CV3	Syndrome CUO3	Syndrome DT3	Syndrome CCD3	Syndrome SP3	41	41	41	41	41	ck

1D.4) CCD envía a CU's sus diagnósticos acumulados (en *broadcast*) (VOTEO)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	1D	73	Syndrome CV4	Syndrome CUO4	Syndrome DT4	Syndrome CCD4	Syndrome SP4	41	41	41	41	41	ck

1D.5) SP envía a CU's sus diagnósticos acumulados (en *broadcast*) (VOTEO)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
0F	1D	74	Syndrome CV5	Syndrome CUO5	Syndrome DT5	Syndrome CCD5	Syndrome SP5	41	41	41	41	41	Ck

### **COMANDOS EN DONDE INTERACTÚAN CV- CU'S**

2.1) CV envía su diagnóstico en broadcasting (VOTEO)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	02	70	Syndrome CV	REP. RAM CV	41	41	41	41	41	41	41	41	ck

1B.1) CV avisa CU's inicio de autodiagnóstico (VOTEO ó Diagnóstico Simple)

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	1B	70	T	41	41	41	41	41	41	41	41	41	ck

Donde : T - 0x55 Diagnóstico simple  
 - 0xAA Diagnóstico por voto

**APÉNDICE A. COMANDOS DE RED INTERNA**

41.1) CV ordena a CU's la generación de señales eléctricas de permisos y de reconfiguración.

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	41	60	T	41	41	41	41	41	41	41	41	41	ck

D: DT realice VOTEO y reconfiguración de CV.

P: Generación de señales de permiso para DT.

**COMANDOS DE INFORMACIÓN A SOFDEVO (CV- SEFDEVO)**

10.1) CV avisa a Sefdevo que el satélite se ha liberado

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	10	50	41	41	41	41	41	41	41	41	41	41	ck

11.1) CV avisa a sefdevo que envió a ET TImN de N orbitas

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	11	50	N	41	41	41	41	41	41	41	41	41	ck

Donde: N – es el número de órbitas enviadas a ET

12.1) CV avisa a Sefdevo que recibió correctamente un nuevo programa

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	12	50	41	41	41	41	41	41	41	41	41	41	ck

13.1) CV avisa a Sefdevo que ha ejecutado el comando X

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	13	50	X	M/L/ RI	41	41	41	41	41	41	41	41	ck

X = número de comando

M/L/RI = 1, indica que es comando en línea

2, comando de misión

3, comando de Red Interna

19.1) CV avisa a Sefdevo que inicia o finaliza el procesamiento en paralelo

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	19	50	I/F	41	41	41	41	41	41	41	41	41	Ck

Si I/F = 0x55 inicia procesamiento en paralelo

I/F = 0xAA finaliza procesamiento en paralelo

1A.1) CV avisa a Sefdevo que inicia o finaliza prueba en Kourou

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	1A	50	I/F	41	41	41	41	41	41	41	41	41	ck

Si I/F = 0x55 inicia prueba

I/F = 0xAA finaliza prueba

2A a 2D) CV informa a Sefdevo sobre variables de la computadora de vuelo.

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	Num Cmd	50	Var0	Var1	Var2	Var3	Var4	Var5	Var6	Var7	Var8	Var9	Ck

**APÉNDICE A. COMANDOS DE RED INTERNA**

Donde

- Var1: Contabilización de Órbita General en el satélite
- Var2: Contabilización de Muestreo General en el satélite
- Var3: Puntero de la órbita de TImN por adquirir
- Var4: Puntero del muestreo de TImN por adquirir
- Var5: Número de órbitas de TImN pendientes por bajar
- Var6: Órbita inicial de adquisición de TImN
- Var7: Número de órbitas bajadas en el último avistamiento
- Var8: Número de órbitas pendientes por adquirir
- Var9: Número de órbitas de adquisición de TImN
- Var10: Bandera que indica si la Estación Terrena está visible o no.

