

9



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

"TRANSMISION INALAMBRICA DE DATOS DE LOS SITIOS DE UNA RED TETRAPOL A UN SERVIDOR DE APLICACION DE DATOS, USANDO EL MICROCONTROLADOR AT90S8515 Y LOS RECURSOS DE LA MISMA RED"

T E S I S
QUE PARA OBTENER EL TITULO PROFESIONAL DE
INGENIERO ELECTRICO Y ELECTRONICO
(AREA ELECTRONICA)
P R E S E N T A N :
CAROLINA CAMARGO ESPINOSA
HECTOR JACQUES FLORES

DIRECTOR DE TESIS: ING. DAVID ERNESTO ARROYO AMARO
CODIRECTORA DE TESIS: ING. BEATRIZ ESLAVA ARELLANES



MEXICO, D. F.

2002

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.

PAGINACIÓN

DISCONTINUA

Autorizo a la Dirección General de Bibliotecas de la UNAM a difundir en formato electrónico e impreso el contenido de mi trabajo académico.

NOMBRE: Camargo Espinosa
Carolina

FECHA: 25 noviembre 2002

FIRMA: [Firma manuscrita]

ESTA TESIS NO SALE
DE LA BIBLIOTECA

MINISTERIO DE CULTURA Y PATRIMONIO

AGRADECIMIENTOS

- **Al Ing. David E. Arroyo Amaro, por sus consejos y contribución al desarrollo de la presente tesis.**
 - **A la Ing. Beatriz Eslava Arellanes, por su colaboración en la culminación de este trabajo.**
 - **Al Ing. Lauro Santiago Cruz y los compañeros de la Coordinación de Instrumentación del Instituto de Ingeniería por su apoyo.**
 - **A la Facultad de Ingeniería.**
 - **A la Universidad Nacional Autónoma de México.**
-

-
- o A mi mamá por su cariño y apoyo incondicional.
 - o A mi papá por su afecto y sus invaluable consejos.
 - o A mi hermano por todo lo que hemos pasado juntos.
 - o A Elizabeth y Leticia por su amistad.
 - o A Héctor y su familia por la comprensión y apoyo en los momentos difíciles.

Carolina Camargo Espinosa

-
- ❖ A mis **padres**, su apoyo y cariño hicieron posible este trabajo.
 - ❖ A mis **hermanas**, ellas son la estructura principal y motor de mi familia.
 - ❖ A mis **abuelos**, por su ejemplo y apoyo incondicional.
 - ❖ A mis demás **familiares**, por su apoyo y presencia a lo largo de mi vida.
 - ❖ A **Carolina**, por su compañía a lo largo de mi vida universitaria y por ser parte integral de este trabajo. A su **familia**, por su apoyo y colaboración.
 - ❖ A todos los **profesores** que con sus conocimientos y consejos contribuyeron en mi formación profesional.
 - ❖ A mis **compañeros y amigos** por todos los momentos que pasamos juntos en la UNAM.

Héctor Jacques Flores

ÍNDICE GENERAL

INTRODUCCIÓN

CAPÍTULO 1 COMUNICACIONES RADIO MÓVILES E INTERCONEXIÓN DE REDES

1.1. Comunicaciones radio móviles	1
1.1.1. Formas de conexión	1
1.1.1.1. Simplex	2
1.1.1.2. Half-dúplex	2
1.1.1.3. Full-dúplex	2
1.1.1.3.1. División de frecuencia dúplex	2
1.1.1.3.2. División de tiempo dúplex	2
1.1.2. Modulación	2
1.1.2.1. Modulación analógica	3
1.1.2.2. Modulación digital	3
1.1.2.2.1. Modulación MSK	3
1.1.2.2.2. Modulación GMSK	3
1.1.2.2.3. Modulación QPSK	4
1.1.2.2.4. Modulación $\pi/4$ QPSK	4
1.1.3. Redes de radio comunicaciones móviles	4
1.1.3.1. Subsistema de terminales móviles	4
1.1.3.2. Subsistema de estaciones base	4
1.1.3.3. Subsistema de conmutación	5
1.1.3.4. Subsistema de manejo	5
1.1.4. Troncalización y técnicas de multi-acceso	5
1.1.4.1. FDMA	6
1.1.4.2. TDMA	6
1.1.4.3. CDMA	7
1.2. Interconexión de redes de comunicaciones	7
1.2.1. Conceptos de redes de computadoras	7
1.2.1.1. Clasificación de redes	7
1.2.1.2. Modelo de referencia OSI	8
1.2.1.2.1. Capas	9
1.2.1.2.2. Encapsulamiento	10
1.2.1.2.3. Tipo de servicio	10
1.2.2. Tecnologías WAN	10
1.2.2.1. Mecanismos de transporte para enlaces WAN	10
1.2.2.1.1. DS-0	11
1.2.2.1.2. T1	11
1.2.2.1.3. E1	11
1.2.2.1.4. SONET/SDH	12
1.2.2.2. Tipos de redes WAN	12
1.2.2.2.1. Redes X.25	12
1.2.2.2.2. Redes ISDN	13
1.2.2.2.3. Redes ATM	15

CAPÍTULO 2 RED TETRAPOL

2.1. Generalidades	18
2.2. Tetrapol en el mundo	20
2.3. Tetrapol en México	22
2.4. Principales características técnicas	23
2.5. Servicios	25
2.6. Infraestructura	26
2.7. Otra tecnología PAMR/PMR digital troncalizada: TETRA	27

CAPÍTULO 3 SISTEMA DE ADQUISICIÓN DE DATOS

3.1. Generalidades	29
3.2. Elementos de un SAD	29
3.2.1. Transductores	30
3.2.2. Acondicionadores de señal	30
3.2.3. Elementos de medición	30
3.2.4. Equipo de cómputo y software	33
3.3. Ruido	34
3.3.1. Inmunidad	34
3.3.2. Emisiones electromagnéticas	35
3.3.3. Ruido en la etapa de acondicionamiento de señal	36
3.4. Hardware	37
3.4.1. Transductores y acondicionamiento de señal	37
3.4.2. Elementos de medición	40

CAPÍTULO 4 PROTOCOLOS DE COMUNICACIÓN DEL MÓDULO ADQUISIDOR DE DATOS

4.1. Definiciones y abreviaciones	42
4.1.1. Abreviaciones	42
4.1.2. Definiciones	43
4.2. Arquitecturas de la UDT	43
4.3. Controlador UDT-ST	45
4.3.1. Nivel físico	45
4.3.1.1. Características generales de las señales RS232	45
4.3.1.2. Características de la comunicación	46
4.3.1.3. Identificación del tipo de UDT conectada	46
4.3.2. Protocolo MPAP	47
4.3.3. Protocolo DTAP	47
4.4. Protocolo Internet (IP)	48
4.4.1. Encabezado	48
4.4.2. El protocolo IP en el sistema Tetrapol	52

4.5. Protocolo de Datagrama de Usuario (UDP)	53
4.5.1. Encabezado	53
4.5.2. Puerto fuente UDP en Tetrapol	53
4.6. Protocolo de aplicación	54

CAPÍTULO 5 SOFTWARE DEL MÓDULO ADQUISIDOR DE DATOS

5.1. Rutina principal	57
5.2. Interrupción por comunicación	60
5.2.1. Rutina de conexión MPAP	61
5.2.2. Rutina de recepción del frame "DATA"	62
5.2.3. Rutina de recepción del frame "ACK"	62
5.2.4. Rutina de protocolo DTAP	62
5.2.5. Rutina de procesamiento de datos procedentes del DAS	65
5.2.6. Rutina de transmisión de mensajes uplink	67
5.3. Interrupción por temporizadores	68
5.3.1. Rutina de temporizador 0	69
5.3.2. Rutina de temporizador 1	70
5.4. Rutinas de muestreo	71
5.4.1. Rutina del convertidor A/D	71
5.4.2. Rutina de almacenamiento en memoria externa	73

CAPÍTULO 6 SOFTWARE DEL SERVIDOR DE APLICACIÓN DE DATOS

6.1. Herramientas del lenguaje	74
6.2. Descripción de las rutinas	75
6.2.1. Menú principal	75
6.2.2. Rutina de modificación	77
6.2.3. Rutina de configuración	79
6.2.4. Rutina de obtención de información de las memorias	82
6.2.5. Rutina de monitoreo de valores actuales	84
6.2.6. Rutina del temporizador 1	86
6.2.7. Rutina del temporizador 2	86
6.2.8. Rutina de recepción del puerto Winsock 2	87
6.2.9. Rutina de recepción del puerto Winsock 3	88

CAPÍTULO 7 INTEGRACIÓN DEL SISTEMA

7.1. Evaluación del Módulo Adquisidor de Datos	91
7.2. Evaluación del software del Servidor de Aplicación de Datos	94
7.3. Operación del sistema en conjunto	96
7.4. Interfaces de la red Tetrapol	97

CONCLUSIONES	99
---------------------	-----------

BIBLIOGRAFÍA	102
---------------------	------------

APÉNDICE A PROGRAMA DEL MÓDULO ADQUISIDOR DE DATOS

APÉNDICE B PROGRAMA DEL SERVIDOR DE APLICACIÓN DE DATOS

**APÉNDICE C HOJAS DE ESPECIFICACIONES DE COMPONENTES
UTILIZADOS**

INTRODUCCIÓN

En la actualidad los sistemas de comunicaciones radio móviles han tenido un intenso desarrollo, permitiendo transmitir no sólo voz sino también datos. Los sistemas Radio Móviles Profesionales o Privados, no han sido la excepción. Desde finales de la década de los 80's surgieron varias tecnologías Profesionales digitales con características troncalizadas (*trunking*). Una de estas tecnologías es Tetrapol, desarrollada en Europa por la compañía francesa MATRA, operacional en México desde hace un par de años.

Una parte fundamental de las redes Tetrapol, como en todos los sistemas radio móviles, la constituyen los sitios de repetición. De ahí la importancia de monitorear las variables que garantizan la ininterrupción del servicio; tales como son el nivel de combustible del generador que provee de energía al sitio cuando el suministro de la red de distribución eléctrica falla y la temperatura que sirve para detectar cualquier fallo en el sistema de aire acondicionado. Además del monitoreo de estas variables se requiere un sistema que sea capaz de transmitir dicha información a un centro de monitoreo remoto, encargado de recibir los datos de varios sitios a la vez.

Existen diversas opciones para transmitir la información concerniente a las variables antes mencionadas, por ejemplo, mediante la red telefónica, la Telefonía Móvil Pública, o vía satélite. Sin embargo, la primera está restringida a la disponibilidad de una línea telefónica, la segunda al área de cobertura, que en diversas áreas rurales es inexistente, y la principal desventaja de usar un satélite es el costo. Es así, que resulta más conveniente aprovechar los recursos de la misma red. Por ello, el objetivo de la presente tesis, es desarrollar un sistema electrónico que mida las variables previamente señaladas, y utilice los servicios de transmisión de datos ofrecidos por la red Tetrapol para transmitir de manera inalámbrica las variables a un Servidor de Aplicación de Datos.

La presente tesis consta de siete capítulos. Los primeros cuatro se enfocados a plantear el marco teórico necesario para comprender plenamente los capítulos cinco a siete, en los que se explica la programación realizada tanto en el Módulo Adquisidor de Datos, como en el Servidor de Aplicación de Datos y finalmente la integración del sistema.

En el capítulo uno se presentan, además de algunos conceptos de las comunicaciones radio móviles, como la definición, formas de conexión, esquemas de modulación más utilizados por los sistemas radio móviles, componentes de las redes de radio comunicación, troncalización y técnicas de multiacceso, también se describen conceptos referentes a la interconexión de redes de computadoras, como el modelo

OSI, mecanismos de transporte que permiten enlazar LAN's, WAN's y redes radio móviles, y finalmente se muestran las principales tecnologías WAN.

En el capítulo dos se presenta un panorama general de la tecnología Tetrapol, cuyo objetivo es dar a conocer al lector la situación actual de estas redes tanto en el mundo como en México, los servicios que ofrece esta tecnología y algunos aspectos técnicos que la caracterizan.

En el capítulo tres se mencionan las principales características de un sistema adquisidor de datos en función de sus componentes, se describen los elementos que fueron seleccionados para realizarlo, y se presentan los conceptos de ruido que se tomaron en consideración para el desarrollo del sistema.

En el capítulo cuatro se describe la arquitectura de comunicación requerida para llevar a cabo la transferencia de información entre el Módulo Adquisidor de Datos y el Servidor de Aplicación de Datos.

En el capítulo cinco se muestra la programación realizada para el Módulo Adquisidor de Datos, que incluye rutinas dedicadas a adquisición de datos, almacenamiento en memoria externa, desplegado y transferencia de datos.

En el capítulo seis se expone el software desarrollado en Visual Basic para el Servidor de Aplicación de Datos, que representa la interfaz con el usuario.

En el capítulo siete se describen las pruebas realizadas a los elementos que constituyen al sistema tanto en forma individual como en su conjunto.

Finalmente, aparecen las conclusiones, la bibliografía, y apéndices. Los apéndices incluyen el listado del software desarrollado y las hojas de especificaciones de los principales componentes utilizados.

CAPÍTULO 1

COMUNICACIONES RADIO MÓVILES E INTERCONEXIÓN DE REDES

En este capítulo se exponen, tanto conceptos de las comunicaciones radio móviles, como de la interconexión de redes de computadoras, útiles en la comprensión del funcionamiento general de una red Tetrapol y sus pasarelas de conexión con sistemas externos.

1.1 COMUNICACIONES RADIO MÓVILES

Las comunicaciones radio móviles se definen donde existe una radio conexión entre una estación de comunicación móvil y una estación de comunicación fija o bien entre estaciones móviles. A la estación de comunicación fija se le conoce como estación base, y a las estaciones móviles como radio terminales móviles. Las estaciones de comunicación pueden ser de transmisión, recepción o transmisión/recepción. La cobertura es el área geográfica dentro de la que las radio terminales móviles pueden establecer comunicación ya sea con la estación fija o entre sí.

1.1.1 Formas de conexión

Los sistemas de comunicaciones radio móviles pueden diseñarse para transmitir en un sentido, en ambos sentidos sólo una a la vez o en ambos sentidos simultáneamente. Según estos criterios se definen tres formas de conexión:

- Símplex
- Half-dúplex
- Full-dúplex

1.1.1.1 Símplex

En este tipo de conexión unidireccional, las transmisiones únicamente pueden ocurrir en un sentido, por lo que en una sesión de comunicación una estación está dedicada a la transmisión y la otra a la recepción. Un servicio de comunicación radio móvil con este tipo de conexión es el conocido como "paging". Este servicio consiste en alertar a alguien que lleve consigo un pequeño dispositivo receptor, haciendo una llamada a un número telefónico específico desde cualquier red telefónica.

1.1.1.2 Half-dúplex

Es una conexión unidireccional que permite la comunicación entre dos equipos en ambos sentidos, pero no admite que el tráfico de entrada y el de salida se lleven a cabo simultáneamente, ya sea que ambos compartan una sola frecuencia, o que cada tráfico utilice una frecuencia diferente. El sentido de la comunicación se determina cuando se pulsa el comando "pulsar para hablar" (PTT, pulse to talk). La función de este comando es conectar el transmisor a la antena y desconectar el receptor de ésta.

1.1.1.3 Full-dúplex

Es una conexión bidireccional que permite las comunicaciones simultáneas en ambas direcciones. Este es el caso de la telefonía móvil. Existen dos técnicas para realizar la conexión full-dúplex: división de frecuencia dúplex (FDD) y división de tiempo dúplex (TDD).

1.1.1.3.1 División de frecuencia dúplex (FDD)

Esta técnica utiliza dos frecuencias distintas por cada sesión, una frecuencia es usada para transmitir información y la otra para recibirla.

1.1.1.3.2 División de tiempo dúplex (TDD)

Esta técnica emplea una frecuencia tanto para transmitir como para recibir información, lo que se alterna es el tiempo en que recibe y en el que transmite, aunque estrictamente no se recibe y transmite simultáneamente, si las terminales móviles y su correspondiente estación base no están muy alejadas, el efecto es imperceptible. Sin embargo, cuando hay grandes distancias involucradas, existen retrasos considerables.

1.1.2 Modulación

Muchas veces las señales eléctricas generadas por diferentes fuentes de información no son adecuadas para la transmisión directa a través de un canal determinado. Para facilitar su transmisión estas señales son modificadas o moduladas. La modulación es la alteración de una onda portadora para representar otra señal (moduladora), que constituye el mensaje. Por el tipo de fuente de información (señal moduladora), existen dos tipos de modulación: analógica y digital.

1.1.2.1 Modulación analógica

Las principales técnicas de modulación analógica son: en amplitud (AM), en frecuencia (FM) y en fase (PM). La modulación en amplitud (AM), consiste en variar la amplitud de una onda portadora, que es una señal senoidal de alta frecuencia, en función de la tensión moduladora. En la modulación en frecuencia (FM), se varía la frecuencia de una onda portadora proporcionalmente a la señal moduladora. En este tipo de modulación, la amplitud de la portadora permanece constante. En la modulación en fase (PM), se modifica la fase de la señal portadora en más o menos grados, en función de la tensión moduladora. En este tipo de modulación, tanto la amplitud como la frecuencia de la portadora son constantes.

1.1.2.2 Modulación digital

En la actualidad, la tendencia de los sistemas de comunicaciones móviles es a utilizar la modulación digital, esto se debe principalmente a que la modulación digital ofrece, entre otras ventajas sobre la modulación analógica, mejor inmunidad al ruido, optimización para la multicanalización y para diferentes tipos de información (voz, datos y video) y mayor seguridad. Las principales técnicas de modulación digital utilizadas en comunicaciones móviles son: MSK (Minimum Shift Keying), GMSK (Gaussian Minimum Shift Keying), QPSK (Quadrature Phase Shift Keying) y $\pi/4$ QPSK.¹

1.1.2.2.1 Modulación MSK

En este tipo de modulación se utilizan dos frecuencias distintas para representar la información binaria a transmitir, donde regularmente $f_2=2f_1$. El método consiste en separar los bits de datos en dos flujos, uno de pares y otro de nones, y compararlos. Si los bits par e impar son '0', entonces se transmite la frecuencia más alta invertida, usualmente 180° . Si el bit par es '1' y el impar '0', entonces se transmite la frecuencia baja invertida. Si el bit par es '0' y el impar es '1', se transmite la frecuencia baja sin cambio de fase. Finalmente, si ambos bits son '1' la frecuencia más alta se transmite sin cambio de fase.

1.1.2.2.2 Modulación GMSK

Este tipo de modulación es una variación de la modulación MSK descrita anteriormente. La diferencia radica en que la modulación GMSK utiliza un filtro gaussiano con un determinado ancho de banda, previo a la etapa de modulación. Este filtro provee un espectro de potencia de salida más compacto que el obtenido de la modulación MSK, y ofrece diversas propiedades requeridas para suprimir los componentes de alta frecuencia, tales como un ancho de banda reducido y un corte preciso.

¹ GMSK: GSM, DECT, Tetrapol;
 $\pi/4$ QPSK: TETRA, D-AMPS(E.E.U.U.A.), JDC(Japón);
QPSK-C: APCO Project 25.

1.1.2.2.3 Modulación QPSK

En este tipo de modulación, la fase de la portadora toma uno de los cuatro valores equidistantes como 0, $\pi/2$, π , y $3\pi/2$, donde cada valor de la fase corresponde a un par único de bits del mensaje. Por ejemplo: 11 está a 45°, 10 está a 135°, 00 está a 225° y 01 está a 315°, por lo que un cambio de fase de 90° causa únicamente un bit de cambio.

1.1.2.2.4 Modulación $\pi/4$ QPSK

Este tipo de modulación es una mejora de la modulación QPSK, porque proporciona el máximo número de transiciones de fase. Las transiciones de la fase de la portadora pueden tomar uno de ocho valores, múltiplos de $\pi/4$ radianes. En $\pi/4$ QPSK el cambio máximo de fase es de +/- 135° mientras que para QPSK es de 180°. Por lo tanto una señal $\pi/4$ QPSK limitada en banda preserva mejor la propiedad de envoltura constante que una señal QPSK limitada en banda.

1.1.3 Redes de radio comunicaciones móviles

Una red de radio comunicaciones móviles es un conjunto de elementos que funcionan bajo los mismos estándares y permiten a un usuario establecer comunicación con otro, vía una interfaz de radio. En general se componen de los siguientes subsistemas:

- Estaciones móviles o terminales móviles
- Estaciones base
- Conmutación (Switching)
- Manejo y operación (Management)

1.1.3.1 Subsistema de Terminales Móviles

Las terminales móviles son los dispositivos de cada usuario de una red que por lo general tienen una identificación almacenada ya sea en el dispositivo o bien en una tarjeta inteligente "smartcard" deslizable en el dispositivo. Existen tres tipos de terminales móviles: las terminales con una interfaz humana, por ejemplo con un micrófono, un pequeño teclado y un display; las terminales que sólo tienen conexión para un dispositivo externo de entrada/salida, por ejemplo con conexión para PC o fax; y por último las terminales que cuentan con ambas posibilidades.

1.1.3.2 Subsistema de Estaciones Base

Una estación base en general, es un módem de radio multicanal y una antena. Existe una estación base por cada célula, que es una pequeña sección del área de cobertura. Las terminales móviles que se encuentran dentro de una célula, se comunican directamente con la estación base, que sirve como una estación

retransmisora de alta potencia. Existen dos tipos de estaciones base: las estaciones base no inteligentes llamadas estaciones base transceptoras (BTS), que son las que se ubican en cada célula, y las estaciones base de control (BCS), que controlan remotamente, mediante líneas telefónicas o radioenlaces a varias BTS. El principal objetivo de las estaciones base de control, es el manejo de frecuencias usadas por todos los transceptores conectados, y la coordinación de las BTS para evitar interrupciones en las sesiones de comunicación por el cambio de célula (handover).

1.1.3.3 Subsistema de conmutación

El subsistema de estaciones base se conecta a este subsistema, que se compone del centro de conmutación móvil, MSC (*mobile switching center*), que es el encargado de proporcionar toda la capacidad de conmutación. Aquí se manejan dos principales problemas originados por la movilidad: localización de una terminal móvil, y comunicaciones vía redes anfitrión (*roaming*). El subsistema de conmutación contiene dos registros o bases de datos: una para almacenar información de sus propios subscriptores y otra dedicada a los datos de los huéspedes que se guardan temporalmente en la red. Estos registros se actualizan cada que una terminal móvil se mueve y ocasiona conmutación.

Un MSC normalmente es suficiente para un área de cobertura de alrededor de 1 millón de habitantes².

1.1.3.4 Subsistema de manejo

Este subsistema se encarga de las siguientes áreas funcionales: manejo de fallas, manejo de costos, manejo de la configuración, manejo de operación y manejo de la seguridad. Para realizar estas tareas, este subsistema se conecta directa o indirectamente con todos los otros subsistemas de la red.

1.1.4 Troncalización y técnicas de multi-acceso

Un requerimiento general para las redes de radio comunicaciones móviles es que varias sesiones de comunicación puedan ocurrir simultáneamente entre varias terminales móviles y una estación base. Debido a que los canales de comunicación comúnmente se encuentran restringidos, es necesario que un número elevado de usuarios potenciales comparta un volumen limitado de recursos. Esto es posible ya que estadísticamente se ha comprobado que aunque el número de usuarios sea elevado, únicamente una fracción reducida de ese número está activa ocupando los recursos en un momento determinado (principio de concentración de enlaces)³. El acceso automático de los usuarios a cualquiera de los recursos, se denomina multi-acceso. Para ello, se requiere del uso de protocolos de señalización⁴ eficaces que regulen los

² "Mobile telecommunications: standards, regulation and applications", Bekkers Rudi, 1998; pag. 112.

³ Para ver el análisis estadístico de este principio consúltese el libro: "Ingeniería de Sistemas Trunking", José María Hernando Rábanos, pp. 35 – 37.

procesos de asignación y liberación de canales. En sus orígenes, los sistemas radio móviles funcionaron con base en la asignación rígida de canales, empleándose la conmutación manual para cambiar de canal. A diferencia de estos sistemas, aquellos que utilizan alguna técnica de multi-acceso, se denominan troncales (*trunking*).

Los tres principales mecanismos de multi-acceso son:

- Acceso Múltiple por División de Frecuencia (FDMA)
- Acceso Múltiple por División de Tiempo (TDMA)
- Acceso Múltiple por División de Código (CDMA)

1.1.4.1 FDMA

En la técnica FDMA, el ancho de banda total disponible con que cuenta una estación base, se divide en radio canales. Cada radio canal se asigna a una sola sesión de comunicación. Las asignaciones son de banda estrecha, del tipo un sólo canal por portadora (SCPC, *Single Channel per Carrier*). Las sesiones, cada una en su radio canal, pueden efectuarse simultáneamente e ininterrumpidamente, en las diferentes frecuencias. Cada receptor o grupo de receptores selecciona, mediante un filtro sintonizable, el radio canal deseado.

1.1.4.2 TDMA

En la técnica TDMA, se asigna a todas las sesiones, una misma frecuencia durante breves intervalos de tiempo de forma periódica, conocidos como *time slots* (TS), de manera que, las sesiones se efectúan simultánea, pero discontinuamente en una sola portadora (multiplexaje). Esta técnica involucra un sistema complejo de temporización y se debe considerar que todo aumento en la capacidad de TDMA (mayor número de sesiones) incrementa proporcionalmente la velocidad de transmisión y, por consiguiente, el ancho de banda necesario de la portadora TDMA. Por esta razón existe un límite al número de canales que se pueden sustentar, que comúnmente es 10.⁵ En la fig. 1.1 se muestra el esquema general de un paquete TDMA.

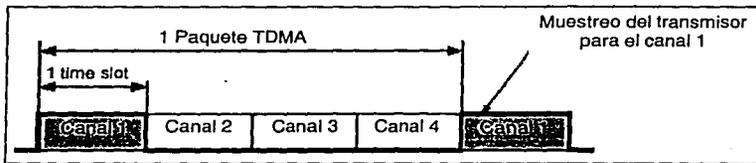


Figura 1.1 Paquete TDMA.

⁴ De todos los canales con los que cuenta una BS, por lo menos uno se destina para que la estación base transmita la información del sistema a las estaciones móviles. Los canales restantes son canales de comunicación y son usados para transmitir voz y datos.

⁵ Fuente: "Comunicaciones móviles", José María Hernando Rábanos, pag. 61.

1.1.4.3 CDMA

En esta técnica varias sesiones de comunicación se llevan a cabo simultánea y continuamente en una sola frecuencia portadora. Esto es posible si para cada canal, la estación base genera un código único que cambia con cada conexión. La estación base, agrupa todas las transmisiones codificadas de los subscriptores en una sola portadora. La estación móvil genera su propio código mediante algoritmos previamente establecidos y lo utiliza para extraer las señales apropiadas. La calidad de la conexión decrece gradualmente con el número de usuarios concurrentes. Esto puede reflejarse en un incremento en el número de errores.

1.2 INTERCONEXIÓN DE REDES DE COMUNICACIONES

En la actualidad la mayoría de las redes de comunicaciones móviles ofrecen a sus usuarios los servicios necesarios para establecer enlaces con usuarios de otras redes de comunicaciones móviles. Además la tendencia es a ofrecer equipos más sofisticados que además de voz, sean capaces de transmitir datos y crear conexiones con computadoras remotas. Por ello muchas redes de comunicaciones móviles cuentan con un centro de conmutación móvil MSC capaz de coordinar todos los enlaces externos, ya sea con otras radio terminales, teléfonos fijos o computadoras.

