

92



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

Tejem

FACULTAD DE INGENIERIA

FALLA DE ORIGEN

RED DE COMUNICACIONES Y PROGRAMACION GLOBAL DE LAS COMPUTADORAS DEL MICROSATELITE SATEX 1.

T E S I S

QUE PARA OBTENER EL TITULO DE INGENIERO MECANICO ELECTRICISTA (AREA ELECTRICA - ELECTRONICA)

P R E S E N T A N:

GALDINO GUTIERREZ PEREZ
MARTIN JUAREZ HERNANDEZ

DIRECTOR DE TESIS: M.I. ESAU VICENTE VIVAS

México, D.F.

1995



TESIS CON FALLA DE ORIGEN



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

A mis padres *Fernando* y *Mercedes* por los sacrificios, la oportunidad y el gran amor brindados para lograr mis metas y objetivos.

A mis hermanos *Arturo*, *Memo*, *Estela*, *Fernando*, *Licha*, *Paty* y *Yoyis* por el apoyo y las alegrías cotidianas.

A mis sobrinos *Fernando*, *Paquito* y *Laura* por el derroche de alegría e ilusiones.

A *Gabriela Coca (Gravy)* por el apoyo y comprensión de todo este tiempo, para cumplir con mis metas y objetivos.

AGRADECIMIENTO:

Quisiera agradecer a todas aquellas personas que consiente o inconscientemente desempeñaron un papel importante durante el desarrollo de mi carrera y la elaboración del presente trabajo, de manera especial :

A *Esau Vicente Vivas* por la valiosa asesoría, la sabia actitud y paciencia brindados en aquellos momentos de incertidumbre, que fueron determinantes para la culminación de este trabajo.

A *Galdino Gutiérrez*, *Martin Santos*, *Victor Manuel Melo*, *Santiago Pérez* e *Iván Aspillaga* por su cooperación y valiosa ayuda para el desarrollo de este trabajo.

Un sincero agradecimiento a *Raúl Chavarria Juárez* por sus consejos y asesorías.

A *Galdino Gutiérrez*, *Juán Dueñas*, *Rafael Burgos* y *Hector Ramón* por haberme permitido formar parte de su grupo de amigos y de trabajo.

A la *UNAM* y *Facultad de Ingeniería* por la formación académica y humana que me dieron a lo largo de mi años de estudio.

A todos mil gracias.

Martín Juárez Hernández.

Agradecimientos

A la Universidad Nacional Autónoma de México y en especial a la Facultad de Ingeniería, por haberme dado las bases de mi formación profesional.

A mis padres por su apoyo moral y económico brindado en mi estancia en la Universidad.

A mis amistades por los momentos de alegría compartidos en nuestra vida de estudiantes.

A los becarios del Departamento de Automatización del Instituto de Ingeniería, por su apoyo y asesoría brindados, en especial a Martín Lopez, Victor Melo, Santiago Pérez e Ivan Aspillaga.

A Martín Juárez por su participación para la realización de este trabajo.

A Esaú Vicente Vivas por la asesoría y paciencia brindados en la realización de este trabajo.

Galdino Gutiérrez Pérez

	pag.
1.- Introducción.	
1.1 Antecedentes	1
1.2 Constitución de un microsatélite	3
1.3 Características globales del Satex1	4
1.4 Principales actividades del Satex1	5
2.- Arquitectura de cómputo distribuido abordo del microsatélite.	
2.1 Introducción	6
2.2 Descripción de la computadora abordo del Satex1	7
2.3 Análisis de la arquitectura de cómputo distribuido en el microsatélite	12
2.4 Acciones de control y monitoreo después de la liberación del satélite	16
2.5 Red de comunicaciones del Satex1	20
3.- Protocolos de comunicación entre las computadoras abordo del microsatélite.	
3.1 Introducción	22
3.2 Protocolos de comunicación Interna para el microsatélite	23
4.- Protocolos de comunicación estación terrena - satélite para la recepción de telemetría, envío de comandos y actualización de programas.	
4.1 Introducción	26
4.2 Protocolos de comunicación con estación terrena	27
5.- Desarrollo de programas para el control de eventos, adquisición de datos y comunicaciones.	
5.1 introducción	38

5.2 Programación desarrollada	39
5.3 Programación para el programa principal del Satex1	40
5.4 Programación de la interrupción para estabilización	41
5.5 Programación de la interrupción para comunicación interna	42
5.6 Programación para interpretación y ejecución de comandos	42
5.7 Programación de la interrupción para adquisición de telemetría	43
5.8 Programación de la interrupción para la comunicación externa	43
6.- Pruebas de validación	
6.1 Introducción	45
6.2 Validación del programa principal	46
6.3 Validación de la rutina para segunda etapa de estabilización	46
6.4 Validación para la rutina de interrupción para comunicación interna	47
6.5 Validación de interrupción para comunicación externa	48
6.6 Validación de la interrupción de la adquisición de telemetría	49
7.- Conclusiones y recomendaciones	
7.1 Introducción	50
7.2 Conclusiones	51
7.3 Recomendaciones	52
Referencias	54
Apéndice A	A1
Apéndice B	B1
Apéndice C	C1

INTRODUCCIÓN

1.1 Antecedentes

Existe una amplia gama de tipos de satélites, los cuales pueden clasificarse atendiendo a varias de sus características. Para los propósitos de este trabajo es suficiente clasificarlos, de acuerdo con su tamaño, en grandes, medianos y pequeños; el presente trabajo está enfocado hacia los pequeños satélites o microsátélites.

Hasta hace algunos años, el campo de los microsátélites no había sido muy amplio, sin embargo, en los últimos años su crecimiento ha sido motivado por un gran número de consideraciones, entre ellas la miniaturización y el alto desempeño del equipo electrónico, el advenimiento de nuevos materiales de alta resistencia y peso reducido y, la enorme capacidad de sistemas de cómputo diminutos.

Otra razón importante para considerar la viabilidad de los microsátélites es la facilidad con que pueden sustituirse, debido a que no representan inversiones tan fuertes como las obligadas en satélites de mayor tamaño; aunado a esto, se encuentra el hecho de que hasta el momento ha sido posible lanzarlos como cargas secundarias a precios relativamente bajos, sin embargo, existen proyectos como el Skipper y el orbex7 [1], cuyo objetivo es desarrollar vehículos especiales para lanzamiento de microsátélites.

En segundo término, está la viabilidad para construir satélites con diseños para periodos de vida útil menores con lo cual se logra una disminución en el proceso de prueba, documentación y fuentes de aseguramiento. Estos factores

contribuyen a reducir los costos en los sistemas de diseño y construcción, así como en la puesta en órbita.

Un factor central en el futuro de los microsátélites es que el avance tecnológico dará aún más la posibilidad de generar sistemas funcionales de alta capacidad de cómputo y altos niveles de confiabilidad en paquetes de tamaño modesto. Los diseños para tiempos de vida útil cortos, permitirán sustituir de manera económica, eficiente y con una mayor frecuencia a los sistemas espaciales, con lo que se logrará una constante actualización en dichos sistemas.

El campo de aplicación de estos satélites es muy amplio y abarca, entre otras, aplicaciones para canales B (canales para voz o datos con ancho de banda de 64 kbps y modulación digital PCM) de Red Digital de Servicios Integrados, redes de telecomunicaciones en banda C, estudios de fenómenos globales de gran escala como son cambios climatológicos, predicción en la actividad sísmica, monitoreo y control ambiental, así como la supervisión aérea, marítima y terrestre en regiones remotas para detección de tráfico de drogas y propósitos militares. Por todo ello, indudablemente, el futuro de los satélites será de los pequeños satélites[1].

El trabajo desarrollado para la presente tesis es parte de un proyecto multi-institucional para diseñar, construir y probar el microsátélite experimental SATEX1, financiado por el Instituto Mexicano de Comunicaciones. Participan instituciones educativas y de investigación de todo el país, entre las cuales se encuentran la Universidad Autónoma de Puebla (UAP), Instituto Politécnico Nacional (IPN), el Centro de Investigación Científica y de Estudios Superiores de Ensenada (CICESE), el Instituto de Astrofísica, Óptica y Electrónica (INAOE), el Centro de Investigación en Matemáticas (CIMAT) y la Universidad Nacional Autónoma de México (UNAM).

Entre algunos de los objetivos del proyecto están la formación de recursos humanos especializados en proyectos espaciales en las diversas instituciones participantes, el aprovechamiento de experiencias nacionales previas en materia aeroespacial, fomentar la participación de jóvenes investigadores y estudiantes, promover convenios de colaboración académica entre instituciones nacionales e internacionales, planear el desarrollo del proyecto para que pueda darse continuidad al trabajo en esta área y poder así utilizar soluciones y equipos desarrollados en futuros proyectos espaciales.

1.2 Constitución de un microsatélite

Un microsatélite está constituido por un grupo de subsistemas con tareas específicas que interactúan estrechamente, dándole así la capacidad de sobrevivencia necesaria para lograr cada una de sus misiones. Entre los subsistemas más importantes están los siguientes:

- **Subsistema de Telemetría y Comando** .- Este subsistema le permite al satélite mantener comunicación con otros satélites y/o con su estación terrena. Con este subsistema el microsatélite es capaz de recibir comandos de su estación terrena (ET), es decir, mandatos o instrucciones para ejecutar acciones. Por medio de él también envía su telemetría, es decir, las mediciones de las variables que adquiere el satélite a través de sus sensores.
- **Subsistema de Potencia**.- Este subsistema es responsable de la captación, y distribución de la corriente y voltajes eléctricos para cada uno de los sistemas que componen al microsatélite.
- **Subsistema de control de altitud y orientación** .- Este subsistema incluye los sensores necesarios para determinar el apuntamiento del vehículo; dependiendo del tipo de estabilización del satélite se pueden tener sensores en uno o más ejes. Asociado a él se encuentra el equipo de control digital que permite automatizar las funciones o tareas del vehículo.
- **Subsistema de propulsión y estabilización** .- Los satélites grandes utilizan sistemas activos compuestos por algún propulsor y válvulas que controlan las salidas de los gases propelente y oxidante; también pueden utilizar ruedas inerciales. La mayor parte de los microsatélites sólo cuentan con sistemas pasivos como magnetotorques, bobinas de campo magnético y gradientes gravitacionales.
- **Subsistema de control térmico**.- Puede ser pasivo o activo, en los primeros se utilizan radiadores y disipadores para reflejar el calor en áreas expuestas a los

rayos solares. En áreas no expuestas al sol se usan materiales aislantes de color oscuro para retener el calor. Un buen control térmico dependerá de aspectos importantes en el diseño de la estructura como son el control de la conducción térmica y la disipación del calor. Estos dos factores juegan un papel importante para mantener la temperatura en los rangos tolerados. Los sistemas activos pueden ser calentadores o "peltiers" para incrementar o reducir temperaturas locales respectivamente.

- Subsistema de comunicaciones.- Está formado por transpondedores, los cuales permiten recibir señales de comunicación (audio, video y datos), amplificarlas, cambiarles la frecuencia y retransmitirlas. Los microsatélites no siempre llevan transpondedores[1],[2],[5].

1.3 Características globales del SATEX1

El microsatélite es un cubo de 47cm. por lado, cuatro de sus caras estarán cubiertas por celdas solares para generar energía eléctrica que se suministrará a sus subsistemas electrónicos. El Satex1 cuenta con dos cargas útiles principales una para comunicaciones ópticas y una cámara de video, aunque debe enfatizarse que por tratarse de una primera experiencia de diseño, integración y validación, cada subsistema construido para el microsatélite representa una carga útil adicional, de las cuales se obtendrán importantes experiencias.

Para controlar el apuntamiento del satélite se cuenta con un sistema pasivo formado por un gradiente gravitacional que una vez desplegado mide 6 metros con una masa terminal de 2.5 kg., además cuenta con tres bobinas de torque magnético con las cuales se generan pares de corrección en la dirección requerida. Para conocer la orientación del satélite se tiene un sensor burdo de sol formado por los paneles solares, también se cuenta con dos sensores finos de sol que permitirán resolver la orientación del vehículo respecto a tierra. Como sensor redundante se tiene un magnetómetro triaxial, el cual establece la realimentación para el control de las bobinas de torque magnético. Se tiene la posibilidad de medir corrientes provenientes de reguladores de voltaje y subsistemas electrónicos importantes como los transmisores VHF, computadoras y electrónica de acondicionamiento de señales; se tienen también sensores de temperatura en sitios estratégicos como baterías, cargas útiles y módulos

electrónicos importantes. Adicionalmente se miden voltajes de operación vitales en las fuentes reguladas de voltaje y batería.

El satélite experimental requiere de un sistema electrónico que permita realizar las operaciones de forma automática para la adquisición de datos locales, ejecución de algoritmos de control y para regular la comunicación entre computadoras abordo, por lo cual debe contar con una arquitectura capaz de manejar las interfaces requeridas por los diversos subsistemas del vehículo y que permita resolver todos los aspectos relacionados con su programación.

1.4 Principales actividades del satex1

La primera acción del satélite será detectar el instante en que es liberado desde su lanzador, después de ocurrir la liberación y de una espera de tiempo definida por el sistema de lanzamiento, el satélite liberará sus antenas. Posteriormente y de acuerdo con un calendario de tareas y restricciones de estabilización, se procederá a liberar el gradiente gravitacional. Una vez realizadas estas primeras tareas, durante la vida útil de vehículo se obtendrá un registro de las variables eléctricas y ambientales en diversos puntos críticos. Se mantendrá comunicación con tierra en puntos estratégicos del territorio nacional, para que Satex1 pueda recibir comandos y transmitir su telemetría e imágenes. También se mantendrá una constante comunicación entre las computadoras abordo con el fin de transferir comandos, diagnóstico, detección de fallas y transferencia de programas para reconfiguración y control.

ARQUITECTURA DE CÓMPUTO DISTRIBUIDO ABORDO DEL MICROSATÉLITE

2.1 Introducción

En este capítulo se hará una descripción general de la arquitectura de cómputo implantada, así como de las interconexiones de los diversos equipos electrónicos dentro del satélite, explicando la interacción entre estos. Para el análisis de las computadoras de abordo se mencionarán las características de la tarjeta de evaluación y el microcontrolador utilizado. En la descripción del microprocesador se detallarán los periféricos utilizados haciendo énfasis en sus características de operación.

Con base en esta arquitectura se describirá como se llevara a cabo las diversas acciones de control sobre los subsistemas y actuadores del satélite, mostrando en diagrama de bloques la electrónica desarrollada para tal fin.

2.2 Descripción de la computadora abordo del Satex1

Para la descripción de la computadora del Satex1 se comenzará explicando el proceso de su elección de acuerdo a los requerimientos del proyecto. Se mencionarán las características generales de su *software* asociado y *hardware* integrado.

2.2.1 Proceso de elección de la computadora de abordo del Satex1

En el proceso de elección de la computadora del Satex1, se tomaron en consideración varios factores, determinados a partir de los requerimientos del proyecto. En el Satex1 se requiere un sistema electrónico que permita realizar operaciones de forma automática entre ellas la adquisición de datos locales, la ejecución de algoritmos de control, así como para la regulación, supervisión y mando del tráfico de las comunicaciones con la estación terrena.

Los objetivos originales de este proyecto contemplaron dos posibles soluciones respecto a la computadora de abordo, la primera consistía en adquirir una microcomputadora de calificación espacial, de preferencia alguna que hubiese volado ya en algún microsatélite experimental; la segunda consistía en encontrar una tarjeta de tipo industrial a la cual se le realizarían modificaciones para poder utilizarla en ambiente espacial de órbita baja.

Respecto a la primera opción, se contactaron a grupos experimentados en el área de microsatélites y, además, se solicitaron cotizaciones a diversas organizaciones de diferentes países para tener un marco de referencia sobre los costos de este equipo. En lo referente a las cotizaciones de equipos calificados para el espacio, todas ellas sobrepasaron por mucho nuestros recursos financieros, el precio de las computadoras oscila desde los 150 mil hasta los 250 mil USD, a los cuales habría que agregarle los costos de interfaces adicionales, y la programación particular que demanda el proyecto Satex1. A esto se debe agregar el hecho de que para poder comprar este tipo de equipo se debe solicitar primeramente un permiso al departamento de estado de los EUA, para saber si el proyecto es calificado positivamente (aprobado). Posteriormente se puede iniciar la compra y por lo general después de haber liquidado un 50% del valor del producto, el tiempo de entrega oscila entre 6 y 15 meses. Como se puede

desprender, la primera opción, aunque atractiva por la robustez del equipo, resultó ser la más alejada de las posibilidades para su adquisición.

Respecto a la segunda opción, se trato de buscar la tarjeta más adecuada, basada en microcontrolador y al mismo tiempo se inició una búsqueda de microcontroladores calificados para vuelo espacial, militar e industrial de rango de temperatura extendida. Se encontraron diversas tarjetas pero la decisión final tomo en cuenta las siguientes consideraciones.

- a) Que el microcontrolador estuviese soldado al impreso y de preferencia de montaje superficial.
- b) Que existiera un sistema de desarrollo asociado para programar la computadora con un lenguaje de alto nivel.
- c) Que tuviera el apoyo de librerías para ejecutar operaciones de punto flotante, las cuales son necesarias para la ejecución de algoritmos de estabilización.
- d) Que el microcontrolador existiera en versión CMOS y de preferencia de grado militar para soportar ciertos niveles de radiación, figura 2.1.
- e) Que la tarjeta integrara recursos importantes para el proyecto, esto es, capacidad de memoria, puertos seriales, temporizadores, líneas entrada/salida, convertidor A/D, etcétera.

No obstante que estas últimas restricciones parecen triviales, se encontró que la mayor parte de los equipos comerciales no cubrían los requisitos, lo cual nos obligo a elegir el equipo con características más cercanas a las ya citadas. Esta situación llevó a la elección de la computadora que se presenta en las secciones siguientes.

La altitud a la cual estará operando el satélite es a una altura aproximada de 800 Km. La radiación promedio entre 200 a 1000 Km es de 1 K - 10 Krad por año [3].

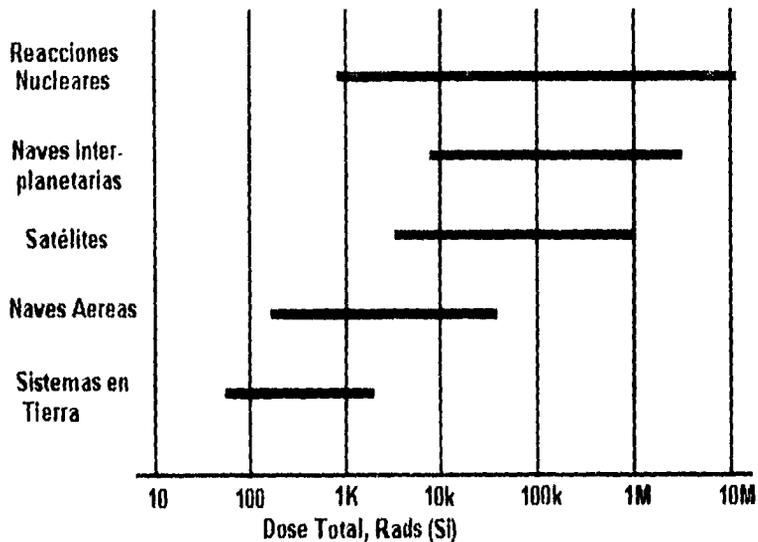


Figura 2.1 Radiación típica en Rads (Si).

2.2.2 Características generales de la computadora elegida

La computadora que se instalará en el Satex1 está basada en el microcontrolador 80C166. Las características generales de la tarjeta son las siguientes:

- La tarjeta puede direccionar 256 Kbytes de memoria RAM y 64 Kbytes de memoria EEPROM. La organización de la memoria se muestra en la figura 2.2.
- Convertidor analógico de 10 canales de 10 bits, con circuitos de muestreo y retención integrados. Este utiliza el método de aproximaciones sucesivas devolviendo el resultado de la conversión en $9.75\mu\text{s}$. Soporta cuatro modos de conversión: simple, simple continua, múltiple y múltiple continua.

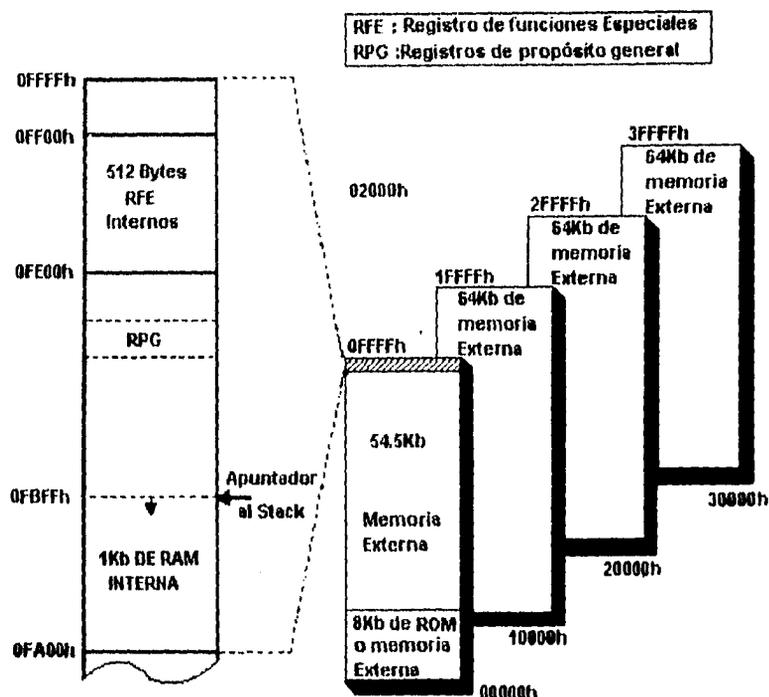


Figura 2.2 Organización de la memoria de la computadora utilizada.

- Dos puertos series con generador de baudaje independiente, soporta comunicación síncrona y asíncrona. Además por cada puerto serie proporciona tres vectores de interrupción separados para transmisión, recepción y recepción errónea.
- 7 temporizadores de 16 bits con 8 niveles de preescalamiento.
- 16 niveles de prioridad programables para interrupciones y 32 fuentes de interrupción.
- Modos de espera y modos programables para bajo consumo de potencia.
- 76 líneas digitales de entrada/salida, organizadas en 4 puertos paralelos de 16 bits (puerto 0 al puerto 3), 2 líneas entrada/salida en el puerto 4, y 10 líneas de

- La computadora soporta también cuatro diferentes configuraciones de ducto: *single chip*, ducto multiplexado de 8 y 16 bits datos, con 16/18 bits de direcciones y ducto no multiplexado con 16 bits de datos y 16/18 bits de direcciones.