NumCmd

- 0x2A : No hay órbitas por bajar.
- 0x2B : No fue posible bajar ni una orbita por falta de tiempo.
- 0x2C : Se ha enviado TImN a tierra
- 0x2D : CV sólo envía a Sefdevo el estado de TLMN

2E.1) CV informa a Sefdevo la configuración de la captura de las 2 primeras imágenes

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	2E	50	Orblmg01	Muestlmg01	Minlmg01	Seglmg01	Orblmg02	Muestlmg02	Minlmg02	Seglmg02	0xE	41	Ck

2E.1) CV envía a Sefdevo la configuración de la captura de las 2 últimas imágenes

0	1	2	3	4	5	6	7	8	9	10	11	12	13
2F	2E	50	Orblmg03	Muestlmg03	Minlmg03	Seglmg03	Orblmg04	Muestlmg04	Minlmg04	Seglmg04	0x11	41	Ck

El commando 0x2E es enviado a Sefdevo cuando desde ET se le envía a satex la misión de capturar máximo 4 imágenes. Por tanto, el tiempo de captura de cada una de las imágenes está dado por:

- Orblmg0x – Órbita de captura de la imagen 0x
- Muestlmg0x – Muestreo de captura de la imagen 0x
- Minlmg0x – Minuto del muestreo de la captura de la imagen 0x
- Seglmg0x – Segundos del muestreo de la captura de la imagen 0x

Donde x es el número de la imagen a capturar (1 a 4)





## APÉNDICE B. SOFTWARE DE CONTROL DEL EDAC

```
for(pagina=12;pagina<16;pagina++) /*se refrescaran 4 paginas de 16K*/
                                   /*que equivalen a 1 segmento de 64K*/
{
    NumErr=RefrescoMemEDAC(pagina)+NumErr;
}
ActivaRamExp(SegExpAct);
return(NumErr);
}

/**** funcion de refresco de memoria por pagina de 16kbytes, ****
***** es independiente del control de la ram expandida *****/

int RefrescoMemEDAC(int Pagina)
{
    #pragma combine nb=A64592 /*64592=FC50H*/
    int PaginaAsm;
    PaginaAsm=Pagina;

                                   /* respaldo de los DPP's*/

    /*en la siguiente rutina se efectua el refresco de la pagina asociada
    a la memoria expandida que esta mapada en la pagina 3 del micro,
    el tamaño de la pagina es 16384. Por tanto para acceder un segmento de
    64K el software llama a esta rutina 4 veces. El software previo a la
    entrada tambien especifica el hardware para definir que segmento de
    Memoria Expandida se desea refrescar.
    No se refresca la ram interna->sfr's,--Solo stack */

    /*Esta rutina se puede utilizar para refrescar la memoria RAM
    cada 10 minutos, pues no destruye datos, solo que el EDAC
    debera de estar habilitado para corregir errores en caso de
    que se encuentren*/

    /*R1 se utilizara como apuntador y por tanto contituye una direccion
    de 16 bits. El micro al procesar esta direccion extrae de los bits
    14 y 15 el DPP (Data Page Pointer) utilizado.
    En el siguiente software se desea utilizar el DPP2 y por tanto según
    pag. 5-25 del manual del micro, bits 15, 14= 1, 0 para DPP2, por
    tanto la direccion 8000h realiza esto.*/

    #pragma asm
    MOV R1,#8000h ;Fija direccion que habilite DPP2
                  ;1000 0000 0000 0000 = 8000h
                  ;bits 10 elijen DPP2
                  ;y 0000 0000 0000 direccionan 16384
                  ;osea 1 pagina
    MOV R4,#0FC50H ;R4=numero de PaginaAsm (variable de entrada)
    MOV R2,[R4] ;R2=PaginaAsm POR REFRESCAR (NUEVO)
    MOV R3,DPP2 ;respalda el DPP2 VIEJO
    MOV DPP2,R2 ;coloca el num de pag NUEVO en el DPP2
    MOV [R4],ZEROS ;variable de entrada=0
                  ;porque ahora se usara como variable de salida
                  ;para reportar errores
```