1.2.1 Conceptos de redes de computadoras

Una red de computadoras es un conjunto de dispositivos, como computadoras, impresoras, escáners, ruteadores, etc. que utilizan un protocolo común de comunicaciones con la finalidad de compartir recursos.

En términos generales, un protocolo es un conjunto aceptado de procedimientos, reglas o especificaciones formales que gobiernan el comportamiento bajo ciertas circunstancias. Cuando se habla de un protocolo de comunicaciones, éste se refiere al acuerdo que especifica el formato y el significado de los mensajes intercambiados por los dispositivos involucrados.

1.2.1.1 Clasificación de redes

Según el número de computadoras implicadas en una red y su separación física, se distinguen principalmente dos tipos de redes:

- LAN
- WAN

Redes de área local LAN

Una red LAN es un sistema de comunicación de datos que permite a un número de dispositivos independientes comunicarse directamente entre ellos, dentro de un área geográfica restringida (la IEEE se refiere a esta área con un radio de 10 [km] o menos), por lo que no requiere de instalaciones externas para conectar sus equipos.

Redes de amplia cobertura WAN

Una red WAN es un sistema de comunicación de datos que tiene una cobertura geográfica relativamente grande (la IEEE establece un radio mayor a 100 [km]), debido a que la constituyen dos o más redes LAN interconectadas mediante las instalaciones de transmisión que ofrecen compañías de servicios de telecomunicaciones públicas como las telefónicas.

La necesidad de interconexión de varias redes LAN surgió primero por el deseo de científicos de compartir información, y luego se sumó la necesidad de organizaciones para intercambiar información entre oficinas esparcidas hasta en diferentes países.

Uno de los retos principales al conectar varias redes es soportar la comunicación entre tecnologías diferentes ya que, por ejemplo, varios lugares pueden utilizar diferentes medios de transmisión, o bien operar a velocidades variables. En 1974 la ISO (Organización Internacional de Estándares) desarrolló el modelo de referencia OSI (*Open Systems Interconnections*) para facilitar tanto la comprensión del complejo funcionamiento de una red como la interconexión de varias redes heterogéneas.

1.2.1.2 Modelo de Referencia OSI

El modelo de referencia OSI divide las funciones implicadas en la transferencia de información, en siete partes denominadas capas, cada una se ocupa de tareas más pequeñas y fáciles de manejar, por lo que diseñar un protocolo independiente para cada capa se simplifica. Las siete capas que el modelo de referencia OSI utiliza se muestran en la Fig. 1.2.

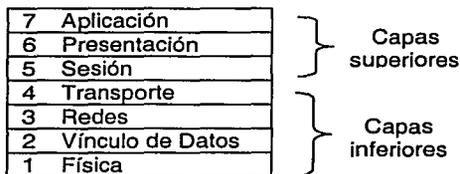


Figura 1.2 Modelo de referencia OSI.

Las capas superiores son responsables de la presentación de la interfaz al usuario, y en general sólo están implementadas en software.

Las cuatro capas inferiores tienen que ver con la transferencia de datos, y se ocupan del empaque, enrutamiento, verificación y transmisión de cada grupo de datos. Estas capas no se ocupan del tipo de datos que transmiten o reciben y no hacen ninguna diferencia entre distintas aplicaciones.

1.2.1.2.1 Capas

Capa de aplicación

La capa de aplicación es la interfaz del sistema OSI con el usuario final. Su función es desplegar la información recibida y enviar los datos introducidos por el usuario a las capas inferiores.

Capa de presentación

La tarea de la capa de presentación es aislar las capas inferiores del formato de datos de la aplicación. Es decir, convierte los datos de la aplicación a un formato común, útil para las capas inferiores. La capa de presentación hace lo inverso para datos de llegada. Estos son convertidos del formato común a los formatos específicos de la aplicación.

Capa de sesión

La capa de sesión es una capa de control de flujo y de sincronización. La capa de sesión está involucrada en la coordinación de las comunicaciones entre diferentes procesos de aplicación, y le permite a cada uno el conocimiento del estado del otro.

Capa de transporte

La capa de transporte proporciona la transferencia de datos del extremo fuente de un sistema al extremo destino de otro sistema. En algunos casos la capa de transporte es responsable de comprobar que los datos enviados coincidan con los recibidos. Este papel es importante para asegurarse de que los datos han sido enviados correctamente, y para hacer un reenvío si se detecta algún error. La capa de transporte administra el envío de datos, y determina su orden y prioridad.

Capa de red

La capa de red proporciona el enrutamiento físico de los datos, examinando la topología de la red y determinando cuál es el mejor camino para el envío de un mensaje.

Capa de vínculo de datos

La capa de vínculo de datos proporciona el control de la capa física, detecta y posiblemente corrige errores que pudieran ocurrir en la transmisión. La capa de vínculo de datos está por lo general ocupada en detectar interferencias de señales en los medios físicos de transmisión, ya sean conductores de cobre, fibra óptica o microondas.

Capa física

La capa física se ocupa de los medios mecánicos, eléctricos, funcionales y de procedimientos que se requieren para la transmisión de los datos, es decir que se refiere al cableado u otra forma de transmisión.

1.2.1.2.2 Encapsulamiento

Por lo general cada capa se comunica con otras tres capas del modelo OSI: la capa inmediata superior, la inmediata inferior y su capa equivalente en otra máquina. Las siete capas del modelo OSI utilizan varios tipos de información de control para comunicarse con sus capas equivalentes en otros sistemas. Cada capa de la computadora transmisora generalmente añade esta información en el encabezado y/o finalizador antes de enviar los datos a una capa inferior y la misma capa de la computadora receptora usa la información adicional para procesar los datos recibidos y la retira. Con lo anterior cualquier capa de la computadora destino recibe el mismo mensaje enviado por la misma capa del equipo fuente.

1.2.1.2.3 Tipo de servicio

El protocolo de cada capa presta un tipo de servicio, que puede ser orientado a conexión o no orientado a conexión. El primero establece una conexión ya sea física o virtual antes de realizar la transferencia de información, que garantiza una transferencia confiable. El segundo consiste en dividir un mensaje en paquetes, cada uno es independiente y puede seguir rutas diferentes para arribar a su destino, por lo que no precisa establecer ninguna conexión física o virtual antes de enviar la información. Por conexión física se entiende un enlace privado que siempre está dedicado a la conectividad de dos terminales. La conexión virtual se refiere a enlaces compartidos entre varias terminales. Existen dos tipos de conexiones virtuales: permanentes (PVC) y conmutadas (SVC). Las PVC establecen una conexión lógica permanente para conectar dos terminales a través de un medio compartido. Las SVC son conexiones que se establecen y terminan según la necesidad de comunicaciones.

1.2.2 Tecnologías WAN

Las tecnologías WAN se clasifican en tecnologías de conmutación de circuitos y conmutación de paquetes. En las primeras, una conexión fija se establece entre los nodos fuente y destino antes del envío de la información, cada paquete toma la misma ruta, y todos los paquetes arriban en orden. En las segundas, las conexiones se establecen durante la transmisión, los paquetes no necesariamente siguen la misma ruta, y pueden llegar en desorden.

1.2.2.1 Mecanismos de transporte para enlaces WAN

Para entender como funcionan las redes WAN, es necesario mencionar primero los mecanismos de transporte que proporcionan las compañías de servicios para establecer enlaces WAN. Los mecanismos más utilizados son: DS-0, T1 en Estados Unidos y Japón, E1 en Europa y México y SONET/SDH.

1.2.2.1.1 DS-0 (Digital Signal at level 0)

DS-0 es el término utilizado para definir el ancho de banda necesario de un canal digital sencillo para transmitir una señal analógica de voz. Dicho ancho de banda es de 64 kbps. El origen de este valor se remonta a los inicios de las redes telefónicas analógicas, que fueron diseñadas para transmitir únicamente señales de voz. El rango de frecuencias de la voz humana es desde 300 Hz hasta 3300 Hz, que da una diferencia de 3000 Hz, que es el ancho de banda mínimo para transmitir voz. Sin embargo, en la práctica el ancho de banda utilizado es de 4000 Hz con filtros a 300 Hz y 3300 Hz. Para transmitir datos fue necesario realizar la conversión A/D y viceversa. La conversión A/D involucra dos pasos: muestreo y codificación. De acuerdo al Teorema de Nyquist, el muestreo de la señal analógica debe realizarse al doble de la máxima frecuencia contenida en la señal. Para este caso sería $4000 \times 2 = 8000$ muestras por segundo. Cada muestra se representa por un código digital de 8 bits, que ofrece un máximo de 256 posibles puntos. Multiplicando 8000 muestras por segundo por los 8 bits por muestra resulta 64000 [bps], con lo que se concluye que la representación digital de una señal analógica de voz requiere 64000 [bps].

1.2.2.1.2 T1

Define el multiplexaje, mediante la técnica de multiplexaje por división de tiempo (TDM, Time Division Multiplexing), de 24 canales separados de voz de 64 kbps, o bien 24 canales DS-0, en una señal digital con amplio ancho de banda. El frame T1 consta de 193 bits, 8 bits por canal más 1 bit de control. El ancho de banda de la señal T1 es de 1.544 Mbps, equivalente a 24 canales de 64 kbps y 1 canal de control de 8 kbps. Varias señales T1 pueden ser multiplexadas en flujos de bits más rápidos. Por ejemplo T1C contiene 2 canales T1, T2 contiene 4 canales T1, T3 contiene 28 canales T1 y T4 contiene 168 canales T1. Esto es lo que se llama jerarquía.

1.2.2.1.3 E1

Define el multiplexaje, mediante la técnica TDM, de 30 canales de voz de 64 kbps más un canal de 64 kbps para control en una señal digital con amplio ancho de banda. El ancho de banda de la portadora E1 es de 2.048 Mbps. Dicha señal portadora consta de un frame de 256 bits. Es decir 32 time slots de 8 bits cada uno. En la Tabla 1.1 se muestra la jerarquía de la portadora E1.

Portadora E	Velocidad de Transmisión [Mbps]	Número de Canales 64 kbps	Número de Canales E-1	Número de Canales E-2	Número de Canales E-3	Número de Canales E-4
	0.064	1				
E-1	2.048	30	1			
E-2	8.448	120	4			
E-3	34.368	480	16	4		
E-4	139.264	1920	64	16	4	
E-5	565.148	7680	256	64	16	4

Tabla 1.1 Jerarquía de la portadora E1.

1.2.2.1.4 SONET/SDH

SONET es un estándar ANSI y SDH es un estándar ITU-T, ambos definen las transmisiones de amplio ancho de banda vía fibra óptica. La terminología incluye OC-n (*Optical Carrier level*), que indica señales ópticas jerárquicas, STS-n y STM-n representan las señales eléctricas jerárquicas para SONET y SDH respectivamente. En la Tabla 1.2 se muestra la jerarquía:

OC-n	STS-n	STM-n	Ancho de banda (Mbps)	Número de canales DS-0
OC-1	STS-1	-	51.84	672
OC-3	STS-3	STM-1	155.52	2016
OC-9	STS-9	STM-3	466.56	6048
OC-12	STS-12	STM-4	622.08	8064
OC-18	STS-18	STM-6	933.12	12096
OC-24	STS-24	STM-8	1244.16	16128

Tabla 1.2 Jerarquía de la portadora OC-n.

1.2.2.2 Tipos de redes WAN

Algunas de las tecnologías WAN que existen son: ISDN, X.25 y ATM. Estas operan en las tres capas más bajas del modelo OSI: capa física, capa de enlace de datos y capa de red.

1.2.2.2.1 Redes X.25

Las redes X.25 son redes de paquetes conmutados que, por lo general funcionan con los siguientes protocolos: PLP (Protocolo de la Capa de Paquetes) correspondiente a la capa de red del modelo OSI, LAP-B (Procedimiento de Acceso al Enlace Balanceado) correspondiente a la capa de enlace de datos del modelo OSI, y entre otras interfaces seriales de la capa física puede utilizar X.21bis, EIA/TIA- 232, EIA/TIA-449 y G.703.

Protocolo PLP

Es el protocolo de la capa de red X.25 que administra el intercambio de paquetes entre terminales a través de circuitos virtuales. Este protocolo se encarga de añadir información de control a los datos del usuario provenientes de las capas superiores en forma de encabezado, que sirve para identificar un circuito virtual particular con el que los datos están asociados, y además provee números de secuencia que sirvan para control de flujo y de errores en los circuitos virtuales. Además de transmitir datos de usuario, PLP transmite paquetes de información relacionados al establecimiento, mantenimiento y terminación de los circuitos virtuales. Por ello existen paquetes de control que incluyen el número del circuito virtual, el tipo de paquete, y cierta información de control.

Protocolo LAP-B

Es un protocolo de la capa de enlace de datos que administra la comunicación y el atrapamiento de paquetes entre terminales. Es también un protocolo orientado a bits, lo que asegura que las tramas estén ordenadas correctamente y libres de errores. Hay tres tipos de tramas: información, supervisión y no numeradas. La trama I (información) transporta información de las capas superiores y de control. Sus funciones son el secuenciamiento, el control de flujo y la detección y recuperación de errores. Estas tramas transportan números de secuencia de envío y recepción. La trama S (supervisión) transporta información de control. Sus funciones son la solicitud y suspensión de transmisiones, el reporte de status y la confirmación de la recepción de tramas I. Las tramas S solamente transportan números de secuencia de recepción. La trama U (no numerada) transporta información de control. Sus funciones son el establecimiento y la desconexión del enlace y el reporte de errores. Estas tramas no transportan números de secuencia.

Las tramas incluyen encabezado, datos encapsulados y un finalizador. El encabezado se compone de los siguientes campos: apuntador de trama (1 byte), dirección (1 byte) y control (1 byte). El apuntador delimita el inicio de la trama; la dirección indica si la trama transporta un comando o una respuesta y el campo de control califica los comandos y las respuestas e indica de qué tipo de trama se trata. El finalizador consta de los campos: FCS (2 bytes) y apuntador (1 byte). El campo FCS maneja la verificación de errores y asegura la integridad de los datos transmitidos; el apuntador indica el fin de la trama.

Protocolo X.21bis

Es un protocolo de la capa física que define los procedimientos mecánicos y eléctricos para usar el medio de transmisión. Soporta conexiones punto a punto, velocidades de hasta 19.2 Kbps y transmisión síncrona, dúplex total por un medio de cuatro alambres.

1.2.2.2.2 Redes ISDN

La red digital de servicios integrados (ISDN) representa el rediseño y reconstrucción de la tradicional red telefónica de un sistema analógico a una red totalmente digital. La red digital integra la transmisión de voz y datos mediante una simple línea telefónica usando el cable convencional y barato de par trenzado. Una red ISDN se compone de las terminales, los adaptadores de terminal (TA), los dispositivos de terminación de red, el equipo de terminación de línea y el equipo de terminación de central. Las terminales ISDN pueden ser de dos tipos: especializadas, y no especializadas. A las terminales ISDN especializadas se les denomina TE1 (Equipo Terminal Tipo 1). A las terminales que salieron antes que los estándares ISDN se les denomina TE2 (Equipo Terminal Tipo 2). Los TE1 se conectan a ISDN a través de un enlace digital de par trenzado de cuatro hilos. Los TE2 se conectan a ISDN a través de un TA. El siguiente punto de conexión son los dispositivos de terminación de red, que pueden ser de dos tipos NTE1 (Terminador de Red Tipo 1) o NTE2 (Terminador de

Red Tipo2). NTE1 y NTE2 son los dispositivos que conectan el cableado de cuatro hilos del subscríptor con el ciclo local convencional de dos hilos.

ISDN mantiene una separación lógica entre los datos del usuario de la información de señalización y control, empleando dos tipos de canales: B y D. El canal B es un canal de 64 kbps usado únicamente para transmitir datos de computadora, voz y vídeo digitalizados, que no incluye ningún tipo de información de señalización o control. Las transmisiones del canal B pueden ser de conmutación tanto de paquetes como de circuitos. El canal D es empleado para transmitir la información de señalización y control, y puede ser de 16 kbps o de 64 kbps.

Los proveedores de servicios ISDN generalmente agrupan cierta cantidad de canales B y D que ofrecen a sus clientes. Existen dos principales formas de agruparlos: BRI (*Basic Rate Interface*) y PRI (*Primary Rate Interface*). El arreglo BRI consiste en 2 canales B y un canal D de 16 kbps (2B+D). El arreglo PRI tiene la configuración general nB+D, donde el canal D es de 64 kbps. Las configuraciones PRI más comunes son: 23B+D, equivalente al estándar T1 de 1.544 Mbps, y 30B+D, y al estándar E1 de 2.048 Mbps.

ISDN basa su funcionamiento en diversos protocolos, que operan de acuerdo al modelo OSI. La Tabla 1.3 muestra la relación entre los protocolos que usa cada canal de ISDN y el modelo OSI.

Capa OSI	Canal D		Canal B	
	Control de llamada	Datos de Paquete	Circuitos conmutados	Paquetes Conmutados
3	ITU-T Q.931/I.451	PLP	N/A	PLP
2	LAP-D		Frame Relay	LAP-B
1	I.430 (BRI) e I.431 (PRI)		I.430 (BRI) e I.431 (PRI)	

Tabla 1.3 Protocolos usados en ISDN

Capa Física

Los formatos de trama de la capa física de ISDN difieren dependiendo si la trama va de la terminal a la red o viceversa. Las tramas tienen una longitud de 48 bits, de los que 36 representan datos. Los bits de la trama de la capa física son los siguientes: F, proporciona sincronización; L, ajusta el valor promedio de bits; E, asegura la resolución de la contención cuando varias terminales contienen por el uso del canal en un bus pasivo; A, es el bit de activación; S no se usa, B son los datos del canal B; y D son los datos del canal D.

LAP-D

Este protocolo sirve para asegurar que la información de control y señalización fluya y sea recibida adecuadamente. La Fig. 1.4 muestra la trama de LAP-D.

Indicador	Direcciones	Control	Datos	CRC	Indicador
-----------	-------------	---------	-------	-----	-----------

Figura 1.4 Trama LAP-D.

Los campos indicador marcan el comienzo y la terminación de la trama. El campo de direcciones sirve para indicar cuál es el protocolo que opera en la capa superior, y también indica si el paquete es un comando o una respuesta. El campo de control añade información de control como el tipo de paquete y los números de secuencia de los paquetes. El campo de datos es la información recibida de las capas superiores. El campo CRC da integridad de datos mediante la suma de verificación.

Frame Relay

Es un protocolo síncrono de paquetes conmutados WAN que provee conectividad LAN-LAN. Confía en que los protocolos de los niveles superiores realicen la corrección de errores y la petición de retransmisión si un paquete fue descartado o perdido. Las mejores cualidades de Frame Relay son: economía y eficiencia. En las redes que utilizan esta tecnología, las estaciones terminales comparten el medio de transmisión de la red de manera dinámica, así como el ancho de banda disponible. Los paquetes de longitud variable se utilizan en transferencias más eficientes y flexibles. Posteriormente, estos paquetes se conmutan entre los diferentes segmentos de la red hasta que llegan a su destino. Las técnicas de multiplexaje estadístico controlan el acceso a la red en una red de conmutación de paquetes. La ventaja de esta técnica es que permite un uso más flexible y eficiente del ancho de banda.

1.2.2.2.3 Redes ATM

Es una tecnología asíncrona de conmutación de paquetes y multiplexaje que reúne los beneficios de la conmutación de circuitos (garantizando capacidad y retardo de transmisión constantes) con las de conmutación de paquetes (flexibilidad y eficiencia para tráfico intermitente). Esto lo logra utilizando paquetes de longitud fija denominados celdas, que contienen exactamente 53 bytes, 48 son para los datos del usuario y 5 para información de control. Puede ser usada para transmitir datos, voz y video, separada o simultáneamente, a través de la misma ruta. En la Fig. 1.4 se muestra el modelo de referencia de ATM.

Capa de Adaptación ATM (AAL)	Subcapa de convergencia
	Subcapa de Segmentación y Reensamble
Capa ATM	
Capa Física	Subcapa de Convergencia de la Transmisión
	Subcapa del Medio Físico

Figura 1.4 Modelo de referencia de ATM.

Capa Física

La capa física ATM corresponde a la capa física del modelo OSI. Transporta celdas de una interfaz a otra vía un canal de comunicaciones. Soporta tanto canales ópticos como eléctricos. Para LAN soporta rangos entre 25 Mbps y 155 Mbps para cobre y fibra de vidrio. Para WAN soporta los estándares SONET/SDH. Contiene dos subcapas: Medio Físico y Convergencia de la Transmisión. La primera se ocupa de las

funciones dependientes del medio, tales como transferencia y alineamiento de bits y conversiones eléctrico-ópticas. La otra realiza funciones orientadas a convertir las celdas en flujos de bits en el nodo de envío, y viceversa en el otro extremo.

Capa ATM

Realiza el multiplexaje/demultiplexaje de las celdas y la conmutación. Además provee las conexiones virtuales entre los puntos terminales, que pueden ser conexiones de canal virtual (VCC) o conexiones de ruta virtual (VPC). Las primeras son circuitos virtuales entre la fuente y el destino, que pueden ser permanentes (PVC) o conmutados (SVC). Las segundas son una colección de VCC's que comparten los mismos nodos terminales. También genera o extrae los encabezados de las celdas dependiendo si se envía o se recibe.

Capa de Adaptación ATM

Parte los datos de usuario de niveles superiores en celdas de 48 bytes más el encabezado. Define 5 diferentes tipos de adaptación para soportar diferentes clases de servicios:

- AAL1: Soporta servicios Clase A, orientados a conexión, velocidad constante de bits (CBR), por ejemplo transmisiones de voz.
- AAL2: Soporta servicios Clase B, orientados a la conexión, velocidad variable de bits (VBR), por ejemplo datos sincronizados, video.
- AAL3/AAL4: Soporta servicios Clase C y Clase D, que son respectivamente orientado a la conexión VBR (transferencias de archivos) y no orientado a la conexión VBR (datos LAN).
- AAL5: Soporta servicios Clase C y Clase D, dando una opción más sencilla y eficiente a AAL3/AAL4.

Consta de dos subcapas. La de Convergencia provee un servicio AAL específico. La de segmentación y reensamble divide los mensajes del nivel superior en celdas de 48 bytes y viceversa según envíe o reciba.

ATM maneja varias interfaces, las principales son UNI y NNI. UNI (*User to Network Interface*) se usa para realizar la comunicación entre DTE's y los conmutadores de ATM. NNI (*Network to Network Interface*) lo usan los conmutadores de ATM para comunicarse entre sí. El formato de la celda para UNI se muestra en la Fig. 1.5.

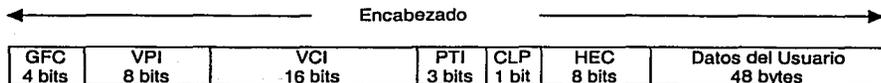


Figura 1.5 Formato de la celda UNI.

GFC (*Generic Flow Control*): Controla el flujo de datos permitiendo a múltiples dispositivos ATM ser añadidos a la misma interfaz de red.

VPI (*Virtual Path Identifier*): Una conexión de ruta virtual es una conexión semipermanente que da una colección lógica de canales virtuales que tienen los mismos puntos terminales. En un ambiente no ATM, se requieren enlaces separados para voz, datos y video. Con ATM una sola ruta virtual puede ser establecida para soportar diferentes conexiones virtuales. El identificador VPI es parte de la dirección de la red que se usa para identificar un grupo de canales.

VCI (*Virtual Channel Identifier*): Una conexión "virtual channel" es un circuito virtual que provee una conexión lógica entre la fuente y el destino. Estos circuitos virtuales pueden ser permanentes (PVC) o conmutados (SVC). El identificador VCI es parte de la dirección usada por ATM y sirve para identificar estos canales.

PTI (*Payload Type Identifier*): Indica el tipo de información contenida en la celda. Un '0' en el bit más significativo indica que la celda transporta información del usuario. Las celdas de información de usuario que incluyen un '1' intermedio indican que la celda ha experimentado congestión. Un '1' en el bit más significativo indica que la celda transporta información de control ('100' y '101') o de manejo de los recursos ('110').

CLP (*Cell Loss Payload*): Especifica si se debe descartar la celda en caso de congestión. Un '1' la descartará, un '0' no lo hará.

HEC (*Header Error Control*): Provee corrección de errores para errores de 1 bit y detección de errores para varios bits en el encabezado de la celda.

La única diferencia entre una celda UNI y una NNI es el campo GFC, que no existe en la interfaz NNI, en su lugar está el campo VPI, que consta de 12 bits en lugar de 8.

CAPITULO 2

RED TETRAPOL

2.1 GENERALIDADES

Las redes de comunicaciones radio móviles se dividen en privadas y públicas. Las redes privadas también conocidas como profesionales¹ (PMR, Private Mobile Radio o Professional Mobile Radio) son aquellas cuya infraestructura pertenece a la organización que la utiliza, y que en sus inicios no proveían a sus usuarios los servicios necesarios para establecer comunicaciones con usuarios de otras redes privadas o públicas. Las redes públicas ofrecen servicios nacionales de comunicaciones radio móviles, por los que se cobra una tarifa a los usuarios para cubrir los costos de comunicación. Éstas comprenden dos tipos: redes de telefonía celular (PMT, Public Mobile Telephony), y redes radio móviles de acceso público (PAMR, Public Acces Mobile Radio), que son redes públicas que se operan en grupos de usuarios, y se comportan como redes privadas.

Los factores decisivos en la selección de una red PMR o PAMR son el tráfico y el costo. Las organizaciones con tráfico intenso optan por redes PMR para disminuir la posibilidad de saturación en situaciones críticas y el costo a largo plazo.

Los primeros intentos para buscar soluciones digitales para sistemas PMR surgieron a principios de los años ochentas. Esto permitió mejorar la confiabilidad técnica y facilitar la encriptación de la conexión. Fue en esta época que los primeros sistemas propietarios PMR digitales aparecieron, incluyendo ASTRO e iDEN (ambos de Motorola), EDACS (Ericsson), SR 440 (Bosch/Ascom) y Radiocom 2000 (MATRA Communication).

Aunado a las mejoras implementadas por los sistemas digitales, las características de los sistemas troncalizados (*trunking*) contribuyeron al incremento de la eficiencia del espectro.

Los sistemas PMR troncalizados (*trunking*) difieren de los sistemas radio móviles públicos PMT, tales como GSM (*Global Systems for Mobile Communications*) o UMTS

¹ El término "profesional" se refiere a los servicios que ofrecen este tipo de redes, descritos en este capítulo, sección 2.5. En la presente tesis se utilizará el término "privado", en lugar de "profesional", debido a que existen sistemas PAMR que ofrecen también algunos de los servicios en cuestión.

(*Universal Mobile Telecommunications System*) básicamente en términos de su mayor velocidad para establecer una llamada, manejo de llamadas de grupo, llamadas con prioridad, encriptación de extremo a extremo, y la posibilidad de llamadas en modo directo (*walkie-talkie*).

Los usuarios potenciales de los sistemas PMR troncalizados incluyen grupos cerrados de usuarios tales como servicios de transporte (taxis, ferrocarriles, autotransportes, etc.), aeropuertos, compañías de energía y servicios de emergencia (policía, bomberos, ambulancias, ejército, defensa civil, policía fronteriza, etc.).

Tetrapol es una tecnología PMR de radio digital, celular troncalizada para comunicaciones de voz y datos. Tetrapol fue desarrollada por la compañía francesa MATRA Communication a partir de 1987, de acuerdo a las demandas de los servicios de Seguridad Pública y las fuerzas de protección, como son: encriptación punto a punto, modo directo (*Talk Around*), amplia cobertura, agrupamiento dinámico, etc.