La configuración empleada por la tarjeta es el ducto no multiplexado, la cual se elige a través de las señales de control EBC0 y EBC1. Los diagramas de la arquitectura de la tarjeta se muestran en el apéndice A.

Otras características generales:

- Voltaje de alimentación variable de 5V a 6.5V máximo.
- Dimensiones reducidas 10x15 cm.
- Bajo peso.
- Tecnología CMOS.
- Equipo de desarrollo para computadoras personales.
- Programable en lenguaje C de alto nivel.

Interfaces adicionales requeridas:

- Multiplexaje para convertidores A/D.
- Acondicionadores de señales (amplificación y filtrado).

Para mayor información acerca de las características generales del microcontrolador consultar la referencia [4].

2.3 Análisis de la arquitectura de computo distribuido en el microsatélite

En esta sección se describen las interacciones de los diversos módulos del Satex1 , haciendo referencia al diagrama de bloques mostrado en la figura 2.4.

Como se puede observar, el satélite cuenta con cinco módulos computacionales , la computadora principal (CP) , la computadora redundante

(CR), la computadora de carga útil óptica (CCUO), la computadora de cámara CCD y un decodificador de tonos basado en un microcontrolador de Intel (DT).

La arquitectura inicial propuesta para el satélite contemplaba la interconexión de las computadoras por medio de una red que permitía que cualquier par de nodos se pudiera comunicar, además de que cualquier módulo tenía acceso al equipo de VHF para comunicarse con la Estación terrena. La arquitectura era muy flexible, pero implicaba una mayor complejidad en su programación y en su validación. En vista del corto tiempo disponible para desarrollar el proyecto se optó por utilizar otro tipo de red, la cual permite la comunicación entre computadoras, y además, la comunicación de CP y CR con la estación terrena a través de un protocolo establecido. Como se puede observar de la arquitectura, la CP y la CR tienen las mismas interconexiones con los demás módulos; por tanto la estrategia de trabajo contempla la operación mutuamente exclusiva de CP y CR, es decir, mantener CR apagada mientras CP trabaja correctamente.

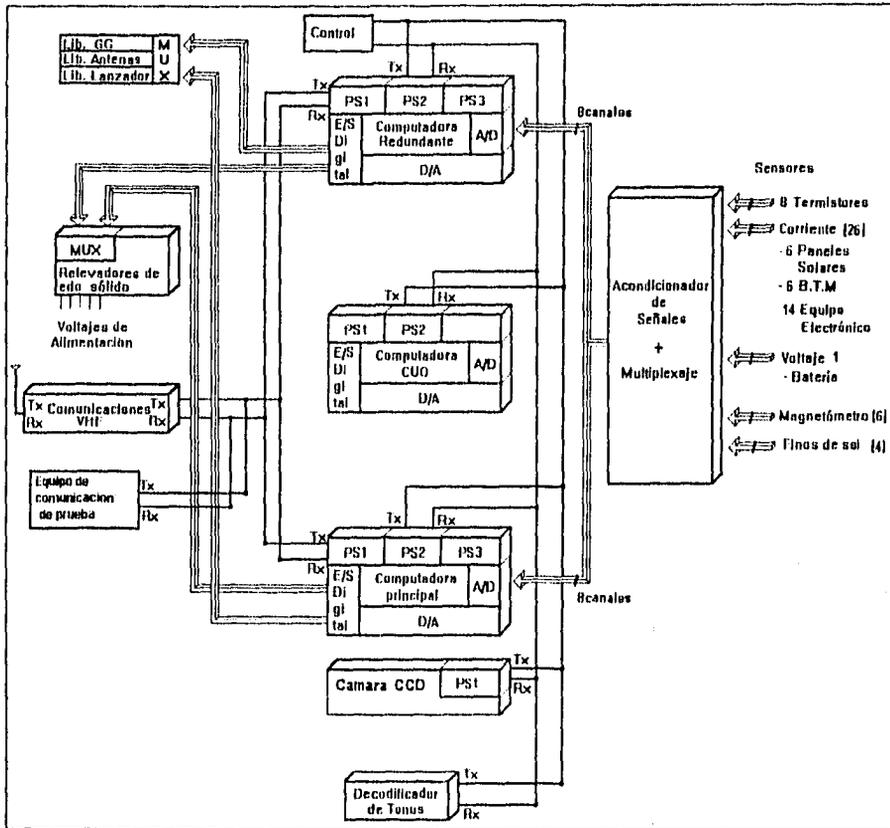
El diseño de la red se basó fundamentalmente en el tipo de operaciones asignadas a la CP, CR, CCUO y al decodificador de tonos. Por esta razón se analizaron las diferentes tareas por realizar en el transcurso de la misión.

2.3.1 Operatividad entre computadora principal y CCUO

La computadora del experimento óptico se activa una vez que el lanzador libera al satélite, una vez activada espera comandos provenientes de la estación terrena para iniciar operaciones. Como se puede observar de la Arquitectura mostrada, la CP es la única que tiene acceso al canal de comunicaciones, por tanto, para la realización del experimento óptico es necesaria la comunicación entre ellas vía puerto serie (a esta red de comunicación se le nombró red interna ver sección 2.5), con la cual se le transmitirán las misiones requeridas por ET.

El experimento óptico consiste en un enlace infrarrojo entre Satex1 y la ET, que requiere un aviso por medio de un comando de reconocimiento por parte de ET, cuando se detecta el haz infrarrojo en tierra; para lograr el apuntamiento del haz infrarrojo se requieren varios parámetros que sirven para direccionar los espejos que reflejan el haz, estos se envían desde la ET y se retransmiten a la CCUO por la red interna a través de CP. Las misiones para el experimento óptico

como ya se mencionó serán programadas por ET, dependiendo de la órbita en la que se trabaje y de las condiciones atmosféricas en tierra. Este experimento se lleva a cabo sólo si la carga en la batería reportada por la telemetría es favorable ya que este experimento consume una gran cantidad de energía.



2.4 Arquitectura de cómputo distribuida a bordo del Satex1.

Otra de las operaciones conjuntas es el diagnóstico mutuo del estado operativo de cada periodo orbital. Durante éste la CCUO pasa información de telemetría a CP para que ésta baje la telemetría de ambas cuando se le requiera. El diagnóstico de CP se utiliza para conmutar a la computadora redundante si ésta llegará a tener alguna falla.

2.3.2 Operatividad entre el decodificador de tonos y CP

Estos dos equipos en condiciones normales efectúan un diagnóstico mutuo, de igual forma el DT tiene la capacidad de conmutar a la CR si el diagnóstico de CP resulta anómalo. Como se observa CCUO y el Decodificador de Tonos tienen la capacidad de conmutar la computadora principal, aunque en el caso del decodificador de tonos se espera un tiempo mayor que el del periodo orbital para conmutar.

Otra de las tareas llevadas a cabo por el Decodificador de Tonos es, en caso de falla del enlace de VHF, la recepción de comandos de ET por medio de un enlace redundante así como la retransmisión de ellos hacia la CP por la red interna ver sección 2.5.

2.3.3 Operatividad entre computadora de cámara CCD y CP

La CP interactúa con este módulo cuando la estación terrena solicita la adquisición de alguna imagen, por tanto parámetros como el instante de adquisición y el formato de la misma se le transmiten por medio de la red interna. Cuando la estación terrena desee bajar alguna imagen, los comandos y sus respectivos argumentos son recibidos por la CP, la cual interacciona con la computadora de la CCD para leer la imagen y posteriormente transmitirla a la estación terrena.

Como se puede apreciar las interacciones de CP con los diversos módulos son independientes entre sí, lo cual facilitó relativamente el desarrollo de los protocolos de comunicación.

2.3.4 Fallas probables en los módulos computacionales

A continuación se indican las fallas probables en los módulos computacionales y su posible solución, así como la forma en que se hace su detección ya sea de forma automática, como las mencionadas anteriormente, o por medio de la ET.

FALLAS PROBABLES	EQUIPO CON FALLA
1) En condiciones normales operan CP y el Decodificador de Tonos de acuerdo a sus diagramas de funcionamiento. La CP envía su telemetría, la de CCUO y la del decodificador de tonos.	Ninguno
2) Cuando falla la CP tierra deja de recibir telemetría y por lo tanto envía comando de conmutación, por medio del Decodificador de tonos, el cual podrá ser ejecutado por el propio DT y/o por CCUO.	CP
3) La falla en el DT se detecta con el diagnóstico realizado por CP. Si falla el DT y en caso de falla de CP, la CCUO activará a la CR por tiempo cuando detecte por diagnóstico que CP no funciona. La perfecta operación de CP es el medio para lograr la estabilización del Satex1 y por lo tanto para permitir el desarrollo del experimento óptico.	DT y probablemente CP
4) Si CCUO falla la CP o la CR lo detectan por medio del diagnóstico y lo reportan a ET.	CCUO
5) Por tiempo, la CCUO detecta falla en la CR (previamente debió haber fallado CP) lo reporta a ET y por lo tanto la comunicación hacia tierra será por medio de comandos a través del DT.	CR
6) ET detecta anomalías y envía comandos al DT para enterarlo de la situación.	CP, CR o CCUO

2.4 Acciones de control y monitoreo después de la liberación del satélite

Aquí se hará una breve descripción de las operaciones a realizar por el satélite, una vez liberado, dando a conocer las acciones por realizar en cada

procedimiento, señalando además su relación con los periféricos de la tarjeta y las interfaces electrónicas.

2.4.1 Inicio de Operaciones del Satex1

Las operaciones se inician al detectar el instante en el que se libera el Satex1 desde su lanzador, esta acción es sensada constantemente mediante la programación, de manera que al darse la apertura del interruptor de liberación, la CP abandona el estado de bajo consumo y procede a continuar con las operaciones restantes

Después de ocurrir la liberación, se espera un tiempo definido por el lanzador para evitar interferencia y posteriormente el Satex1 puede proceder a liberar antenas de VHF e iniciar la comunicación con Tierra. Para liberar las antenas se necesita accionar un circuito electrónico que mediante la aplicación de una corriente eléctrica funde un cinturón de material termosensible. Para este propósito se asigna una salida digital de la CP que ordena su liberación, la cual no se duplica debido a que si llegara a fallar el procedimiento de liberación las antenas se liberarán por efectos de la radiación en el material termosensible, la cual causará su degradación y por tanto el despliegue de antenas.

Una vez iniciadas las comunicaciones con la ET se proceden a ejecutar las misiones programadas, las cuales se basan en el control de eventos para realizar el muestreo de sensores que se describirá en el punto 2.4.3. Para enviar telemetría se requiere del encendido de electrónica de acondicionamiento de señales y el encendido del equipo de transmisión.

Por lo tanto , el inicio de las operaciones del Satex1 requiere de la CP los siguientes atributos:

- 1) Dos entradas digitales para detectar la liberación del Lanzador Ariane.
- 2) Una salida digital para ordenar el despliegue de antenas de VHF.
- 3) 1 salida digital para activar el receptor de VHF.
- 4) 3 salidas digitales para multiplexaje.
- 5) 3 salidas digitales para encendido de electrónica.

6) 2 salidas digitales para encendido del equipo transmisor de VHF(uno de redundancia).

2.4.2 Estabilización

Para lograr los objetivos de apuntamiento del satélite se utilizan dos técnicas de control, una activa y otra pasiva. La técnica pasiva está constituida por un gradiente gravitacional (GG), el cual se libera cuando se cumplen condiciones orbitales y restricciones de estabilización del satélite. Para liberar el GG se necesitan activar dos tornillos explosivos, con lo cual se inicia su despliegue, es decir, se necesitan dos salidas digitales provenientes de la CP para éste propósito. Después del despliegue el GG no requiere de ningún tipo de actuador o sensor para realizar acciones de control, por lo cual este subsistema no demanda entradas o salidas (E/S) adicionales de la CP.

La técnica activa está compuesta por tres bobinas de torque magnético (BTM) las cuales aprovechan el campo magnético terrestre y la inyección de corrientes eléctricas en la bobinas para producir torques de estabilización en tres ejes. Para este subsistema se requieren tres salidas digitales provenientes de la CP que ordenan la aplicación de corrientes deseadas, así como un magnetómetro triaxial para medir la intensidad de campo magnético terrestre y por tanto para calcular la intensidad de torque generado en los algoritmos de la estabilización. Por el lado del magnetómetro triaxial se necesitan tres canales de conversión A/D para que la CP pueda adquirir datos de este sensor; por llevar un magnetómetro triaxial redundante el número de líneas A/D requeridas se duplica.

Adicionalmente, para retroalimentar los algoritmos de estabilización (diseñados por el CIMAT) con datos de posición tridimensional del satélite, se utilizarán dos sensores finos de sol, los cuales requieren 2 canales de conversión A/D por cada uno.

Por tanto para la estabilización se requieren los siguientes atributos de la CP:

- 1) 2 salidas digitales para liberar el gradiente gravitacional.
- 2) 6 salidas digitales para aplicar corrientes a las BTM.

- 3) 6 canales A/D para capturar datos de los magnetómetros triaxiales.
- 4) 4 canales A/D para capturar datos de sensores finos de sol.

2.4.3 Medición de temperaturas, corrientes y voltajes de operación

Durante la vida útil del satélite se obtendrá un registro de valores de temperatura de diversos puntos críticos del vehículo, también se medirán las corrientes eléctricas generadas por los paneles solares y diversos subsistemas electrónicos (reguladores, batería, computadora, módulos de adecuación de señales, cargas útiles, etcétera).

En cuanto a sensores de temperatura se incluyen 8 termistores, los cuales requieren de 8 canales A/D para pasar información a la CP. Respecto a los sensores de corriente se incluyen 6 para paneles solares, 14 para subsistemas electrónicos y 6 para las BTM, los cuales en total demandan 34 canales A/D de la CP.

De sensores de voltaje, solo se requiere 1 para que la CP realice las lecturas correspondientes para la batería.

Por lo tanto, para medir temperaturas, corrientes y voltajes con la CP se requiere:

- 1) 8 canales A/D para medición de temperaturas
- 2) 34 canales A/D para medición de corrientes eléctricas.
- 3) 1 canal A/D para medir voltaje de batería.

Debido a los requerimientos expuestos en los puntos 2.4.1, 2.4.2 y 2.4.3, los canales de conversión A/D se multiplexan requiriendo de tan solo 8 canales A/D.

2.5 Red de comunicaciones del Satex1

Durante las fases de planeación y diseño de la instrumentación del microsatélite quedó de manifiesto la necesidad de utilizar técnicas de tolerancia a fallas para incrementar la confiabilidad operativa del satélite y por lo tanto su vida útil. Se idearon diversas arquitecturas con diferentes tipos de redundancia y lógica de control para detectar, aislar y reconfigurar el equipo en caso de presentarse fallas en algunos de los subsistemas, incluso en aquellos de tipo prioritario. Se pensó añadir redundancias en transmisores, antenas, computadoras, sensores y electrónica vital; lo cual llevó a generar esquemas de instrumentación versátiles y atractivos, aunque algunos de ellos comprometían aspectos de peso, volumen y potencia, que en este caso constituyen restricciones importantes

Después de analizar las diversas opciones, y considerando restricciones (entre ellas el tiempo de desarrollo) se optó por escoger la arquitectura más simple posible, pero que brindara los objetivos de comunicación perseguidos.

2.5.1 Red de comunicación externa

Para las tareas de telemetría y comando se requiere un puerto serie por parte de la CP que será conectado al equipo transmisor de VHF, este equipo será utilizado para la recepción de comandos y programas, o bien para transmitir bloques de telemetría. A la comunicación realizada a través de este puerto serie se le denominó comunicación externa, los protocolos utilizados en ella serán explicados en detalle en el capítulo 4. Debe subrayarse que la CP y la CR estarán conectadas al equipo de comunicaciones por medio de un puerto serie, para prevenir conflictos entre sus líneas se utiliza un arreglo electrónico especial que les permite compartir el mismo canal teniendo solo uno activo a la vez.

El equipo de comunicaciones tiene una derivación hacia un equipo de prueba que actuará cuando el satélite esté en la base de lanzamiento y se requiera realizar un diagnóstico al equipo por medio de comunicación serial. Por medio de éste se permitirá detectar alguna falla y por tanto cambiar algún módulo abordo antes de ser lanzado; una vez verificado el buen funcionamiento del equipo se removerá este módulo para que no cause interferencia con el equipo de VHF.

2.5.2 Red de comunicación interna.

La instrumentación del satélite cuenta con un decodificador de tonos completamente independiente del equipo de VHF, el cual constituye un enlace redundante con tierra, adicionalmente cuenta con una computadora de uso exclusivo para el experimento óptico la CCUO y otra para la cámara CCD. De acuerdo con estas consideraciones se observó la necesidad de conectar tanto el decodificador de tonos como la CCUO con las computadoras CP o CR ya que por medio de éstas se realiza el acceso al canal de comunicaciones. La decisión de utilizar un segundo puerto serie para comunicación interna, permitió reducir la complejidad de los protocolos, con lo cual se reduce el tiempo de desarrollo y la validación del *software*. Por esta razón se diseñó una red de comunicación interna bajo el esquema de que todos pueden hablar y todos pueden escuchar; pero con una programación que evita colisiones. Este enlace de comunicación cubre las necesidades de transferencia de información entre computadoras dentro de la cual estarán los diagnósticos continuos que permitirán detectar fallas como las descritas en el punto 2.3.

Los protocolos utilizados en esta red de comunicaciones se describirán con detalle en el capítulo siguiente.

PROTOCOLOS DE COMUNICACIÓN ENTRE LAS COMPUTADORAS ABORDO DEL MICROSATÉLITE

3.1 Introducción

Cuando un grupo de computadoras se conectan en red, tienen la capacidad de intercambiar información entre ellas, sin embargo, para este intercambio se requiere de un conjunto de reglas y convenciones que sea válido y reconocido por todos los nodos de la red, este conjunto de reglas es comúnmente conocido como *protocolo*, en otras palabras, el protocolo es un conjunto de reglas que establecen el orden, el formato y el significado de las tramas que serán utilizadas por un nodo para solicitar algún servicio o para ordenar a otro nodo la ejecución de alguna operación. Para evitar conflictos en el acceso al canal de comunicación debe existir un servidor, es decir, una computadora responsable de administrar el flujo de información a través del canal, de tal manera que haciendo uso del protocolo indique a los demás integrantes de la red el momento en el cual pueden acceder a dicho canal.

La arquitectura de cómputo del Satex1 contempla una interconexión en red de las computadoras abordo, por lo cual deberá contar con un servidor (CP y/o CR) y con protocolos para la administración del canal de comunicaciones.

En este capítulo se describe el trabajo realizado en protocolos para comunicación entre computadoras abordo.

3.2 Protocolo de comunicación interna para el Microsatélite

Inicialmente se pensó en utilizar o adaptar algún tipo de protocolo ya existente para redes de computadoras, que satisficiera los requisitos de la red implantada en el satélite y que evitara colisiones cuando eventualmente existieran dos o más computadoras intentando tener acceso al canal de comunicación; así pues, se analizaron protocolos como el "token ring", "token bus" y algunos protocolos normalizados como el X21 y X25 [7].

Sin embargo, estos protocolos resultan demasiado sofisticados y de difícil implantación dentro de la red, por lo cual se pensó en diseñar protocolos con base en las características de la red interna del Satex1 y la forma de acceso a la misma.

Como se desprende de la sección anterior, la red interna permite la comunicación entre cualquier nodo existente. De esta forma, para entablar comunicaciones debe existir un protocolo que permita definir sin dudas quién o quienes son los receptores, más algún tipo de argumento o campo de datos para transferir información o comandos.

En el caso del microsatélite el protocolo tiene una palabra de control, en la cual se codifica el origen, el destino y la función que deba ejecutarse. La función puede dirigirse a todas las computadoras o a una en particular; de tal forma que al decodificar los bits de la función queda determinado el número de bytes que acompañan a la palabra de control, con lo cual queda completamente definida la acción de comunicación.

Junto con el protocolo expuesto, se ejecutan periódicamente rutinas en las que la CP o CR solicitan autodiagnóstico al decodificador de tonos y a la CCUO, y viceversa. Con ésto cada controlador recaba información del diagnóstico de los equipos vecinos.

Adicionalmente, cuando se solicita el diagnóstico de algún equipo, el programa contiene rutinas de tiempo que acumulan el tiempo transcurrido, de tal forma que si después de un tiempo determinado (1.4 veces el periodo orbital) CP no se ha comunicado con CCUO, ésta realiza la conmutación hacia CR y avisa al decodificador de tonos para que éste mande un aviso a ET al pasar por México.

Respecto a las solicitudes de diagnóstico, éstas se realizan cada periodo orbital después de pasar por México.

Entre los datos adicionales que se pueden especificar en el protocolo están:

- 1 byte para sincronización con evento, usado para indicar el paso del satélite sobre el territorio nacional; puede ser de cualquier controlador a otro. También permite indicar cuando el vehículo pasa por el ecuador magnético.
- X número de bytes de programa; estos bytes constituyen una nueva versión de programa o algoritmo para cualquier computadora abordo.
- Y número de bytes de comando para especificar algún tipo de control requerido por tierra.
- Palabra de tiempo de adquisición de imagen. Esta palabra es usada por la CP (o CR) para indicar al controlador de la cámara el momento en el cual debe adquirir la imagen.

La palabra de control se integra por los siguientes grupos de bits :

a) *Origen (bits b7 y b6)*: Indican cual de las cuatro computadoras envía el mensaje.

b7	b6	Origen
0	0	computadora principal (C.P./ CR)
0	1	computadora de la cámara
1	0	computadora de carga útil óptica (CCUO)
1	1	decodificador de tonos (8751)

b) *Destino (bits B5 y B4)*: indican cual de las cuatro computadoras deberá recibir el mensaje.

B5	B4	Destino
0	0	computadora principal (C.P./ CR)
0	1	computadora de cámara
1	0	computadora de carga útil óptica (CCUO)
1	1	decodificador de tonos (8751)

c) *Función (bits b3,b2,b1 y b0)*: Indican cual es el propósito de la palabra de control.

b3	b2	b1	b0	FUNCIÓN
0	0	0	0	Solicita al destino ejecutar su rutina de

				diagnóstico
0	0	0	1	Diagnóstico correcto
0	0	1	0	Diagnóstico incorrecto
0	0	1	1	CP o CR piden a la cámara transferir imagen
0	1	0	0	Reset
0	1	0	1	Optativa.
0	1	1	0	Retransferir bloque de alguna imagen
0	1	1	1	Desactivar interrupción por puerto serie para comunicación interna por un tiempo dado
1	0	0	0	Optativa
1	0	0	1	Sincronizar temporizadores con algún evento
1	0	1	0	Avisa a CCUO que E.T. detectó enlace óptico
1	0	1	1	Aviso de que CP o CR está en mal estado
1	1	0	0	Transferir nuevo programa
1	1	0	1	Transferir comando
1	1	1	0	Poner en bajo consumo de potencia
1	1	1	1	Salir de bajo consumo de potencia o ejecutar rutina de interrupción por puerto serie

PALABRA DE CONTROL: ORIGEN(b7-b6) + DESTINO (b5-b4) + FUNCIÓN (b3-b0).