## APÉNDICE B. SOFTWARE DE CONTROL DEL EDAC

```

;Lo que haga el EDAC, depende si esta habilitado para corregir
;(Mod. Correc.) o no (Mod. Detec.)

RefIniRam:
MOV R2,[R1] ;Con DPP2 se accesaran 16384 datos
;R2=*R1 LECTURA de RAM expandida
;EDAC genera sindrome nuevo y lo
;compara con el viejo
MOV [R1],R2 ;*R1=R2 ESCRITURA sin alterar bancos RAM
;EDAC calcula nuevo sindrome y lo
CMP R2,[R1] ;?*R1==R2
JMPR CC_NE,ERRAM

RetERRAM:
SRVWDT ;servicio al wdt
CMPI2 R1,#0BFFEh ;FIN DE PAGINA (de 16384)
;FFFF-BFFF=4001H
;65536-49151=16384
JMPR CC_NE,RefIniRam ;EJECUTA LOOP HASTA QUE SE
;ALCANCE
JMPR CC_UC,FINRAMREF ;SALTA AL FIN DE REFRESCO RAM

ERRAM:
MOV R2,[R4] ;R2=(variable de entrada)=0
ADD R2,#01 ;R2++
MOV [R4],R2 ;variable de salida llevara=Numero de Err
JMPR CC_UC,RetERRAM ;Continua con ciclo

FINRAMREF:
MOV DPP2,R3 ;RESTAURA EL NUMERO DE PAG VIEJO
NOP ; FIN DE RUTINA

#pragma endasm

return(PaginaAsm);

}

/*****
/***** Refresca RAM Normal de 256K *****/
/*Refresca los tres primeros segmentos de RAM convencional para que el EDAC
pueda realizar la correccion de errores en la RAM basica del MICRO*/

void Refre3primSegmtsRamNormEDAC()
{
unsigned int IndSegExp;
unsigned int IndOffset;
unsigned int dato;
int huge * hpMemExp;
for(IndSegExp=0;IndSegExp<3;IndSegExp++)
{
for(IndOffset=0;IndOffset<65534;IndOffset)
{
hpMemExp=_mkhp(IndOffset,IndSegExp);
/*direccion de segmentos 0, 1, 2 mas offset*/
if(IndSegExp==0) /*Si esta en el segmento 0*/
{
/*La region de SFR's no se escribe, pues hace locuras el software.*/

```

## APÉNDICE B. SOFTWARE DE CONTROL DEL EDAC

```
if(IndOffset<65024) /*Region SFR's FE00-FFFF=65024-65635*/
    {
        dato=*hpMemExp; /*se refrescaran los tres primeros segmentos*/
        *hpMemExp=dato; /*de RAM convencional*/
    }
}
else
    {
        dato=*hpMemExp; /*se refrescaran los tres primeros segmentos*/
        *hpMemExp=dato; /*de RAM convencional*/
    }
    IndOffset=IndOffset+2;
} /* Accesa Nuevo segmento*/

}

void refrescaRAMnormalYexp()
{
int i;

Refre3primSegmtsRamNormEDAC();

InitHwCtrlRamExp();          /*inicializa ram expandida */

for(i=0;i<16;i++)           /*refresca 16 segmentos RAM expand */
{
    /* _srvwdt();*/
    Refres_1seg_RamExp_ConEDAC(i);
    /*esta rutina refresca un segmento de 64K
    por medio de 4 barridos de 16K para el
    segmento correspondiente en ENSAMBLADOR*/
}

/*SE EJEMPLIFICA EL USO DE MEMORIA EXPANDIDA
EN LENGUAJE C, PARA PROPOSITOS DE USO EN EL
SOFTWARE DE VUELO*/
}
```