Actualmente, Tetrapol cuenta con dos organizaciones encargadas de brindar soporte y continuar desarrollando la tecnología: el Foro Tetrapol, integrado predominantemente por fabricantes, y el Club de Usuarios Tetrapol, integrado por organizaciones de usuarios.

El foro Tetrapol es una asociación registrada en Francia en 1994. La membresía está abierta a todos los fabricantes de equipo de comunicaciones. En la tabla 2.1 se muestra una lista completa de los miembros del Foro, quienes con la finalidad de dar soporte y crecimiento al mercado PMR, deciden colectivamente los cambios y adiciones a las especificaciones técnicas de Tetrapol. El foro ofrece especificaciones abiertas para las interfaces técnicas, acción que permite a los fabricantes ofrecer productos compatibles con los de otros proveedores. Actualmente Tetrapol cuenta con 3000 páginas de especificaciones técnicas denominadas PAS (*Publicly Available Specifications*), que cumplen con las reglas definidas por ETSI (*European Telecommunications Standards Institute*).

Proveedores	País	Integrador	PMR infraestructura	PMR terminales y repetidores portátiles	Aplicaciones de datos y CAD	Periféricos accesorios y herramientas de prueba	Validación y entrenamiento
Atmel	E.E.U.U.A.						
Bascom	Moldavia						
CeoTronics	Alemania						
CET	Tailandia						
CIM Systems	Reino Unido						
Cogent	Reino Unido						
DASA	Alemania						
Davies I. C.	Reino Unido						
Debis System.	Alemania						
EADS Telecom	Alemania						
EADS Telecom	España						
EDSN	Francia						
ELCON M.	Alemania						
EMCOM	Sudáfrica						

Framatome Co.	Francia						
Grundig Plettac	Alemania						
Intergraph PS	E.E.U.U.A.						
IT Centre	Rusia						
ITRONIX	E.E.U.U.A.						
Ironix-Husky	Reino Unido						
Kathrein	Alemania						
LET	Francia						
Loxley Int. Ltd	Tailandia						
Maxon	Korea del Sur						
Médium Soft	Rep. Checa						
Mier	España						
Nortel Networks	Canadá						
ntl	Reino Unido						
Orbacom	E.E.U.U.A.						
Orga	Francia						
POSITRON	Canadá						
Prescom	Francia						
RCD	Rep. Checa						
Rockwell	E.E.U.U.A.						
Sagem	Francia						
Schlumberger	Francia						
SCHOMANDL	Reino Unido						
Siemens	Suiza						
Sigma	Irlanda						
Sonda	Chile						
Sonic	Francia						
TAIT	Nueva Zelanda						
Tecsi	Francia						
Tektronix	E.E.U.U.A.						
TELECA	Francia						
TESLA	Rep. Checa						
THALES C.S	Reino Unido						
TOP Business	Alemania						
TSA Telecom	Suiza						
Wavetek WG	Alemania						
ZETRON	Reino Unido						

Tabla 2.1 Miembros del Foro Tetrapol²

2.2 TETRAPOL EN EL MUNDO

El mercado para los sistemas PMR a nivel mundial en el año 2000, era aproximadamente de 4.6 billones de euros por año³. Aproximadamente 90% de los sistemas se basan en tecnología analógica, que será reemplazada por modernos sistemas digitales. Esto aunado a la creciente demanda de los sistemas de alta seguridad, principalmente en aeropuertos, abre grandes expectativas de crecimiento para los proveedores de sistemas PMR, entre ellos aquellos que venden redes Tetrapol.

² Información del Foro Tetrapol (www.tetrapol.com octubre de 2002)

³ Fuente: www.tetrapol.com (2001)

Los primeros usuarios de la tecnología Tetrapol fueron la Gendarmería (1992) y la Policía (1995) francesas. Desde entonces se han instalado varias redes en el mundo, y así a finales del año 2001 Tetrapol contaba con 57 redes bajo construcción o en operación en 28 países, más de 300,000 usuarios (750,000 al finalizar contratos firmados), y un área de cobertura de más de 1,000,000 de km² (4,000,000 km² al finalizar contratos firmados)⁴.

Las redes Tetrapol instaladas se enfocan básicamente a tres sectores: Civil (Industria Privada), Defensa y Seguridad Pública

Ejemplos de redes en el Sector Civil:

- SNFC Sistema de Ferrocarriles Franceses
- Planta de AUDI, Inglostadt, Alemania
- Bayernoil, Refinería, Alemania
- BMW, Seguridad de Planta, Alemania
- BVG, Autobuses de Berlín
- Servicios de Seguridad del Aeropuerto, Frankfurt, Alemania

Ejemplos de redes de Defensa:

- Base Aérea de la OTAN en Alemania
- Fuerzas de Paz de la ONU en Kosovo
- Ejército Italiano
- Ministerio de Defensa Británico
- Ejército y Fuerza Aérea Francesa

Ejemplos de redes de Seguridad Pública:

- Red ACROPOL de la Policía Nacional de Francia
- Red RUBIS de la Gendarmería Nacional Francesa
- Red Nacional IRIS de la Secretaría de Gobernación de México
- Red SIRDEE de la Policía Nacional y Guardia Civil de España
- Red POLYCOM de las fuerzas de Seguridad de Suiza
- Red PEGAS del Ministerio de Interior y de la Defensa de la República Checa
- Red SITNO del Ministerio de la Defensa de la República Eslovaca
- Red PHOENIX de las Fuerzas de Seguridad de Rumania
- Red de la Fuerzas Policiacas y de Defensa Civil de Singapur
- Red de las Fuerzas Policiacas de Tailandia

⁴ Fuente: <http://www.eads-telecom.net/1000/3192.asp> (diciembre de 2001)

2.3 TETRAPOL EN MÉXICO

La creación de la Red Nacional de Telecomunicaciones fue una de las estrategias del Sistema Nacional de Seguridad Pública del sexenio 1994-2000, que comprende, la renovación y actualización tecnológica de las distintas instituciones de seguridad en el país para mejorar su coordinación.⁵ Por esta razón, a Matra Nortel Communications se le contrató en 1998⁶, para instalar el Sistema Nacional de Radio comunicación. Dicha red denominada IRIS (Interconexión de Redes Integrales de Seguridad Pública) incluye computadoras portátiles y equipos de radio comunicación en oficinas, patrullas y ambulancias, conectados a la red de comunicación Tetrapol. La instalación de dicha red requirió de una inversión de aproximadamente 200 millones de dólares⁷.

El proyecto permitirá a las instituciones policíacas tener acceso a bases de datos de personas buscadas o desaparecidas, huellas dactilares, vehículos robados y licencias de conducir, mismas que serán actualizadas de manera constante.⁸ Actualmente se pretende agilizar el intercambio de voz, datos e imágenes para permitir que las instituciones relacionadas con la seguridad pública actúen de manera eficaz. El proyecto tiene como objetivo tender 32 redes estatales de radiocomunicación independientes, todas ellas interconectadas para crear una gran red nacional.⁹ De las 32 redes estatales, por lo menos 30 serán Tetrapol, ya que en Guanajuato y el Distrito Federal, Motorola y Ericsson¹⁰, respectivamente son los proveedores que enlazan sus cuerpos de seguridad. Las bases de datos se alimentarán de todas las dependencias policíacas del país y su contenido dependerá de las aportaciones que realice cada estado de sus archivos. Además, se contempla realizar transferencia de datos desde cualquier parte del país, utilizando dispositivos móviles.

En un futuro, es probable que en México, diversos sectores de servicios y de la Industria Privada decidan incorporar sistemas PMR troncalizados (*trunking*), como podría ser el caso de las empresas aeroportuarias y Comisión Federal de Electricidad¹¹.

⁵ Fuente: "Entregarán en 2002 red antidelinuencia", Luis Alegre, Reforma, 8 septiembre 2001.

⁶ Fuente: "Entregarán en 2002 red antidelinuencia", Luis Alegre, Reforma, 8 septiembre 2001.

⁷ Fuente: "Red de 200 millones de dólares para atacar inseguridad y terrorismo", El Financiero, 7 septiembre 2001.

⁸ Fuente: "Red de 200 millones de dólares para atacar inseguridad y terrorismo", El Financiero, 7 septiembre 2001.

⁹ Fuente: "Aparatos de alta tecnología para el combate al hampa", Reforma, 2 septiembre 2001.

¹⁰ Fuente: "Entregarán en 2002 red antidelinuencia", Luis Alegre, Reforma, 8 septiembre 2001.

¹¹ Fuente: "Redefinen Matra como EADS Telecom y visualizan fuerte crecimiento en México", Alberto Aguilar, Reforma, 14 septiembre 2001

2.4 PRINCIPALES CARACTERÍSTICAS TÉCNICAS

El desarrollo de Tetrapol se inició en 1987, motivado por la invitación para competir en la solución del sistema de radio digital troncalizado nacional para la gendarmería francesa. El método de acceso a canal elegido fue la técnica de acceso múltiple por división de frecuencia, FDMA (*Frequency Division Multiple Access*), y Matra Communication fue seleccionada para construir la red. FDMA es el método de acceso a canal, en que a cada usuario se le asigna una frecuencia específica para llamar. Con el método de acceso a canal FDMA se alcanza un amplio rango y por ende un área de cobertura extensa.

En principio, Tetrapol puede ser usada en frecuencias entre los 70 y 520 MHz. En la práctica sólo las frecuencias típicas autorizadas para PMR, en las bandas de 80, 160 y 400 MHz son utilizadas.

Como con la mayoría de los sistemas de radio, el método de frecuencia dúplex también es usado por Tetrapol. La transmisión y la recepción son manejadas en dos frecuencias diferentes, que están distanciadas una de otra por una separación llamada dúplex. El tamaño de la separación dúplex, depende de la banda de frecuencia en la que el sistema opera. Las terminales Tetrapol (como las de la mayoría de los sistemas PMR) generalmente también trabajan en un modo *half-dúplex*, es decir que no es posible transmitir y recibir simultáneamente.

El método de modulación empleado es GMSK (*Gaussian Minimum Shift Keying*). Este método también es usado por GSM y tiene la ventaja de que transmisores simples y relativamente baratos pueden ser usados.

Los sistemas digitales de radio troncalizados para PMR comparados con los sistemas radio móviles públicos como GSM o UMTS generalmente tienen un menor número de usuarios con llamadas de corta duración. Consecuentemente el tráfico es menor y se pueden construir células más grandes. La potencia de transmisión radiada por frecuencia portadora de estación base es del orden de 25 W ERP.

Los diferentes sistemas PMR difieren en términos del número de usuarios, área de cobertura, volumen de tráfico y servicios proveídos. Algunos sistemas son limitados en ruido (los límites del sistema están determinados por el ruido del receptor) o limitados en interferencia (presencia de alta interferencia de canal común producida por canales). La eficiencia de espectro depende en gran medida de estos parámetros.

Para cada estación base, se instalan típicamente de 4 a 8 (puede extenderse hasta 12) canales de radio. Un canal se utiliza como señalización, emitiendo ininterrumpidamente una portadora específica para transmitir la información del sistema de la estación base a las radio terminales móviles. Los canales restantes son canales de comunicación y pueden ser utilizados para transmitir tanto voz como datos.

Tetrapol puede ser operado en modo de red o modo directo. En el caso de modo de red, la radio terminal móvil está ligada a la estación base (infraestructura) y es

monitoreada por ésta. En este modo, cuando dos radio terminales móviles se comunican entre sí la llamada siempre se conduce vía la estación base. En modo directo dos o más radio terminales móviles se comunican sin involucrar a la estación base (*walkie talkie*). Por ende, el modo directo puede ser usado en áreas donde no hay cobertura de radio (por ejemplo en un túnel o en el sótano de un edificio).

Si el tráfico es bajo y la cobertura es amplia, Tetrapol permite el uso de una tecnología de canal común (radio co-canal, *simulcast*). En este caso, todas las estaciones base transmiten en exactamente la misma frecuencia. La señal de alta frecuencia y la posición temporal de la señal de modulación son enviadas sincronizadamente por las estaciones base. La red puede ser considerada como una macrocélula, que es alimentada por múltiples estaciones base. Con este procedimiento, es posible alcanzar una excelente cobertura con una gran área de bajo tráfico, simultáneamente con economía de alta frecuencia.

Parámetro	Valor
Ancho de canal	10 kHz, 12.5 kHz
Potencia de transmisión de una estación base por frecuencia portadora típicamente	25 W ERP (Effective Radiated Power)
Potencia de transmisión de equipo móvil	1 W, 2 W, 10 W
Sensibilidad de receptor, estática (<i>Bit Error Rate</i> =1.5%)	Terminal móvil: -119 dBm Estación base: -121 dBm
Sensibilidad de receptor, dinámica (Urbano típico 50 km/h; <i>Bit Error Rate</i> =1.5%)	Terminal móvil: -111 dBm Estación base: -113 dBm
Modo	Semidúplex
Método de acceso a canal	FDMA
Modulación	GMSK, <i>Relative filter bandwidth</i> , BT=0.25
Velocidad de transmisión de canal	8 kbit/s
Velocidad neta de transmisión	Protegida: 4.8 kbit/s No protegida: 7.2 kbit/s
Codificación de voz	RP-CELP (<i>Regular Pulse-Code Excited Linear Predictive</i>): 6 kbit/s
Eficiencia de espectro en un ambiente con interferencia limitada (mucho tráfico, muchas células)	43 bit/(seg*kHz*célula)
Eficiencia de espectro en un ambiente con ruido limitada (una célula aislada)	192 bit/(seg*kHz)
Rango del radio de célula	Rural: 20 km Suburbano: 6 km
Co-existencia estándar	ETS 300 113

Tabla 2.2 Principales parámetros de Tetrapol.¹²

¹² Tetrapol *Factsheet*, versión 1.3, Marzo, 26 2001.

<http://www.num2002.ch/en/telekommunikation/forschung/tetrapol/>

2.5 SERVICIOS

Tetrapol ofrece una gran variedad de servicios considerados "profesionales", que no se encuentran disponibles en sistemas PMT como GSM o UMT. Los principales son:

Teleservicios:

- **Llamada individual:** Corresponde a una llamada en un sistema radio móvil público (GSM, UMTS). Un usuario llama a otro usuario individual, estableciéndose una conexión entre ellos.
- **Llamada de grupo:** Un usuario llama a un grupo predefinido. Cada miembro del grupo puede escuchar todo y puede hablar.
- **Modo directo:** Dos o más terminales móviles se comunican entre sí, sin involucrar a una estación base (*walkie-talkie*).
- **Llamada de *broadcast*:** Es una llamada unidireccional punto a multipunto en un área específica. El área y los usuarios se definen anticipadamente.
- **Llamada de emergencia:** Establece una llamada de alta prioridad al despachador o a un grupo predefinido de usuarios.
- **Llamada inclusiva:** Hace posible solicitar a uno o más usuarios adicionales que se incluyan en una llamada en curso.
- **Talk Group:** Un grupo de usuarios puede conversar entre sí en un canal específico durante un periodo determinado. Dentro del grupo, todos los participantes pueden escucharse unos a otros y pueden hablar en cualquier momento. Para formar parte de la conversación el usuario debe introducir el número del grupo de conversación. Los números de los grupos de conversación activos en un momento dado son transmitidos en el canal de señalización y son del conocimiento de todos los usuarios de la red.

Servicios de datos:

- **Paging:** Mensajes no predefinidos que pueden ser enviados de un despachador a una radio terminal móvil. Los mensajes no son confirmados.
- **Status de transmisión:** Mensajes predefinidos que pueden ser transmitidos de un despachador a una radio terminal móvil y viceversa, o entre radio terminales móviles.
- **Mensajería de datos cortos:** Permite a los usuarios intercambiar mensajes cortos.
- **Servicio de paquetes de datos X.400:** Permite que una llamada X.400 pueda ser establecida entre dos terminales. Esto también hace posible una conexión de una radio terminal móvil a una Red de datos empaquetados, PDN (*Packet Data Network*).
- **Acceso TCP-UDP/IP:** Permite que una radio terminal móvil acceda a servidores que soporten los protocolos TCP-UDP/IP.

Servicios suplementarios:

- **Llamada prioritaria:** Permite al usuario asignar una prioridad a la llamada para que ésta sea procesada antes que las llamadas de menor prioridad. Cuando la red se satura, los recursos necesarios pueden ser liberados mediante este proceso, abortando las llamadas con menor prioridad.
- **Entrada tardía:** Permite al usuario conectarse a una llamada de grupo, después de que ésta se ha iniciado. Servicio que resulta útil por ejemplo, si el usuario está enlazado con otra llamada o no había encendido su equipo en ese momento.

2.6 INFRAESTRUCTURA

Una red Tetrapol está conformada por varios estratos o planos de funcionamiento, que principalmente son: plano de usuario, plano de terminales, plano de red básica y plano de redes.

En el plano de usuario se encuentran los SIM (*Subscriber Identity Module*), que son módulos removibles portadores de la información del suscriptor y de los algoritmos de seguridad.

En el plano de las terminales se hallan las terminales del sistema (ST, *System Terminal*), las terminales de datos de usuario (UDT, *User Data Terminal*), y los puestos de despacho autónomos (SADP, *Stand Alone Dispatch Position*). Las terminales del sistema pueden ser radio móviles o alámbricas (físicamente conectadas). Las radio terminales móviles se conectan a la infraestructura mediante un enlace de radio, y las terminales alámbricas se conectan local o remotamente al sistema mediante una conexión física. Las UDT son las terminales de datos, que conectadas a las terminales del sistema se usan para proveer servicios de datos. El SADP es un puesto aislado dedicado a proveer funciones de despacho.

El plano de red básica comprende subdivisiones geográficas de cobertura de la red que representan entidades capaces de operar en el modo normal de red, incluyen un *radioswitch* o radiocommutador de red (RSW, *Radio Switch*), una o más estaciones base (BS, *Base Station*), y un centro de operación y mantenimiento (OMC, *Operational Maintenance Center*).

En el plano de redes se hallan aquellos elementos que enlazados al SwMI (*Switching and Management Infrastructure*) permiten acceder a redes externas como las redes X.25, redes TCP-UDP/IP y X400. En este plano se encuentran los PABX (*Public Access Branch Exchange*) y las compuertas o *gateways*. El PABX es el acceso a la red pública telefónica, y las compuertas o *gateways* son los puntos de enlace entre una red Tetrapol y una o varias redes LAN o WAN de computadoras.

2.7 OTRA TECNOLOGÍA PAMR/PMR DIGITAL TRONCALIZADA: TETRA (Trans European Trunked Radio)

TETRA es el estándar europeo de comunicaciones radio móviles troncalizadas, aprobado por ETSI (European Telecommunications Standardization Institute). En sus inicios, TETRA se enfocó en la aplicación de redes PAMR (Public Access Mobile Radio). Cuando estas redes no cubrieron las expectativas de los usuarios, se cambió el objetivo de TETRA, principalmente al mercado de la seguridad pública - policía, ejército, departamento de bomberos, ambulancias, etc. -, que debido a que la OTAN (Organización del Tratado del Atlántico Norte) en 1994 liberó la banda de frecuencia de 380 a 400 MHz a lo largo de toda Europa solamente para servicios de seguridad pública, este segmento tuvo la oportunidad de instalar sus propias redes.

Otros mercados en los que TETRA se ha enfocado son redes de radio telefonía celular, redes nacionales cerradas y pequeñas redes locales. La única concesión que tienen de una red PAMR es TETRALINK en el Reino Unido.¹³ Los principales promotores de TETRA son: MOTOROLA, NOKIA, SIMOCO y el consorcio TETRACOM formado por la división de radio comunicación móvil de PHILIPS, Rohde & Schwartz y DeTeWe.

TETRA soporta aplicaciones para comunicaciones radio móviles tanto comunes como profesionales, tales como: llamadas de grupo, conversación/datos, fax, transferencia de archivos, acceso a bases de datos, mensajería, servicios de localización de vehículos, manejo de flotas, etc.

El método de acceso a canal usado por TETRA es Acceso Múltiple por División de Tiempo, TDMA (*Time Division Multiple Access*). Con este método, TETRA ofrece cuatro canales independientes de comunicación dentro de un radio canal de 25 kHz. Un canal individual ocupa una cuarta parte de la duración de una *time slot*. Durante el periodo restante, el canal de radio puede ser empleado por otros usuarios. En una llamada, el transmisor es encendido y apagado aproximadamente 18 veces por segundo. Sin embargo, si se necesita una transmisión de alta capacidad, un usuario puede incluso ocupar múltiples *time slots* (hasta cuatro).

En el sistema TETRA el canal de señalización es transmitido en una portadora específica en el primer *slot* de cada paquete. Este canal de señalización es usado para transmitir la información del sistema de red a las terminales y así permitir la sincronización entre las terminales y la estación base.

La potencia de transmisión del equipo móvil es continuamente regulada por comandos de la estación base para obtener la potencia mínima requerida para que la estación base detecte la información transmitida por el equipo móvil (control de potencia). Lo que significa que este control de potencia, por una parte permite minimizar la interferencia y por otra parte maximizar la vida de la batería del equipo móvil. La potencia de transmisión de la estación base no es regulada por TETRA.

¹³ Fuente: "Mobile telecommunications: standards, regulation and applications", Bekkers Rudi, 1998.

El método de modulación usado es $\pi/4$ -DQPSK (Cambio de fase diferencial cuaternaria). A través de este método de modulación lineal se obtiene eficiencia de espectro, aunque requiere transmisores de alto grado de linealidad, y por lo tanto costosos.

En principio, TETRA puede ser utilizada en todas las frecuencias por debajo de 1 GHz, sin embargo, en la práctica sólo las frecuencias típicas autorizadas para PMR en las bandas de 160, 400 y 800 MHz son empleadas.

CAPÍTULO 3

SISTEMA DE ADQUISICIÓN DE DATOS

En el presente capítulo se muestran las características principales de un sistema adquisidor en función de sus componentes, se presentan los conceptos de ruido que se tomaron en consideración para el desarrollo del sistema y finalmente, se describen los elementos que fueron seleccionados para realizarlo.

3.1 GENERALIDADES

Debido a la necesidad de conocer el comportamiento de diversas variables físicas para dar seguimiento y/o controlar procesos, surgieron los Sistemas de Adquisición de Datos (SAD), que además de medir, son capaces de almacenar los valores y en algunos casos de procesarlos.

En sus inicios los SAD operaban mecánicamente, realizando, en la mayoría de los casos, el respaldo de la información en rodillos de papel. Con la evolución de la electrónica, aparecieron los SAD que basaron su funcionamiento en esta tecnología, con una consecuente mejora en la precisión de las mediciones, así como una reducción en el tamaño de los sistemas. Posteriormente con el surgimiento y desarrollo de las computadoras, los SAD incorporaron esta tecnología para realizar algunas de las tareas del sistema, tales como el análisis y la presentación de resultados, desplazando en muchos casos, a los SAD que integraban los módulos de adquisición, almacenamiento, procesamiento y despliegado de datos en una misma plataforma. El uso de los equipos de cómputo aplicado a los SAD tiene como beneficio una mejor precisión, resolución y procesamiento de la información, además de aumentar la flexibilidad de los sistemas, por consiguiente, la rentabilidad de éstos.

3.2 ELEMENTOS DE UN SAD

La adecuada obtención de resultados en un SAD basado en una plataforma computacional depende de la calidad y capacidad de cada uno de los siguientes elementos:

- Transductores
- Acondicionadores de señal
- Elementos de medición
- *Software*
- Equipo de cómputo

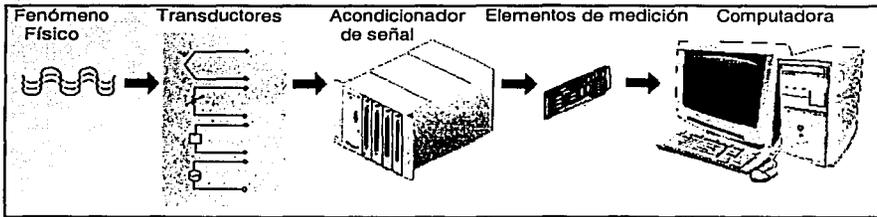


Fig. 3.1 Elementos de un SAD basado en una plataforma computacional.

3.2.1 Transductores

Los transductores son dispositivos capaces de convertir las magnitudes a medir en señales eléctricas, por ejemplo, sensores de temperatura encapsulados, termistores y termopares convierten la temperatura en un voltaje o corriente eléctrica. Otros ejemplos son los transductores de presión o flujo que convierten fuerza o flujo en una señal eléctrica. En cada caso, las señales eléctricas producidas varían de acuerdo con los parámetros físicos que monitorean.

3.2.2 Acondicionadores de señal

Generalmente las salidas de los transductores deben ser acondicionadas para proveer de señales apropiadas al dispositivo de medición. La adecuación consiste en aislar, filtrar y amplificar señales de bajo nivel. Los acondicionadores de señal pueden estar integrados al dispositivo de medición, ser módulos dedicados e independientes o puntas de prueba como las usadas en los osciloscopios.

La señal eléctrica equivalente generada por el circuito de acondicionamiento de señal, asociado a los transductores proporciona el nivel variable de voltaje o de corriente.

3.2.3 Elementos de medición

Los elementos de medición permiten representar una señal eléctrica analógica o digital en forma binaria para su posterior procesamiento. En el caso de señales analógicas se emplean como elementos de medición un convertidor analógico digital y un módulo adquirente usualmente basado en un microcontrolador (μC).

Convertidor Analógico Digital

Las principales características de un Convertidor Analógico Digital (ADC, *Analog to Digital Converter*) son rango de entrada, sensibilidad vertical, resolución, frecuencia de muestreo, ancho de banda analógico y no linealidad.

El rango de entrada se refiere al mínimo y máximo niveles de voltaje con los cuales el ADC puede digitalizar una señal.

La sensibilidad vertical define el mínimo cambio de nivel o voltaje detectable, representado por el bit menos significativo (LSB, Less Significant Bit). La sensibilidad vertical se puede encontrar dividiendo el rango de voltaje del convertidor entre el producto de la ganancia por dos elevado al número de bits de la resolución.

$$\text{Sensibilidad vertical} = \text{Rango de voltaje} / (\text{Ganancia} * 2^{\text{bits de resolución}})$$

La resolución es el número de bits que un convertidor utiliza para representar una señal analógica. Entre más alta sea la resolución, mayor será el número de niveles en los que el rango de voltaje se dividirá, y por ende, más pequeño el cambio de nivel detectable (sensibilidad vertical).

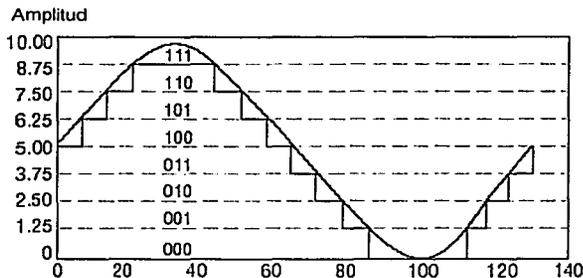


Fig. 3.2 Conversión Analógica Digital

La frecuencia de muestreo es la frecuencia a la que una señal es adquirida y digitalizada por un ADC. De acuerdo con el teorema de Nyquist, la mínima frecuencia de muestreo debe ser al menos el doble de la frecuencia de la señal muestreada. La frecuencia a la mitad de la frecuencia de muestreo es conocida como frecuencia de Nyquist. Teóricamente, es posible recuperar información de señales con, o por debajo de la frecuencia de Nyquist.