PROTOCOLOS DE COMUNICACIÓN ESTACIÓN TERRENA - SATÉLITE PARA LA RECEPCIÓN DE TELEMETRÍA, ENVÍO DE COMANDOS Y ACTUALIZACIÓN DE PROGRAMAS

4.1 Introducción

Para establecer la comunicación entre las computadoras de abordo y la ET, se desarrolló un protocolo capaz de permitir el reconocimiento de diferentes tramas de control y que permite transmitir misiones a los diferentes módulos, además de contar con protección para un enlace deficiente.

En este capítulo se describirán los protocolos utilizados para establecer la comunicación entre el satélite y tierra. Se incluye un análisis de los requisitos de control y comunicaciones entre tierra y satélite para definir los comandos del sistema. Para el enlace descendente se detalla el diseño de tramas que transportan datos de los diferentes sensores del satélite, así como parámetros de diagnóstico de los diferentes módulos electrónicos instalados abordo. Para el enlace ascendente se describirán los comandos diseñados para ordenar acciones

deseables en el satélite, tales como la sincronización del tiempo, control de actuadores y subsistemas, así como la activación de sistemas redundantes; se describirán también los procedimientos desarrollados para actualizar programas y/o subrutinas de control en las computadoras de abordo.

4.2 Protocolos de comunicación con Estación Terrena

Los protocolos de comunicación externa del Satex1 integran la formulación lógica con la que se transmite y se recibe información desde cualquiera de los nodos (Satex1 o ET). El diseño de los mismos se basa en la planeación de las diferentes actividades que se asignaron al microsatélite experimental (ME).

Por medio de estas estructuras se hace posible la comunicación entre computadoras via los enlaces de subida y de bajada. De bajada se espera la telemetría enviada por el satélite para capturar las variables que definen su estado de operación en un momento dado, en ellas se incluyen los resultados del diagnóstico del equipo y muestras de todos los sensores abordo.

De subida se permite el envío de comandos para coordinar acciones de control o tareas del satélite, además de permitir la transmisión de nuevas versiones de programas a las computadoras de vuelo.

Los comandos están diseñados para poder realizar una programación de las actividades del satélite de forma parcial, de acuerdo a sus características orbitales y de estabilización. El primer gran obstáculo presentado en la programación es la sincronización; como no se cuenta con un mecanismo apropiado abordo para conocer la posición del satélite en tiempo real, se tuvo que crear una técnica que pudiera sincronizar al satélite teniendo en cuenta que el periodo orbital es aproximadamente constante y que las órbitas del satélite no siguen un patrón que permita tener contacto con el satélite en todas las órbitas. La sincronización entonces se logra con el comando 14, que define el número de órbita en la cual se empezará a tomar telemetría y la órbita en que se hará contacto nuevamente con la ET. Este comando es uno de los principales pues sus parámetros permiten adquirir telemetría normal y/o especial definida en los comandos 12 y 13. La telemetría especial surge de la necesidad de muestrear uno o varios sensores con un periodo de muestreo más corto que el de la

telemetría normal, para hacer análisis del comportamiento de alguna variable física a lo largo de la órbita, como pudiera ser el campo magnético o la temperatura. El envío de telemetría y/o retransmisión de telemetría especial (comandos 10 y 17) se efectúa por comando, puesto que el satélite no detecta el paso por México y necesita de esta señal para comenzar a transmitir. El envío de la telemetría se hará de acuerdo al tiempo en que el satélite se encuentre en línea de vista, ese tiempo estará codificado en el comando 14.

Otro de los comandos prioritarios para el satélite es el comando 11, éste controla las etapas de estabilización descritas en el capítulo 2 cuenta con múltiples opciones para manipular los algoritmos de control de estabilización. Como ya se mencionó la estabilización se compone de dos etapas, la etapa inicial y final, de tal forma que con las opciones de este comando se puede iniciar o finalizar cualquier etapa de control. Se implantaron varias opciones pues los algoritmos de estabilización no han sido probados en algún vuelo anterior y es necesario efectuar experimentos en órbita de manera exhaustiva.

Para prevenir al satélite en caso de eclipse se cuenta con un comando que ordena al satélite iniciar el modo de bajo consumo de energía (comando 7), dentro de este no se tomarán muestreos de los sensores para ahorrar energía. Al reanudar operaciones cada vez que se quiera hacer un muestreo se revisará la carga en la batería y si ésta se encuentra en el nivel adecuado de carga se procede a realizar la adquisición.

Una vez integrado el satélite en su totalidad se utilizará el comando para diagnosticar el equipo en la base de lanzamiento, permitiendo este detectar una posible falla para corregirla antes de colocar el satélite en el sistema de lanzamiento. El comando se enviará a través del equipo de comunicaciones de prueba.

Los comandos faltantes realizan diferentes acciones, como encender algún equipo o conmutar a otro en caso de falla detectada por la ET. Se tiene además la opción de transmitir todos los comandos descritos anteriormente por el enlace redundante y algunos de ellos sólo pueden enviarse por este enlace, como es el caso del comando 16.

4.2.1 Definición de comandos para el control de funciones en el ME

Los comandos diseñados integran el conjunto de tareas que el personal de tierra podrá programar en el satélite. Como se verá más adelante, la programación de todo el sistema permite el envío de programas completos a cualquiera de las computadoras de abordo, por lo cual de ser necesaria la inclusión de un nuevo o nuevos comandos en el satélite, será posible reprogramarlo aún después de haberlo colocado en órbita.

Los comandos programados se listan a continuación:

1) Salto a nuevo programa

Este comando transfiere el control al nuevo programa previamente transmitido.

- a) Para computadora principal (CP)
- b) Para computadora de respaldo (CR)
- c) Para computadora de carga útil óptica (CCUO)

2) Orden para liberar el gradiente gravitacional por medio de algún controlador abordo

- a) Computadora redundante
- b) Computadora de respaldo
- c) Decodificador de Tonos (8751)

3) Conmutar equipo de comunicaciones abordo

- a) Activar equipo transmisor VHF1
- b) Activar equipo transmisor VHF2
- c) Utilizar equipo receptor VHF2
- d) Utilizar equipo receptor VHF2

4) Indicación de que la estación terrena ya detectó el experimento óptico

5) Reservado para el experimento óptico

6) Programación de la cámara CCD para adquirir imagen

En este comando se le indica, vía comunicación interna, a la cámara CCD el tiempo en el cual debe adquirir la imagen.

7) Modo mínimo de bajo consumo energético para Satex1

a) Por tiempo indicado

b) Hasta que la estación terrena lo indique

8) Abandona modo de mínimo consumo energético

9) Orden de inicialización para las computadoras

a) Para CP o CR

b) Para CCUO

c) Para Decodificador de tonos

d) Cámara

e) General

10) Orden en línea para que Satex1 transmita trama

a) Telemetría Normal (TLM-N)

b) Telemetría Especial (TLM-E)

c) Ambas (TLM-N y TLM-E)

d) Imagen

11) Control de etapas de estabilización

a) Termina primera etapa

b) Termina primera etapa y comienza segunda etapa en forma simultánea

c) Comienza segunda etapa

d) Termina segunda etapa

e) Comienza primera etapa

12) Telemetría especial

Existe un modo fijo de adquisición para todos los sensores, sin embargo, para hacer más flexible la captura de datos, la programación del Satex1 permite el muestreo a frecuencia variable de algunos sensores.

Para implantar estos modos de adquisición, se dividió la memoria de almacenamiento de datos en dos regiones, una de tamaño constante en la que se guardan los diversos muestreos (de todos los sensores) capturados a lo largo de un periodo orbital, en esta sección se tiene capacidad para almacenar la telemetría de hasta cinco periodos orbitales durante los cuales no se tenga contacto con el satélite (esto se programa con el comando 13). En la segunda, aunque también de tamaño fijo se almacenan muestras a frecuencias variables. En la figura 4.1 se muestra gráficamente el manejo de memoria.

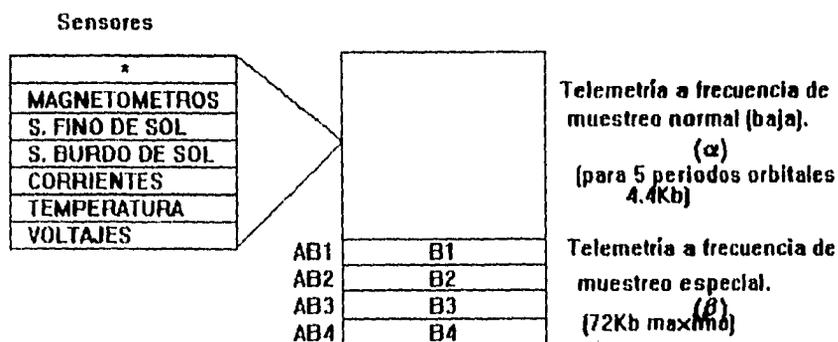


Figura 4.1 Particionamiento de la memoria de datos para muestrear señales a frecuencia fija y variable.

La estructura del comando se diseñó para permitir la definición del tipo de sensores que se desean muestrear a frecuencias especiales. Al variar el tamaño de las regiones B_i del espacio β se pueden adquirir más muestras de algún sensor. Específicamente, se pueden definir dos frecuencias de muestreo, cada

una para un par de sensores escogidos, además de proporcionar el tiempo inicial y final en el cual se hará el muestreo, estos tiempos están referidos al tiempo de inicio de la órbita, figura 4.2.

Estructura del comando

Tiempo inicial	Tiempo final
# de sensores para muestreo especial	
#1 sensor	
#2 sensor	
#3 sensor	
#4 sensor	
Frecuencia de muestreo sensores #1,2	
Frecuencia de muestreo sensores #3,4	

Figura 4.2 Argumentos que definen el comando.

El área de RAM (β) se usará para almacenar datos de telemetría a frecuencia de muestreo rápido. Esta área se particiona dinámicamente de acuerdo al número de sensores por muestrear (máximo 4).

La expresión que define al particionamiento está dada por:

$$\frac{\beta}{\# \text{ sensores para muestreo especial}} = \gamma$$

donde: γ = Espacio de memoria asignado a cada sensor por lo tanto los apuntadores de memoria están dados por:

- AB1 = α
- AB2 = $\alpha + \gamma$
- AB3 = $\alpha + 2\gamma$
- AB4 = $\alpha + 3\gamma$

13) Muestreo especial para algoritmos de estabilización

Este comando tiene la misma estructura que el comando 12 y utiliza la misma área de memoria; los únicos sensores considerados son magnetómetros y sensores finos de sol. Las muestras se almacenan consecutivamente sin particionar la memoria hasta agotar el tiempo indicado por el comando o cuando ya no haya capacidad de almacenamiento.

14) Actualización de tiempo al Satex1

Con la recepción de este comando se inicializa el tiempo en el Satex1 programándose a partir de este la nueva misión a realizar. En este comando se programa la órbita en la cual se tomará telemetría y opcionalmente se puede configurar para que guarde telemetría de hasta cuatro órbitas en las cuales no se tenga contacto con ET. Otro de los datos que proporciona es el tiempo que Satex estará libre y en línea de vista durante el avistamiento actual.

15) Prueba en Kourou, comando para diagnosticar la operatividad de algunos subsistemas de Satex1 en la base de lanzamiento.

16) Habilitar la computadora redundante

a) DT

b) CCUO

NOTA: Este comando se enviará por el decodificador de tonos, la opción b) se realiza por comunicación interna (DT- CCUO) sólo en el caso de que las líneas digitales del DT no puedan realizar el cambio.

17) Retransmisión de trama de telemetría especial

Este comando especifica el sensor y el número de bloque a retransmitir, en el caso de magnetómetros el número de sensor deberá ser codificado como el uno.

4.2.2 Protocolo de comunicaciones Satex1-ET

Debe subrayarse que la mayor parte de los comandos mencionados son unidireccionales, los cuales al ser recibidos por el ME se verifican para invalidar aquellos que presenten errores por transmisión. Algunos de ellos (comandos 10 y 17) generan una respuesta, por lo cual la estructura de los protocolos es flexible para permitir esta interacción.

En seguida se describirá el trabajo realizado en cuanto a protocolos de comunicación externa del satélite.

4.2.2.1 Protocolos de comunicaciones para el enlace ascendente

Las tramas de información ascendente se han dividido en dos clases las tramas de comandos y las tramas de programas, cada una con su propio formato que ayuda a su decodificación en la CP. Respecto a la información en las tramas de comandos ésta ya se describió en los puntos anteriores. En cuanto a las tramas de programas, se codifican los programas por ejecutarse abordo (en alguno de los módulos computacionales) cabe además señalar que también a través de estas tramas se envían las acciones de control para estabilización del Satex1. La información codificada en estas tramas se desglosará en los siguientes puntos.

4.2.2.1.1 Protocolo de comunicaciones para el envío de comandos al satélite

El formato de los comandos está compuesto por una palabra de control, una palabra que especifica el comando, los argumentos o datos asociados y una palabra codificada para detectar errores de transmisión.

Cuando el bit más significativo de la palabra de control es uno, indica la presencia de un comando. Los bits 0 al 4 indican las opciones o alternativas de un comando en particular; en tanto que los bits 5 y 6 indican a cual computadora va dirigido el comando.

La palabra que especifica al comando contiene el código de operación que identifica al comando en cuestión, el cual permite definir hasta 256 comandos diferentes, cada uno con sus alternativas particulares.

Los posibles datos relacionados con el comando que se envía pueden abarcar desde 0 hasta 25 palabras; adicionalmente, después de la palabra de detección de error se agregan 2 palabras de sincronía.

4.2.2.1.2 Protocolo para el envío de programas al Satex1

En el caso de la transmisión de programas vía el enlace ascendente, se utiliza otro protocolo. El formato contiene una palabra de control, una palabra que define el número total de bloques a transmitir (de acuerdo con la longitud del programa por enviar), los datos del iésimo bloque transmitido (bloque corriente) y una palabra para detectar posibles errores de transmisión en el bloque transmitido. El bit más significativo de la palabra total de bloques y los bits 0 al 4 de la palabra de control permiten definir (y por tanto transmitir) hasta 64 bloques de datos cada uno constituido por 500 palabras, para formar un total de 32K palabras. Los bits 5 y 6 definen cuál será la computadora a la que irá dirigida la transmisión del nuevo programa.

El bit más significativo de la palabra de control indica la transmisión de programa cuando tiene el valor de 0.

En caso de que el programa deba transmitirse en dos o más avistamientos del Satex1, la CP esperará el envío total y en cada envío de telemetría informará de los bloques faltantes o los bloques con errores, utilizando para ello el formato que se describirá en el siguiente punto.

El envío de los datos de control de estabilización se realiza en bloques, pero una vez recibidos se colocan en arreglos especiales.

4.2.2.2 Protocolos de comunicaciones para el enlace descendente

Las tramas de información descendente se han dividido en dos clases las tramas de repetición de bloques de programas y las tramas de envío de telemetría e imagen. La información contenida en las tramas de repetición de bloques de programa, informa si los bloques de programas enviados llegan en mal estado y si es así, sirve para que la ET proceda a la retransmisión de éstos. Respecto a las tramas de telemetría e imagen como su nombre lo indica,

contienen información codificada de datos adquiridos por telemetría y la cámara CCD. La información codificada en estas tramas se desglosará en los siguientes puntos.

4.2.2.2.1 Trama de Repetición de Bloques de programa

La trama de retransmisión está formada por una palabra de control, en la cual se indica el tipo de datos por transmitir. Además se indica cual computadora del Satex1 realiza la transmisión. Las palabras de estado de los bloques indican cuales bloques habrán de retransmitirse.

4.2.2.2.2 Trama de envío de telemetría e imagen

Satex1 utiliza el mismo protocolo para transmitir telemetría normal o especial, el formato del protocolo contiene una palabra de control y una palabra que indica el número de bloque corriente, seguido por los datos del bloque. Los bits de la palabra de control definen el tipo de telemetría que la computadora envía. Para el caso de transmisión de telemetría especial, se indica además el número de sensor del cual se envía su telemetría.

Donde:

TLM-N= Telemetría Normal

TLM-E= Telemetría Especial

La palabra de control de bloque indica el número del bloque que se transmite. La telemetría normal se transmite en un solo bloque por lo cual esta palabra será igual a cero. Posteriormente aparecen 7 palabras, las cuales tienen información del diagnóstico de los diferentes subsistemas incluyendo el de la computadoras, estos datos se obtienen una vez por periodo orbital.

Cuando se transmite telemetría normal los datos están compuestos por una serie de paquetes, cada paquete proporciona información de un muestreo (el satex1 efectúa aproximadamente 1 muestreo cada 10 minutos) por lo que 10 paquetes proporcionan información correspondiente a un periodo orbital (el periodo orbital es de 100 min). Debe subrayarse que la palabra de detección de

errores del paquete 1 incluye las palabras de CTRL. , CTRL de bloque y las palabras de estado (diagnóstico).

Respecto a la palabra de control en la trama de TLM-N los cuatro bits menos significativos indican el número de la órbita a la cual pertenece la telemetría, a los demás casos se les asigna un 1 lógico. Los bits 2 y 3 indican el número de sensor cuando se transmite telemetría especial. Los bits 4 y 5 indican cual de las computadoras de abordo envía la telemetría. Los bits 6 y 7 definen el tipo de telemetría enviada , esto es : TLM-N, TLM-E, retransmisión o imagen.

Los datos de telemetría especial son adquiridos de acuerdo a la frecuencia especificada en el comando correspondiente y almacenados en un sector de memoria especial, de donde se extraen para formar la trama de TLM-E; los datos se envían de forma análoga a los bloques de TLM-N, sólo que los datos son de un mismo sensor y no se envían las palabras de estado. El envío de la imagen se lleva a cabo de acuerdo a lo anterior sólo cambiando los bits del tipo.

DESARROLLO DE PROGRAMAS PARA EL CONTROL DE EVENTOS, ADQUISICIÓN DE DATOS Y COMUNICACIONES

5.1 Introducción

En esta sección se presentan los procedimientos, diagramas de flujo y programas desarrollados para las computadoras CP y CR . Para evitar largas explicaciones, se remite al lector al análisis de los diagramas de flujo, en donde se encuentran los procedimientos lógicos para el microsatélite desde el inicio de su vida. La programación realizada se hizo en lenguaje C, ANSI C [8], [12], además de la ayuda de librerías especiales para interactuar con los puertos, periféricos, registros y números flotantes [9], [10], [11].

Una vez que se ha editado el código fuente en lenguaje C, es necesario crear el programa ejecutable, para lo cual se deben seguir los siguientes pasos (para correrlo en memoria RAM) :

- **Compilar.**- Se hace uso del programa **C166** para generar un archivo fuente en lenguaje ensamblador a partir del código en lenguaje C.
- **Ensamblar.**-Se utiliza el programa **A166** para generar un archivo objeto con sufijo .obj.
- **Ligar .-** Mediante el programa **I166** se ligan las librerías especiales con el archivo objeto y se genera un nuevo archivo con sufijo .Ino .
- **Localizar.** - Se emplea nuevamente el archivo **I166** para definir las localidades de memoria en donde será cargada la aplicación.

- Formato- En este paso se emplea el programa **ieee166** para generar un archivo con el formato **XVW16G**, con sufijo **.abs**.
- Cargar.- Finalmente, se emplea el programa **XVW166**, con el cual es posible cargar, ejecutar y depurar al programa, así como, monitorear sus variables.

Para cargar el programa en memoria EPROM es necesario seguir todos los pasos anteriores, con excepción del último, además de seguir los dos pasos siguientes :

- Formatear.-Mediante el programa **ihex** se genera un archivo en hexadecimal con formato intel, con sufijo **.hex**.
- Grabar.-Debido a que el microprocesador trabaja con 16 líneas para datos es necesario grabar el programa en 2 segmentos, el segmento par en una memoria de 8 bits y el segmento impar en otra memoria de 8 bits.

El programa global fue dividido en subrutinas activadas por interrupción o llamadas por programa, con esto se logró desarrollar y validar en forma independiente, la programación correspondiente a cada una de las tareas que deberá realizar el satélite durante su vida útil.

5.2 Programación desarrollada

La instrumentación que se ha descrito para el Satex1 permite desarrollar las operaciones autónomas del satélite, su supervisión y el control desde una estación terrena. El complemento vital de esta instrumentación son los programas instalados en las computadoras, los cuales integran las rutinas de inicio de vida del satélite, los protocolos entre computadoras, comunicación entre CP (o CR) y la estación terrena para transmitir telemetría o recibir comando (o bien programas actualizados), así como los procedimientos para lectura de sensores, control de actuadores, empaquetamiento de datos, captura de imágenes, etcétera. Algunas de las actividades específicas implantadas en la programación son las siguientes:

- Reconocimiento de comandos enviados desde la estación terrena (E.T) para realizar operaciones con alguna de las cargas útiles o con la instrumentación de alguna de las computadoras.
- La adquisición de parámetros de estado de los diferentes sensores instalados en el satélite durante su vida útil.
- El empaquetamiento y el envío de telemetría hacia la(s) estación(es) terrena(s) de supervisión y control.
- La recepción de comandos de control o de nuevos programas de operación para las computadoras de abordo.
- Conocer el estado de operación de celdas solares, baterías, bobinas electromagnéticas, magnetómetro y diversos subsistemas electrónicos.
- Conocer periódicamente los cambios de temperatura de las paredes internas de paneles solares y otros sitios internos específicos del satélite.
- Encender o apagar los equipos de comunicaciones abordo.
- Activar flujo de corriente en las bobinas electromagnéticas para control de estabilización.
- Controlar la cámara CCD de acuerdo con comandos enviados desde tierra para la captura y almacenamiento de imágenes durante cada periodo orbital.

5.3 Programación para el programa principal del satélite

Los siguientes diagramas describen el programa principal, en el cual se indican a groso modo las acciones que desarrollará el Satex1 después de ser liberado por el sistema de lanzamiento. Dentro de los procedimientos se prevé la detección de liberación, diagnóstico de equipo abordo, liberación de antenas y programación de interrupciones.

Al iniciar el programa principal, llega a un ciclo en el cual, después de sensar las líneas de liberación, entra a un estado de bajo consumo de potencia . El programa puede salir de éste estado mediante la activación de una de dos interrupciones previamente programadas. Una de las interrupciones es activada por tiempo, es decir, el programa se interrumpe cada 2 minutos y sale del estado de bajo consumo de potencia para volver a sensar las líneas de liberación, en caso de detectarse la liberación, sale del ciclo y entra a un segundo ciclo en el cual se debe confirmar la liberación; en caso de ser confirmada el programa sale de este segundo ciclo, de lo contrario el programa regresará al primer ciclo. La

otra interrupción se activa mediante la recepción de algún dato por el puerto serie asignado a la comunicación externa, para este caso, el programa realiza los mismos procedimientos que en la interrupción anterior, sin embargo, antes de sensar las líneas de liberación, efectuará un autodiagnóstico comprendiendo diagnóstico a memoria RAM, puertos serie, CPU y un muestreo en los sensores, así mismo, enviará por la red de comunicación interna palabras de control solicitando autodiagnóstico para CCUO y decodificador de tonos. Esta segunda interrupción tiene la finalidad de contar con un medio que permita conocer el estado de las computadoras abordo del satélite antes de ser lanzado, de esta forma, esta rutina de diagnóstico será ejecutada cada vez que el personal en la base de lanzamiento lo solicite mediante el conector serie externo.

Una vez que se ha detectado la liberación, se tiene un retraso (definido por el lanzador) y posteriormente el programa libera las antenas del equipo de VHF, configura el resto de los periféricos del microcontrolador y programa interrupciones. A partir de ese momento comienzan a funcionar los temporizadores y contadores responsables de activar las rutinas de interrupción para muestreo de sensores y estabilización.

Finalmente, el programa entra a un ciclo infinito en el cual después de ejecutar comandos, entrará a un estado de mínimo consumo energético del que saldrá mediante alguna de las interrupciones programadas, ver figura 5.1 del apéndice C.

5.4 Programación de la interrupción para estabilización

Esta interrupción está programada inicialmente para activarse cada 240 milisegundos, sin embargo, es posible cambiar esta frecuencia, mediante un comando, para interrumpir cada 0.42, 0.84 ó 1.68 segundos.

La estabilización del microsátélite se efectúa en dos etapas. La primera etapa funciona en base a un algoritmo que sensa el campo magnético y aplica corriente a las bobinas, de tal forma que, se obtenga un efecto favorable para el control del movimiento rotativo del satélite, debe subrayarse que esta etapa opera solo durante las primeras semanas de operación.

Para la segunda etapa de estabilización, es necesario contar con un grupo de parámetros transmitidos desde la estación terrena al SATEX1 y calculados en tierra en base a la telemetría previamente proporcionada por el satélite. Dichos

parámetros constituyen el nuevo algoritmo de control para estabilización, los cuales serán cambiados periódicamente de acuerdo con nuevos cálculos basados en la telemetría.

El fin de cada una de las etapas y el inicio de la segunda serán indicados al satélite mediante el envío del comando adecuado, ver figura 5.2 del apéndice C.

5.5 Programación de la interrupción para comunicación interna

Esta interrupción se activa cuando llega algún dato al puerto serie asignado a la comunicación interna, sin embargo, se debe tomar en cuenta que dicho dato no será asimilado por el CPU, por lo tanto únicamente servirá para activar la interrupción y después se perderá.

Esta subrutina de interrupción tiene como objetivo responder a los requerimientos de comunicación de algún microprocesador abordo con la CP (o CR), contempla además responder a la solicitud de diagnóstico, recibir datos de algún comando o programa enviado por algún microcontrolador distinto de CP O CR, y/o responder al requerimiento de sincronización de contadores y temporizadores con algún evento.

La CP o CR al ser interrumpida recibe y analiza la palabra de control para saber si realmente el requerimiento de comunicación va dirigido a ella, además de determinar el número de datos que debe recibir. Esta subrutina no debe quedar esperando el arribo de datos por tiempos indefinidos, por lo cual, se da un tiempo máximo de espera para la recepción de uno y otro dato, de tal forma que si el dato esperado tarda demasiado en arribar, el proceso se suspende y el programa sale automáticamente de esta interrupción, consultar figura 5.3 del apéndice C.

5.6 Programación para la interpretación y ejecución de comandos

Estos diagramas describen los pasos seguidos por las computadoras abordo para identificar, verificar, así como para ejecutar o traspasar los diferentes comandos que acepta el microsatélite experimental, ver figura 5.4 del apéndice C.

5.7 Programación de la interrupción para adquisición de telemetría

La programación de eventos tales como la adquisición de variables de telemetría se efectúa a través del comando de actualización de tiempo para el Satex1 (comando 14). Considerando los casos en los que no se pueda tener contacto con el mirosatélite en algunas órbitas en este comando se puede programar la adquisición de telemetría de hasta 4 órbitas del satex1.

La adquisición de telemetría se lleva a cabo mediante interrupción vía temporizador, ésta es programada para que muestree sensores cada 10 minutos, el número de muestreos en 1 órbita es de 10 por tener un periodo de 100 minutos aproximadamente. Dentro del diagrama de flujo queda contemplado un diagnóstico entre computadoras que se realiza 10 minutos antes de terminar el periodo orbital, ver figura 5.5 del apéndice C.

5.8 Programación de la interrupción para la comunicación externa

Las comunicaciones se controlan por medio de interrupciones en los receptores seriales, los cuales son programados para operar a una velocidad de 1200 bauds. En lo que respecta a la comunicación externa, está programada para recibir tramas de comandos y de programas, los comandos son analizados de acuerdo a prioridades, de tal forma que los de mayor prioridad se ejecutan en el mismo instante en que se reciben. Los de menor prioridad son almacenados en una pila de comandos para su posterior ejecución, la mayoría de estos comandos se transmiten a las demás computadoras; entre otros parámetros contienen datos para la actualización de eventos de las cargas útiles.

En lo que se refiere a las tramas de programas estas son almacenadas por el programa principal para su posterior análisis o traslado. En el caso de programas para control de estabilización, los datos son almacenados en arreglos especiales para su posterior ejecución. En el caso de programas para la CCUO y la CP los datos son puestos en arreglos independientes para su traslado y ejecución. Cuando el programa es para CP se tiene el comando 1 que obliga a realizar un salto al nuevo programa cuando concluye la transmisión del programa.

La verificación de los datos enviados se hace por medio del procesamiento de las tramas tanto de programa como de comando. En el caso de programa se aprovechará cada avistamiento del Satex1 para informar a la ET los bloques en

mal estado y los bloques faltantes, esto por medio de la trama de retransmisión descrita en el capítulo 2.

Para evitar problemas de interrupción en la transmisión, se contabiliza el tiempo de recepción de palabras y se programa un tiempo finito de espera para evitar posibles estancamientos del programa, ver figura 5.6 del apéndice C.

PRUEBAS DE VALIDACIÓN

6.1 Introducción

Debido a que durante el desarrollo de la presente tesis aún no se contaba con la instrumentación total ni con el resto del equipo que deberá ir a bordo del microsatélite, fue necesario idear pruebas de validación sencillas, pero que garantizaran el buen funcionamiento de la programación.

El programa está compuesto por varios procedimientos, por lo que fue necesario probar y validar cada uno de ellos en forma independiente y, finalmente, validar la integración total del programa global.

Las pruebas de validación de los algoritmos de las computadoras a bordo, se realizaron a través de la simulación de los diferentes módulos de cómputo y cargas útiles. Para la validación se probaron en su totalidad las subrutinas que componen estos algoritmos.

En este capítulo se describe la validación de las rutinas más críticas e importantes del programa. Los programas de prueba realizados para la verificación de periféricos (convertidor A/D, puertos serie, temporizador y líneas E/S) se muestran en el apéndice B.

6.2 Validación del programa principal

Para la validación del programa principal, se midió el consumo de corriente en las distintas etapas del programa, con el fin de verificar que entrara y saliera del estado de bajo consumo de energía.

Durante la ejecución del programa, se suceden varias acciones claramente observables, de esta forma, mediante la observación de la ejecución de estas acciones y del consumo de corriente, es fácil determinar la etapa en que se encuentra el programa.

Durante la primera etapa (antes de la liberación), se pudo comprobar que mientras se mantuviera un nivel bajo en las líneas digitales asignadas a la detección de liberación, el programa entraba en bajo consumo de energía, del cual sale cada 2 minutos o bien cuando recibe algún dato por el puerto para comunicación externa. Si el dato que llega es el comando 16, se ejecuta una rutina para diagnóstico. Al finalizar la rutina de interrupción, el programa revisa el nivel en las líneas de liberación y regresa al estado de bajo consumo si el nivel sigue siendo bajo.

Una vez cambiado el nivel (a nivel alto) en las líneas de liberación, se comprobó que el programa saliera del ciclo para continuar su ejecución, es decir, configurar los dispositivos periféricos que no habían sido configurados, poner niveles altos en las líneas digitales correspondientes a la liberación de antenas, habilitar interrupciones restantes e iniciar el conteo con los temporizadores y contadores adecuados para activar las rutinas de interrupción para muestreo y estabilización.

La validación del programa principal, fue aceptada hasta comprobar que todas las acciones programadas se ejecutaban correctamente con la secuencia y tiempos correspondientes.

6.3 Validación de la rutina para la segunda etapa de estabilización

Básicamente, esta rutina debe activar o desactivar el estado de las salidas digitales asignadas para controlar las bobinas de torque magnético y liberación

del gradiente gravitacional, siguiendo rigurosamente una secuencia predeterminada por cálculos en la estación terrena y en tiempos bien establecidos por los mismos cálculos.

Todas las acciones de control efectuadas por ésta rutina son calculadas por la ET considerando los periodos orbitales del satélite, de tal forma que ella suministra las cadenas de control para estabilización para uno o varios periodos orbitales, al termino de los cuales, iniciará la ejecución de las acciones de control para el siguiente periodo orbital.

Las acciones de control para estabilización consisten en conmutar las líneas digitales asignadas a las bobinas electromagnéticas. Se asignaron dos salidas digitales para cada bobina una de las cuales habilita un circuito que suministra corriente continua en un sentido, mientras que la otra salida digital habilitará a otro circuito para proporcionar corriente a la bobina en sentido inverso. Además fueron asignadas dos líneas digitales para liberar al gradiente gravitacional, una principal y otra para redundancia.

Para realizar las pruebas correspondientes a esta rutina, fue necesario reducir el tiempo de duración de cada periodo orbital de 100 minutos a 100 segundos y conectar LED's de color rojo a cada una las líneas digitales asignadas a la habilitación de los circuitos de corriente positiva, LED's verdes a cada una de las líneas asignadas a circuitos de corriente negativa y LED's de color amarillo a las líneas asignadas a los circuitos de liberación del Gradiente Gravitacional.

Se formó una cadena de control, la cual como se indicó en el capítulo anterior, debe señalar el número de periodo orbital, el número de acciones de control para cada órbita y el tipo de acción de control (encender alguna bobina con corriente positiva, negativa o apagarla). Posteriormente se transfirió dicha cadena de control mediante puerto serie desde una computadora personal (simulando a la ET) a la tarjeta de la computadora principal. Una vez finalizada la transferencia se envió el comando para iniciar la ejecución de la rutina, posteriormente se comprobaron la secuencia, los tiempos y el cambio de periodo orbital establecidos por la cadena de control.

6.4 Validación para la rutina de interrupción para comunicación interna

Como se explicó anteriormente, la rutina de interrupción para comunicación interna se ejecuta cuando llega un dato por el receptor del puerto serie dos y,

tiene como principal objetivo responder a los requerimientos de las otras computadoras, entre los que están: recibir comando, realizar diagnóstico, recibir un nuevo programa o algoritmo y actualizar temporizadores y contadores.

Con la finalidad de validar esta parte del programa, se programó una rutina que, cargada en una tarjeta distinta a la de CP, fuera capaz de establecer un diálogo con CP, y que mediante el protocolo de comunicación interna le requiriera diagnóstico, le transfiriera comandos o programas y le indicara actualizar su tiempo.

Una vez terminado el diálogo se envió por el otro puerto serie, el resultado del diagnóstico, se ejecutaron los comandos transferidos y se verificó mediante eventos, que la actualización del tiempo se realizara en forma correcta.

Una parte importante en la programación de esta rutina, es la protección contra interrupción en el intercambio de datos, es decir, proteger la programación de la CP, ya que ésta no puede quedar esperando el arribo de todos los datos por tiempo indefinido. Entonces, en caso de que algún dato tarde más tiempo del requerido, la CP suspende el diálogo e indica en el arreglo de "estado" que ha habido una falla en la comunicación con la otra computadora.

Con la finalidad de comprobar el buen funcionamiento de esta rutina, se provocaron fallas en la comunicación interna, de tal forma que la CP detectara un retraso excesivo en el arribo de los datos, y así, obligarla a responder a dicha falla.

La validación de esta rutina fue aceptada hasta que se comprobó que el intercambio de datos y su almacenamiento fueran correctos y además, hasta que la respuesta a falla por retraso en el arribo de datos fuera correcta.

6.5 Validación de interrupción para comunicación externa

La validación de la interrupción del receptor para comunicación externa fue a través de la comunicación vía puerto serie, con una PC simulando la ET. Aquí se validó la recepción y ejecución de todos los comandos descritos en el capítulo 2, así como la recepción de tramas de programa, en éstas se aseguró que las tramas llegaran a la sección de memoria previamente definidas en arreglos. En el caso de comandos su validación fue a través de su ejecución por la CP, verificando los tiempos de ejecución y su resultado final. Con respecto a las tramas de programas, como ya se mencionó, se aseguró que estas llegaran a su

destino final; como en el caso de transmisión de programas a la CCUO donde se verificó la transmisión vía puerto serie a través de la red interna; o en el caso de control de estabilización, donde se validó la operación de las bobinas de torque magnético por medio de tiempos programados. En el caso de nuevo programa para la CP su validación se realizó al operar el comando de salto a nuevo programa.

En la recepción de comandos y programas se simularon fallas en el equipo receptor como son la caída del enlace de comunicaciones y el mal estado de las palabras recibidas (tramas erróneas). En estos casos se utilizaron algoritmos de detección de fallas, para verificar que no se afectara la continuidad de la programación abordo.

6.6 Validación de la interrupción de adquisición de telemetría

Para su validación fue necesario reducir el tiempo de los periodos orbitales, los tiempos de muestreo de sensores y otros parámetros con el fin de verificar contra un cronómetro los tiempos fijados por los temporizadores. En esta interrupción se observó que se ejecutaran las misiones programadas por la ET para la adquisición de la telemetría descritas en el comando 14. Con esta interrupción se puede validar gran parte de los algoritmos faltantes como la adquisición de telemetría especial, telemetría de magnetómetros y el diagnóstico realizado periódicamente, esto es posible ya que en base a los temporizadores de esta interrupción se lleva a cabo la sincronización de las tareas mencionadas.

Adicionalmente se realizaron pruebas para corroborar el buen funcionamiento del convertidor A/D, necesario para la adquisición de datos. Para esto se desarrollaron programas en lenguaje C para decodificar los datos obtenidos de sensores simulados (por medio de potenciómetros), para almacenarlos en memoria y para enviarlos vía puerto serie a una PC.

**ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA**

CONCLUSIONES Y RECOMENDACIONES

7.1 Introducción

En esta tesis se ha presentado una arquitectura de cómputo distribuida para un microsatélite experimental, la cual, permitirá realizar una serie de actividades con los diversos módulos del sistema. Para las computadoras principales de la arquitectura se desarrollaron programas que ejecutan las tareas de control y de comunicaciones. El diseño global ha tomado en cuenta aspectos críticos del proyecto como son el tiempo de desarrollo, tolerancia a fallas, confiabilidad y el presupuesto disponible.

En este capítulo, se exponen las conclusiones que describen los logros alcanzados, así como las observaciones y recomendaciones pertinentes que permitan continuar con el desarrollo del presente trabajo .

7.2 Conclusiones

- 1) Se diseñó una arquitectura de cómputo distribuido y de comunicaciones acorde con las necesidades de control y de comunicaciones necesarias en el Satex1.
- 2) Se elaboró un programa que cubre las necesidades operativas de la computadora principal abordo del Satex1, entre las cuales se tiene la capacidad de interactuar con otros módulos del sistema mediante una red interna y con una estación terrena a través de enlaces de telecomunicaciones.
- 3) Los procesos de control y de comunicaciones cuentan con protección tolerante a fallas que incrementa los índices de seguridad de la programación desarrollada.
- 4) La tolerancia a fallas descrita en el punto anterior se encuentra apoyada en electrónica que permite un perfecto desempeño de las protecciones por software. Este apoyo se encuentra en la aplicación de módulos redundantes y electrónica para prevenir casos de falla.
- 5) El diseño de la red y los protocolos para la comunicación interna fueron desarrollados especialmente para satisfacer los requerimientos de intercomunicación del equipo de cómputo abordo del Satex1. No están basados en protocolos ni topologías existentes comercialmente, debido a que estos no satisfacen los requerimientos de comunicación o en su defecto, son tan elaborados y complejos que sobrepasan la capacidad de procesamiento que se les puede asignar.
- 6) La segmentación del programa global en procedimientos, facilitó el desarrollo de la programación, ya que fue posible programar cada una de las tareas en forma independiente.
- 7) La segmentación del programa, también permitió la validación de las rutinas en forma independiente, lo cual facilitó enormemente la validación del programa global, una vez realizada su integración total.
- 8) Se comprobó que la elección de la tarjeta utilizada para el Satex1, fue acertada pues el *hardware* integrado a ellas así como su software de desarrollo,

constituyó una herramienta esencial para realizar la programación de los procedimientos requeridos y la ejecución de tareas de control y comunicaciones deseadas.

7.3 Recomendaciones

Las recomendaciones pertinentes que se pueden citar en este trabajo son las siguientes.

1) En caso de que existan futuros proyectos satelitales, es de vital importancia apoyar un proyecto para desarrollar nuestra propia computadora calificada para el espacio de órbita alta, así como planear con anticipación la adquisición de componentes de calificación espacial (por los que hay que esperar meses para su entrega). Este proyecto se tendría que iniciar cuando menos con un año de anticipación al inicio del siguiente proyecto satelital.

2) Dentro de los beneficios que se tendrían al desarrollar nuestra propia computadora se encuentran: la reducción del costo del subsistema de cómputo debido a que algunos productos terminados de este tipo exceden los 200 mil USD; los cambios requeridos por misiones especiales se realizarán de manera más rápida; se puede tener un stock de 2 ó 3 computadoras listas y probadas para vuelo espacial; y además se tendría la posibilidad de vender este equipo a otros países que como el nuestro inicien actividades en materia espacial.

3) Uno de los principales obstáculos que se presentaron durante el desarrollo del proyecto, fue la dificultad para el intercambio de información con algunos grupos participantes, por lo anterior recomendamos ampliamente para futuros proyectos idear un plan de trabajo para mantener una constante comunicación entre todos los participantes, de tal forma, que sea posible un rápido, eficiente y veráz intercambio de información en el momento requerido.

Finalmente, todo el grupo de trabajo del IIUNAM expresa un gran reconocimiento al Instituto Mexicano de Comunicaciones y a sus directivos que han hecho posible el inicio de proyectos tecnológicos espaciales, los cuales en la actualidad constituyen un pilar imprescindible en la generación de conocimiento y

soluciones en las áreas de comunicaciones, cómputo y electrónica, entre otras. Su labor es todavía más importante al fomentar el trabajo de equipo entre instituciones, lo cual contribuye a concretar esfuerzos colectivos en rubros tecnológico y académico a todo lo largo de nuestro país.

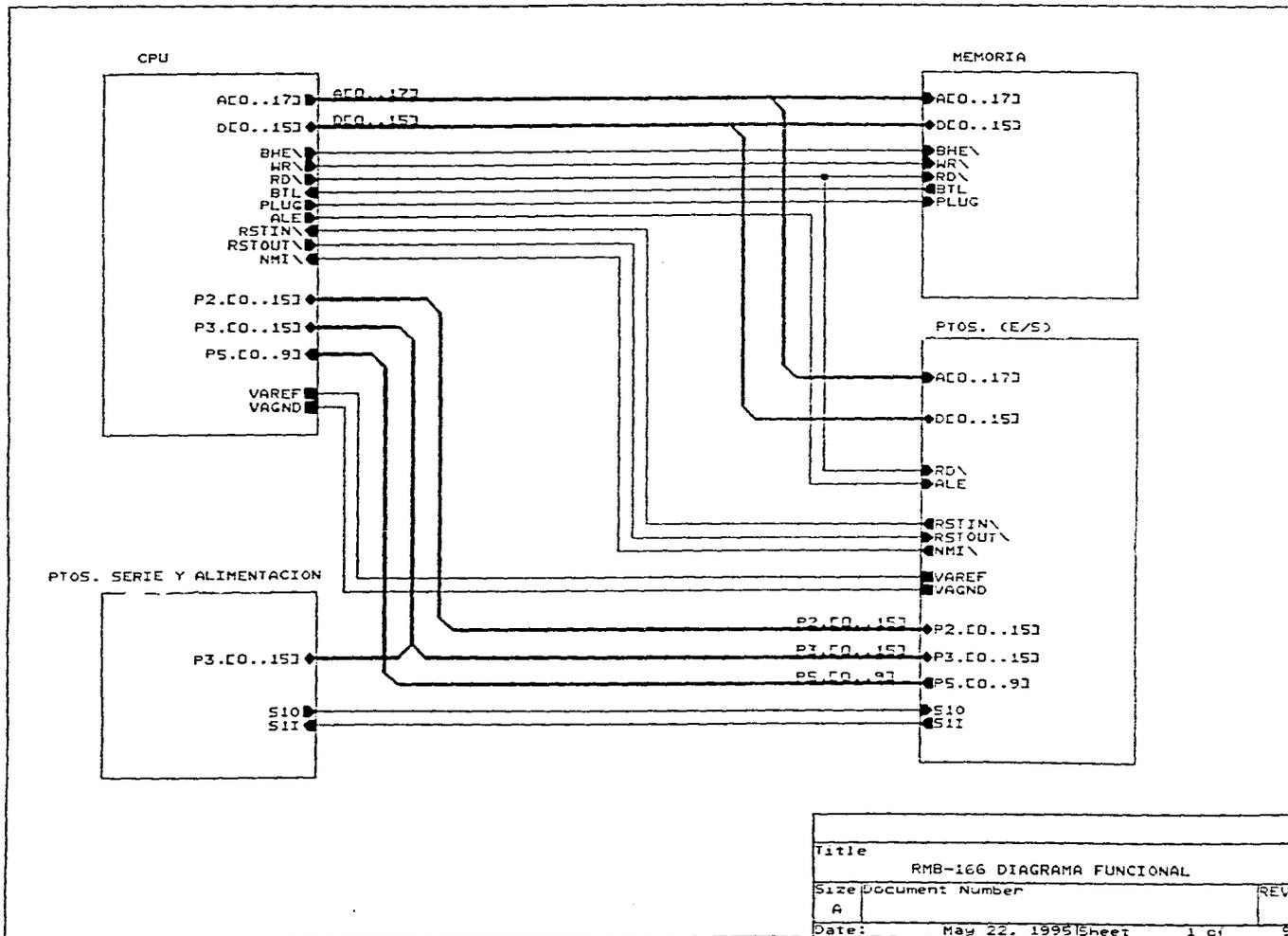
1. M.C. Herrmann. " Small Spacecraft Applications for Future Science Missions", Conference on Small Satellites, UTAH State University, Septemper13-16, 1993.
2. Walter L. Morgan, Gary D. Communications Satellites Handbook., tercera edición, Mc Graw Hill, 1985.
3. Eric J. Hoffman and Robert C. Moore, et al. Satellite Communications, Tracking & Control Course section 3. The Jhons Hopkins University, 1990.
4. SAB 801166/83C166 User's Manual 1990.
5. Walter L. Morgan, Gary D. Principles of Satellite Comunications., segunda edición, Mc Graw Hill, 1985.
6. Adrew S. Tanenbaum, Computers Networks, segunda edición, Prentice hall.
7. Jhon Schwartz, Telecommunications Networks: Protocols, Modeling and Analisis . 3a de. Prentice Hall.
8. Brian W. Kernigham, El lenguaje de Programación C, segunda edición, Prentice hall, 1991.
9. Manual del C-166 (Compilador).
10. Manual del A-166 (Ensamblador).
11. Manual del XVW-166 (Debbuger).