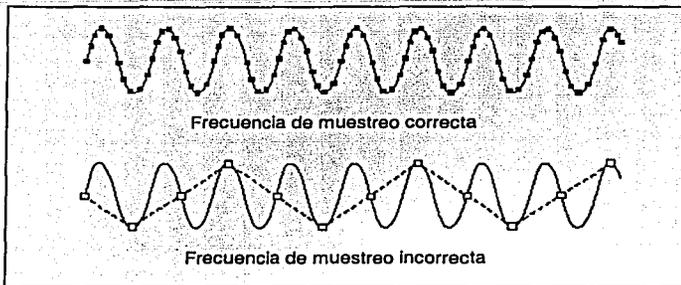


Fig. 3.3 Teorema de Nyquist.

El ancho de banda analógico se refiere al rango de frecuencias (en hertz) en el que una señal es digitalizada adecuadamente.

La no linealidad es una medida aplicable al bit menos significativo que indica la peor desviación posible, idealmente, a medida que se incrementa el nivel de voltaje aplicado a un convertidor o digitalizador, los valores digitales obtenidos también deben incrementarse linealmente. Si se grafica el voltaje aplicado contra los valores digitales obtenidos se obtiene una línea recta. Un digitalizador perfecto debería tener una medida de no linealidad de cero, en la práctica, un buen digitalizador tiene una medida de ± 0.5 LSB.

Otro parámetro de igual importancia es la configuración de los canales de entrada.

Los canales se configuran de tres formas: entrada sencilla, pseudodiferencial y diferencial.

En la configuración "entrada sencilla", todas las entradas de los canales se refieren a una tierra común, que es la misma del ADC. En los canales pseudodiferenciales, todas las entradas se refieren a una misma tierra común, sin embargo, ésta no es común al ADC. Finalmente en los canales diferenciales cada entrada se refiere a una tierra independiente de los demás canales y del ADC. El uso de canales diferenciales requiere de más recursos del ADC, puesto que por cada dos canales sencillos se obtiene un solo canal diferencial, sin embargo, los canales diferenciales presentan una mejor inmunidad al ruido. La utilización de uno u otro tipo de entrada de canal, dependerá de la calidad que se requiera en la medición, de las condiciones que le rodean y de la señal a medir. Si la distancia entre el sensor y el ADC no es muy grande y no existe interferencia generada por otros equipos, tal vez sea suficiente un canal sencillo para obtener una medición correcta.

Por otra parte, es importante considerar en la selección de un tipo de canal, la relación entre la señal medida y su referencia. Como se mencionó, la señal generada por el conjunto de transductores y por el circuito acondicionador de señal se puede representar como una fuente de voltaje variable, que está comprendida en alguna de las dos categorías siguientes: referenciadas o no referenciadas a tierra (flotantes).

Las señales referenciadas a tierra son aquellas en las que la señal de voltaje es referenciada a la tierra del sistema, es decir, está conectada a un punto de tierra común al ADC.

La diferencia en el potencial de tierra entre dos instrumentos conectados al mismo sistema de alimentación, típicamente se encuentra entre 1 y 100 [mV], pero puede ser mayor si los circuitos de distribución de la alimentación no están debidamente conectados. Si una señal fuente referenciada a tierra es medida inadecuadamente, esta diferencia puede resultar un error en la medición.

Las señales no referidas a tierra son aquellas en las que la señal de voltaje no tiene una referencia absoluta, por ejemplo, la tierra física o la tierra del sistema. Algunos ejemplos comunes de este tipo de señales son baterías, señales fuente alimentadas con baterías, termo acopladores, transformadores y amplificadores de aislamiento. Es importante fijar la referencia de una señal este tipo a la tierra de la entrada analógica del ADC para establecer una referencia local.

Módulo adquirente basado en un microcontrolador

El módulo basado en un microcontrolador se encarga de obtener los datos adquiridos por el ADC, almacenarlos en la memoria y transmitirlos a la computadora.

Un microcontrolador (μC) es un sistema digital, que maneja datos, efectuando operaciones aritmético-lógicas dictadas por un programa. Además se encarga de la administración de los periféricos internos, generalmente memoria ROM, RAM, y puertos entrada/salida. Los puertos de entrada/salida permiten al μC una comunicación directa con el exterior, es decir, es posible interconectar el μC con variables físicas y/o variables de control externas al μC . El número de las distintas operaciones que se pueden realizar en un sistema dado es finito y la cantidad de operaciones a procesar, depende directamente de la complejidad de su diseño.

3.2.4 Equipo de cómputo y software

Los elementos del equipo de cómputo como microprocesador y memoria determinan la velocidad de procesamiento de datos provenientes de los elementos de medición y la precisión de resultados. El software determina el tipo de procesamiento (algoritmos) y las características de despliegue de resultados. Además permite exportar los datos a otras aplicaciones. El uso de periféricos proporciona al SAD la capacidad de almacenar los datos en discos compactos, discos flexibles, papel, acetatos, etc.

Cabe mencionar que dadas las características inalámbricas del SAD, los elementos que lo conforman se agrupan en dos subsistemas: Módulo Adquisidor de Datos, conformado por transductores, acondicionadores de señal y elementos de medición; y el Servidor de Aplicación de Datos, compuesto por el equipo de cómputo y su software de aplicación.

3.3 RUIDO

Uno de los factores que más influyen en el diseño de un sistema es el ruido. Por esta razón a continuación se mencionan sus principales características.

El ruido puede presentarse en diferentes formas; puede ser acústico, eléctrico o incluso óptico. En general, la palabra ruido tiene una connotación que implica algo indeseable, y aunque en algunas ocasiones es algo aleatorio e insignificante, en otras se convierte en un fenómeno considerable.

En el diseño de un circuito electrónico se deben considerar los efectos del ruido eléctrico, el cual se define como cualquier señal eléctrica no deseada presente en un circuito.

En algunos casos las principales causas de ruido eléctrico son inherentes a los dispositivos utilizados para realizar el trabajo, como son las fuentes de poder y su acoplamiento al sistema, el ruido generado por los transistores, etc. Sin embargo, en la mayoría de los casos, las fuentes de ruido que generan más interferencia al sistema son fenómenos electromagnéticos, provenientes del entorno de trabajo, que incluye interferencia por las líneas de alimentación o distribución, cableado, motores, transmisores de diversa índole, etc. Es por ello, que dada la incursión de equipos eléctricos y electrónicos en cada vez más variadas áreas de la actividad humana, los niveles de tolerancia electromagnética se han vuelto más estrechos, esto a medida que la separación física entre dispositivos se ha reducido, y se ha convertido en un factor considerable lograr una adecuada compatibilidad electromagnética (EMC, *Electromagnetic Compatibility*)¹, que se define como la capacidad de un dispositivo para operar dentro de ciertos límites de emisiones e inmunidad a la interferencia electromagnética.

3.3.1 Inmunidad

El ambiente electromagnético en el que un equipo debe operar, determina el nivel de tolerancia a la interferencia y puede variar considerablemente. Por ejemplo, la magnitud de los campos de radiación que afectan al sistema depende de la distancia a la fuente de los mismos. Campos de radiofrecuencia intensos ocurren en las cercanías de transmisores, radares e incluso de hornos de microondas caseros. Los transmisores portátiles no tienen una alta potencia de radiación, sin embargo, si se

¹ En algunas áreas de la electrónica la EMC ha sido un requerimiento desde los años 60, como es el caso de los dispositivos de uso militar.

acercan demasiado a equipos susceptibles generan interferencia. Por otra parte el incremento en el uso de radio comunicadores ha llevado a establecer niveles de mínima inmunidad a interferencia por radiofrecuencia sobre todo en sistemas de mediciones críticas.

Las descargas electrostáticas son otra fuente de transitorios generados principalmente cuando una persona que ha sido cargada con un alto potencial eléctrico toca una parte susceptible del equipo, descargando todo su potencial a través de éste, de manera que miles de voltios pueden fluir en un lapso corto, esto aunado a un tiempo de levantamiento por debajo de los nanosegundos. Aún cuando la energía descargada sea poca y ésta sea conducida a tierra por medio del chasis, este pulso de energía se puede acoplar fácilmente a la circuitería interna ocasionando interferencia y posibles daños.

La inmunidad a los transitorios ha cobrado mayor importancia debido a que los equipos basados en microprocesadores son más susceptibles a los transitorios que los equipos analógicos. Se ha comprobado que incluso los transitorios generados por interruptores producen múltiples transitorios con tiempos de ascenso en nanosegundos y amplitudes de cientos de volts. Otras fuentes de transitorios son los automotores, el ambiente transitorio automotriz es particularmente severo con respecto a su rango nominal de voltaje, en él intervienen los transitorios generados por el alternador, el sistema de ignición y el apagado del mismo. Las alteraciones electromagnéticas pueden ocasionar la corrupción en la ejecución del programa base de un microprocesador, generando una falla sensible en el sistema. Si el sistema cuenta con un circuito de recuperación de programa, la falla parecerá un error normal y podría caer dentro de un rango aceptable de funcionamiento.

El rango aceptable de funcionamiento, y por ende, de inmunidad electromagnética de los equipos electrónicos depende de su aplicación, por ejemplo, los equipos de entretenimiento y dispositivos cuyo funcionamiento no comprometa la eficacia o seguridad en algún evento, se encuentran al final de la escala de inmunidad. Sin embargo, los equipos de cómputo e instrumentación para el control de sistemas críticos como aeronaves y plantas de energía, se encuentran al inicio de la escala, y es en estos equipos en donde las autoridades han considerado rangos y tolerancias de inmunidad electromagnética, en el caso de otros equipos de menor riesgo, las tolerancias a la interferencia electromagnética se establecen por medio de factores de mercadotecnia.

3.3.2 Emisiones electromagnéticas

En comparación con los requerimientos de inmunidad, las emisiones son relativamente más fáciles de caracterizar. La mayoría de las emisiones de equipo electrónico son ocasionadas por switches u otra operación electromecánica y por líneas con señales de reloj. Estos tipos de emisiones han sido reguladas por muchos años, ya que afectan directamente a las transmisiones de señales de AM y de comunicaciones en general.

Las emisiones de equipo digital pueden ser generadas por señales de reloj de alta-frecuencia del orden de hasta cientos de megahertz. Los sistemas de reloj y sus armónicas son las principales fuentes de ruido, debido a que la energía se concentra en un estrecho ancho de banda.

Por otra parte, en los sistemas basados en μC o μP , el ruido de amplio ancho de banda debido a las líneas de datos y direccionamiento también está presente. La amplitud del ruido y su distribución espectral varía dependiendo del modo de operación del circuito o del software residente en el mismo.

Las emisiones electromagnéticas pueden ser conductivas o radiadas. Los estándares de emisiones comerciales diferencian una de la otra en función de la frecuencia, el punto de división es universalmente aceptado a los 30 [MHz]. Esto puede parecer arbitrario, pero se ha demostrado empíricamente que los mecanismos de acoplamiento suceden predominantemente por conducción por debajo de los 30 [MHz], y arriba de estos por radiación.

3.3.3 Ruido en la etapa de acondicionamiento de señal

La etapa de acondicionamiento de señal incluye el filtrado, acoplamiento de impedancia y amplificación. Esta última juega un papel muy importante, ya que tiene por objeto otorgar el nivel adecuado de voltaje para que la etapa de medición del SAD obtenga una lectura óptima. Sin embargo, la amplificación o ganancia tiene además otras implicaciones en cuanto a ruido se refiere, ya que la existencia de ganancia de voltaje entre dos puntos cualesquiera de un circuito, favorece la presencia de ruido en el mismo. En un circuito bien diseñado la ganancia de cualquier punto interno del circuito con respecto a la salida debe ser cercana a cero, y la ganancia entre la entrada y la salida debe estar bien definida. Idealmente, la interferencia que entra al circuito donde existe ganancia unitaria no afecta al circuito.

La interferencia o ruido entrante en un circuito será tratado como cualquier otra señal, por lo tanto, será amplificado. La única forma de rechazar esta interferencia se da debido a las características de ancho de banda del circuito, de manera que si la señal de interferencia está fuera del ancho de banda del sistema, ésta puede ser eliminada. Así, es posible limitar el rango de señales de interferencia que afectan al sistema aplicando filtros a la entrada.

El ruido en un semiconductor es una combinación del ruido térmico y del ruido de impacto (*shot*). El ruido térmico es producido por el movimiento termal aleatorio de los electrones en un conductor o resistencia. El ruido de impacto (*shot*) es el resultado del arribo de cargas discretas a un flujo de corriente, esta señal de ruido es proporcional al flujo de corriente en el dispositivo. Es importante considerar que no es posible evitar el ruido térmico, dado que es intrínseco al sistema.

3.4 HARDWARE

3.4.1 Transductores y acondicionamiento de señal

Nivel de combustible

Dada la importancia de garantizar el funcionamiento de un sitio de repetición incluso en caso de falla en el suministro de energía eléctrica de la red local de distribución, los sitios cuentan con fuentes alternas de energía como lo son el banco de baterías y un generador que se activa cuando las baterías están por debajo de cierto nivel de carga. La función del generador en esta situación no sólo es proveer al sitio de energía, sino que también se encarga de recargar las baterías. De ahí la importancia de medir los parámetros de operación del generador y de transmitirlos a un Servidor de Aplicación de Datos. Uno de los parámetros más importantes es el nivel de combustible del generador, ya que de agotarse antes de que se restablezca el suministro normal de energía, el sitio no contará con la energía necesaria para operar.

Los sitios Tetrapol utilizan un generador de la serie B propia de la marca Cummins, que está equipado con un tanque de combustible de 120 [lt]. El generador cuenta con un sistema propio de mediciones, que monitorea las diferentes variables críticas para su funcionamiento. Entre estas variables se encuentran la temperatura interna, nivel y presión del aceite, revoluciones por minuto, consumo y nivel de combustible.

Este sistema de medición se encuentra integrado en un panel, que para los fines de la presente tesis, se considera como una caja negra, de la que solo se conocen las entradas y las salidas, las últimas conformadas por un conjunto de indicadores gráficos que representan los valores de las respectivas variables.

El sistema se basa en un conjunto de transductores, entre los que se encuentra un transductor de nivel de combustible. El tanque de combustible tiene integrado un sensor analógico de nivel tipo potenciométrico compuesto por un flotador sujeto a una varilla de metal, que a su vez mueve un cursor metálico que contacta de manera deslizante sobre una resistencia. Para obtener la medición, el sistema propio del generador aplica una diferencia de potencial de 5[V] en los extremos de esta resistencia, por lo que se tiene un circuito divisor de voltaje, de manera que en el cursor deslizante se obtiene un voltaje entre 0 [V] y 5 [V] relacionado directamente con el nivel de combustible en el tanque. En la fig. 3.4 se muestra el sensor de nivel de combustible descrito.

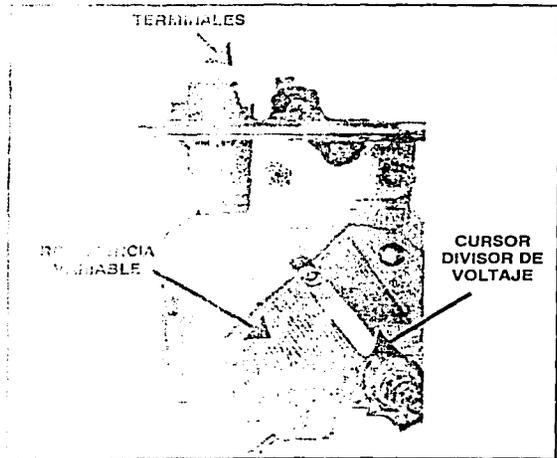


Fig. 3.4 Sensor de nivel de combustible.

El sensor de nivel de combustible es un divisor de voltaje a diferencia de los sensores de temperatura que se conectan directamente a una electrónica como variable de medición. Lo anterior permite una gran confiabilidad y a que no es necesario acondicionar la señal de corriente para obtener una señal de voltaje que el SAD sea capaz de medir.

El sensor de combustible es parte del sistema de medición propio del generador, que digitaliza la información directamente. El sistema de medición genera información no digitalizada directamente del sensor de combustible, que es enviada al SAD para procesarla de manera adecuada a través de un convertidor.

Debido que la señal es una resistencia, el ruido que pueda inducirse no se considera necesario implementar un filtro de ruido y en algunos casos adicionales para eliminar el ruido, el único elemento que se implementa es un zener, cuya finalidad es evitar un sobrevoltaje del sensor. El zener se conecta al GND, el dicho zener actúa como regulador de voltaje a carga. La conexión de este circuito se muestra en la fig. 3.5.

Debido que el sensor, la señal del transductor de combustible se transmite por medio de un par de cables, trazado al SAD y directamente a la entrada de dos canales del ADC. Cuando se genera un momento el ADC recibe la orden de tratar ambos canales como entrada diferencial y automáticamente realiza la resta interna de las señales de los canales para eliminar una posible señal de ruido.

TEJIS CON
FALLA DE ORIGEN

TEJIS CON
FALLA DE ORIGEN

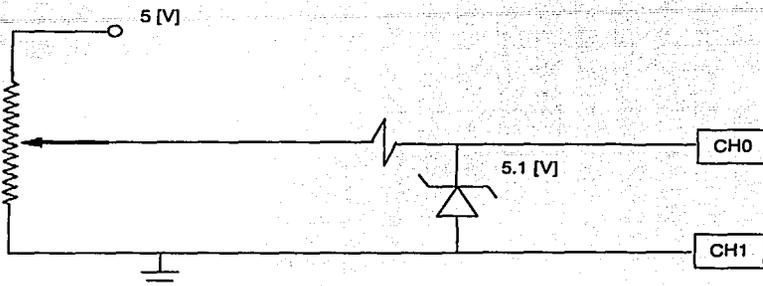


Fig. 3.5 Circuito regulador de voltaje.

Temperatura ambiente

Para garantizar la adecuada operación del equipo presente en un sitio de repetición, éste cuenta con un sistema de aire acondicionado. Un fallo en este sistema significaría un sobrecalentamiento del equipo, que podría dañarlo. Con la finalidad de detectar posibles fallos en el sistema de aire acondicionado, el SAD mide la temperatura ambiente del sitio, que en condiciones normales no debe sobrepasar los 20 [°C], cualquier valor superior generará la transmisión de una alarma².

Para realizar la medición de esta variable se utilizó el sensor LM35, que permite medir temperaturas dentro del rango de 0 a 100 [°C]. Este sensor se eligió principalmente por el comportamiento lineal que presenta, su calibración inherente y bajo costo. Debido a que el rango de temperaturas que se espera medir se encuentra entre 0 y 50 [°C] y puesto que el sensor entrega 10 [mV/°C] se decidió añadir el amplificador de instrumentación AD620 de ANALOG DEVICES con una ganancia de 10, que permite acotar la señal de entrada dentro del rango del convertidor y por consiguiente, obtener una mayor precisión y eliminar el ruido inducido en el cableado del sensor al SAD. El diagrama de conexión se muestra en la fig. 3.6.

² Véase Capítulo 5

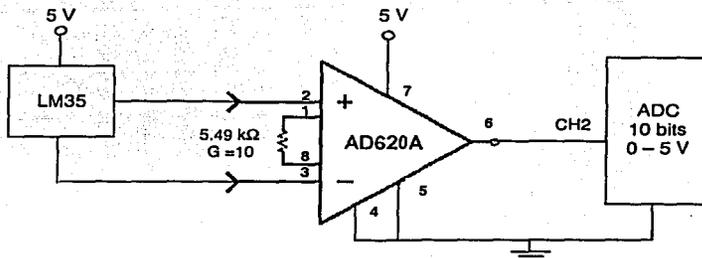


Fig. 3.6 Diagrama de conexión del sensor de temperatura.

3.4.2 Elementos de medición

Microcontrolador

Actualmente existe en el mercado una gran variedad de fabricantes de microcontroladores, entre los que se encuentran: Atmel, Intel, National Semiconductors, Microchip, Motorola, Zilog, etc. En el presente trabajo se decidió utilizar el μ C AT90S8515 de Atmel, debido a que se contaba con el sistema de desarrollo de Atmel, y porque dentro de la familia AVR de Atmel, el AT90S8515 ofrece mayor capacidad de memoria de programa (8 kbytes de memoria programable tipo *Flash*), mayor memoria SRAM (512 bytes) y mayor capacidad de memoria EEPROM interna (512 bytes). Factores decisivos considerando la complejidad de los protocolos incluidos en el programa fuente³. Otras características importantes del AT90S8515 son: μ C RISC *like* de 8 bits, bajo consumo, dos temporizadores, uno de 8 bits y otro de 16 bits, temporizador *watchdog* programable, un puerto serial tipo UART, 12 interrupciones y cuatro puertos de entrada/salida de 8 bits cada uno.

Con la finalidad de incrementar la capacidad de almacenamiento del módulo adquirente basado en el μ C AT90S8515, se decidió añadir cuatro circuitos integrados de memoria EEPROM externa Microchip 24LC64 de 8 kbytes cada uno. Aunque existe una amplia gama de memorias externas, los factores decisivos fueron su bajo costo, su forma de acceso: serial, y la posibilidad de incrementar fácilmente la capacidad de almacenamiento del adquirente. Sea aumentando la cantidad de circuitos integrados hasta ocho, o cambiar las memorias de 8 kbytes por memorias de 16, 32, 64 kbytes o hasta 1 Mbyte, que operan de manera similar a las elegidas originalmente. Por lo que las modificaciones serían mínimas. La única desventaja que presentan las memorias con mayor capacidad es un costo elevado debido a que aún no se venden en México.

³ Véase Capítulo 4 Protocolos de comunicación del Módulo Adquirente de Datos.

Convertidor A/D

Aunque existen en el mercado diversos microcontroladores que incluyen un ADC (HC11, Z180, etc.) el AT90S8515 no cuenta con uno integrado, por lo que fue necesario emplear uno externo.

Considerando que las variables a medir no requieren una precisión estricta, pero que por otra parte el SAD puede ser empleado en otras aplicaciones que sí lo requieran, se decidió utilizar un convertidor de 10 bits. Comparando los convertidores existentes en el mercado mexicano, se eligió el ADC10158 de National Semiconductors, el cual cuenta con 8 canales sencillos (4 diferenciales) con resolución de hasta 10 bits más signo.⁴

El convertidor A/D ADC10158, ofrece la ventaja de que puede ser programado al momento de realizar una conversión, así los canales pueden ser tratados como entradas diferenciales, pseudodiferenciales (todos en referencia a una entrada común diferente de la tierra), o simples (todos en referencia a la tierra). La capacidad del ADC para tratar los canales de acuerdo con la aplicación, permite para señales de no muy alta frecuencia, evitar el uso de elementos externos, como retenedores (*sample and hold*) e incluso amplificadores diferenciales. El ADC realiza la resta de dos canales diferenciales obteniendo un resultado sin la intervención del microcontrolador, liberando a este último de procesos que consumen tiempo.

Por otra parte, los datos digitalizados se transmiten al microcontrolador de manera paralela en varios posibles formatos relativos al orden de los bits significativos, al número de bits y al signo. Así el ADC10158 es un convertidor A/D versátil y fácil de manejar, además de que por medio de la característica *chip enable*, es posible colocar otro ADC en paralelo, duplicando el número de canales.

⁴ Resolución configurable de 8 o 10 bits con o sin signo.

CAPÍTULO 4

PROTOCOLOS DE COMUNICACIÓN DEL MÓDULO ADQUISIDOR DE DATOS

Para que el Módulo Adquisidor de Datos, envíe la información al Servidor de Aplicación de Datos, usando los recursos de la Red Tetrapol, el Módulo Adquisidor de Datos, primero debe establecer comunicación con la Radio Terminal del sistema, que debe identificar al Módulo Adquisidor de Datos como una Terminal de Datos de Usuario (UDT). Es por ello, que en el presente capítulo se describe la arquitectura con la que opera una UDT, así como los protocolos que emplea el Módulo Adquisidor de Datos para comunicarse.

4.1 DEFINICIONES Y ABREVIACIONES

Con la finalidad de ser coherente con las especificaciones de Tetrapol y de esta manera facilitar al lector la consulta de documentación más extensa referente a estos sistemas, el presente capítulo utiliza las siguientes abreviaciones y definiciones:

4.1.1 Abreviaciones

DNC	Controlador de Red de datos
DAS	Servidor de Aplicación de Datos
MSW	Conmutador Principal
NDIS	Especificación de interfaz del controlador de red
NIC	Tarjeta de interfaz de red
OMC	Centro de operación y mantenimiento
RT	Radio terminal
SAD	Sistema de Adquisición de Datos
ST	Terminal del Sistema
SwMI	Infraestructura de Manejo y Conmutación
UDT	Terminal de Datos del Usuario

4.1.2 Definiciones

- DAS:** Servidor de Aplicación de Datos externo a la red Tetrapol, que maneja aplicaciones basadas en IP.
- DNC:** Es una entidad funcional del SwMI que realiza las funciones de ruteo a redes externas que trabajen con IP.
- Mensaje downlink:** Mensaje transmitido del SwMI a la ST y de ésta a la UDT.
- Mensaje uplink:** Mensaje transmitido de la UDT hacia la ST, y de ésta al SwMI.
- MSW:** Es el conmutador principal de la red.
- RT:** Son las radio terminales móviles que se conectan a la red vía un enlace de radio.
- ST:** Son las terminales del sistema Tetrapol que permiten al usuario tener acceso a la red; pueden ser de dos tipos: radio terminales (RT) o terminales alámbricas.
- SwMI:** Es toda la infraestructura fija con la que cuenta la red, comprende las estaciones base, el radio switch y el OMC (Centro de operación y mantenimiento).
- UDT:** Terminal de datos que se conecta a una ST para emplear los recursos del sistema Tetrapol.

4.2 ARQUITECTURAS DE LA UDT

Una UDT es un equipo terminal de datos que al conectarse a una ST puede emplear los recursos de transmisión de datos del sistema Tetrapol. La transmisión de datos puede dirigirse a otra UDT o bien, a un Servidor de Aplicación de Datos (DAS, *Data Application Server*), que es un equipo externo al sistema Tetrapol, que se accede mediante un controlador de red (DNC), conectado al MSW.

Una UDT puede ser una PC o cualquier otro equipo electrónico que cumpla con alguna de las dos arquitecturas de UDT que reconoce Tetrapol. Una de ellas se basa en los protocolos STUTEL¹ y SDP (*Submit Delivery Protocol*), y la otra en el ambiente Microsoft, incorporando un controlador NDIS de capa inferior, varios controladores de protocolo y las aplicaciones de Windows NT. La figura 4.1 muestra estas dos arquitecturas y sus respectivos protocolos.

¹ Nombre comercial del ETS 300 075.

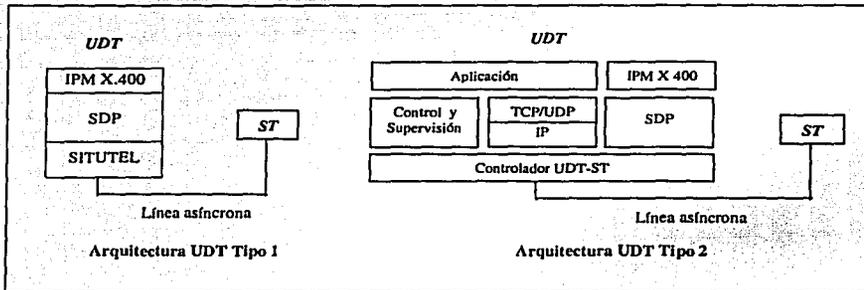


Fig. 4.1 Arquitecturas de UDT.