12. Herbert Schildt , Turbo C/C++, The Complete Reference, Mc. Graw-hill, 1992.

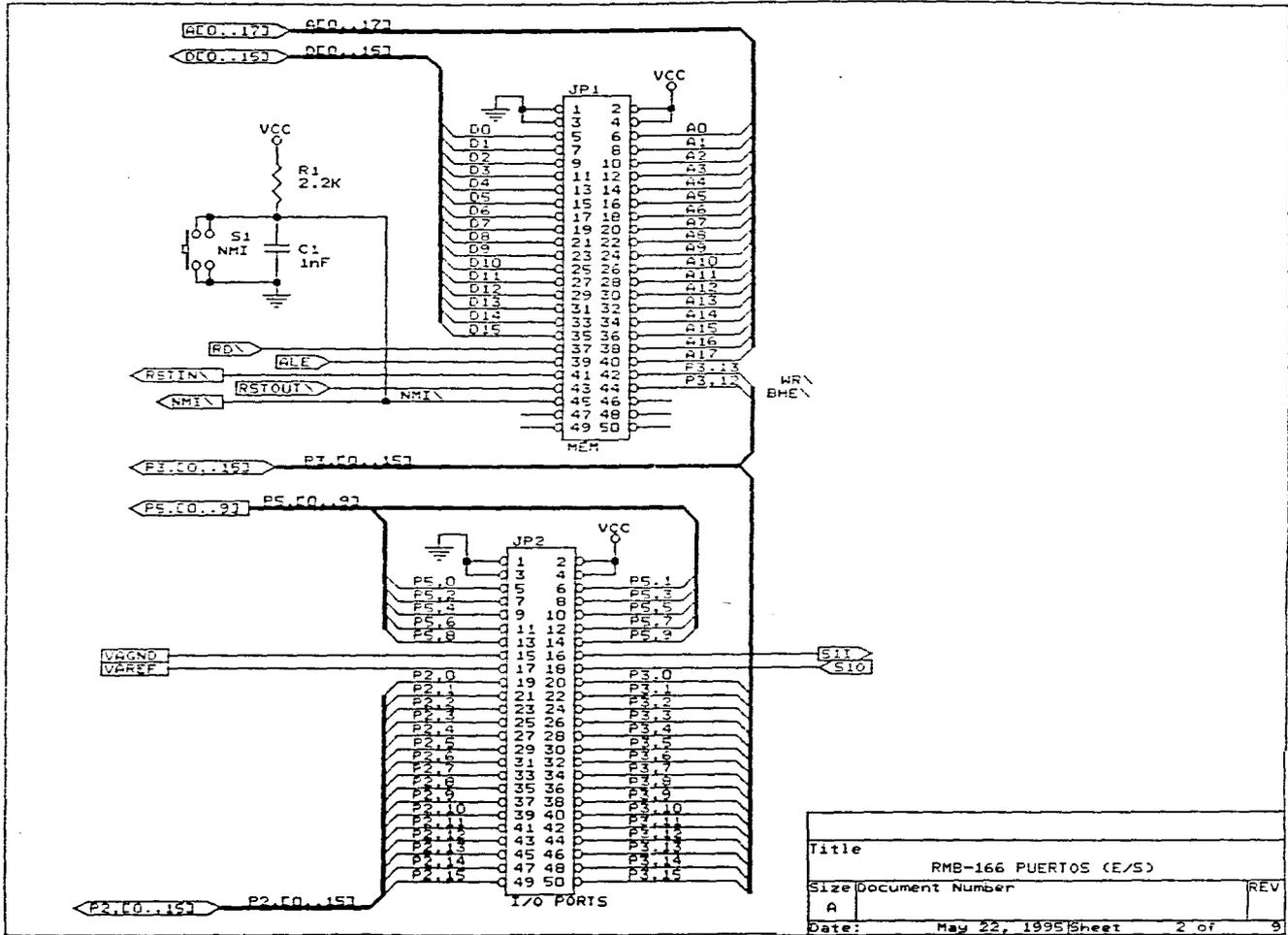
Esquemáticos y diagramas electrónicos de la tarjeta utilizada RMB-166

PAGINACION VARIA

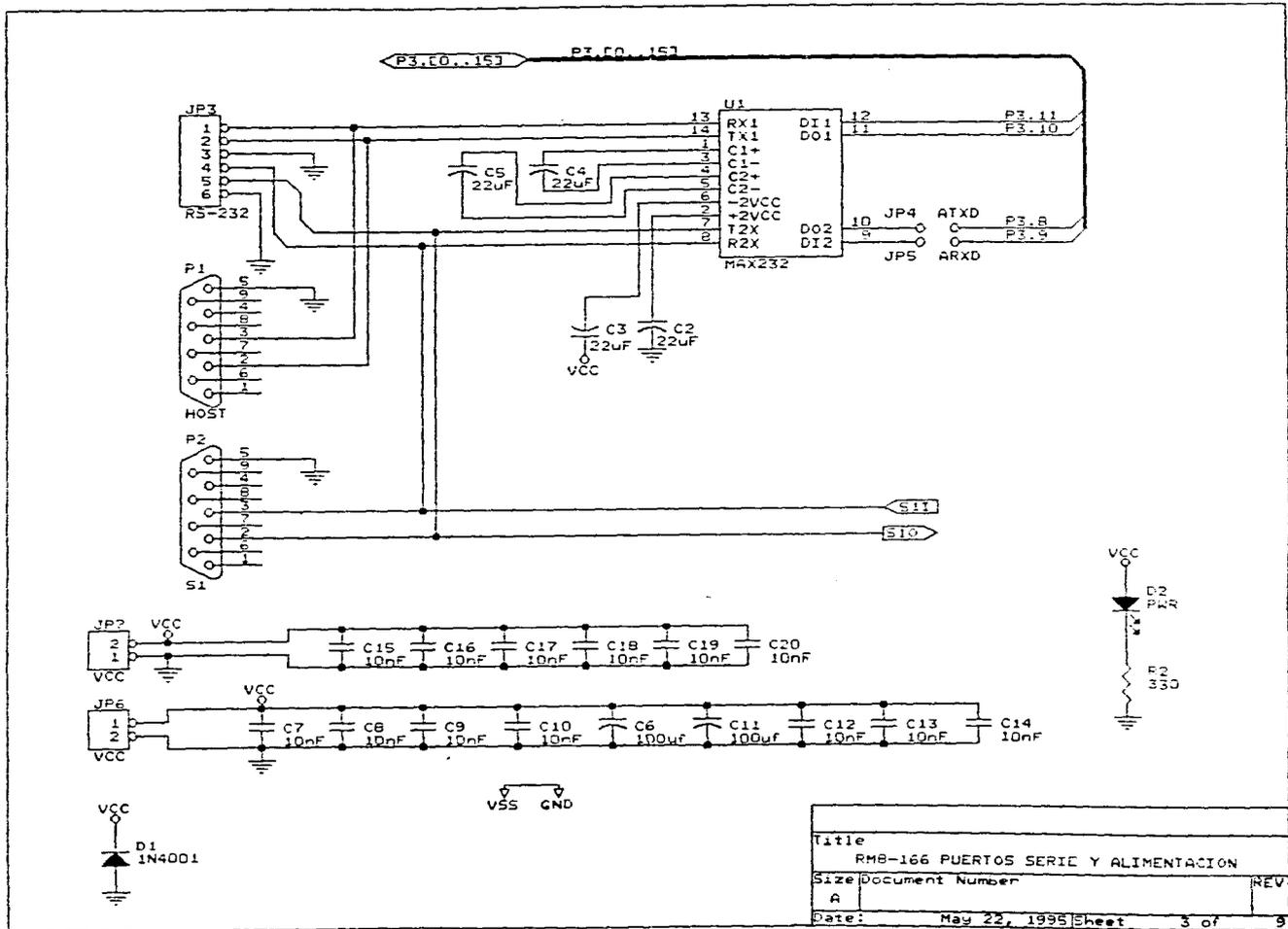
COMPLETA LA INFORMACION



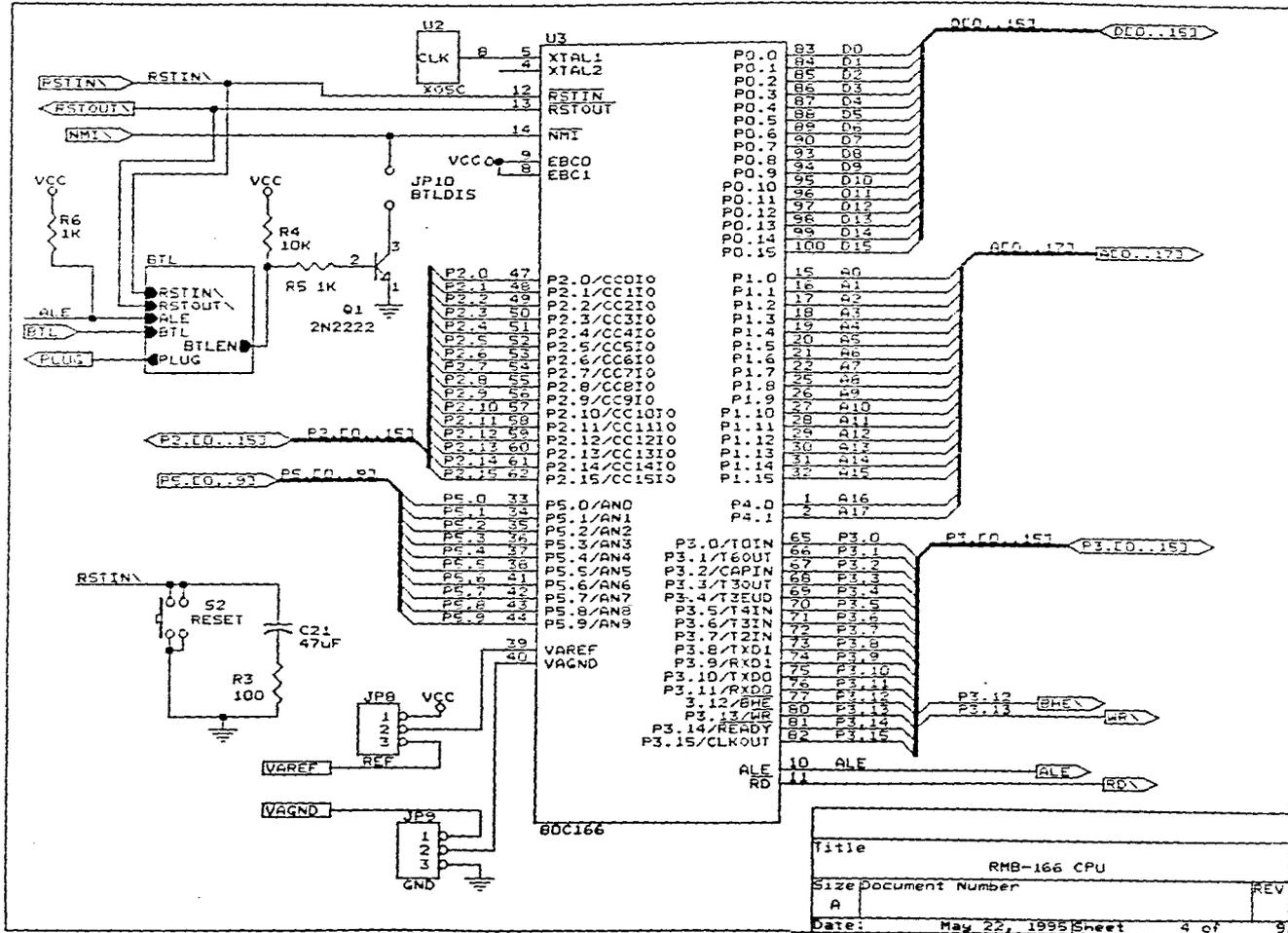
Title		
RMB-166 DIAGRAMA FUNCIONAL		
Size Document Number		REV
A		
Date:	May 22, 1995	Sheet 1 of 9

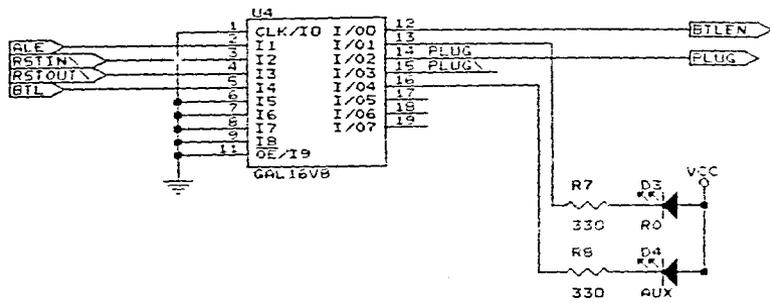


Title		
RMB-166 PUERTOS (E/S)		
Size	Document Number	REV
A		
Date:	May 22, 1995	Sheet 2 of 9

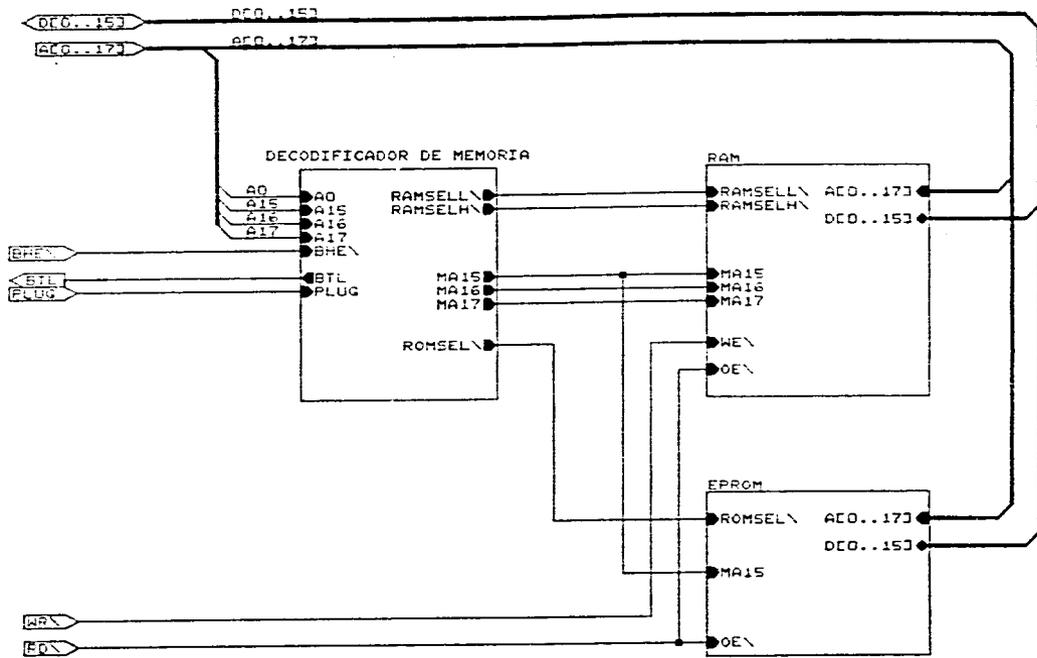


Title		
RMB-166 PUERTOS SERIE Y ALIMENTACION		
Size	Document Number	REV
A		
Date:	May 22, 1995	Sheet 3 of 9

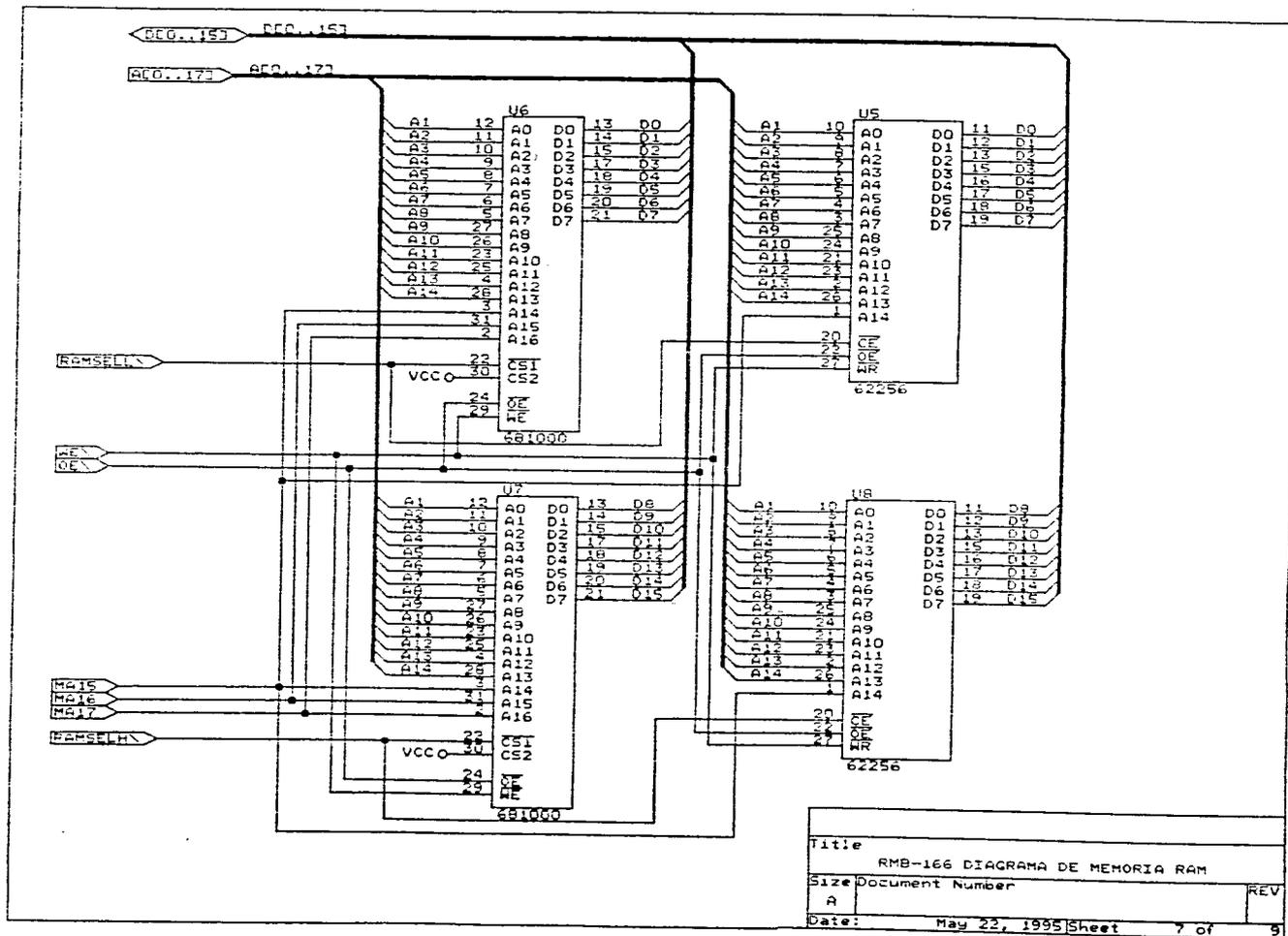




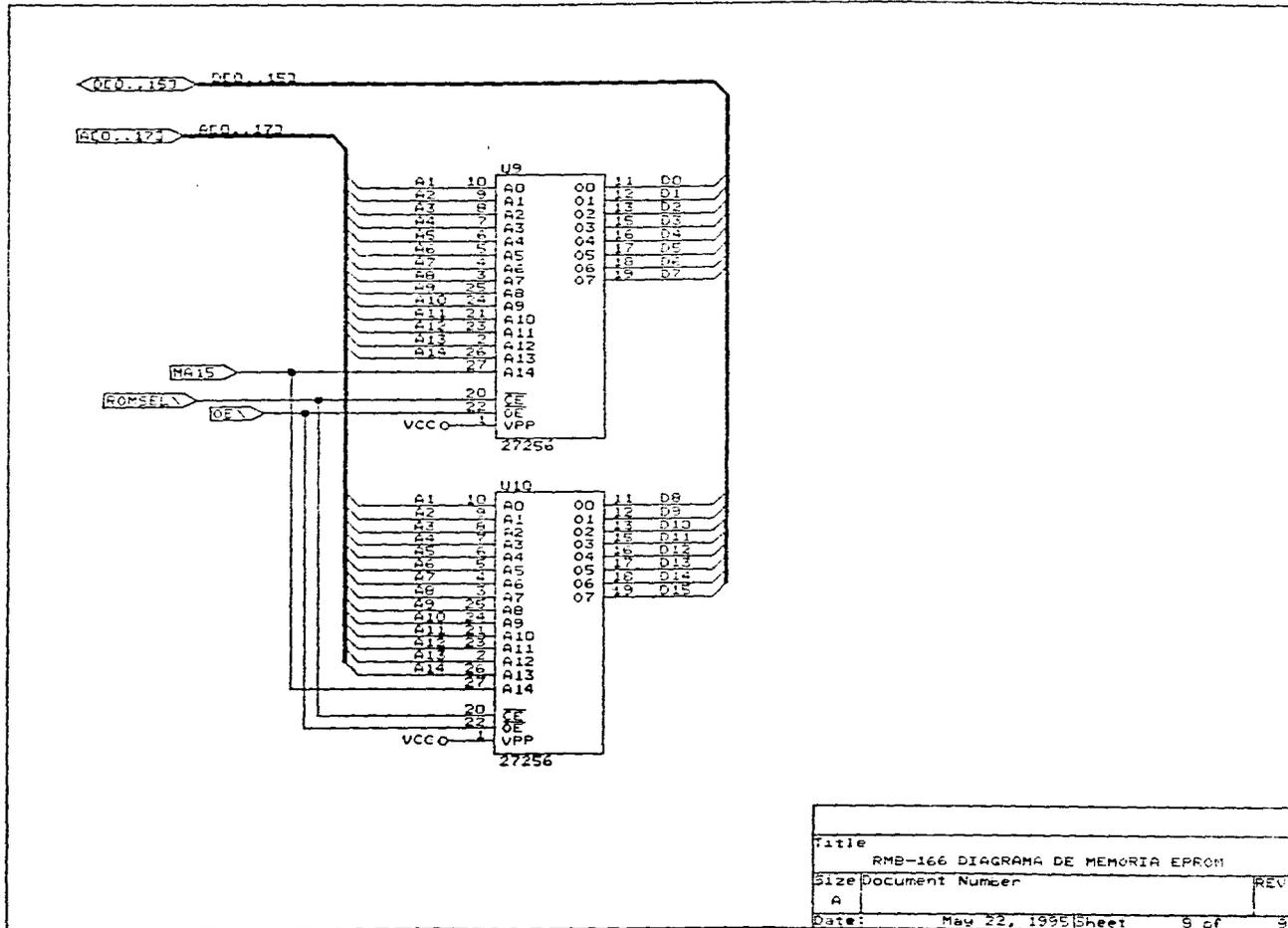
Title		
RMB-166 CIRCUITO DE BOOTSTRAP		
Size	Document Number	REV
A		
Date:	May 22, 1995	Sheet 5 of 9