La arquitectura "tipo 1"² utiliza los protocolos SDP y STUTEL para manejar las aplicaciones de mensajería que proporciona el protocolo IPM (*Interpersonal Messaging*) X.400. Por otra parte, la arquitectura UDT "tipo 2" comprende 3 niveles principales. El primer nivel es el controlador UDT-ST, que permite establecer la comunicación entre una UDT y una ST (interfaz abierta UDT-ST). El segundo nivel comprende tres conjuntos de protocolos: Control y supervisión, TCP-UDP/IP y SDP. El protocolo de control y supervisión realiza funciones que le permiten al usuario tener un mayor control sobre la interfaz de radio. Los protocolos TCP-UDP/IP realizan las funciones de transporte y direccionamiento de datos. El protocolo SDP maneja las funciones de entrega y recepción de mensajes y es específico para trabajar con el protocolo de mensajería IPM X.400. Finalmente, el tercer nivel, correspondiente a la aplicación, es muy variado.

El Módulo Adquisidor de Datos emula una arquitectura UDT tipo 2 para comunicarse con la ST. Los protocolos que emplea son: el controlador UDT-ST, UDP/IP y un protocolo de aplicación específico para el adquisidor. Estos protocolos se implementaron en el Módulo Adquisidor de Datos completamente mediante software, éste se describe en el capítulo 5.

En la fig. 4.2 aparecen los protocolos empleados por la UDT (Módulo Adquisidor de Datos), por el DAS y por el DNC.

² Para mayor información véase PAS Tetrapol.

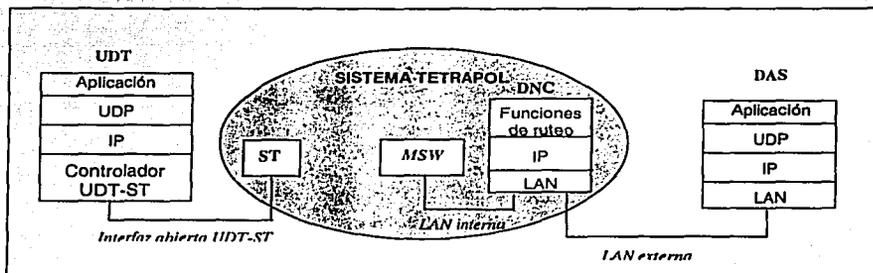


Fig. 4.2 Esquema de la comunicación con la arquitectura UDT tipo 2.³

4.3 CONTROLADOR UDT-ST

Este controlador sirve para establecer la interfaz abierta UDT-ST, que permite conectar la UDT a la ST, y consta de los tres niveles siguientes:

- Nivel físico
- Nivel de enlace: protocolo MPAP (*Mobile PC Asynchronous Protocol*)
- Nivel de aplicación: protocolo DTAP (*Data Transmission Applicative Protocol*)

4.3.1 Nivel físico

El nivel físico comprende las características físicas y eléctricas de la conexión, y un protocolo para identificar el tipo de UDT. La conexión se establece vía una línea serial en modo asíncrono, de acuerdo al protocolo RS232 de la EIA (*Electronic Industries Association*) o a las normas V24 y V28 de la ITU (*International Telecommunications Union*).

4.3.1.1 Características generales de las señales RS232⁴

El estándar RS232 utiliza una lógica negativa bipolar, en que un voltaje negativo representa un '1' lógico, y un voltaje positivo representa un '0' lógico. En la fig. 4.3 se muestra esta lógica.

³ Fuente: PAS Tetrapol.

⁴ Para ver la información completa de las normas V.24 y V.28 de ITU (RS-232), véase www.itu.com.

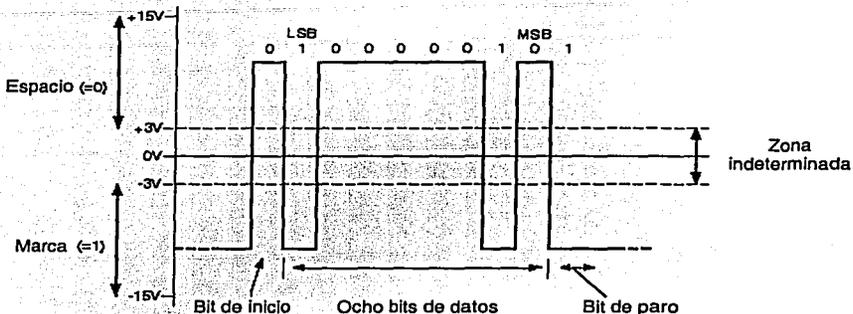


Fig. 4.3 Características del estándar RS232 utilizado.

Las únicas señales empleadas en la conexión UDT-ST son SD "send data" y RD "receive data".

La señal SD está activa cuando los datos son transmitidos desde el dispositivo DTE hacia el dispositivo DCE. Cuando no se están transmitiendo datos, la señal se mantiene con un voltaje negativo ('1' lógico).

La señal RD está activa cuando el dispositivo DTE recibe datos provenientes del dispositivo DCE. Cuando ningún dato se está transmitiendo, la señal se mantiene con un voltaje negativo ('1' lógico).

4.3.1.2 Características de la comunicación

Las características de la comunicación serial asíncrona son:

- Velocidad de transmisión a 4800 bauds
- 8 bits de datos
- Sin paridad
- Un bit de inicio
- Un bit de paro
- Primero el bit menos significativo

4.3.1.3 Identificación del tipo de UDT conectada

Como una ST puede trabajar ya sea con una UDT "tipo 1" o con una UDT "tipo 2". La ST debe identificar el tipo de UDT conectada y ajustarse a los protocolos soportados por ésta.

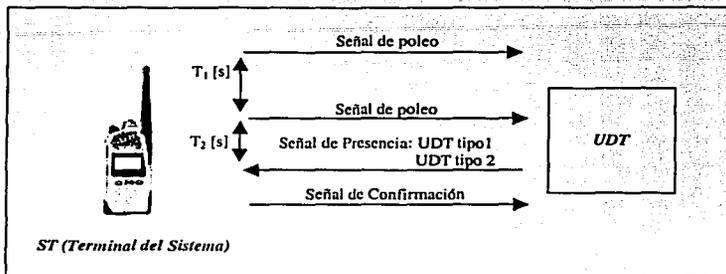


Fig. 4.4 Identificación de la arquitectura de la UDT

La conexión se realiza a iniciativa de la ST, que envía cada T_1 [s] una señal de poleo. La UDT debe contestar en menos de T_2 [s] con la señal de presencia, que indica el tipo de UDT. Cuando la ST recibe esta señal, envía la señal de confirmación, que sirve para detener la emisión de la señal de presencia. Entonces cada lado reinicia el enlace de acuerdo con el protocolo del nivel superior.⁵ La identificación de UDT se muestra en la fig. 4.4.

4.3.2 Protocolo MPAP (*Mobile PC Asynchronous Protocol*)

El protocolo MPAP corresponde al nivel de enlace de la interfaz abierta UDT-ST. Este protocolo se encarga de establecer la conexión y realizar el intercambio de datos.⁶ Para ello, este protocolo utiliza cuatro tipos de paquetes o *frames*:

- Paquetes de conexión (CNX)
- Paquetes de confirmación de conexión (ACQ_CNX)
- Paquetes de datos (DATA)
- Paquetes de confirmación de datos (ACQ_DATA)

4.3.3 Protocolo DTAP (*Data Transmission Applicative Protocol*)

El protocolo DTAP correspondiente al nivel de aplicación de la interfaz abierta UDT-ST, se compone de dos flujos: flujo interno de señalización y flujo de intercambio de mensajes de datos.

El flujo de señalización se utiliza para establecer la conexión DTAP y dirigir la transmisión y la supervisión de la información de señalización y control entre la ST y la UDT. Este flujo comprende los mensajes: STU_CONNECT, UTS_CONNECT_ACK,

⁵ Para mayor información véase PAS Tetrapol.

⁶ Para mayor información del protocolo MPAP véase PAS Tetrapol.

STU_ST_PRESENCE, UTS_ST_PRESENCE_ACK, STU_NETWORK_INFORMATION, UTS_RINGING y STU_SERVICE_MESSAGE.

El flujo de intercambio de mensajes de datos permite el intercambio de la información necesaria para la transmisión *uplink* (UDT hacia ST) o *downlink* (ST hacia UDT) de los mensajes de datos. Existen 2 flujos diferentes para la transmisión *uplink* de los mensajes de datos: transmisión de datos con el servicio de poleo y transmisión de datos con otro servicio. Ambos flujos son completamente independientes.

La transmisión *uplink* con otro servicio comprende dos mensajes: UTS_DATA_U y STU_RADIO_TRANSMISSION_ACK.

La transmisión *downlink* únicamente utiliza un mensaje: STU_DATA_D.⁷

4.4 Protocolo Internet (IP)

El Protocolo Internet es el protocolo principal de la capa de red no orientado a conexión. IP trabaja con paquetes de información denominados datagramas, que consisten en la información proveniente de la capa de transporte más el encabezado IP. Las tareas principales del Protocolo Internet son el direccionamiento de los datagramas de información entre computadoras y la administración del proceso de fragmentación de los datagramas. IP es responsable del enrutamiento de los datagramas, determinado a dónde se enviarán, así como las rutas alternas en caso de problema.

4.4.1 Encabezado

El encabezado del Protocolo Internet utiliza por lo menos 20 bytes agrupados en 5 palabras de cuatro bytes. Su longitud máxima es de seis palabras de 32 bits (24 bytes) cuando se incluyen todos los campos opcionales.

Versión 4 bits	Longitud del encabezado 4 bits	Tipo de servicio 8 bits	Longitud total del datagrama 16 bits	
Identificación 16 bits			Banderas 3 bits	Desplazamiento del fragmento 13 bits
Tiempo de vida 8 bits		Protocolo 8 bits	Suma de verificación del encabezado 16 bits	
Dirección IP del origen 32 bits				
Dirección IP del destino 32 bits				
Opciones IP hasta de 32 bits				
Porción de datos del datagrama				

Fig. 4.5 Formato del datagrama IP.

⁷ Para mayor información del protocolo DTAP véase PAS Tetrapol.

Número de versión

El número de versión es un campo de 4 bits que contiene el número de versión de IP que soporta el software de protocolo. Este campo es requerido para decodificar el resto del encabezado, que varía en cada edición de estándares de IP. Si la versión no coincide, el datagrama se descarta.

Longitud del encabezado

La longitud del encabezado es un campo de 4 bits que indica la longitud total del encabezado IP, indicado en palabras de 32 bits. El encabezado más corto es de cinco palabras, pero el empleo de campos adicionales puede aumentar el tamaño.

Tipo de servicio

El campo de tipo de servicio consta de 8 bits y le indica a IP cómo procesar correctamente el datagrama. Los primeros 3 bits indican la precedencia del datagrama, que puede ir desde cero hasta 7, mientras más alto sea el número mayor es su importancia. Las siguientes 3 banderas de 1 bit controlan el retraso, el rendimiento y la confiabilidad del datagrama. Los 2 últimos bits del campo no se utilizan.

Longitud del datagrama

Este campo da la longitud total del datagrama en bytes, incluye el encabezado. La longitud del área de datos se puede calcular sustrayendo de este valor la longitud del encabezado. El tamaño del campo de longitud total del datagrama es de 16 bits, por lo que la longitud máxima de un datagrama es 65,535 bytes (incluyendo al encabezado).

Identificación

Este campo contiene un número que es un identificador único creado por el nodo emisor. Este número se requiere al volver a ensamblar mensajes fragmentados, para asegurar que los fragmentos de un mensaje no se mezclen con los de otros mensajes.

Banderas

El campo de banderas es un campo de 3 bits, el primero no se utiliza. Los dos bits siguientes corresponden a las banderas conocidas como DF (no fragmentar) y MF (más fragmentos), las cuales controlan el manejo de los datagramas cuando la fragmentación es deseable.

Si la bandera DF tiene un '1', el datagrama no se puede fragmentar bajo ninguna circunstancia. Si el nodo actual no puede enviarlo sin fragmentarlo y este bit está en '1', el datagrama se descarta y se envía un mensaje de error al emisor.

Si la bandera MF está en '1', el datagrama actual vendrá seguido por más paquetes, que deben ser reensamblados para entender el mensaje completo. El último

fragmento enviado tiene la bandera MF en '0', con esto el receptor sabe cuando detener la espera de más fragmentos.

Desplazamiento del fragmento

Este campo le permite al IP reensamblar un mensaje fragmentado en el orden apropiado. Si la bandera MF se encuentra en '1', el desplazamiento del fragmento indica la posición que este fragmento ocupa en el mensaje completo.

Este es un campo de 13 bits, por lo que los desplazamientos se calculan en unidades de 8 bytes, que corresponden a la longitud máxima del paquete de 65,535 bytes.

Tiempo de vida (TTL)

Este campo proporciona el tiempo en segundos que un datagrama puede permanecer en la red antes de descartarse. El nodo emisor establece este tiempo, que normalmente es de 15 o 30 segundos.

Los estándares IP estipulan que el campo debe disminuirse por lo menos un segundo por cada nodo que procese el paquete, aún si el proceso resultó menor de un segundo. También cuando un datagrama se recibe en una compuerta, el tiempo de llegada se registra, de tal forma que si el datagrama debe esperar para su proceso, este tiempo se descontará de su TTL.

Protocolo de transporte

Este campo contiene el número de identificación del protocolo de transporte con el que se ha manejado el paquete. Actualmente hay alrededor de 50 protocolos. TCP es el número 6 y UDP el número 17.

Suma de verificación del encabezado

El número que aparece en este campo es la suma de verificación del encabezado del protocolo, pero no de los campos de datos. Puesto que el campo TTL disminuye en cada nodo, la suma de verificación cambia también. El algoritmo que obtiene la suma de verificación toma el complemento a uno de la suma de 16 bits de todas las palabras de 16 bits.

Direcciones del emisor y del destino

Los campos de direcciones del emisor y del destino del encabezado IP contienen las direcciones IP de 32 bits de los dispositivos emisor y destino. Estos campos se establecen al crearse el datagrama, y no se alteran durante el enrutamiento.

Direcciones IP

IP utiliza una dirección de 32 bits para identificar una máquina y la red a la que pertenece. Hay cuatro formatos para la dirección IP, denominados desde Clase A hasta Clase D, cada uno se utiliza dependiendo del tamaño de la red. La clase se identifica mediante las primeras secuencias de bits, para la Clase A un bit en '0', para la Clase B un par de bits en '10', para la Clase C tres bits en '110' y para la Clase D cuatro bits en '1110'. Las direcciones de Clase A corresponden a redes grandes que utilizan 24 bits para indicar la dirección local. La dirección de red se conserva en 7 bits, lo que limita el número de redes que se pueden identificar. Las direcciones de Clase B sirven para redes de tamaño intermedio, con direcciones locales o de anfitrión de 16 bits y direcciones de red de 14 bits. Las direcciones de Clase C son para redes que sólo emplean 8 bits para la dirección local, limitando el número de dispositivos a 256. Para la dirección de red hay 21 bits. Las redes Clase D se usan con fines de multidifusión. La fig. 4.6 muestra el formato de las cuatro clases de direcciones IP y la tabla 4.1 el número de redes y equipos que cada clase puede contener.

Clase A	0	Identificador de Red (7 bits)	Identificador de equipo (24 bits)
Clase B	10	Identificador de Red (14 bits)	Identificador de equipo (16 bits)
Clase C	110	Identificador de Red (21 bits)	Identificador de equipo (8 bits)
Clase D	1110	Identificador de Red (28 bits)	

Fig. 4.6 Formato de las direcciones IP.

Clase	Direcciones IP	Número de redes	Número de equipos
A	1-126	126	16777216
B	128-191	16384	65536
C	192-223	2097152	256
D	224-240	268435456	0

Tabla 4.1 Cantidad de redes y equipos de cada clase IP.

Opciones

Existen 8 posibles opciones que sirven principalmente para tener un mayor control de la red.

Relleno

El contenido del área de relleno depende de las opciones seleccionadas. Por lo general se utiliza para asegurar que la longitud del encabezado corresponda a un número redondeado de bytes.

4.4.2 El protocolo IP en el sistema Tetrapol

El protocolo IP juega un papel muy importante en el acceso a redes externas al sistema Tetrapol. Sin embargo se debe considerar que no todos los elementos del encabezado IP son soportados y otros son utilizados para transmitir información propia del sistema Tetrapol. De esta manera se encuentran tres tipos de elementos: restablecidos, no soportados y procesados. "Restablecido" significa que los elementos IP no son transmitidos dentro del sistema Tetrapol, pero se restablecen en la interfaz externa. Mientras que "No soportado" significa que ese elemento IP no se procesa en el sistema Tetrapol y no debe ser usado por la aplicación. Finalmente "Procesado" significa que los elementos IP son usados para enrutar la información por el sistema Tetrapol. La tabla 4.2 muestra si los elementos del encabezado IP son soportados, restablecidos o procesados por Tetrapol.

Elemento	Restablecido	No soportado	Procesado
Versión	X(V4)		
Longitud del encabezado	X		
Tipo de servicio	X		
Longitud total	X		
Identificación	X		
Banderas	X		
Desplazamiento del fragmento		X	
Tiempo de vida		X	
Protocolo	X		
Suma de verificación del encabezado	X		
Dirección IP fuente			X
Dirección IP destino			X
Opciones		X	

Tabla 4.2 Status de los elementos del encabezado IP en Tetrapol⁸

Como se muestra en la tabla 4.2, los campos desplazamiento del fragmento y opciones no son soportados, esto se debe a que Tetrapol no admite la fragmentación IP. Por otra parte, los elementos "procesados" por Tetrapol son las direcciones fuente y destino, las cuales sirven para especificar la dirección RFSI⁹ que identifica a una ST y a su asociada UDT, con el formato de una dirección IP Clase A, la que ofrece el mayor rango de valores para el direccionamiento de estaciones.

Las direcciones IP Clase A que contengan una dirección de red diferente de una de Tetrapol, las direcciones IP Clase B y las direcciones IP Clase C están autorizadas para designar un receptor en una red IP externa.¹⁰

⁸ Fuente: PAS Tetrapol.

⁹ Para mayor información véase PAS Tetrapol.

¹⁰ Fuente: PAS Tetrapol.

4.5 Protocolo de Datagrama de Usuario (UDP)

Es el protocolo de la capa de transporte no orientado a conexión que forma parte de la familia de protocolos TCP/IP. Las comunicaciones sin conexión no proporcionan confiabilidad, lo que significa que no hay indicación para el dispositivo emisor de que un mensaje se haya recibido correctamente. UDP es por tanto un protocolo simple que intercambia datagramas sin reconocimientos o entregas garantizadas, y requiere por ello que además del procesamiento de errores, la retransmisión también sea manejada por otros protocolos.

4.5.1 Encabezado

El encabezado UDP se compone de cuatro campos de 16 bits: puerto origen, puerto destino, longitud del datagrama y suma de verificación.

Puerto origen 16 bits	Puerto destino 16 bits
Longitud 16 bits	Suma de verificación 16 bits

Fig. 4.7 Encabezado UDP.

Los puertos origen y destino constan de números de puerto del protocolo UDP de 16 bits que se utilizan para demultiplexar datagramas para los procesos de recepción de la capa de aplicación. El campo de longitud especifica la longitud de los datos y del encabezado UDP. La suma de verificación ofrece una verificación de integridad (opcional) del encabezado y los datos de UDP.

4.5.2 Puerto fuente UDP en Tetrapol

El sistema Tetrapol hace un uso específico del puerto fuente UDP, para identificar el servicio de radio que se usará en la interfaz aérea, la prioridad y la encriptación del mensaje.¹¹

El servicio de radio especifica las características con las que la red debe transmitir el mensaje en la interfaz aérea de acuerdo a los protocolos de Tetrapol.¹²

La prioridad indica la preferencia que se da a un mensaje sobre otros. Existen tres tipos: rutina, urgente y flash.

La encriptación indica si un mensaje puede o no ser transmitido sin encriptar. Existen dos tipos de encriptación: obligatoria y opcional.

¹¹ La codificación del puerto fuente UDP aparece en PAS Tetrapol.

¹² La descripción de los servicios de radio aparece en PAS Tetrapol.

4.6 PROTOCOLO DE APLICACIÓN

El DAS y los módulos adquirentes de datos intercambian cinco tipos de mensajes: presencia, información almacenada en las memorias, valores actuales, alarmas y configuración.

Los mensajes de presencia, son enviados por cada Módulo Adquisidor de Datos en un lapso configurable por el usuario (el tiempo predefinido es igual a 10 [min]). Este tipo de mensaje le indica al DAS que dicho módulo se encuentra actualmente conectado a una RT, por lo que es posible establecer comunicación con él.

Los mensajes involucrados en la transmisión de información de las memorias son de dos clases: petición y respuesta. La petición es un mensaje *downlink* (DAS-Módulo Adquisidor de Datos) que aparece cuando un usuario elige obtener la información de las memorias de un módulo adquirente activo. La respuesta es un mensaje *uplink* (Módulo Adquisidor de Datos-DAS), que puede enviarse en uno o varios paquetes de información. El primero de estos paquetes incluirá la fecha y hora en que se inició el muestreo.

Los mensajes relacionados con la transmisión de valores actuales también son de dos clases: petición y respuesta. La petición es un mensaje *downlink* que aparece cuando un usuario decide conocer qué valores tienen en ese momento las variables que mide un dispositivo activo. La respuesta es un mensaje *uplink* contenido en un solo paquete de información que le transmite al DAS los valores actuales.

Existen dos tipos de mensaje de alarma, uno de ellos le indica al DAS qué variable ha sobrepasado los límites permisibles y, por tanto ha originado la emisión de este mensaje, y el otro informa al DAS que la máxima capacidad de almacenamiento de datos se ha alcanzado. Cuando el DAS recibe alguno de estos mensajes, envía la confirmación de alarma, cuya finalidad es notificar al Módulo Adquisidor de Datos que recibió la alarma y con esto evitar la retransmisión.

El mensaje de configuración es un mensaje *downlink*, con el que, el DAS indica al dispositivo el tiempo de muestreo, el tiempo de presencia y qué canales debe activar. Además, debido a que los módulos adquirentes de datos no cuentan con reloj de tiempo real, este mensaje les indica la fecha y hora del inicio del muestreo.

Los formatos de estos mensajes se muestran en la fig. 4.8.

M	E	# mensaje (1 byte)	Puerto UDP (2 bytes)
---	---	--------------------	----------------------

Mensaje de petición de la información de las memorias

V	A	# mensaje (1 byte)	Puerto UDP (2 bytes)
---	---	--------------------	----------------------

Mensaje de petición de valores actuales

C	O	Fecha y hora (24 bytes)	Tiempo de muestreo (1 byte)	Canales seleccionados (1 byte)	Tiempo de presencia (1 byte)
---	---	-------------------------	-----------------------------	--------------------------------	------------------------------

Mensaje de configuración

A	L	R
---	---	---

Mensaje de confirmación de alarma recibida

MENSAJES DOWNLINK (DAS - Módulo Adquisidor de Datos)

P	R	E	S	Nombre del dispositivo (5 bytes)
---	---	---	---	----------------------------------

Mensaje de presencia

V	A	# mensaje (1 byte)	Datos (Longitud variable)
---	---	--------------------	---------------------------

Mensaje de valores actuales

M	E	# mensaje (1 byte)	# segmento (1 byte)	Fecha y hora (24 bytes)	Tiempo de muestreo	Canales (1 byte)	Datos (Longitud variable)	Indicador último segmento
---	---	--------------------	---------------------	-------------------------	--------------------	------------------	---------------------------	---------------------------

Mensaje de información de las memorias, si es el primer segmento.

M	E	# mensaje (1 byte)	# segmento (1 byte)	Datos (Longitud variable)	Indicador último segmento
---	---	--------------------	---------------------	---------------------------	---------------------------

Mensaje de Información de las memorias, si no es el primer segmento.

A	L	Nombre del dispositivo (5 bytes)	Valor leído y canal (2 bytes)
---	---	----------------------------------	-------------------------------

Mensaje de alarma.

F	M	Nombre del dispositivo (5 bytes)
---	---	----------------------------------

Mensaje de memoria llena

MENSAJES UPLINK (Módulo Adquisidor de Datos - DAS)

Fig. 4.8 Mensajes del protocolo de aplicación.

Los mensajes se componen de dos tipos de campos: información y control. Los campos de información incluyen el campo datos de los mensajes *uplink*, y los bytes de tiempo de muestreo, tiempo de presencia y canales seleccionados del mensaje *downlink* de configuración. Los campos de control incluyen tipo de mensaje, número de mensaje, número de segmento e indicador de último segmento.

El número de mensaje permite autenticar el mensaje. El número de segmento y el indicador de último segmento sirven para indicar el orden de los fragmentos que se incluyen en los mensajes *uplink* de información de memorias.¹³

¹³ La segmentación se maneja en la capa de aplicación debido a que el sistema Tetrapol no soporta la fragmentación IP (PAS Tetrapol).

CAPÍTULO 5

SOFTWARE DEL MÓDULO ADQUISIDOR DE DATOS

En este capítulo se explica la programación realizada para el Módulo Adquisidor de Datos, cuyo objetivo es realizar la adquisición y el almacenamiento de datos, establecer comunicación con una ST¹ para transmitir inalámbricamente los datos adquiridos, detectar y emitir alarmas, así como manipular un *display* para indicar el modo de operación de la red Tetrapol (normal, directo o fallback), si ha ocurrido una falla en la transmisión de datos o algún error en la memoria.

A continuación se describen cada una de las rutinas que componen al software.

5.1 RUTINA PRINCIPAL

Como se muestra en el diagrama de flujo de la fig. 5.1, la rutina principal establece los parámetros de inicio de la comunicación, del convertidor y del *display*. Posteriormente configura y habilita las interrupciones tanto de la comunicación, como de los dos temporizadores con los que cuenta, y verifica constantemente si alguna bandera ha sido activada.

La comunicación se configura de acuerdo al estándar RS-232, con los siguientes parámetros:

- velocidad de 4800 *bauds*
- 8 bits de datos
- 1 bit de inicio
- 1 bit de paro
- sin paridad.

El estado de la comunicación se establece en: "físicamente no conectado", es decir en espera de la petición de conexión física por parte de la RT.

¹ Cabe mencionar que aunque las Terminales del Sistema (ST) pueden ser Terminales Conectadas Físicamente o Radio Terminales (RT), en este capítulo el término ST se refiere principalmente a las RT, puesto que son éstas últimas las que se utilizan en el presente proyecto.

Los dos temporizadores con los que cuenta el μC AT90S8515 (de 8 bits y 16 bits) se configuran para interrumpir cada 0.05 [s] y cada 12 [s].

El programa verifica constantemente el estado de las banderas siguientes: memoria llena, alarma, tiempo ACK, tiempo de muestreo, tiempo Radio_Trans_ACK, tiempo de presencia UDT-DAS y *display*.

La bandera de memoria llena informa que se ha agotado la máxima capacidad de almacenamiento, por que es necesario emitir el mensaje *uplink* correspondiente.

La bandera de alarma indica que alguna de las variables que se están monitoreando ha alcanzado un valor crítico, y por ello cuando se activa, el programa construye el mensaje² de alarma, llama a la rutina de transmisión de mensajes *uplink* para su envío al DAS y, desactiva esta bandera.

La bandera de tiempo ACK señala que el tiempo que la UDT debe esperar por la trama MPAP ACK se ha agotado. Dicha trama confirma que la ST recibió el mensaje de datos. Por tal motivo, cuando esta bandera se encuentra activa, el programa retransmite el mensaje (hasta 3 veces), activa nuevamente el tiempo ACK y deshabilita esta bandera.

Cuando la bandera de tiempo de muestreo está habilitada, el programa llama a las rutinas del convertidor y de almacenamiento en memoria externa para realizar la adquisición de datos.

La bandera de tiempo Radio_Trans_ACK indica que se agotó el lapso que la UDT debe esperar por el mensaje DTAP STU_RADIO_TRANS_ACK, que informa a la UDT el éxito o fracaso de la transmisión de un mensaje *uplink*. Cuando esta bandera se activa, el programa llama a la rutina de transmisión de mensajes *uplink* para retransmitir el mensaje en cuestión y desactiva esta bandera.

La bandera de tiempo de presencia UDT-DAS indica que el Módulo Adquisidor de Datos debe transmitir su mensaje de presencia DTAP, por lo cual llama a la rutina de transmisión de mensajes *uplink* y deshabilita esta bandera.

La bandera de *display* indica que se debe desplegar alguno de los mensajes siguientes: modo de operación de la red, la falla ocurrida en la transmisión de datos o un error en las memorias. Por ello el programa llama a la rutina del *display* y desactiva esta bandera.

² Véase capítulo 4 sección 5 "Descripción del protocolo de aplicación".

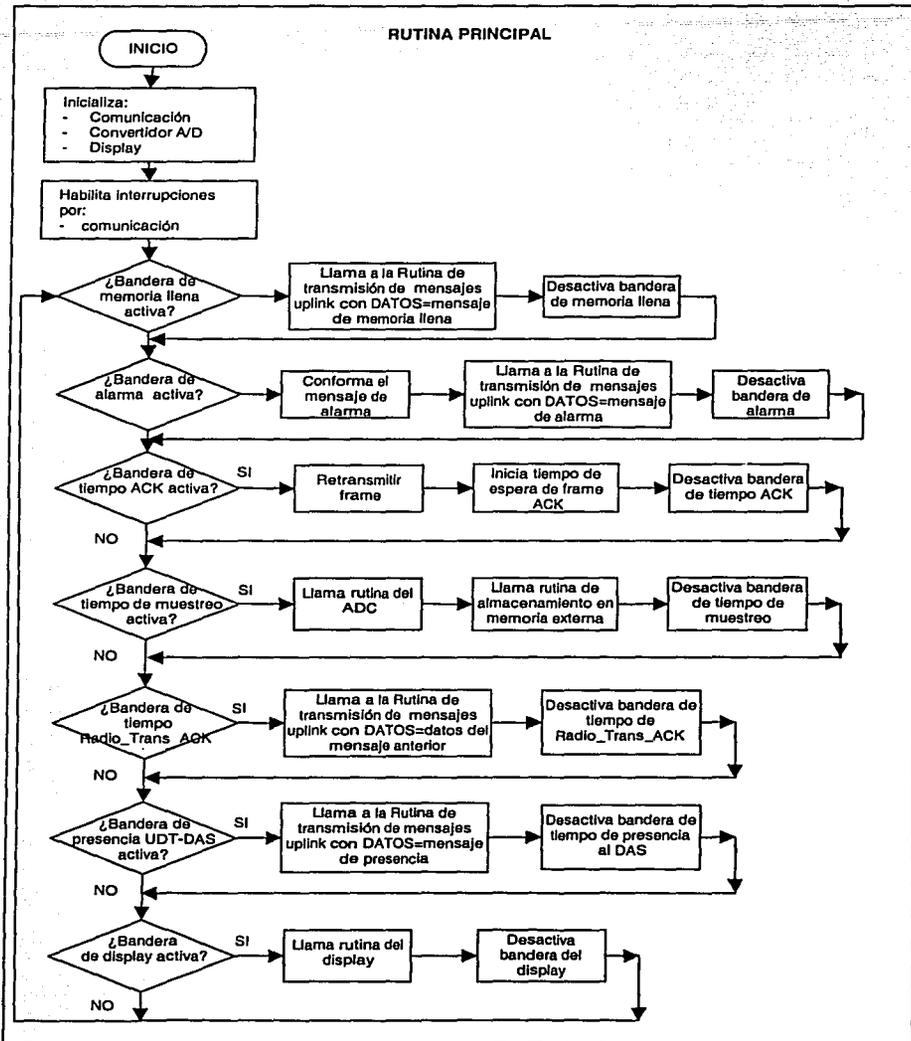


Fig. 5.1 Diagrama de flujo de la rutina principal.

5.2 INTERRUPCIÓN POR COMUNICACIÓN

Como se muestra en el diagrama de flujo de la fig. 5.2, cuando ocurre una interrupción por comunicación, el programa verifica si el carácter recibido corresponde a un mensaje MPAP, o a una petición de conexión física, e ignora cualquier otro carácter recibido.

La petición de conexión física, se contesta de acuerdo al protocolo de identificación de UDT que se describió en el capítulo 4.

Cuando se trata de un mensaje MPAP, el programa verifica si la conexión física ya se encuentra establecida, ya que de no estarlo ignorará el paquete. Si por el contrario, ya se estableció la conexión física, el programa guarda en RAM todo el paquete o *frame*, realiza la suma de verificación de los caracteres recibidos y la compara con la indicada en el paquete. En caso de error, el paquete se descarta, y regresa a la rutina principal. Si la suma de verificación fue correcta, el programa identifica el estado de la comunicación MPAP en el que se encuentra, en función de dicho estado llama a la rutina correspondiente para que se realice el procesamiento de la información recibida. La conexión MPAP tiene tres estados: desconectado, conectado (espera datos) y en espera de ACK.

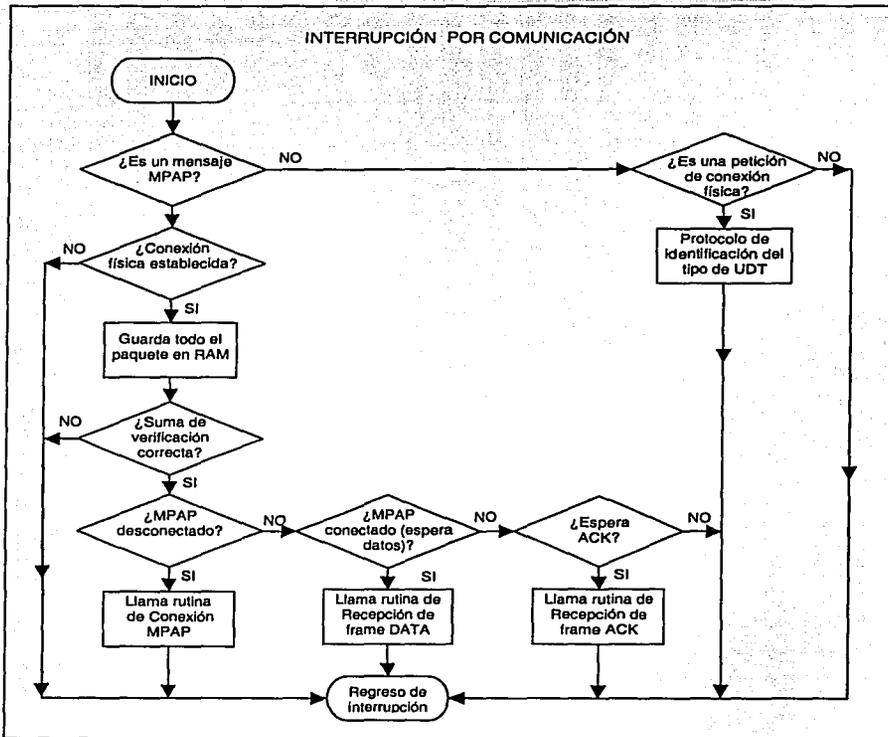


Fig. 5.2 Diagrama de flujo de la interrupción por comunicación.

5.2.1 Rutina de Conexión MPAP

Cuando el nivel MPAP se encuentra en estado "desconectado", únicamente acepta el mensaje de conexión CNX, descartando cualquier otro tipo de mensaje MPAP. El mensaje CNX, indica que la ST desea establecer el nivel de enlace con la UDT (Módulo Adquisidor de Datos), que contesta con el mensaje ACQ_CNX, inicia en '0' los contadores de secuencia (COUNTERn1 y COUNTERn2), y establece el estado MPAP "conectado".

5.2.2 Rutina de Recepción del Frame DATA

Cuando el nivel MPAP se encuentra en estado "conectado", acepta los mensajes CNX y DATA. El mensaje CNX indica que la ST ha concluido el nivel de enlace y está tratando de establecer la conexión MPAP de nuevo, por lo que, la UDT sigue el procedimiento descrito en la rutina de conexión MPAP. Por otra parte, si se trata del mensaje DATA, la rutina realiza la autenticación del paquete mediante los números de secuencia, llama a la rutina de Protocolo DTAP para decodificar el mensaje encapsulado en el *frame* DATA, actualiza los contadores de secuencia y contesta con el mensaje ACQ_DATA para indicar que recibió el mensaje.

En caso de que el número de secuencia corresponda con el valor previo, la rutina envía el *frame* ACQ_DATA correspondiente al mensaje atrasado.³

5.2.3 Rutina de Recepción de Frame ACK

Cuando el nivel MPAP se encuentra en espera de ACK, acepta los mensajes CNX, DATA y ACQ_DATA. Los mensajes CNX y DATA se procesan como se describió en las rutinas de conexión MPAP y de recepción del *frame* DATA.

Si se trata de un *frame* ACQ_DATA, la rutina autentica el mensaje mediante los números de secuencia, desactiva el tiempo de espera de mensaje ACK (ACK time) y establece el estado MPAP "conectado".⁴

5.2.4 Rutina de Protocolo DTAP

Esta rutina decodifica los mensajes del nivel de aplicación de la Interfaz abierta UDT-ST, que vienen encapsulados en el *frame* MPAP DATA. Como se muestra en la fig. 5.3, esta rutina reinicia el tiempo de presencia de la terminal⁵, ya que cualquier mensaje DTAP proveniente de la ST indica que la conexión del nivel de aplicación sigue activa. Posteriormente, la rutina identifica cual de los cinco posibles mensajes DTAP ha recibido: STU_CONNECT, STU_ST_PRESENCE, STU_NETWORK_INFORMATION, STU_RADIO_TRANSMISSION_ACK o STU_DATA_D.

En el caso del mensaje STU_CONNECT, la rutina respalda en la EEPROM interna del μ C los parámetros recibidos para un futuro uso, y contesta a la RT con el mensaje DTAP de confirmación o *acknowledgement*: UTS_CONNECT_ACK.

Para el mensaje STU_ST_PRESENCE, la rutina responde con el mensaje DTAP de confirmación: UTS_ST_PRESENCE_ACK.

³ La rutina "recepción del frame DATA" se pega al diagrama de flujo que aparece en PAS Tetrapol.

⁴ La rutina "recepción del frame ACK" se pega al diagrama de flujo que aparece en PAS Tetrapol.

⁵ Este tiempo se encuentra en PAS Tetrapol como T_Presence.

Cabe mencionar, que para transmitir los mensajes de confirmación, se llama a una subrutina denominada "loadbuffer", que se encarga de añadir el encabezado y finalizadores correspondientes al protocolo MPAP, además se encarga de establecer el estado MPAP en espera de *frame ACK* y de activar el tiempo de espera de *frame ACK* (ACK time).⁶

Cuando se trata del mensaje STU_NETWORK_INFORMATION, la rutina respalda en la EEPROM interna del μ C el modo de operación de la red, y habilita la bandera de *display*, para mostrar cuál es el modo de operación vigente.

En el caso del mensaje STU_RADIO_TRANSMISSION_ACK, desactiva el tiempo de espera correspondiente a este mensaje y verifica si la confirmación fue positiva o negativa. Si es positiva informa que la BS recibió correctamente el mensaje UTS_DATA_U. En caso contrario, el programa revisa qué tipo de falla se presentó: excepcional, temporal, red sobresaturada, problema en la red y problema del subscriptor. Para desplegar el mensaje de falla ocurrida, se activa la bandera de *display*, y exceptuando si la falla fue un problema del subscriptor, el programa retransmitirá el mensaje hasta 3 veces.

Cuando se trata del mensaje STU_DATA_D, el programa llama a la rutina de procesamiento de datos procedentes del DAS, la cual se encarga de decodificar el mensaje enviado por el servidor.

⁶ La subrutina "loadbuffer" se apega al diagrama de flujo que aparece en PAS Tetrapol.

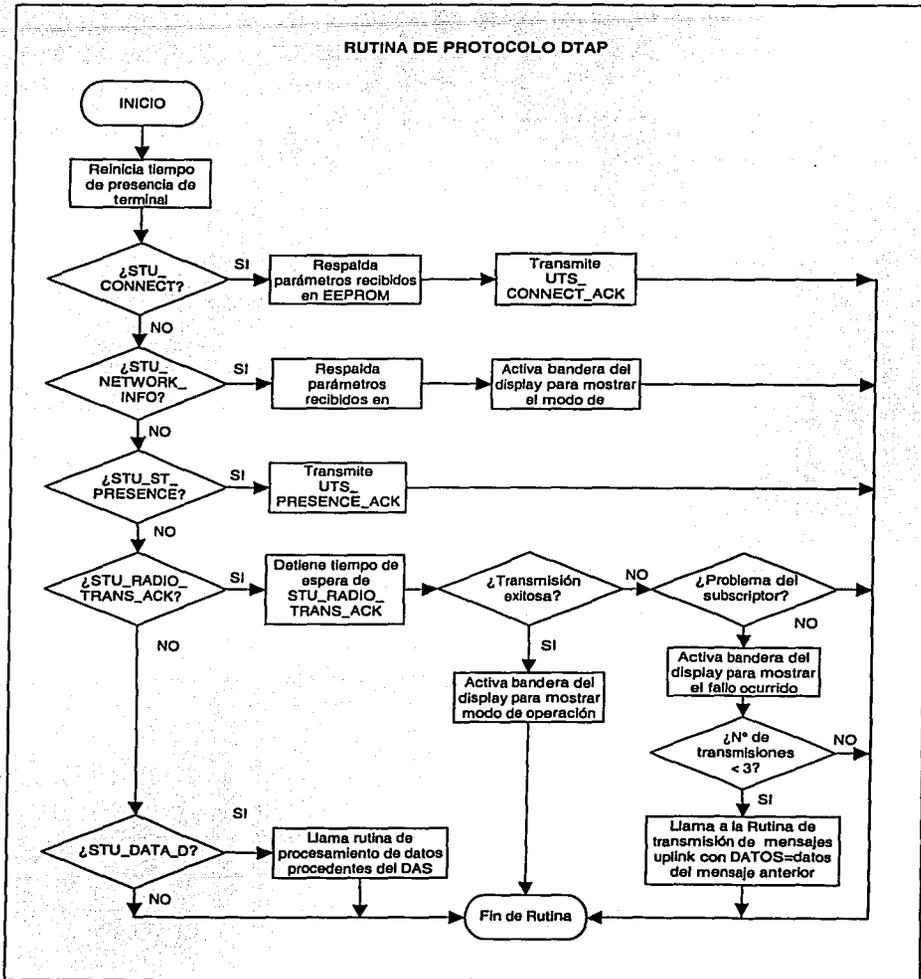


Fig. 5.3 Diagrama de flujo de la rutina del protocolo DTAP.

5.2.5 Rutina de Procesamiento de Datos procedentes del DAS

Como se muestra en la fig. 5.4, esta rutina comprueba que la dirección IP fuente corresponda con la del DAS que el Módulo Adquisidor de Datos tiene predeterminada, y verifica de qué mensaje se trata: petición de envío de información almacenada en memoria externa, petición de envío de valores actuales o configuración.

Cuando se trata de una petición de envío de datos de las memorias, la rutina revisa si la cantidad de información almacenada excede la longitud del *buffer* de transmisión (440 bytes incluyendo encabezados y finalizadores)⁷, en este caso realiza la fragmentación correspondiente, añade el encabezado del protocolo de aplicación⁸ y llama a la rutina de transmisión de mensajes *uplink* para enviar cada uno de los segmentos requeridos.

Cuando se trata de una petición de envío de valores actuales, el programa llama a la rutina del convertidor, añade el encabezado de la capa de aplicación correspondiente, y llama a la rutina de transmisión de mensajes *uplink*.

Cuando se trata de un mensaje de configuración, se guardan en RAM los parámetros recibidos: fecha y hora de inicio de la adquisición, tiempo de muestreo, canales seleccionados y tiempo de presencia. La fecha y hora se incluirán en el mensaje *uplink* de información de las memorias. Debido a que este mensaje indica el inicio de la adquisición, el apuntador de dirección de las memorias se coloca indicando la primera localidad de memoria, y con base en los parámetros: tiempo de muestreo y canales seleccionados, se configura e inicia la adquisición de datos.

⁷ El *buffer* de transmisión está limitado por la capacidad de la memoria RAM interna del μC , que es de 512 bytes, sin embargo las primeras 32 localidades se emplean como variables y las últimas 40 localidades constituyen el STACK.

⁸ Véase Capítulo 4 para mayor información del formato de los mensajes del protocolo de aplicación.

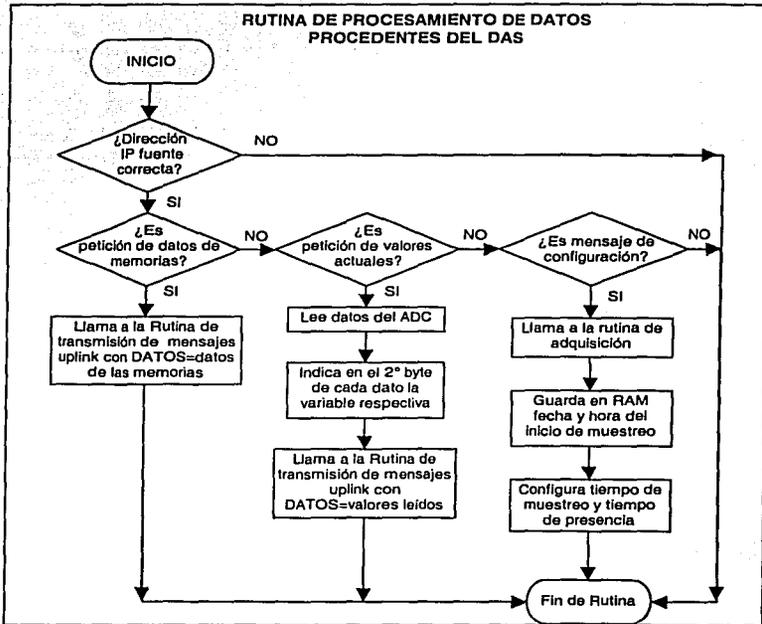


Fig. 5.4 Diagrama de flujo de de la rutina de procesamiento de datos procedentes del DAS.

5.2.6 Rutina de transmisión de mensajes *uplink*

Esta rutina, se encarga de transmitir a la ST los mensajes con dirección al DAS. Antes de enviar un mensaje *uplink*, se corrobora que la red se encuentre operando en modo normal, ya que sólo en este modo es posible la transmisión de datos, y también que no se esté esperando la confirmación RADIO_TRANS_ACK, puesto que no es posible transmitir ningún mensaje *uplink* hasta que no se haya recibido la confirmación (positiva o negativa del anterior) o bien se haya agotado el lapso de espera (300 [s]). Posteriormente, la rutina construye y transmite el mensaje DTAP UTS_DATA_U, establece el estado MPAP en espera de ACK, y activa tanto el tiempo ACK como el tiempo STU_Radio_Trans_ACK. En la fig. 5.5 se muestra el diagrama de flujo de esta rutina.

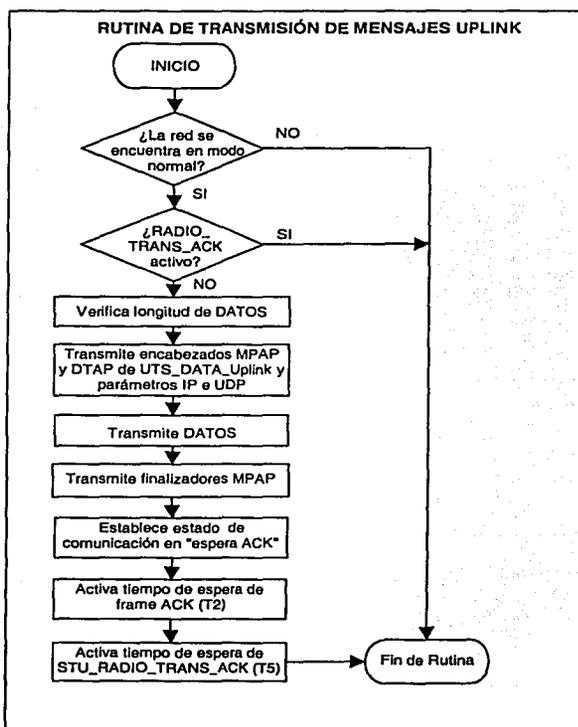


Fig. 5.5 Diagrama de flujo de la rutina de transmisión de mensajes *uplink*.

5.3 INTERRUPCIÓN POR TEMPORIZADORES

El μ C AT90S8515 cuenta con dos temporizadores, uno de 8 y otro de 16 bits, que deben ser compartidos entre 5 tiempos, uno de ellos destinado al muestreo, tres relacionados con la comunicación ST-UDT, y un último involucrado directamente con la comunicación UDT-DAS. En la Tabla 5.1 se muestran estos tiempos.

Tiempo	Descripción	Duración
T1	Tiempo de muestreo	1 a 255 [min]
T2	Tiempo de espera del <i>frame</i> de ACK (ACK time)	2 [s]
T3	Tiempo de espera del mensaje DTAP STU_PRESENCE (T_Presence)	6 [s] ⁹
T4	Tiempo de envío del mensaje de presencia UDT-DAS	5 a 255 [min] (<i>default: 10 [min]</i>)
T5	Tiempo de espera por el mensaje DTAP STU_RADIO_TRANS_ACK	5 [min]

Tabla 5.1 Descripción de tiempos.

El temporizador 0 (8 bits) es compartido por el tiempo MPAP de espera del *frame* ACK (T2) y por el tiempo de espera del mensaje DTAP STU_PRESENCE (T3).

- T2 se activa cada que el Módulo Adquisidor de Datos envía un *frame* de datos y se desactiva cuando recibe el *frame* ACK o bien se termina la conexión MPAP.
- T3 se activa cuando se establece la conexión MPAP, y permanece activo mientras no se concluya este enlace.

El temporizador 1 (16 bits) es compartido por el tiempo de muestreo (T1), el tiempo de envío del mensaje de presencia UDT-DAS (T4) y el tiempo de espera del mensaje DTAP STU_RADIO_TRANS_ACK (T5).

- T1 se activa cuando se recibe el mensaje *downlink* de configuración, y se desactiva cuando el Módulo Adquisidor de Datos transmite los datos de las memorias al DAS.
- T4 se activa con el establecimiento de la conexión MPAP, y se desactiva cuando el Módulo Adquisidor de Datos transmite información del ADC o de las memorias al DAS, o bien cuando se termina la conexión MPAP.
- T5 se activa cuando se envía el mensaje UTS_DATA_U, y se desactiva cuando se recibe la confirmación STU_RADIO_TRANS_ACK.

⁹ Este tiempo lo fija la ST y transmite su valor en el mensaje STU_CONNECT, normalmente es 3 [s], sin embargo se emplea el doble de dicho tiempo, 6 [s] (PAS Tetrapol).

Como se puede apreciar en la Tabla 5.1 estos tiempos son múltiplos de un minuto, y aún cuando no se activen simultáneamente es posible compartir temporizadores, considerando un error, para el temporizador 0 de hasta 0.05 [s], mientras que para el temporizador 1 el error en la primera lectura puede ser hasta 12 [s].

5.3.1 Rutina de Temporizador 0 (8 bits)

Este temporizador genera una interrupción cada 0.05 [s], y cada que ocurre, el programa verifica si ya transcurrió T3, en cuyo caso cierra el nivel de enlace. Si no se ha cumplido T3, verifica si T2 está activo y ya transcurrió, de ser así, la rutina revisa si el mensaje que originó su activación, ya ha sido transmitido 3 veces. Si todavía no se alcanza este límite, la rutina habilita la bandera de tiempo ACK para retransmitir el mensaje. Si por el contrario el mensaje ya ha sido transmitido 3 veces, el Módulo Adquisidor de Datos termina la conexión MPAP y desactiva esta interrupción. El diagrama de flujo de esta rutina se muestra en la fig. 5.6.

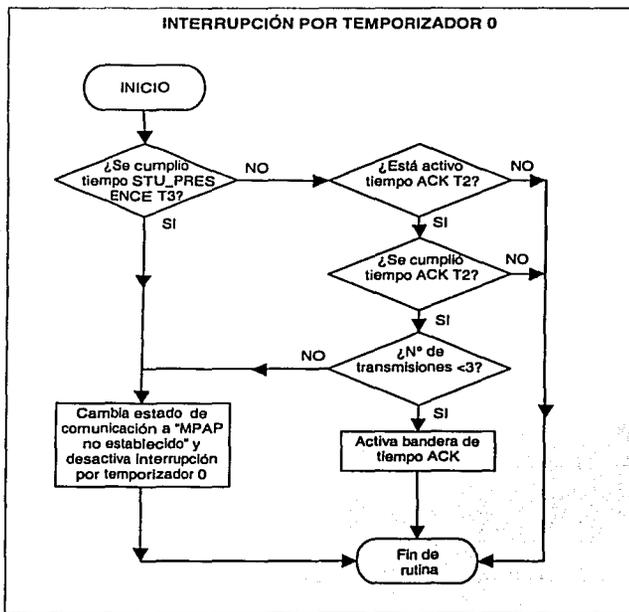


Fig. 5.6 Diagrama de flujo de la interrupción por temporizador 0.

5.3.2 Rutina de Temporizador 1 (16 bits)

Este temporizador genera una interrupción cada 12 [s], y cada que ocurre, el programa verifica si están activos los tiempos T1, T4 y T5. En caso de estar activo alguno de estos tiempos, la rutina revisa si ya se completó el tiempo activo. Si ya transcurrió T1, es decir que ya se cumplió el tiempo de muestreo, esta rutina habilita la bandera correspondiente e inicia el contador de este tiempo. Si ya transcurrió T4, el Módulo Adquisidor de Datos verifica que la conexión MPAP se encuentre establecida y que no esté esperando el mensaje RADIO_TRANS_ACK (confirmación de un mensaje enviado anteriormente), si se cumplen estas dos condiciones habilita la bandera del mensaje de presencia UDT-DAS e inicia el contador de T4. Si ya se cumplió T5, el μ C limpia el contador y verifica si el mensaje *uplink* que generó la activación de este tiempo ya ha sido transmitido 3 veces, si no es así, el Módulo Adquisidor de Datos retransmite el mensaje. Si por el contrario, el mensaje *uplink* ya ha sido transmitido 3 veces, o si ningún tiempo (T1, T4 o T5) se cumplió, la rutina regresa de la interrupción. El diagrama de flujo se muestra en la fig. 5.7.