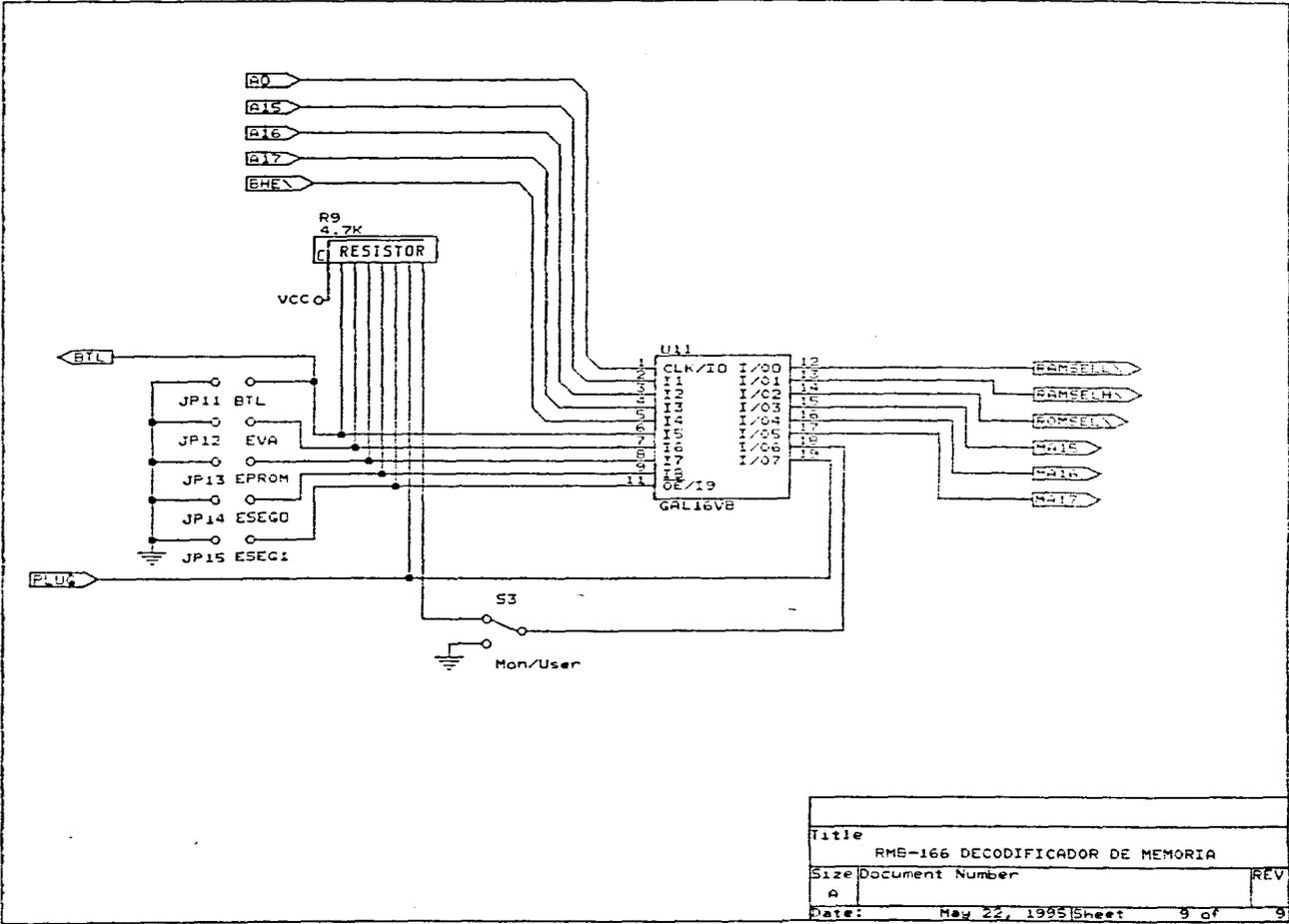
Title		
RMB-166 DECODIFICADOR DE MEMORIA DIAGRAMA		
Size Document Number		REV
A		
Date:	May 22, 1995	Sheet 6 of 9



Title		
RMB-166 DIAGRAMA DE MEMORIA RAM		
Size Document Number		REV
A		
Date:	May 22, 1995 Sheet	7 of 9



Title	
RMB-166 DIAGRAMA DE MEMORIA EPROM	
Size	Document Number
A	REV
Date:	May 22, 1995 Sheet 5 of 5



Title		
RM5-166 DECODIFICADOR DE MEMORIA		
Size Document Number		REV
A		
Date:	May 22, 1995	Sheet 9 of 9

Programas de prueba realizados para la tarjeta RMB-166

ESTÉ PROGRAMA ES UNA SUBROUTINA DE LA PROGRAMACIÓN EN LA
COMPUTADORA DE CARGA ÚTIL ÓPTICA Y FUE UTILIZADO PARA VALIDAR LAS
SUBROUTINAS PARA COMUNICACIÓN INTERNA DE LA
COMPUTADORA PRINCIPAL

```
#include <stdio.h>
#include <c166.h>
#include <reg166.h>
#include "serio2.h"
#define z 1
```

LOCALIZACIÓN DE DATOS PARA AUTODIAGNÓSTICO

```
#pragma combine nb=A16384
int dato1=3;
#pragma combine nb=A16386
int dato2=5;
#pragma combine nb=A16388
int dato3=21;
#pragma combine nb=A16390
int dato4=7;
#pragma combine nb=A16392
int dato5=6;
#pragma default_attributes
unsigned int CARGA;
int retar(unsigned int p);
void cmd20(void);
void intser(void);
unsigned int status[1][10]; /* status[in1][10] */
int comint=0,comext=0,nvodata,in1=1,r=0,r1,pcint=0;
int resul1,resul2,resul3,resul4,resula;
```

VARIABLES PROVENIENTES DE OTRAS RUTINAS

```
int cmd[1][27]; /* arreglo para comandos cmd[incmd][27] */
int n=0,bajo=0,incmd=0;
void audiag(void);
char bcmd='N'; /* PUEDE O NO ESTAR ACTIVADA */
```

```
/****** MAIN () *****/
```

```
main() {
    int di;
    /* INICIALIZACIONES PARA LA INTERRUPCIÓN */

    DP2=0x0FFFF; /* PUERTO P2 TODO DE ESCRITURA */
}
```

```

init_serio();      /* INICIALIZACION DEL PUERTO SERIE 0 */
init_serio2();    /* INICIALIZACION DEL PUERTO SERIE 1 */
S1RIC=0x09,      /* PRIODAD DE INTERRUCION: NIVEL 2 GRUPO 0 */
S1RIE=1;        /* HABILITACION DE LA INTERRUCION POR RECEPTOR
                DEL PUERTO SERIE 1 (S2) */

/* S0RIC=0x0A;
S0RIE=1;*/
T01CON=0x7;      /*timer T0 */
IEN=1;          /* HABILITACION GLOBAL DE INTERRUPCIONES */

                /* LOOP INFINITO */
while( z ) {
  CARGA=65332;
  retar(CARGA);  /**** cmd prueba*/
  if(r1)
  {
    switch(nvodato)
    {
      case 1 :
        intser();
        break;
      case 2:
        cmd20();
        break;
      case 3:
        putch(status[in1][4]);
        putch(45);
        break;
    }
  }
}
if(comext)
{
  S1RIR=0;
  _putbit(1,P2,1);
  while( !kbit2());
  putch2(0x08F);
  CARGA=65039;
  retar(CARGA);
  putch2(0x08D);
  retar(CARGA);
  for(di=0;di<28;di++)
    putch2(cmd[incmd][di]);
  _putbit(0,P2,1);
  S1RIR=1;
  comext=0;
}
if(comint)

```

```

    {
        for(di=0;di<28;di++)
            putchar(cmd[incmd][di]);
        comint=0;
    }
}

```

```

/*****
/* INICIA RUTINA DE INTERRUPCION PARA COM. EXTERNA */

```

```

/**#pragma noframe
interrupt( 0x02B ) */
void
intser (void)
{
    int ec,i=76;
    CARGA=64648;
    retar(CARGA);
    for(ec=0;ec<28;ec++)
    {
        if(r1)
        {
            i--;
            _putbit(0,P2,0);
            cmd[incmd][ec]=nvodato;
            comext=1;
        }
        else
        {
            _putbit(1,P2,0);
            comext=0;
            break;
        }

        CARGA=65332;
        retar(CARGA);
    }
    putchar(i);
}

```

```

/*****
/* INICIA RUTINA DE INTERRUPCION PARA COM. INTERNA */

```

```

#pragma noframe
interrupt( 0x02E ) void
intser2 (void)

```

```

{
int ic;
S1RIE=0;
if(n>=bajo)
{
if(incmd<5)
{
CARGA=64648;
r=retar(CARGA);
if(r)
{
if(nvodato==0x2D) /*OPCION: TRASPASO DE COMANDO*/
{
CARGA=64648;
retar(CARGA);
CARGA=65332;
for(ic=0;ic<=3;ic++)
{
if(r)
{
cmd[incmd][ic]=nvodato;
comint=1;
}
else
{
comint=0;
break;
}
}
r=retar(CARGA);
}
if(comint)
{
bcmd='S';
incmd++;
}
}
}
else if(nvodato==0x020) /*OPCION: AUTODIAGNOSTICO*/
{
audiag();
}
}
}
}
S1RIR=0;
S1RIE=1;
}

```

/* INICIA SUBROUTINA PARA COMANDO 20: DIAGNOSTICO DE EQUIPO */

void

```

cmd20(void)
{
    S1RIE=0; /* DESHABILITA INTERRUPCION POR PTO. SERIE INTERNO */
            /***** COMIENZA DIAGNOSTICO DE ZONAS DE MEMORIA DE CP*****/

            /* INICIA DIAGNOSTICO PARA CCUO */

    status[in1][4]=48; /*cambiar 0x48 por 0x0*/
    pcint=0x8F;
    putch2(pcint);
    pcint=0x80;
    putch2(pcint);
    CARGA=64648; /* ESPERAR 50 mseg PARA QUE CCUO CODIFIQUE A PCINT. */
    r=retar(CARGA);
    if(r)
    {
        if(nvodato!=3)
        {
            status[in1][4]=status[in1][4] | 1;
        }
        CARGA=65332; /* CARGA PARA ESPERAR 15 mseg  $\phi$  NUEVO DATO */
        r=retar(CARGA);
        if(r)
        {
            if(nvodato!=5)
            {
                status[in1][4]=status[in1][4] | 2;
            }
            r=retar(CARGA);
            if(r)
            {
                if(nvodato!=21)
                {
                    status[in1][4]=status[in1][4] | 4;
                }
                r=retar(CARGA);
                if(r)
                {
                    if(nvodato!=7)
                    {
                        status[in1][4]=status[in1][4] | 8;
                    }
                    r=retar(CARGA);
                    if(r)
                    {
                        if(nvodato!=6)
                        {
                            status[in1][4]=status[in1][4] | 16;
                        }
                    }
                }
            }
        }
    }
}

```



```

{
    T0=p;
    TOR=1;
    while( ! ( S1RIR | T0IR | S0RIR ) );
    if(S1RIR)
    {
        nvodato=getch2();
        r=1;
    }
    else
        r=0;
    if(S0RIR)
    {
        nvodato=getch();
        r1=1;
    }
    else
        r1=0;
    /* S0RIR=0;
    S1RIR=0; */
    T0IR=0;
    TOR=0;
    return(r);
}

```

ESTE PROGRAMA ENVÍA UN TREN DE PULSOS POR EL PUERTO P2.2, EL CUAL SE INTERRUPE POR DESBORDAMIENTO EN EL TIMER 6. UNA VEZ DADA LA INTERRUPCIÓN ENTRA A LA FUNCIÓN INTTIMER LA CUAL ENVÍA UN PULSO POR EL PUERTO P2.3

```

#include<c166.h>
#include<reg166.h>
#include<stdio.h>
#include<stddef.h>
#include<time.h>
void retar(void);
void retar1(void);
#define si 1
main()
{
    T6CON=0x0007; /* T6 MODO DE CONTEO ASCENDENTE Y PREESCALAMIENTO
                  PARA Fosc/1024 */
    T6IC=0x08; /* INTERRUPTON NIVEL 2, GRUPO 0 */
    T6IE=1; /* HABILITACION DE INTERRUPTON */
    T6=0; /* LIMPIA CONTADOR T6 */
    IEN=1; /* HABILITACION GLOBAL DE INTERRUPTONES */
    T6R=1; /* ACTIVACION DE TIMER */
    DP2=0x0FFFF; /* HABILITA AL PUERTO 2 TODO PARA ESCRITURA */
    while(si)
    {

```

```

/* PROGRAMA PRINCIPAL : ENVIO DE PULSO POR PTO P2.2 */

    _putbit(1,P2,2);
    retar();
    _putbit(0,P2,2);

}
}

```

```

/* RUTINA DE RETARDO PARA EL PROGRAMA PRINCIPAL */

void retar(void)
{
    int i,j,k;
    for(i=0;i< 2;i++)
    {
        for(j=0;j<253;j++)
        {
            for(k=0;k<253;k++)
            {
                _nop();
            }
        }
    }
}

```

```

/* RUTINA DE RETARDO QUE UTILIZA LA FUNCION DE INTERRUPCION */

void retar1(void)
{
    int m,n,o;
    for(m=0;m< 30;m++)
    {
        for(n=0;n<253;n++)
        {
            for(o=0;o<253;o++)
            {
                _nop();
            }
        }
    }
}

```

```

/* FUNCION DE INTERRUPCION ACTIVADA POR DESBORDAMIENTO EN EL
TIMER T6 (INTERRUPCION NUMERO 26h) */

interrupt( 0x26 ) void
inttimer(void)

```

```

{
    T6R=0;      /* SE DETIENE EL TIMER T6 (EN CASO DE
                NO INCLUIRSE ESTE COMANDO EL TIMER
                CONTINUARIA SU CONTEO Y ACTIVARIA
                UNA NUEVA INTERRUPCION INMEDIATAMENTE
                DESPUES DE SALIR DE ESTA FUNCION Y
                ALCANZAR OTRO DESBORDAMIENTO). */

    /* ENVIO DE PULSO POR EL PUERTO P2.3 (TAREA ASIGNADA A LA
       FUNCION DE INTERRUPCION). */
    _putbit(1,P2,3);
    retar1();
    _putbit(0,P2,3);

    T6R=1;      /* SE ACTIVA EL TIMER UNA VEZ TERMINADA LA
                FUNCION DE INTERRUPCION, LO CUAL PROVO-
                VOCARA UNA NUEVA INTERRUPCION CUANDO SE
                DE OTRO DESBORDAMIENTO. */
}

```

ESTE PROGRAMA MUESTRA LA INTERRUPCIÓN POR RECEPCIÓN EN EL PUERTO SERIE 1 (S2)

```

#include <stdio.h>
#include <c166.h>
#include <reg166.h>
#include "serio2.h"
#define z 1

void retar(void);
main() {
    int c,ca;
    DP2=0x0FFFF; /* PUERTO P2 TODO DE ESCRITURA */
    init_serio(); /* INICIALIZACION DEL PUERTO SERIE 0 */
    init_serio2(); /* INICIALIZACION DEL PUERTO SERIE 1 */
    S1RIC=0x008; /* PRIODIDAD DE INTERRUPCION: NIVEL 2 GRUPO 0 */
    S1RIE=1; /* HABILITACION DE LA INTERRUPCION POR RECEPTOR
              DEL PUERTO SERIE 1 (S2) */
}

```

```

IEN=1;          /* HABILITACION GLOBAL DE INTERRUPCIONES */
P2=0x0FFFF;    /* PONE UN NIVEL ALTO EN TODO EL PUERTO 2 */
while( ! kbhit() ); /* ESPERA UN DATO EN EL RECEPTOR DEL PTO. S 0*/
c=getch();      /* LEE EL DATO POR PUERTO SERIE 0 (S1) */
ca=c+1;        /* INCREMENTA EN UNO EL VALOR DEL DATO */
putch(ca);     /* ENVIA EL DATO POR PUERTO SERIE 0 (S1) */
P2=0x0;        /* LIMPIA EL PUERTO 2 */

/* LOOP INFINITO */

while( z ) {

    /* SE ENVIA UN TREN DE PULSOS AL PIN 15 DEL PTO. P2, EL CUAL
    ES DETENIDO AL ACTIVARSE LA INTERRUPCION PROGRAMADA, Y
    VUELVE AL TERMINAR DICHA INTERRUPCION */

    _putbit(0,P2,15);
    retar();
    _putbit(1,P2,15);
    retar();
}

}

/* RUTINA DE INTERRUPCION: PONE UN "1" EN EL P2.14, RECIBE Y REGRESA
5 CARACTERES POR EL PUERTO SERIAL 1 (S2) Y FINALMENTE PONE UN
"0"
EN EL PUERTO P2.14 */

#pragma nframe
interrupt( 0x02E ) void
intser (void)
{
    int c,i;
    _putbit(1,P2,14);
    for(i=0;i<5;i++){
        while( ! kbhit2() );
        c=getch2();
        putch2(c);
        P2=i | 0x4000;
    }
    _putbit(0,P2,14);
}

/* RUTINA DE RETARDO */

void retar(void)
{
    int i,j,k;

```

```

for(i=0;i<6;i++)
{
for(j=0;j<253;j++)
{
for(k=0;k<253;k++)
{
_nop();
}
}
}
}

```

PROGRAMA DE PRUEBA PARA LA PROTECCIÓN POR TIEMPO EN LA RECEPCIÓN DE DATOS

```

#include<c166.h>
#include "serio2.h"
#include<reg166.h>
#include<stdio.h>
#include <time.h>
#define si 1

/* *****
 * VALOR DE CARGA PARA TIEMPO DESEADO, ESTE VALOR *
 * SE CALCULA MEDIANTE LA FORMULA: *
 * " CARGA=65625-19531*t ? *
 * DONDE t ES EL TIEMPO DESEADO (MAXIMO 3.35 SEG) *
 * EN SEGUNDOS. */
    unsigned int CARGA=65332;
/*
 * IMPORTANTE: EL VALOR DE CARGA SIEMPRE DEBE SER *
 * ENTERO, POR LO CUAL DEBERA TRUNCARSE EN CASO *
 * SE DESEEN VALORES DECIMALES PARA t. *
 * POR EJEMPLO PARA t=15 miliseg, CARGA=65332 *
 *****/

int nvodato,r,a[15];
int retar(unsigned int p);

main()
{
    init_serio2();
    T01CON=0x7;
    DP2=0x0FFFF;

    while(si==1)

```

```

{
    int i;
    _putbit(1,P2,0);
    while(!kbit2());
    S1RIR=0;
    _putbit(0,P2,0);
    for(i=0;i<15;i++)
    {
        retar(CARGA);
        if(r)
        {
            a[i]=nvodato;
            putch2(a[i]);
        }
        else
            putch2(64);

        /* retar(CARGA);*/
    }
}
}

```

/* ESTA SUBROUTINA SALE CUANDO LLEGA UN DATO A ALGUN PUERTO SERIE O SE TERMINA EL TIEMPO ESPECIFICADO */

```

int
retar(unsigned int p)
{
    T0=p;
    TOR=1;
    _putbit(1,P2,15);
    while( ! ( S1RIR | T0IR ) );
    if(S1RIR)
    {
        nvodato=getch2();
        r=1;
    }
    else
        r=0;
    T0IR=0;
    TOR=0;
    _putbit(1,P2,14);
    return(r);
}

```

PROGRAMA DE PRUEBA PARA LA ACTIVACIÓN DE LINEAS DIGITALES

```
#include<c166.h>
```

```

#include "serio2.h"
#include<reg166.h>
#include<stdio.h>
#include <time.h>
#define si 1

/* *****
 * VALOR DE CARGA PARA TIEMPO DESEADO, ESTE VALOR *
 * SE CALCULA MEDIANTE LA FORMULA: *
 * " CARGA=65625-19531*t ? *
 * DONDE t ES EL TIEMPO DESEADO (MAXIMO 3.35 SEG) *
 * EN SEGUNDOS. */
    unsigned int CARGA=46094;
/*
 * IMPORTANTE: EL VALOR DE CARGA SIEMPRE DEBE SER *
 * ENTERO, POR LO CUAL DEBERA REDONDEARSE EN CASO *
 * SE DESEEN VALORES DECIMALES PARA t. *
 * POR EJEMPLO PARA t=1 seg CARGA=35860 *
 *****/

int nvodato,r;
int retar(unsigned int p);

main()
{
    init_serio2();
    T01CON=0x7;
    DP2=0x0FFFF;

    while(si==1)
    {
        _putbit(1,P2,0);
        retar(CARGA);
        if(r)
            putch2(nvodato);
        else
            putch2(255);
        _putbit(0,P2,0);
        retar(CARGA);
    }
}

/* ESTA SUBROUTINA SALE CUANDO LLEGA UN DATO A ALGUN PUERTO SERIE O
SE TERMINA EL TIEMPO ESPECIFICADO */

int retar(unsigned int p)
{
    T0=p;
    TOR=1;
    _putbit(1,P2,15);
}