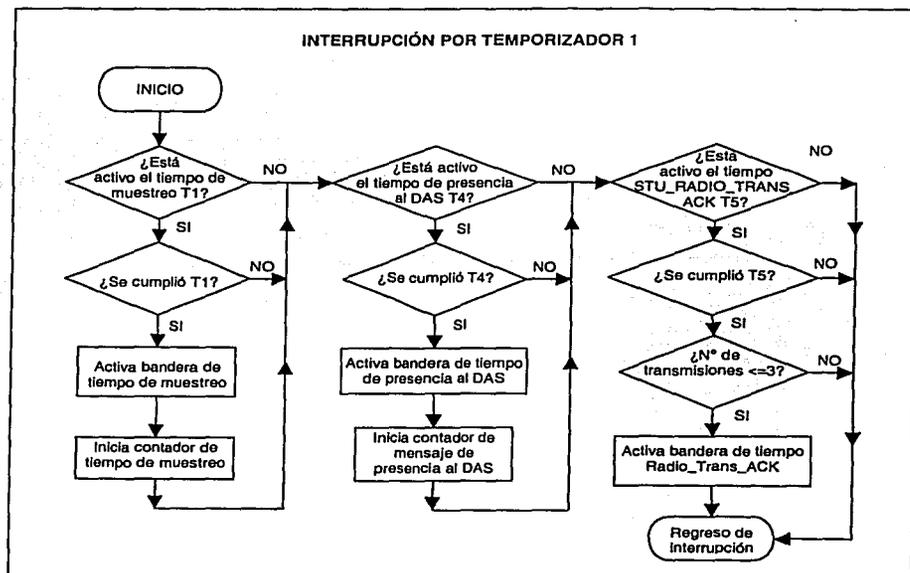


Fig. 5.7 Diagrama de flujo de la interrupción por temporizador 1.

5.4 RUTINAS DE MUESTREO

En esta sección se incluyen las rutinas que se relacionan directamente con la adquisición de datos, que son: rutina del convertidor y rutina de almacenamiento en memoria externa.

5.4.1 Rutina del Convertidor A/D

Esta rutina es la encargada de leer los valores obtenidos de la conversión A/D y almacenarlos temporalmente en memoria RAM. Como se muestra en el diagrama de flujo de la fig. 5.8 esta rutina primero revisa si existe algún canal activo. De ser así, determina secuencialmente qué canal está habilitado y no ha sido leído, realiza la conversión e indica en los cuatro bits más significativos del 2° byte (debido a que la precisión del ADC es de 10 bits se requieren dos bytes para representar cada valor) a qué canal corresponde el valor leído, y almacena estos dos bytes temporalmente en memoria RAM. Después revisa si existe otro canal activo que no haya sido leído, y repite el proceso.

Para determinar la emisión de un mensaje de alarma, dependiendo del canal correspondiente, el valor leído se compara con un valor patrón. En el caso del nivel de combustible este valor patrón es 1 [V], equivalente al 20% de la capacidad del tanque. En el caso de la temperatura el valor patrón es 2 [v], equivalente a 20 [°C].¹⁰

¹⁰ Véase Capítulo 3.

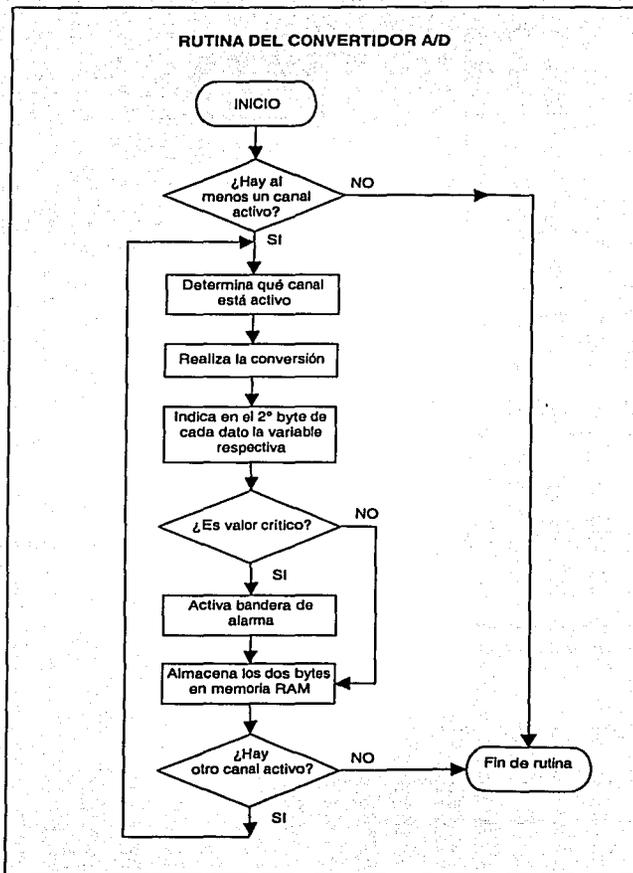


Fig. 5.8 Diagrama de flujo de la rutina del convertidor.

5.4.2 Rutina de almacenamiento en memoria externa

Esta rutina es la encargada de extraer de RAM los valores leídos por el ADC y almacenarlos en la memoria EEPROM externa. Para ello, verifica si no se ha agotado la memoria, ya que de estarlo, no se realiza el respaldo y se activan la bandera de memoria llena y la bandera de *display*. Cuando la memoria tiene espacio disponible, la rutina extrae de la memoria RAM el dato y lo guarda en la memoria EEPROM externa. Actualiza los apuntadores de dirección indicando la siguiente localidad, revisa si existe otro dato en RAM que deba ser almacenado en la EEPROM externa, y de haberlo repite el proceso. El diagrama de flujo de la rutina aparece en la fig. 5.9.

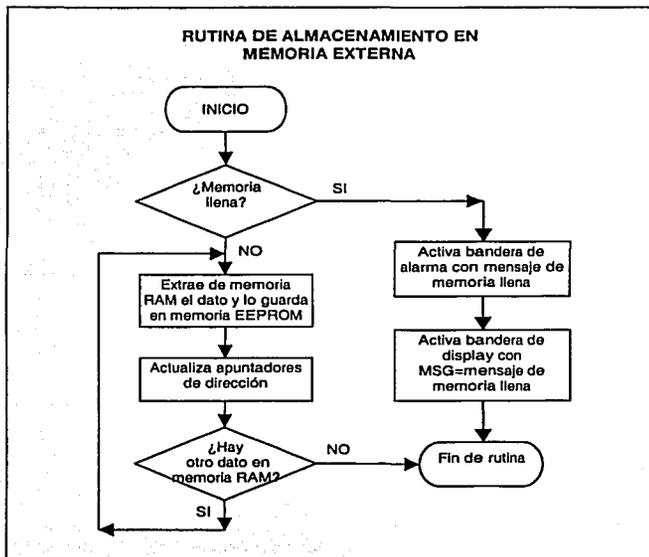


Fig. 5.9 Diagrama de flujo de la rutina de almacenamiento en memoria externa.

CAPÍTULO 6

SOFTWARE DEL SERVIDOR DE APLICACIÓN DE DATOS

En este capítulo se explica la programación realizada en Visual Basic para que el Servidor de Aplicación de Datos (DAS) se comuniquen con los dispositivos adquisidores, reciba los datos registrados por éstos y los muestre en una hoja de cálculo. A continuación se describen cada una de las nueve rutinas empleadas para llevar a cabo la comunicación.

6.1 HERRAMIENTAS DEL LENGUAJE

Visual Basic es un lenguaje de programación de alto nivel diseñado para crear aplicaciones con interfaz gráfica. Algunas de las ventajas que ofrece son el acceso y la manipulación tanto de datos como de objetos de otras aplicaciones, por ejemplo bases de datos, hojas de cálculo, archivos de sonido, aplicaciones de internet, y acceso a redes, entre otras. En particular, las hojas de cálculo y el acceso a redes resultaron útiles al proyecto.

Las hojas de cálculo permiten mostrar los datos recibidos en un ambiente amigable, que facilita al usuario almacenar la información y realizar algún procesamiento, como cálculos, gráficos, etc.

El acceso a redes se realiza mediante el control Winsock, que ofrece los servicios de la capa de transporte del modelo OSI, ya sea de acuerdo al protocolo TCP o en este caso UDP. Con el fin de permitir la comunicación del DAS con diferentes dispositivos y evitar la saturación de un mismo puerto Winsock, se utilizan tres puertos fijos y un arreglo dinámico, cuyo límite es el número de puertos disponibles del DAS.

El puerto fijo Winsock1 es el encargado de transmitir los mensajes *downlink* (petición de la información almacenada en las memorias, petición de valores actuales, configuración y confirmación de alarma recibida)¹, el puerto fijo Winsock2 únicamente

¹ Véase Capítulo 4 sección 4.5.

recibe los mensajes de presencia de los adquisidores activos, y el puerto fijo Winsock3 recibe las alarmas.

El arreglo dinámico de puertos Winsock se genera para asignar a cada adquisidor un puerto UDP destinado a recibir la información almacenada en las memorias y los valores actuales. En los mensajes de petición se indica el puerto UDP por el que el DAS espera recibir los mensajes de datos. El puerto se libera cuando los mensajes esperados son recibidos, o cuando ha transcurrido un lapso de 10 [min]. Para evitar confundir un mensaje esperado con un mensaje retrasado, se utiliza el número de mensaje.

6.2 DESCRIPCIÓN DE LAS RUTINAS

6.2.1 Menú principal

El menú principal que aparece en la fig. 6.1, muestra los dispositivos que se encuentran activos, es decir, aquellos que han enviado periódicamente su mensaje de presencia. Cuando el usuario selecciona alguno de los dispositivos activos, en el menú aparece la información referente al dispositivo, como el nombre del sitio en el que se encuentra, su ubicación, la dirección IP con la que se está comunicando, etc. Además se habilitan las opciones: "Configurar equipo", "Obtener información de las memorias" y "Valores actuales". La fig. 6.2 muestra el menú principal con estas opciones habilitadas y la fig. 6.3 muestra el diagrama de flujo de esta rutina.

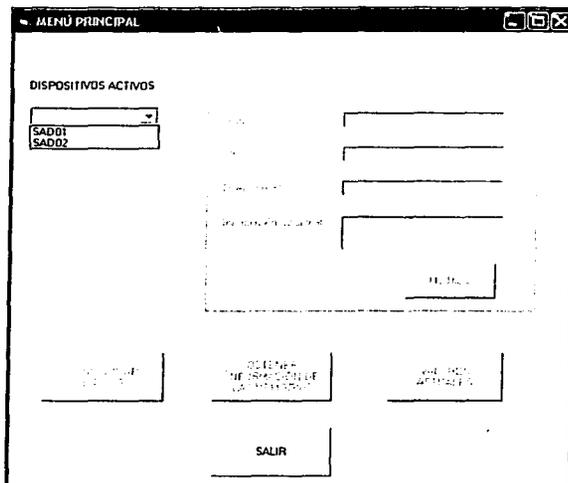


Fig. 6.1 Menú principal

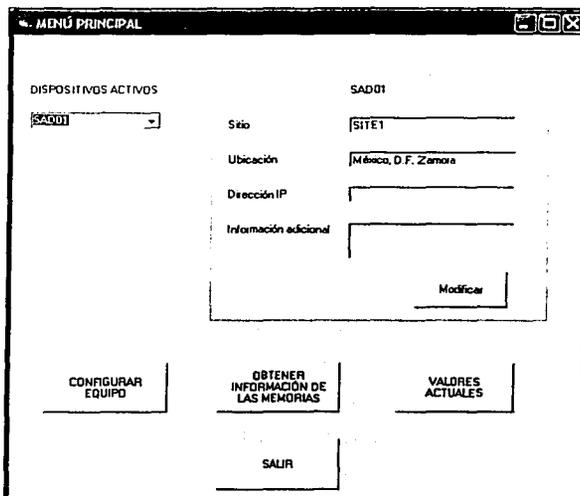


Fig. 6.2 Menú principal con todas las opciones habilitadas.

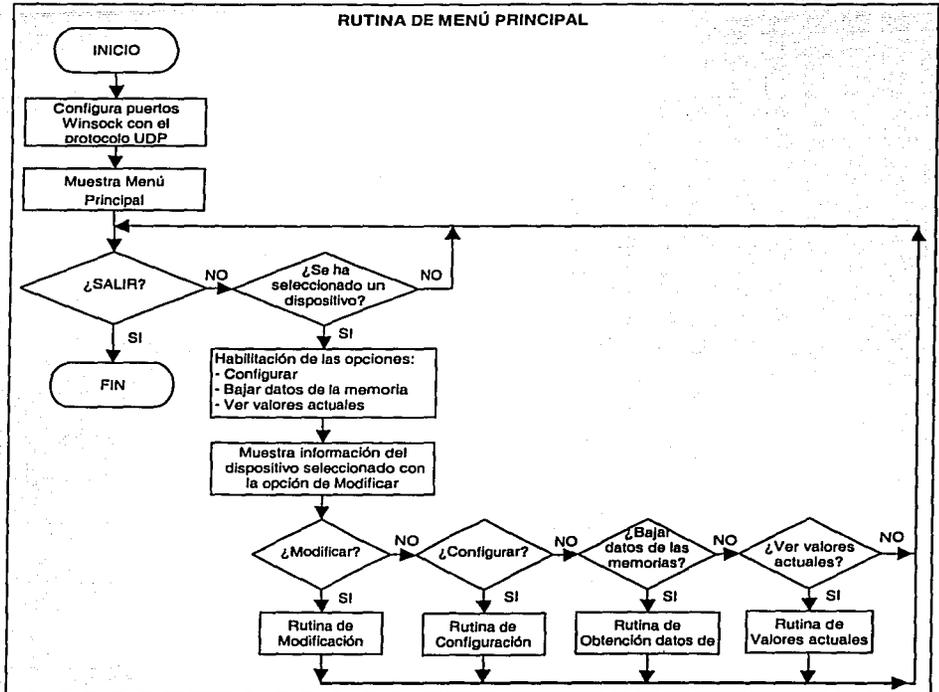
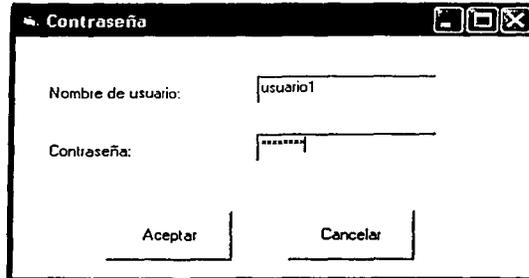


Fig. 6.3 Diagrama de flujo del menú principal.

6.2.2 Rutina de modificación

Cuando se selecciona la opción modificar del menú principal aparece la ventana que se muestra en la fig. 6.4, en la que se pide al usuario introduzca su clave y contraseña. En caso de ser correctos, se desplegará el menú que aparece en la fig. 6.5, que permite al usuario cambiar el texto de cualquiera de los campos de información que se muestran, además aparecerán las opciones "Aceptar", "Cancelar" y "Añadir". La opción "Aceptar" guarda permanentemente los cambios que se hayan realizado en cualquier campo de información y despliega el menú principal de la fig. 6.2. La opción "Cancelar" ignora los cambios y al igual que la opción anterior, despliega el menú principal de la fig. 6.2. Finalmente la opción "Añadir", muestra además de los campos ya mencionados y las opciones "Aceptar" y "Cancelar", el campo "Nombre de

dispositivo", que permite al usuario introducir en la base de datos, un nuevo adquisidor. El diagrama de flujo de esta rutina se muestra en la fig. 6.6.



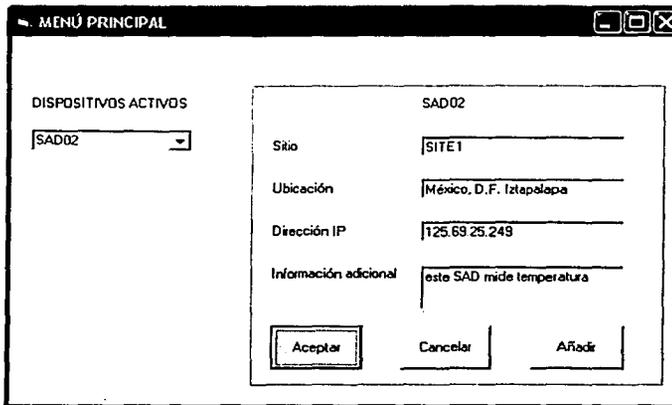
Contraseña

Nombre de usuario: usuario1

Contraseña: *****

Aceptar Cancelar

Fig. 6.4 Ventana de contraseña.



MENÚ PRINCIPAL

DISPOSITIVOS ACTIVOS

SAD02

SAD02

Sitio SITE1

Ubicación México, D.F. Iztapalapa

Dirección IP 125.69.25.249

Información adicional este SAD mide temperatura

Aceptar Cancelar Añadir

Fig. 6.5 Ventana de modificar.

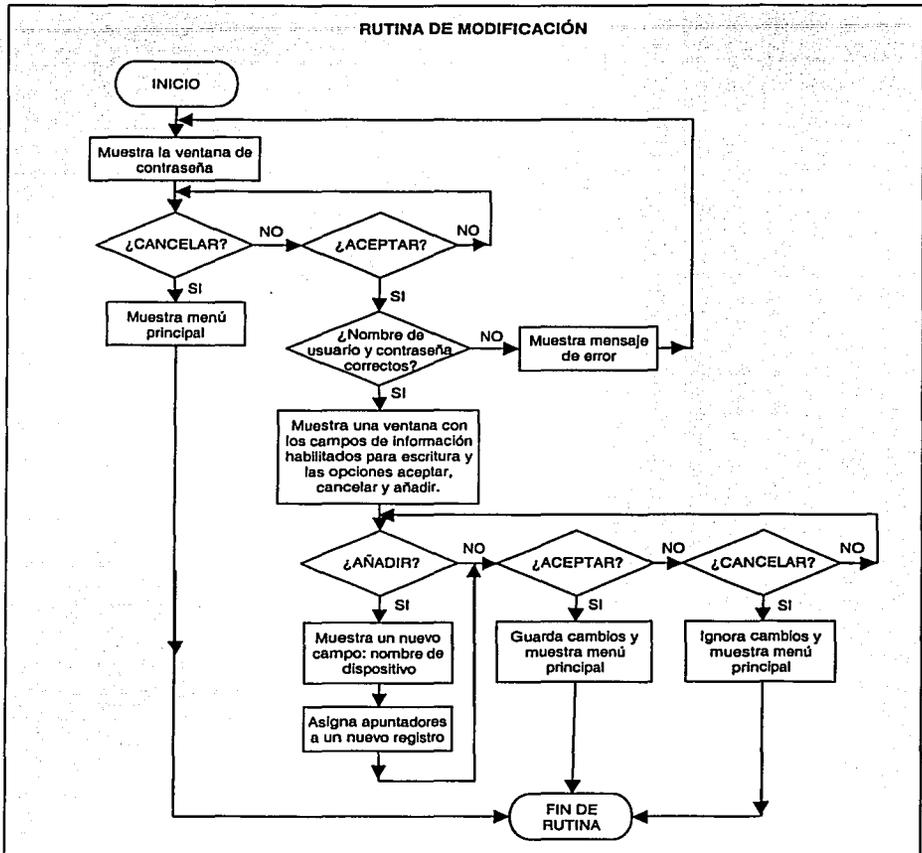


Fig. 6.6 Diagrama de flujo de la rutina de modificar.

6.2.3 Rutina de configuración

Cuando se selecciona la opción "Configurar", aparece la ventana de contraseña que se muestra en la fig. 6.4, cuya finalidad es verificar que el usuario tenga el permiso de realizar cambios en la configuración de un adquisidor. Si el nombre de usuario y la contraseña son correctos, entonces se desplegará la ventana de la fig. 6.7. En esta ventana, el usuario deberá introducir el tiempo de muestreo, el tiempo de presencia y

elegir al menos un canal. Cabe mencionar que el tiempo de muestreo debe encontrarse dentro del rango admisible desde 1 y hasta 255 [min], y el tiempo de presencia debe ser un valor entre 5 y 255 [min]. Cuando se selecciona "Aceptar", el programa verifica que tanto el tiempo de muestreo como el tiempo presencia tengan valores válidos, además revisa que al menos un canal haya sido seleccionado. De cumplirse estas tres condiciones, el DAS envía el mensaje de configuración descrito en el capítulo 4, sección 4.5. Si por el contrario, no se cumplieron estos requisitos, se despliega la ventana de la fig. 6.8. Finalmente, si el usuario selecciona la opción "Cancelar", aparece el menú principal. El diagrama de flujo de esta rutina se muestra en la fig. 6.9.

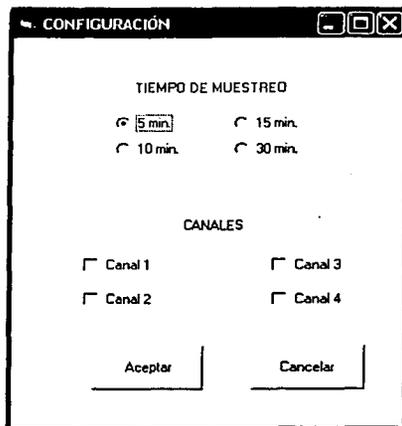


Fig. 6.7 Ventana de configuración.

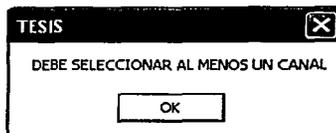


Fig. 6.8 Ventana de mensaje de error.

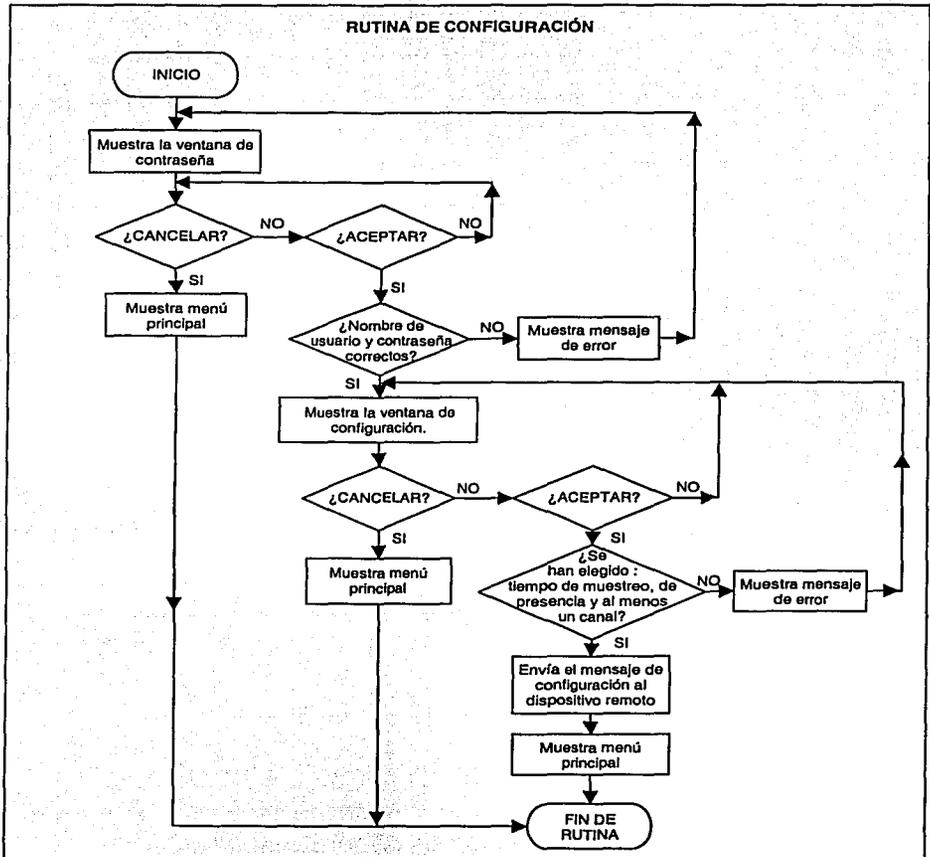


Fig. 6.9 Diagrama de flujo de la rutina de configuración.

6.2.4 Rutina de obtención de información de las memorias

Cuando se elige esta opción, el DAS transmite el mensaje de petición de los datos almacenados por el dispositivo seleccionado. Como se mencionó en la sección 6.1, en este mensaje se indica el puerto UDP al que el Módulo Adquisidor de Datos debe dirigir la información solicitada. Este puerto se asigna mediante un arreglo dinámico de puertos Winsock, que determina qué puertos se encuentran disponibles. Lo anterior permite solicitar información a varios adquirentes simultáneamente.

Una vez que el puerto Winsock asignado ha recibido el mensaje de respuesta, el programa realiza su autenticación mediante el número de mensaje. Si es correcta, el mensaje se procesa, en caso contrario se descarta.

El procesamiento consiste en registrar el tiempo en que el DAS recibió el mensaje (cualquier mensaje *uplink* es considerado por el DAS como mensaje de presencia), ordenar la información recibida de acuerdo al segmento en el que se encuentra, y si se trata del primer segmento extraer y guardar la fecha y hora indicados en el paquete. Cuando se han recibido todos los segmentos, la rutina muestra la ventana de la fig. 6.10, que indica que la recepción se ha completado y libera el puerto Winsock utilizado. Cuando se selecciona la opción "Aceptar" que aparece en esta ventana, se realiza la conversión entre el valor medido y el valor real según la variable correspondiente, y el valor real se muestra en una hoja de cálculo. El diagrama de flujo se muestra en la fig. 6.11.



Fig. 6.10 Ventana de recepción completa

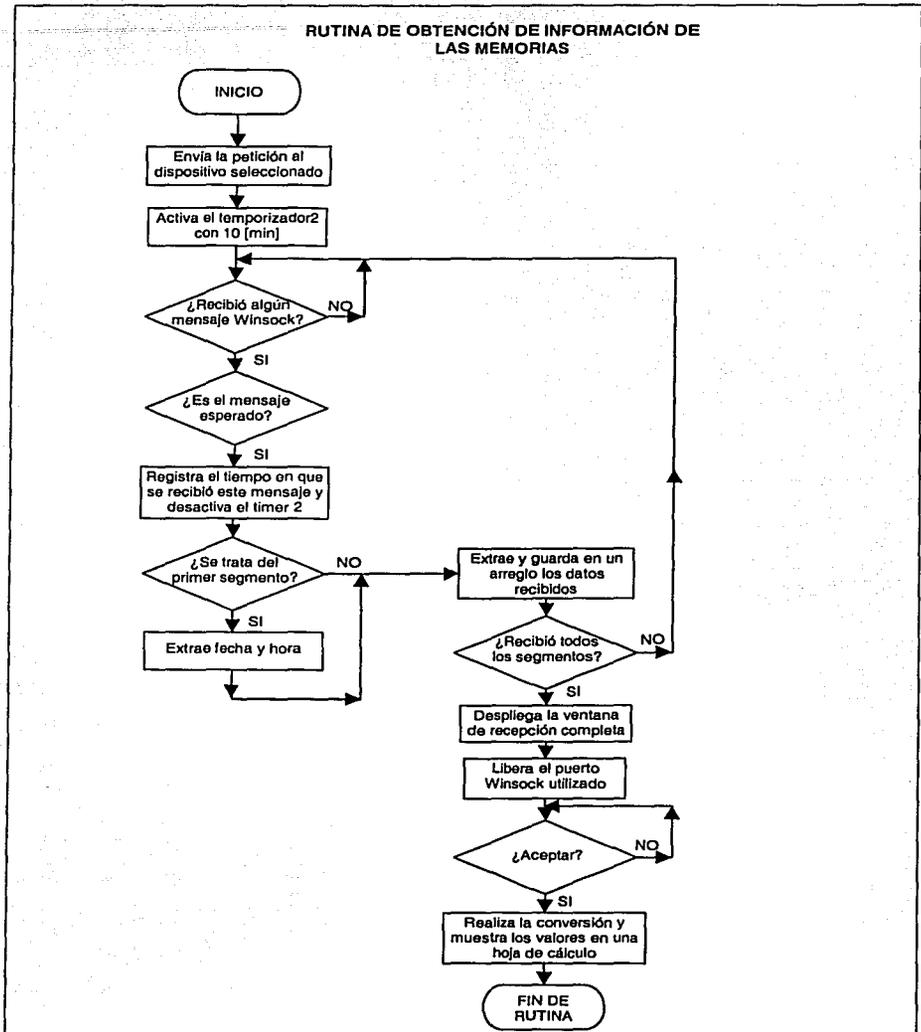


Fig. 6.11 Diagrama de flujo de la rutina de obtención de información de las memorias.