```

```

while( ! ( S1RIR | T0IR) );
if(S1RIR)
{
    nvodato=getch2();
    r=1;
}
else
    r=0;
T0IR=0;
T0R=0;
_putbit(1,P2,14);
return(r);
}

```

PROGRAMA QUE ALMACENA LOS DATOS DEL CONVERTIDOR ANALÓGICO EN UN ARREGLO EN MEMORIA RAM PARA DESPUES SER EVIADOS VIA PUERTO SERIE A UNA PC

```

#include<c166.h>
#include<reg166.h>
#include<stdio.h>
#include "serio2.h"
#define z 1
#define ADCON_CON 0x0024
#define a 0x078
#define b 0x0209
#define c 0x0FA00
#define DP2_P2 0x0FFFF
#define SRCP0_L 0x0FEA0
void retar(void);
#pragma combine nb=A64000
int aa[5]; /*ARREGLO EN EL CUAL SE ALMACENAN LOS DATOS DEL CONVERTIDOR*/
#pragma default_attributes
int i,o;

main()
{
    SRCP0= SRCP0_L; /* DIRECCION DEL BUFFER DEL CONVERTIDOR A/D */
    ADCON=ADCON_CON; /* CONFIGURACION DEL CONVERTIDOR PARA REALIZAR EL MUESTREO DE 5 CANALES */
    ADCIC=a; /* CONFIGURA EL REGISTRO DE INTERRUPCIONES */

    IEN=1; /* HABILITACION DE INTERRUPCIONES */

    o=0x7041; /* BANDERA DE INICIO */
    init_serio();/* INICIALIZACIãN DEL PUERTO SERIE */
}

```

```

while(z){
    PECC0=b; /* CONFIGURA REGISTRO DEL CONTROLADOR DE
              EVENTOS PERIFERICOS */
    DSTP0=c; /* DIRECCION DE DESTINO A LA CUAL SE ENVIARAN
              LOS DATOS */
    _putbit(1,ADCON,7); /* SEÑAL DE ARRANQUE AL CONVERTIDOR */
    while(ADST!=1);
    putch01(o);

    for(i=0;i<5;i++) /* ENVIO DE DATOS DEL CONVERTIDOR A/D
                     A LA PC */
        putch01(aa[i]);

    retar();

}
/* RUTINA DE RETARDO PARA EL ENVIO DE DATOS A LA PC */
}
void retar(void)
{
    int i,j,k;

    for(i=0;i<3;i++)
    {
        for(j=0;j<253;j++)
        {
            for(k=0;k<253;k++)
            {
                _nop();
            }
        }
    }
}
}

```

PROGRAMA QUE RECIBE Y DESPLIEGA LOS DATOS DEL CONVERTIDOR A/D

```

#include <conio.h>
#include <stdio.h>
#include <bios.h>
#include <dos.h>
#include <stdio.h>

void ini_port(void); /* inicializa puerto serie */

```

```

void pantalla(void); /*dibuja la pantalla de reporte*/
void ventana(int upx,int upy,int downx,int downy);
void window(int left, int top, int right, int bottom);
void lee_buffer(char *buffer);
void procesa(char *buffer);

void main(void) /* Programa para recepcion de datos en serie */
{
    char buffer[10]={0};
    unsigned char buff=0;

    ini_port();
    clrscr();
    pantalla();

    /******/
orig: while((inp(0x2fd) & 0x01)==0x00);
    buff=inp(0x2f8);
prin:  if(buff=='A')
    {
        while((inp(0x2fd) & 0x01)==0x00);
        buff=inp(0x2f8);
        if(buff=='p')
        {
            lee_buffer(buffer);
            procesa(buffer);goto orig;
        }
        else
        {
            goto prin;
        }
    }
    else
    {
        goto orig;
    }
    /******/
}

void lee_buffer(buffer) /*lee los siguientes 10 bytes*/
char *buffer;
{
    register int i;
    for(i=0;i<10;i++){
        while((inp(0x2fd) & 0x01)==0x00);
        buffer[i]=inp(0x2f8);}
}

```

```

}

void procesa(buffer)
char *buffer;
{
    char code=0;
    unsigned char tempo1=0,tempo2=0;
    register int i;
    unsigned int palabra=0;
    float resulta=0;

    for(j=0;i<10;i+=2)
    {
        tempo1=buffer[i];
        tempo2=buffer[i+1]&3;
        palabra=(tempo2*256)+tempo1;
        code=(buffer[i+1]>4)&0x0F;
        resulta=palabra*(0.0048828);
        gotoxy(3,6);
        if(code==4){
            gotoxy(3,wherey());
            printf("%f\n",resulta);
        }
        if(code==3){
            gotoxy(20,wherey());
            printf("%f",resulta);}
        if(code==2){
            gotoxy(36,wherey());
            printf("%f",resulta);}
        if(code==1){
            gotoxy(52,wherey());
            printf("%f",resulta);}
        if(code==0){
            gotoxy(68,wherey());
            printf("%f \r \n",resulta);}
    }
}

```

```

void ini_port(void) /*funcion para inicializar el puerto serie*/
{
    outp(0x2fb,0x80); /* Para baudaje MBS */
    outp(0x2f8,0x0c); /* Para baudaje LSB */
    outp(0x2f9,0x00);
    outp(0x2fb,0x03); /* 8-bits, 1 stop-bit, no paridad, */
    outp(0x2fc,0x0b); /* DTR y RTS comunicacion normal usando interrup.*/
    outp(0x2f9,0x0f); /* interrupciones */
    outp(0x2fd,0x60); /* inicializa el registro*/
}

```

```

void ventana(int upx, int upy, int downx, int downy)
{ int i=0;
  gotoxy(upx,upy);          /*dibuja un marco de ventana*/
  printf("%c",201);
  for (i=upx+1;!i==downx);i++)
  { gotoxy(i,upy);
    printf("%c",205);
  }
  gotoxy(downx,upy);
  printf("%c",187);
  for (i=upy+1;!i==downy);i++)
  { gotoxy(downx,i);
    printf("%c",186);
  }
  gotoxy(downx,downy);
  printf("%c",188);
  for (i=downx-1;!i==upx);i--)
  { gotoxy(i,downy);
    printf("%c",205);
  }
  gotoxy(upx,downy);
  printf("%c",200);
  for (i=downy-1;!i==upy);i--)
  { gotoxy(upx,i);
    printf("%c",186);
  }
}

void pantalla(void) /* imprime la pantalla de reporte */
{
  gotoxy(8,2);printf("V0");gotoxy(25,2);printf("V1");
  gotoxy(41,2);printf("V2");gotoxy(57,2);printf("V3");
  gotoxy(73,2);printf("V4");
  ventana(1,1,17,3);ventana(18,1,33,3);
  ventana(34,1,49,3);ventana(50,1,65,3);
  ventana(66,1,79,3);ventana(1,4,79,24);
  window(2,5,78,23);
}

```

**DIAGRAMAS DE FLUJO DE LOS PROGRAMAS PARA EL
CONTROL DE EVENTOS, ADQUISICIÓN DE DATOS Y
COMUNICACIONES**

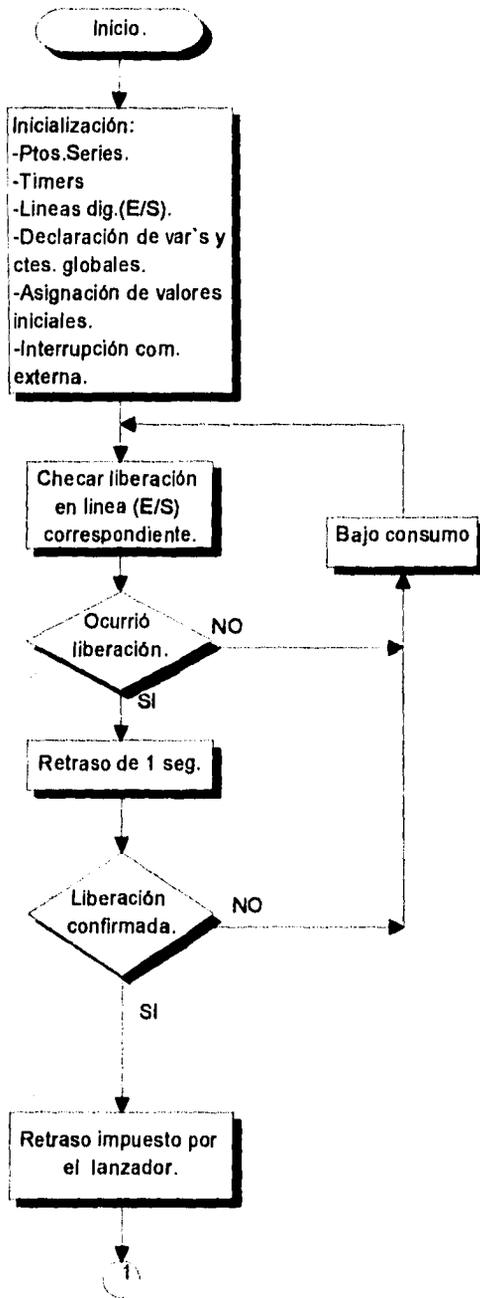


Figura 5.1 Diagrama de flujo del programa principal.

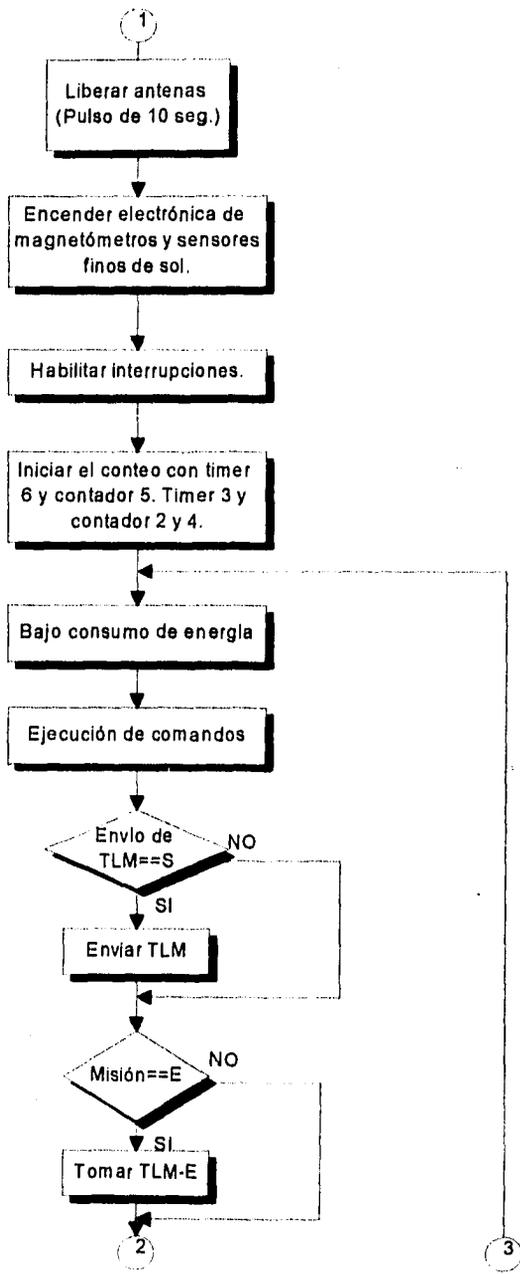


Figura 5.1 Continuación.

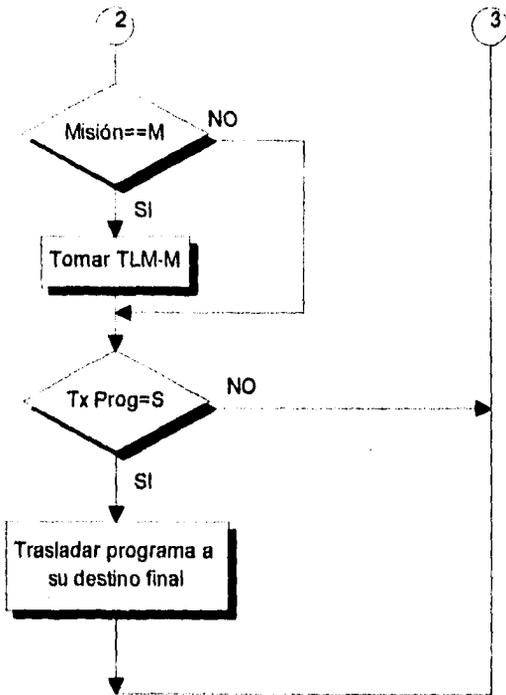


Figura 5.1 Continuación.

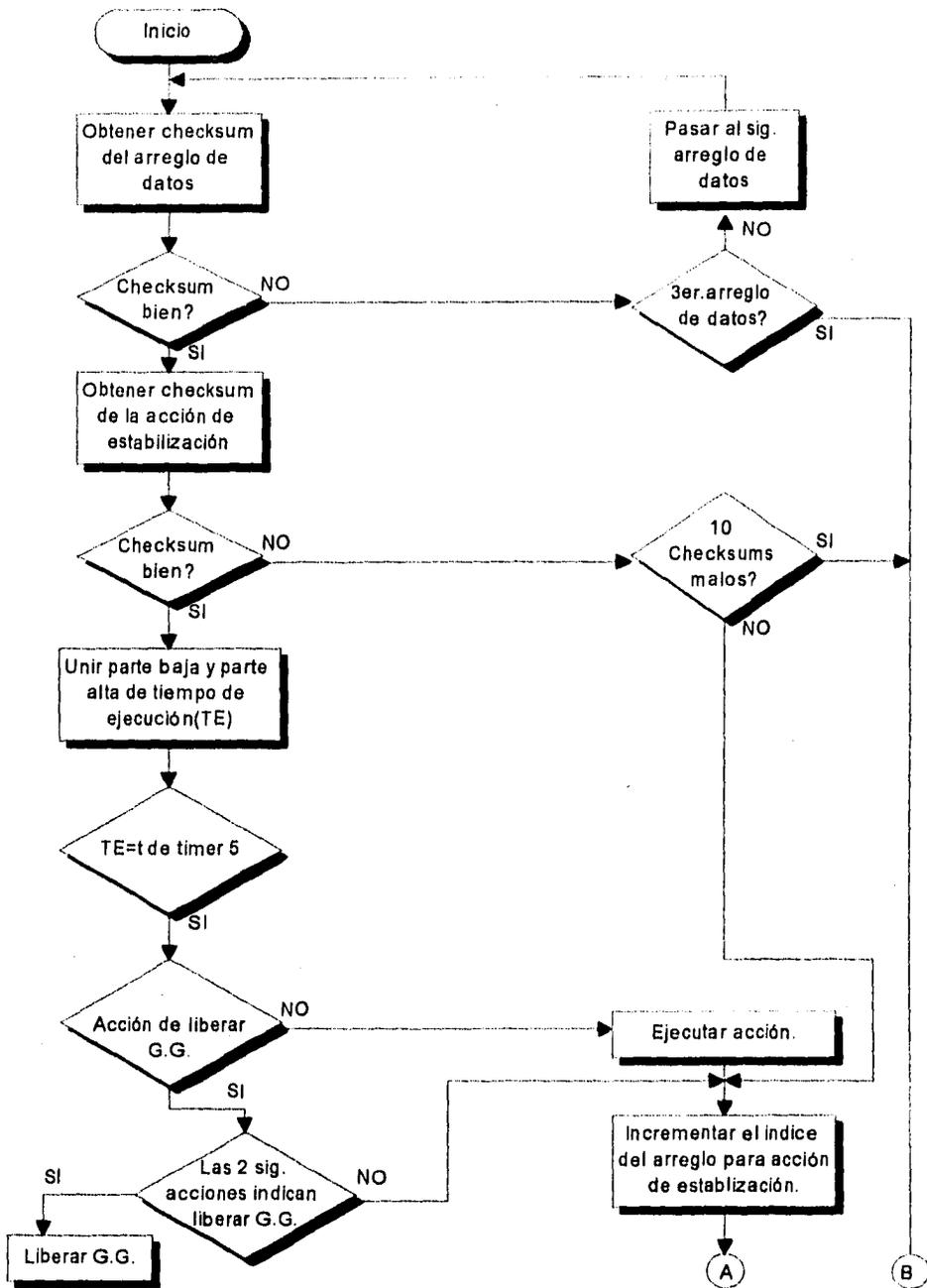


Figura 5.2 Diagrama de flujo para la Interrupción de Estabilización.

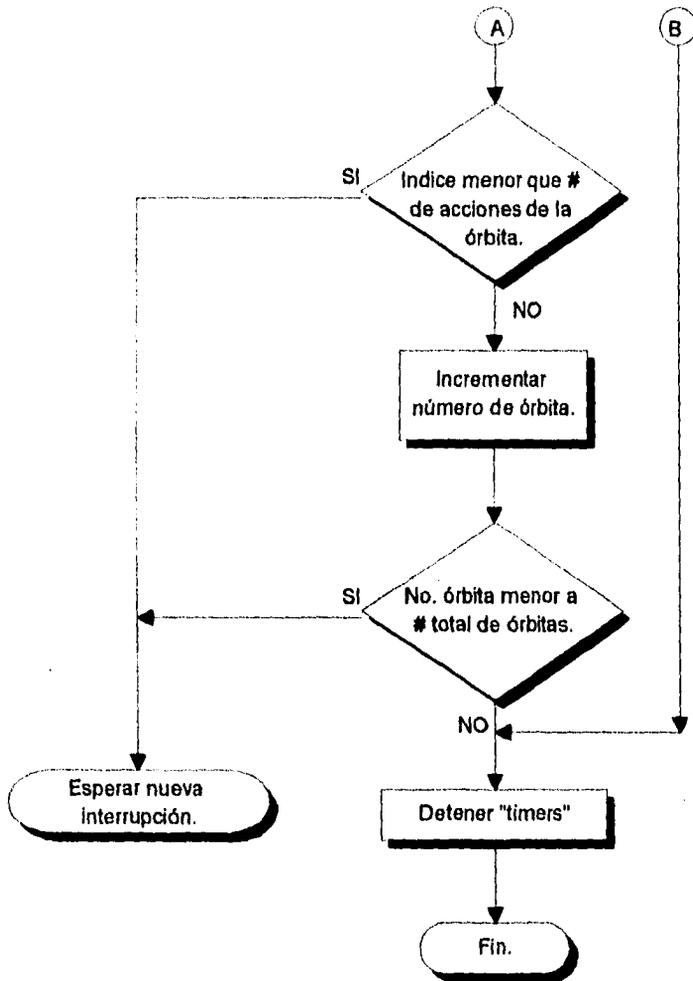


Figura 5.2 Continuación.

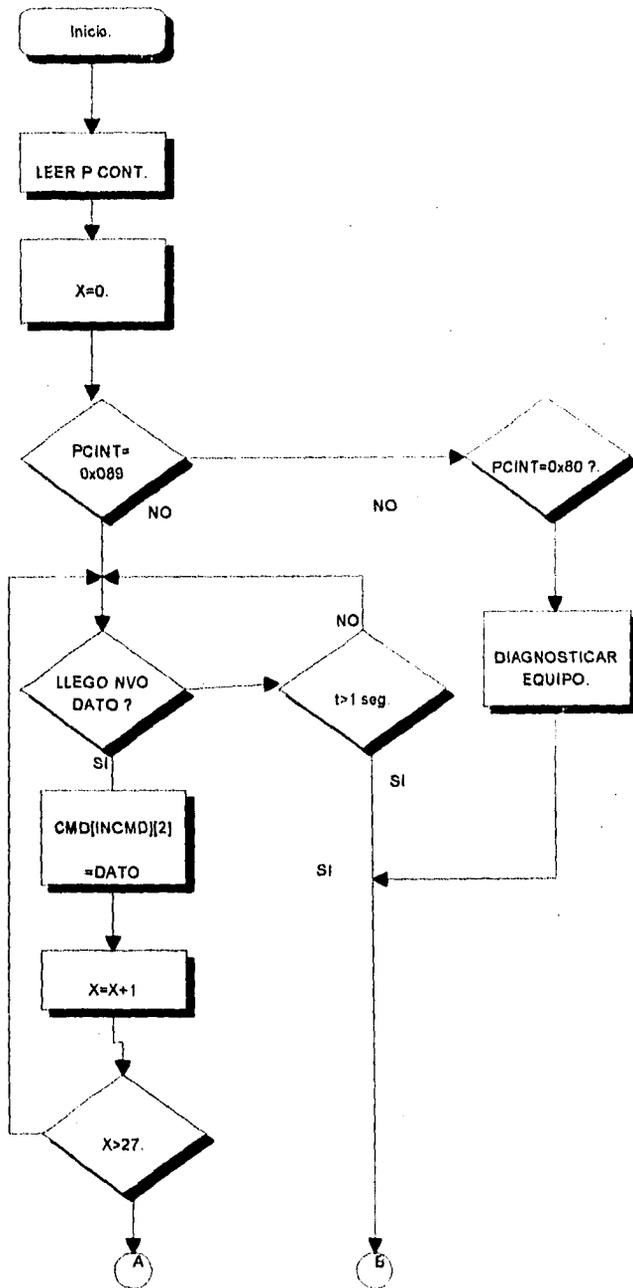


Figura 5.3 Diagrama de flujo de la Interrupción para Comunicación Interna

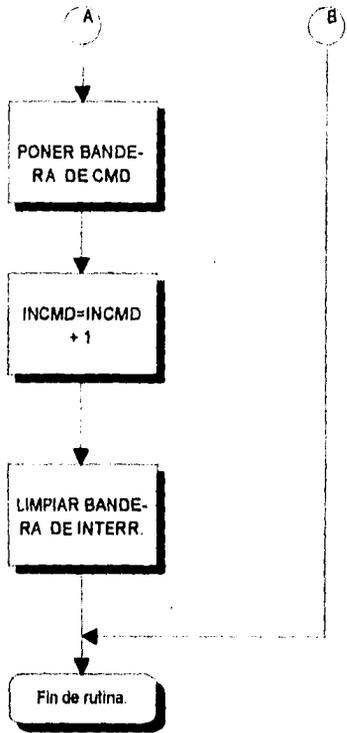


Figura 5.3 Continuación.

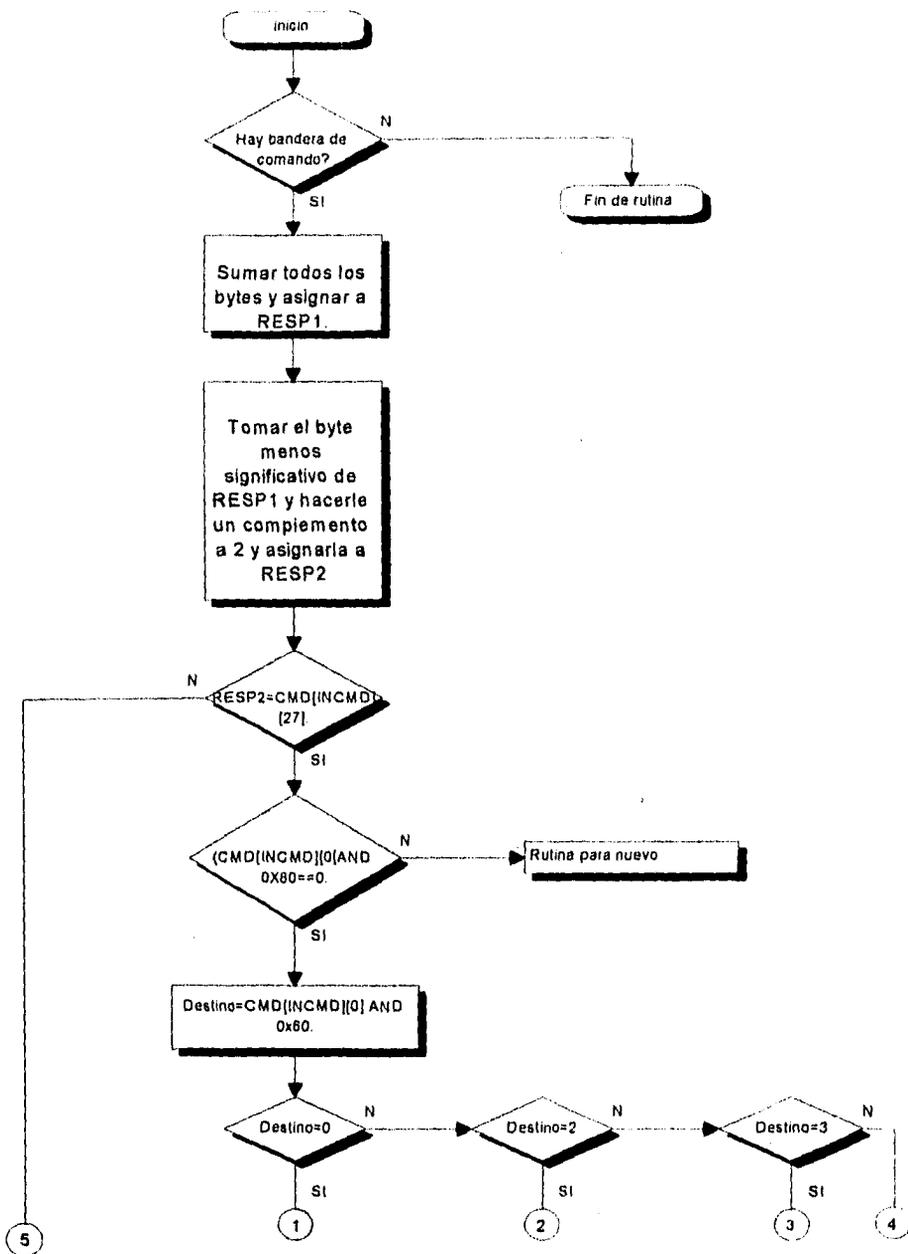


Figura 5.4 Diagrama de flujo para la Rutina de Análisis de Tramas de Comando.

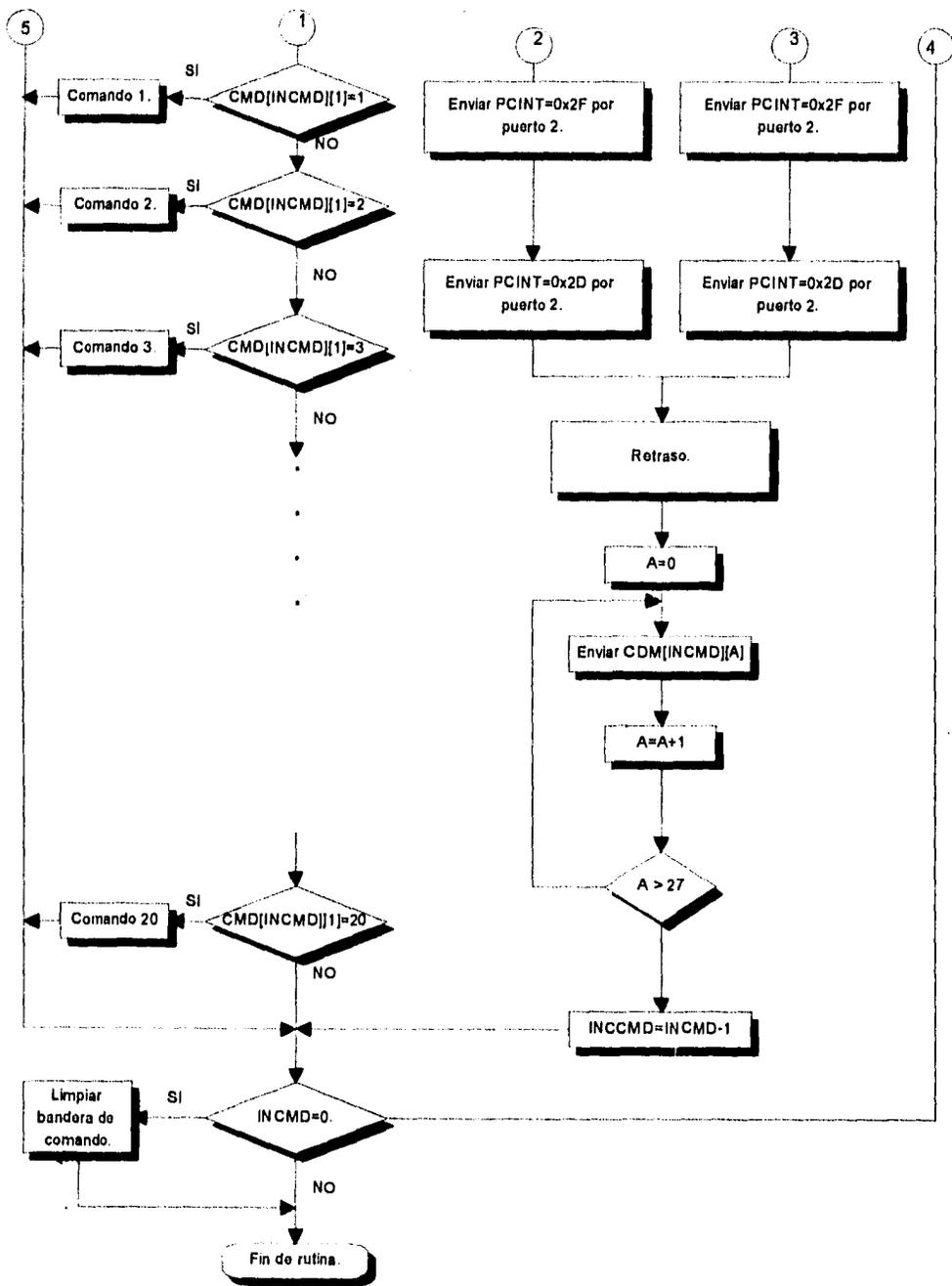


Figura 5.4 Continuación.

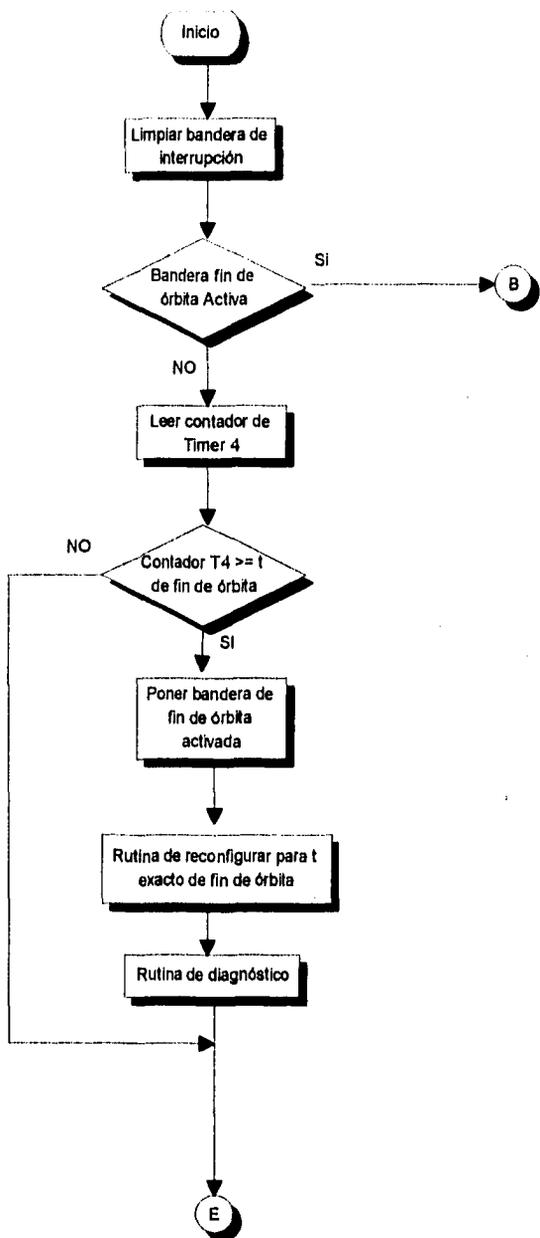


Figura 5.5 Diagrama de flujo de la interrupción para la adquisición de telemetría.

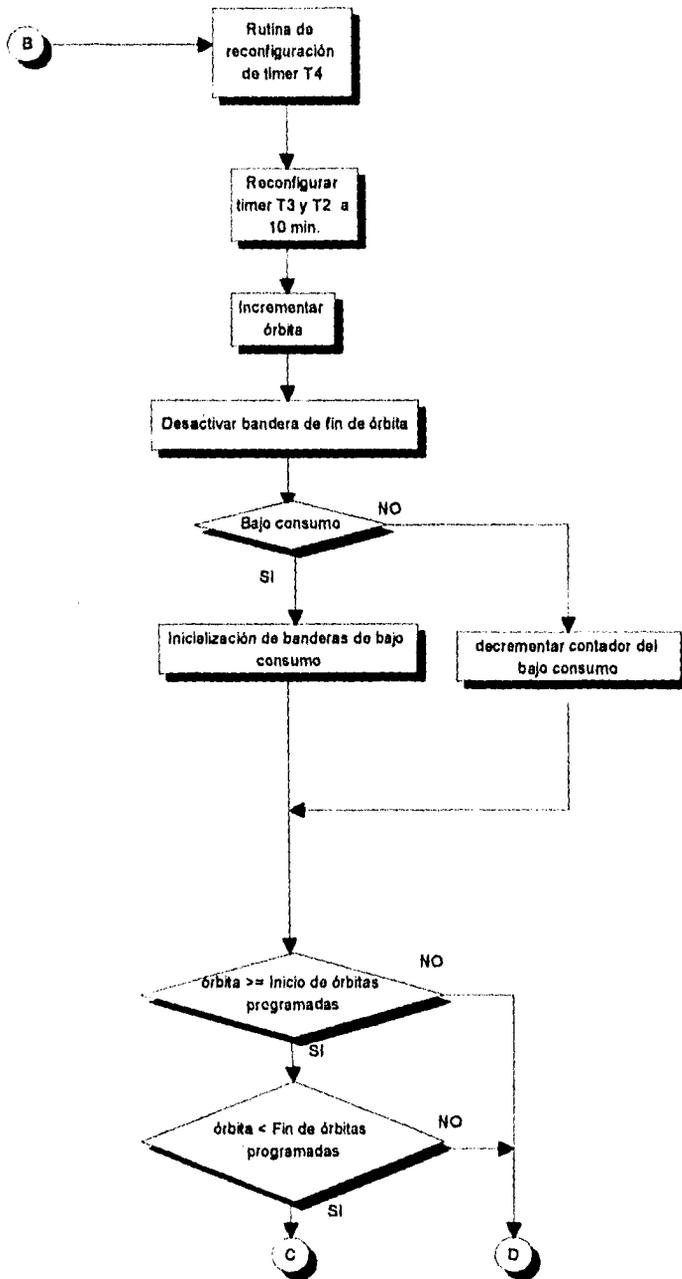


Figura 5.5 Continuación

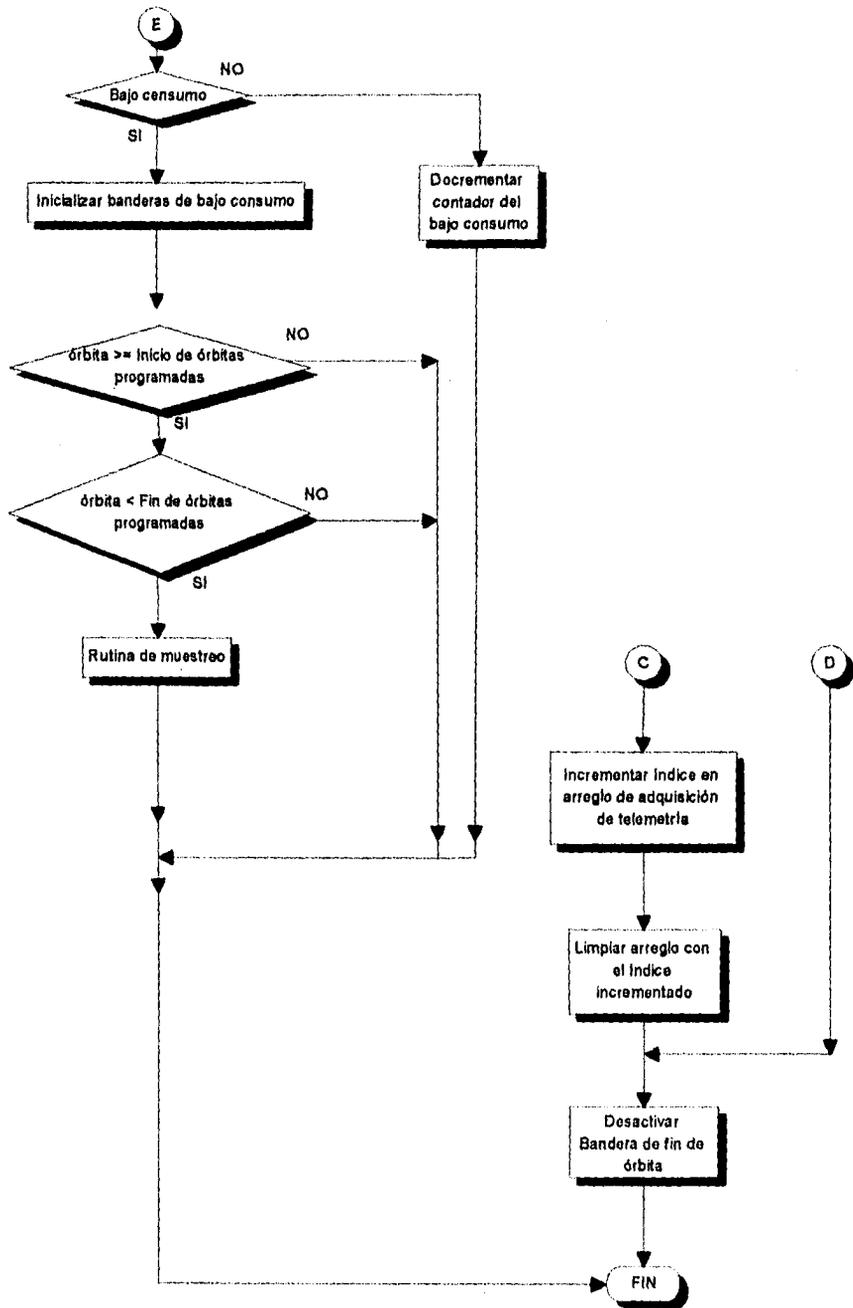


Figura 5.5 Continuación

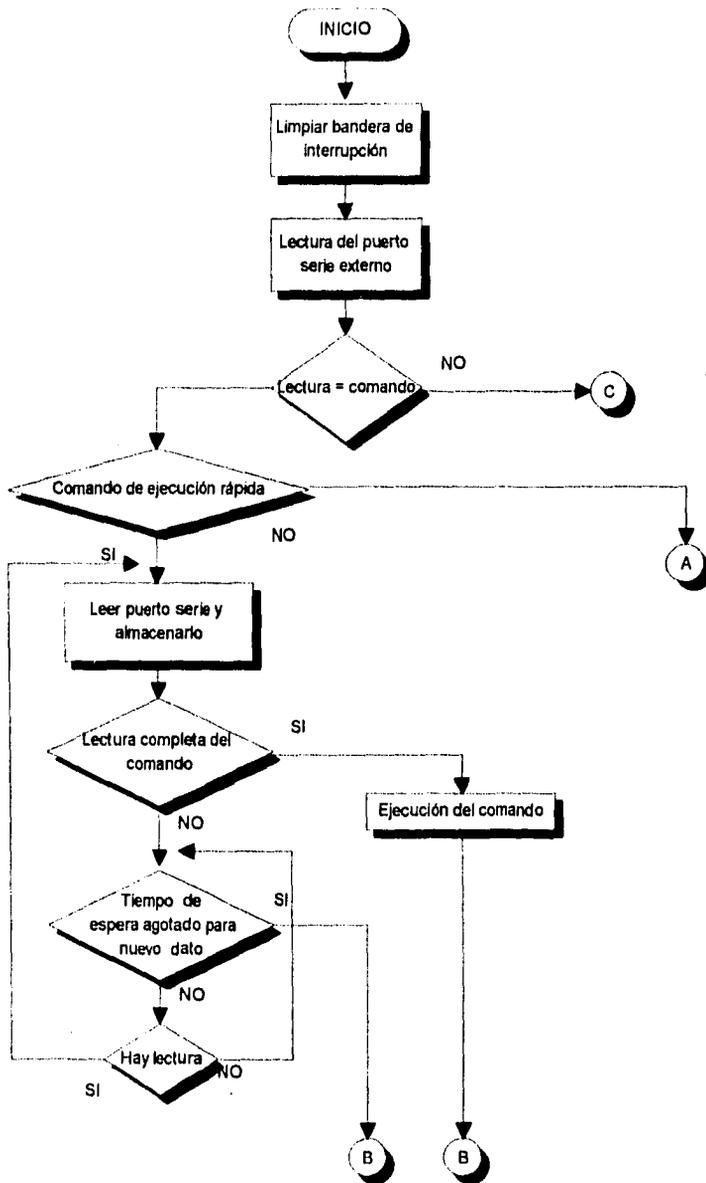
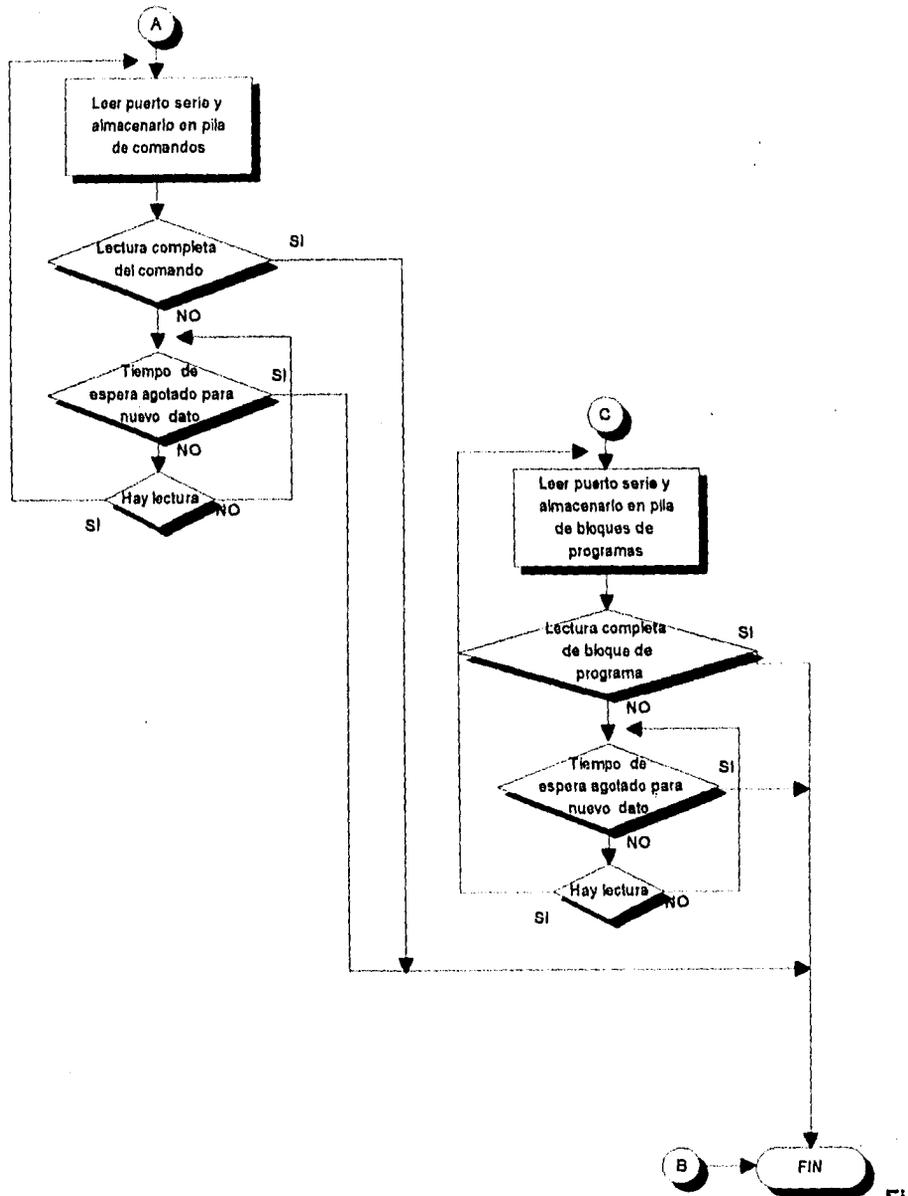


Figura 5.6 Diagrama de flujo de la Interrupción para la Comunicación Externa.



gura 5.6 Continuación