6.2.5 Rutina de monitoreo de valores actuales

Cuando se selecciona esta opción, el DAS transmite el mensaje al dispositivo seleccionado para realizar la lectura de todas las variables habilitadas y enviar la información al DAS. Como en el caso de la petición de información de las memorias, la petición de valores actuales indica el puerto UDP al que el Módulo Adquisidor de Datos debe dirigir la información solicitada.

Cuando el puerto Winsock asignado recibe un mensaje, el programa autentica el mensaje mediante el número de segmento. Si es el esperado se procesa, en caso contrario se descarta.

El procesamiento consiste en registrar el tiempo en que llegó este mensaje (cualquier mensaje *uplink* es considerado por el DAS como mensaje de presencia), en realizar la conversión entre el valor medido y el valor real de cada variable, en mostrar los valores leídos en la ventana que aparece en la fig. 6.12, y finalmente en liberar el puerto Winsock empleado. La ventana de valores actuales cuenta con las opciones: "Mostrar en hoja de cálculo" y "Regresar". La primera permite ver los valores en una hoja de cálculo, y de esta manera guardar la información en un archivo, y la segunda despliega el menú principal. El diagrama de flujo se muestra en la fig. 6.13.

The screenshot shows a window titled "VALORES ACTUALES" with a standard Windows-style title bar. The window content is as follows:

- Site ID: SAD01
- SITIO: SITE1
- UBICACIÓN: México, D.F.
- Temperatura Ambiente: 15 [°C]
- Combustible: 75 [%]
- Buttons: "Regresar" and "Mostrar en Hoja de Cálculo"

Fig. 6.12 Ventana de valores actuales.

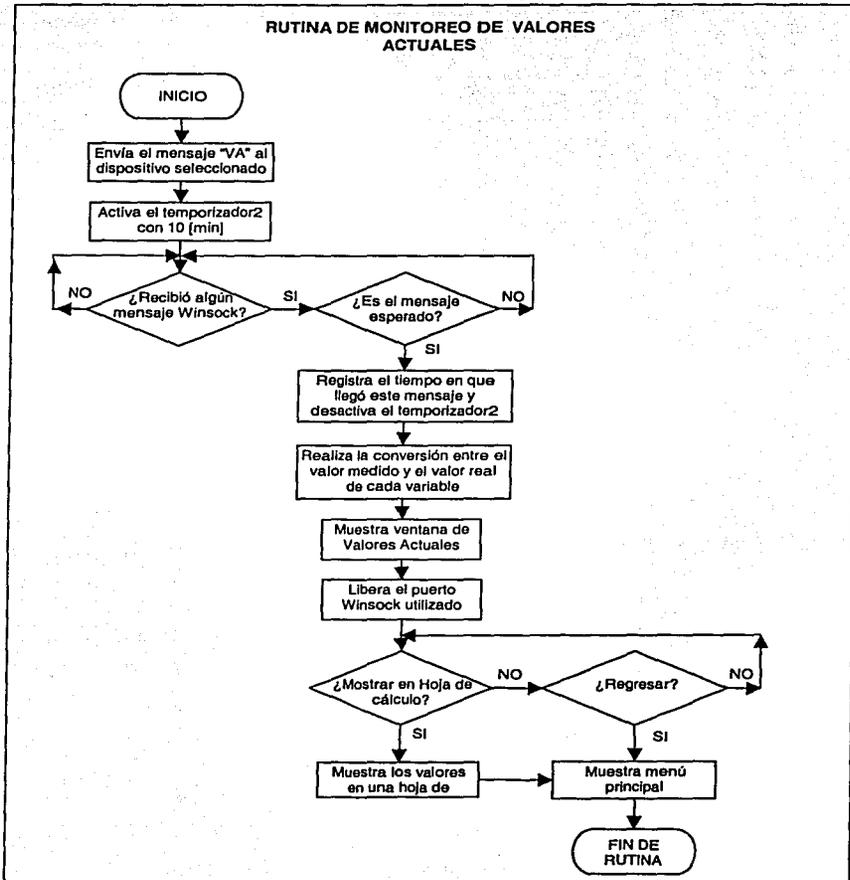


Fig. 6.13 Diagrama de flujo de la rutina valores actuales.

6.2.6 Rutina del Temporizador 1

El temporizador 1 está configurado para interrumpir cada minuto, al hacerlo el DAS verifica si ya transcurrió el tiempo de presencia de cada adquisidor que aparece en la lista de dispositivos activos. En caso de que el tiempo se haya agotado, se elimina de la lista el adquisidor en cuestión, impidiendo al usuario establecer comunicación con el dispositivo hasta que éste vuelva a enviar un mensaje de presencia. El diagrama de flujo se muestra en la fig. 6.14.

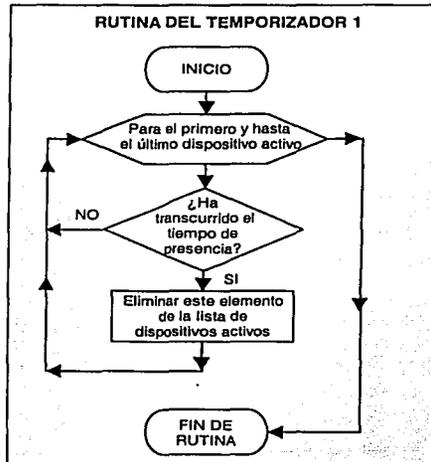


Fig. 6.14 Diagrama de flujo del temporizador 1.

6.2.7 Rutina del Temporizador 2

El temporizador 2 consta de un arreglo de *timers*, cuya función es indicar el tiempo de espera de un mensaje de datos (información almacenada en la memoria o valores actuales). Por ello se encuentra configurado para interrumpir a los 10 [min] de su activación, emitir un mensaje de error al usuario cuando este lapso se cumple y liberar al puerto UDP destinado a recibir la información.

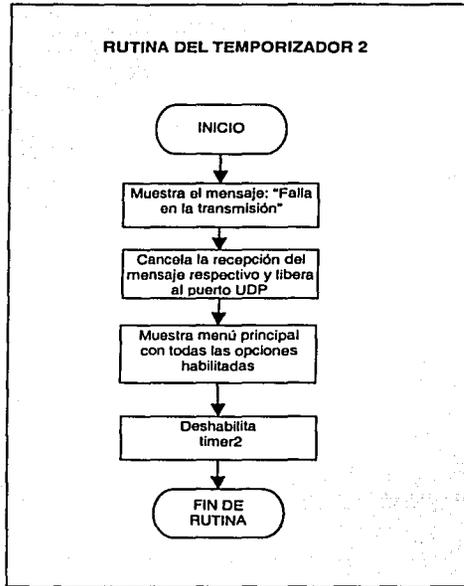


Fig. 6.15 Diagrama de flujo del temporizador 2.

6.2.8 Rutina de recepción del puerto Winsock 2

Este puerto está destinado únicamente a recibir los mensajes de presencia. Cuando este puerto recibe un paquete, obtiene el nombre del dispositivo, con la finalidad de verificar que éste se encuentre dado de alta en la base de datos general, si no es así, el DAS ignora el paquete recibido. En caso contrario registra el tiempo en que llegó el mensaje, actualiza en la base de datos la dirección IP con la que se comunicó el dispositivo, finalmente si el dispositivo no se encuentra en la lista de elementos activos lo añade. El diagrama de flujo se muestra en la fig. 6.16.

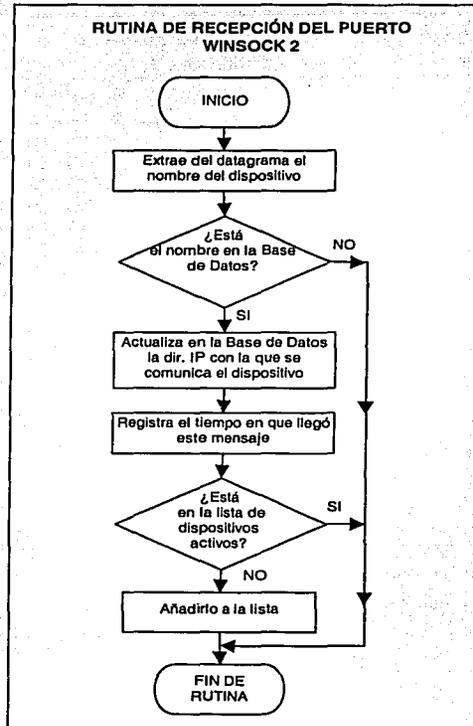


Fig. 6.16 Diagrama de flujo del Winsock 2.

6.2.9 Rutina de recepción del puerto Winsock 3

Este puerto se encarga únicamente de recibir los mensajes de alarma. Al llegar un paquete a este puerto, la rutina obtiene el nombre del dispositivo para verificar que esté dado de alta en la base de datos, si no es así el DAS ignora el paquete recibido. En caso contrario, registra el tiempo en que llegó el mensaje, transmite el mensaje *downlink* de confirmación, actualiza en la base de datos la dirección IP con la que se está comunicando el dispositivo, verifica de qué variable se trata, realiza la conversión

de la figura 6.17 y el valor de la alarma se ve en la ventana de alarma que aparece en la fig. 6.18. Esta ventana puede ser en las opciones: "Mostrar en hoja de cálculo" y "Mostrar en primera", cuando se elige la opción de cálculo además del valor que originó la alarma, se muestra la intensidad de la alarma en cuestión (nombre del dispositivo, ubicación, etc.) y la ubicación de la alarma en la segunda desfoliada nuevamente en el diagrama. El diagrama de alarma que se muestra en la fig. 6.18.



Ventana de alarma.

TESIS CON FALLA DE ORIGEN

TESIS CON FALLA DE ORIGEN

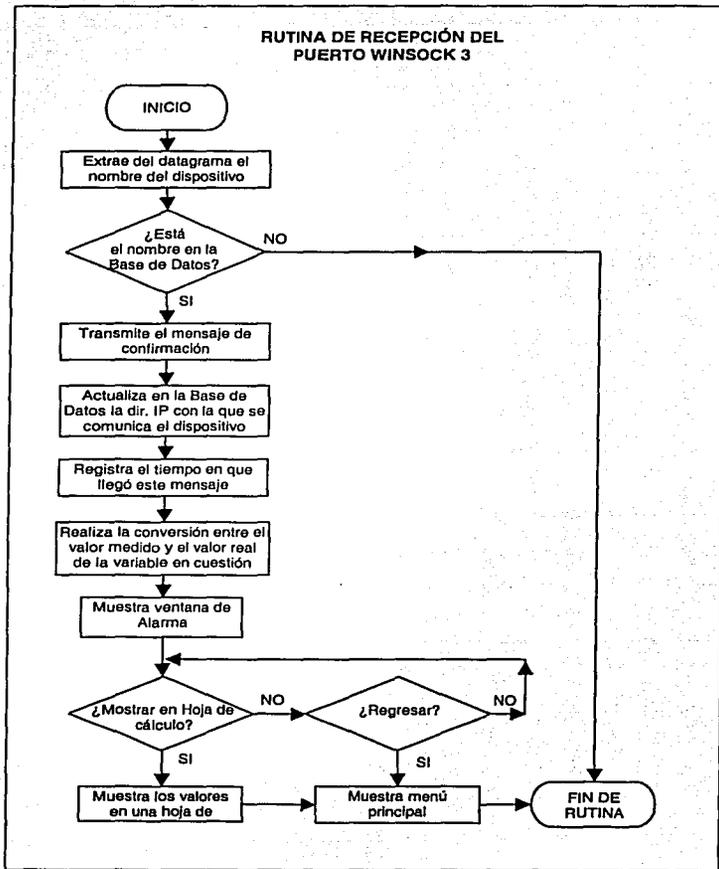


Fig. 6.18 Diagrama de flujo del Winsock 3.

CAPÍTULO 7

INTEGRACIÓN DEL SISTEMA

El diseño de los sistemas modernos se realiza estructuradamente, es decir que la solución de un problema se divide en funciones más simples que proporcionan una solución parcial. Así, el Sistema de Adquisición de Datos (SAD) se estructuró en dos sistemas principales: Módulo Adquisidor de Datos y software del DAS, cada uno integrando varios subsistemas.

La estructuración del proyecto en varias entidades independientes permite realizar un programa de pruebas sistemático, que inicia con la evaluación individual de los subsistemas, continúa con la integración en sistemas y finaliza evaluando el proyecto en su conjunto.

Este seguimiento de pruebas facilita la identificación y aislamiento de fallas en elementos independientes, agilizando la corrección e integración del sistema. Esto es particularmente conveniente, ya que al estar el SAD conformado por dos sistemas distantes enlazados por la red Tetrapol, resulta complejo realizar una identificación de fallas y por ende su depuración al operar el SAD en conjunto.

De acuerdo con la estructuración del proyecto, el programa de pruebas se dividió en dos partes principales: evaluación del Módulo Adquisidor de Datos y evaluación del software del DAS. En el presente capítulo se describe en qué consistieron las evaluaciones, se explica la operación del sistema en conjunto, y finalmente se describen las interfaces empleadas por la red Tetrapol para establecer la comunicación Módulo Adquisidor de Datos - DAS.

7.1 EVALUACIÓN DEL MÓDULO ADQUISIDOR DE DATOS

Para realizar la evaluación al Módulo Adquisidor de Datos se consideró al sistema constituido por los siguientes subsistemas funcionales: software de comunicación, módulo de sensores y acondicionamiento, módulo de conversión A/D, módulo de memorias y módulo de desplegado.

La primera prueba realizada al módulo de sensores y acondicionamiento consistió en determinar experimentalmente el valor de voltaje aplicado a las terminales del sensor de combustible, que como se mencionó en el capítulo 3 está alimentado por el

sistema de monitoreo propio del generador. Una vez que se determinó que se aplicaban 5 [V], se confirmó que el valor de voltaje entregado por el circuito divisor de voltaje variara linealmente entre 0.3 [V] y 5 [V], en relación directa con el nivel de combustible en el tanque, correspondiendo 0.3 [V] a tanque vacío y 5 [V] a tanque lleno. Posteriormente se comprobó que los niveles de voltaje no variarían ante diferentes situaciones como lo fue la medición con el generador apagado y la medición con el generador operando. La única alteración que se observó ocurrió durante el encendido del generador a causa del transitorio originado por la bobina de arranque. Sin embargo esta alteración no fue significativa, y dado que el intervalo de muestreo es de 1 a 255 [min], es probable que este transitorio no sea registrado por el Módulo Adquisidor de Datos. Como prevención al transitorio mencionado, y de esta manera evitar un posible sobrevoltaje en el ADC, se utiliza un circuito comparador que limita el voltaje de entrada al canal del ADC a 5 [V].

Para caracterizar el circuito de temperatura se tomaron 15 lecturas de un recipiente con agua con el sensor LM35 y se compararon con los valores obtenidos con un multímetro. Con esto se comprobó el comportamiento lineal y confiable del sensor.

A diferencia del módulo de sensores y acondicionamiento de señales, los demás subsistemas son controlados por el μC del Módulo Adquisidor de Datos, lo que implica que la mayoría de las fallas en estos subsistemas se debieron a errores en la programación del μC . De ahí la importancia de las pruebas realizadas a cada una de las rutinas del software. Para efectuar estas pruebas se utilizó el emulador ICE-200 de Atmel. Esta herramienta permitió ejecutar el programa paso a paso, observar e incluso modificar el contenido de diversos registros en tiempo de ejecución y principalmente interactuar por medio de los puertos de entrada/salida con variables físicas previamente digitalizadas.

Con el apoyo del emulador, las pruebas al módulo de conversión A/D se realizaron empleando un circuito divisor de voltaje que permitió obtener un voltaje variable de 0.3 a 5 [V], y para comparar los valores observados en el emulador se utilizó como patrón un multímetro digital de 5½ dígitos.

Las pruebas realizadas al módulo de almacenamiento consistieron primero en verificar que la secuencia de lectura y escritura funcionara correctamente, y segundo en revisar que cuando una memoria agotara su capacidad, los apuntadores indicaran a la primera localidad de la siguiente memoria, y que en caso de haberse agotado la capacidad de las cuatro memorias, se dejara de grabar y se emitiera la alarma adecuada al DAS. Además, debido a que esta rutina guarda una estrecha relación con la comunicación Módulo Adquisidor de Datos-DAS, se confirmó que determinara con exactitud la cantidad de datos almacenados y el procedimiento en que los mismos serían transmitidos en cada trama.

Para evaluar el *display*, primero se elaboró un programa encargado de corroborar que el *display* respondiera adecuadamente a los comandos de escritura. Una vez que esto se logró, se incorporó la rutina al programa principal, interrelacionándola con

aquellas rutinas que necesitan desplegar información, y finalmente, se verificó que no interfiriera con el funcionamiento del mismo.

A fin de cumplir con el principal objetivo del proyecto, que es realizar un sistema capaz de utilizar los recursos de transmisión de datos de una red Tetrapol, se hizo particular hincapié en las pruebas realizadas a la comunicación, primero Módulo Adquisidor de Datos - RT y posteriormente, Módulo Adquisidor de Datos-DAS.

Para probar el funcionamiento de las rutinas de comunicación del Módulo Adquisidor de Datos en los niveles físico, de enlace de datos (MPAP) y de aplicación (DTAP), se creó un software en Visual Basic encargado de simular las funciones de comunicación de una RT. Este software permitió observar el comportamiento del Módulo Adquisidor de Datos ante fallas en la comunicación, como recibir un mensaje repetido, un mensaje erróneo o un mensaje extemporáneo. El menú principal de este software se muestra en la fig. 7.1.

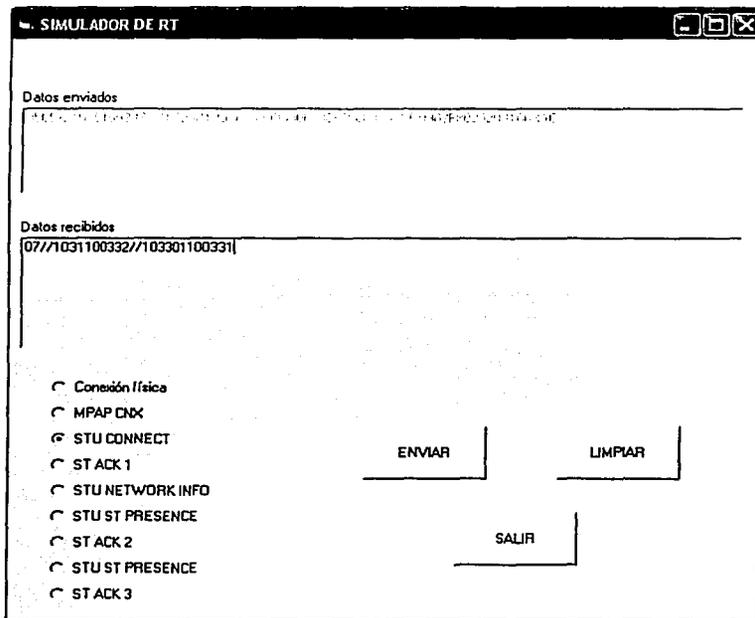


Fig. 7.1 Menú principal del software simulador de la comunicación de una RT.

La siguiente prueba consistió en establecer la comunicación entre el Módulo Adquisidor de Datos y la RT, utilizando un analizador de tramas, que permitió observar las tramas en ambos sentidos de la comunicación, y con ello depurar el software del Módulo Adquisidor de Datos. Además con esta prueba fue posible ajustar los temporizadores.

Una vez que se realizaron las pruebas individuales antes descritas, se llevó a cabo la integración del software, que se realizó en función de la prioridad de cada una de las rutinas. Uno de los retos que se presentaron en esta etapa del proyecto fue garantizar que cada una de las rutinas, desempeñara su labor correctamente sin interferir con las demás, optimizando los recursos del μC .

Cabe mencionar, que las pruebas se realizaron en la tarjeta de desarrollo del μC , misma que cuenta con una fuente de alimentación, por lo que en esta etapa del proyecto no fue necesario diseñar una fuente de alimentación para los circuitos integrados que componen al Módulo Adquisidor de Datos (μC , memorias EEPROM y ADC).

7.2 EVALUACIÓN DEL SOFTWARE DEL SERVIDOR DE APLICACIÓN DE DATOS

El software del DAS comprende las funciones siguientes: comunicación, procesamiento y despliegado de datos.

Para verificar el correcto funcionamiento del software orientado a la comunicación del DAS se creó un programa auxiliar en Visual Basic, cuyo objetivo fue simular las funciones de comunicación del Módulo Adquisidor de Datos. La primera prueba consistió en establecer la comunicación entre el programa auxiliar y el software del DAS, instalados cada uno en una terminal diferente de una red LAN. Con esto se comprobó el correcto funcionamiento de la comunicación uno a uno.

Para analizar el comportamiento del software del DAS bajo condiciones de tráfico se abrieron cinco aplicaciones simultáneas del programa auxiliar en cada una de las tres terminales que se emplearon con este propósito. Los resultados obtenidos con esta prueba demostraron que el software del DAS es capaz de soportar al menos 15 adquirentes simultáneos eficientemente. Sin embargo, las características intrínsecas de la red (tráfico, velocidad, etc.) y aquellas inherentes al DAS (capacidad de procesamiento y los recursos asignados a la aplicación, principalmente la cantidad de puertos UDP disponibles) influyen directamente en el desempeño. En las pruebas se utilizó un equipo pentium III a 500 MHz destinado exclusivamente a ejecutar el software del DAS. La red fue Ethernet 100/10 basada en el sistema operativo Windows 2000 Server.

En la fig. 7.2 se muestra el menú principal del software auxiliar que se utilizó en estas pruebas.

Para realizar las pruebas a la rutina de procesamiento y desplegado se introdujo una serie de datos, cuya finalidad fue asegurar que las ecuaciones encargadas de realizar la conversión del valor leído y el valor real funcionaran correctamente, además se verificó el manejo adecuado de la hoja de cálculo para desplegar los resultados obtenidos. Estas pruebas además permitieron depurar la programación.

SOFTWARE AUXILIAR

Datos enviados
PRESSAD05ALSAD051

Datos recibidos
ALR

Presencia

Alarma

Valores Actuales

Información de memorias (segmento 1)

Información de memorias (segmento 2)

Información de memorias (segmento final)

Puerto UDP asignado 9900

Nombre de SAD SAD05

Dirección IP cliente 02.00.01.01

Dirección IP servidor 02.00.01.07

ENVIAR SALIR LIMPIAR

Fig. 7.2 Menú principal del Software auxiliar.

7.3 OPERACIÓN DEL SISTEMA EN CONJUNTO

Una vez que se comprobó que cada una de las partes principales (Módulo Adquisidor de Datos y DAS), y por ende sus respectivos subsistemas funcionaron correctamente, se realizaron pruebas al sistema en su conjunto. Debido a las características de seguridad del sitio (restricciones de acceso), las pruebas primero se llevaron a cabo desde el Centro de Monitoreo Remoto, es decir en las instalaciones que albergan al DAS. La cercanía entre el DAS y el Módulo Adquisidor de Datos facilitó observar el desempeño de ambos sistemas simultáneamente. A fin de simular el comportamiento de las señales provenientes del módulo de sensores y acondicionamiento de señales, se conectó una fuente de voltaje a la entrada del convertidor A/D.

Esta etapa de evaluación permitió realizar los ajustes pertinentes a las rutinas de comunicación del Módulo Adquisidor de Datos, recepción y transmisión de datos en el DAS, particularmente resultó muy útil para corregir y depurar diversas fallas en la fragmentación de la información.

Una vez que se consideró que el sistema funcionaba adecuadamente en un ambiente controlado, se instaló el Módulo Adquisidor de Datos en un sitio de repetición propio de la red Tetrapol. Sin embargo, como se mencionó antes, dadas las características de seguridad en el sitio no fue posible realizar el número de pruebas deseadas desde el interior del mismo, por lo que solo se realizaron dos pruebas de muestreo. Éstas consistieron en tomar mediciones tanto del nivel de combustible como de la temperatura interna del sitio, en dos diferentes intervalos de muestreo (1min y 2min).

Por otra parte, para verificar el adecuado funcionamiento de las alarmas se implementaron dos pruebas. La primera consistió en suministrar directamente con una fuente un voltaje decreciente desde un valor de 5 [V] hasta el valor al que se genera la alarma (1 [V], equivalente al 20% de capacidad del tanque). Una vez que se constató que el mensaje de alarma fue emitido por el Módulo Adquisidor de Datos y recibido por el DAS se realizó una segunda prueba, esta vez con el sensor de temperatura. El sensor se calentó gradualmente con una fuente de calor (encendedor). Las variaciones en el voltaje de entrada al convertidor A/D originadas por el sensor de temperatura se midieron con un multímetro. De esta manera fue posible ratificar que el mensaje de alarma se generara al alcanzar una temperatura de 20 [°C].

7.4 INTERFACES DE LA RED TETRAPOL

Dada la trascendencia de la red Tetrapol en el presente proyecto, a continuación se describen brevemente las interfaces involucradas en la comunicación Módulo Adquisidor de Datos-DAS.

En la fig. 7.3 se muestran las interfaces que intervienen en la transferencia de información Módulo Adquisidor de Datos - DAS. La primera corresponde al enlace Módulo Adquisidor de Datos - RT, en la que intervienen los protocolos RS232, MPAP y DTAP. La segunda corresponde a la interfaz aérea RT-BS¹. La siguiente interfaz concierne al enlace BS-RSw que puede establecerse mediante un enlace E1 o bien, por medio de una red ATM. El RSw cuenta con diversas herramientas que le permiten establecer enlaces externos hacia la red telefónica pública (PSTN, *Public Service Telephony Network*), hacia redes de telefonía celular (PMT, *Public Mobile Telephony*) como GSM, y hacia redes de computadoras (internet), entre otros.

La herramienta que permite enlazar al RSw con redes de computadoras es el Controlador de Red de Datos (DNC, *Data Network Controller*), encargado de realizar las funciones de ruteo de acuerdo al protocolo IP, además está equipado para soportar diferentes tipos de redes externas LAN: Ethernet, DIX Ethernet, Token Ring, FDDI, etc.

De esta manera, un mensaje *uplink* (Módulo Adquisidor de Datos-DAS), generado en el Módulo Adquisidor de Datos se transmite a una RT mediante una conexión física RS232, y de acuerdo a los protocolos MPAP/DTAP. La RT elimina los encabezados y finalizadores MPAP/DTAP una vez que los ha utilizado, añade los encabezados propios de la interfaz aérea, y se encarga de codificar, modular (GMSK) y transmitir el mensaje. Éste es recibido por la BS que proporcione cobertura en el área en la que se encuentre la RT, en este caso la misma del sitio de repetición. La BS elimina los encabezados que emplea para procesar la información recibida y mantiene aquellos añadidos por la RT destinados al RSw. Para que el mensaje llegue al RSw, se utiliza ya sea un enlace E1, o bien una red ATM, dependiendo de la infraestructura Tetrapol.

¹ Los protocolos correspondientes a la interfaz aérea RT-BS se describen en PAS Tetrapol.

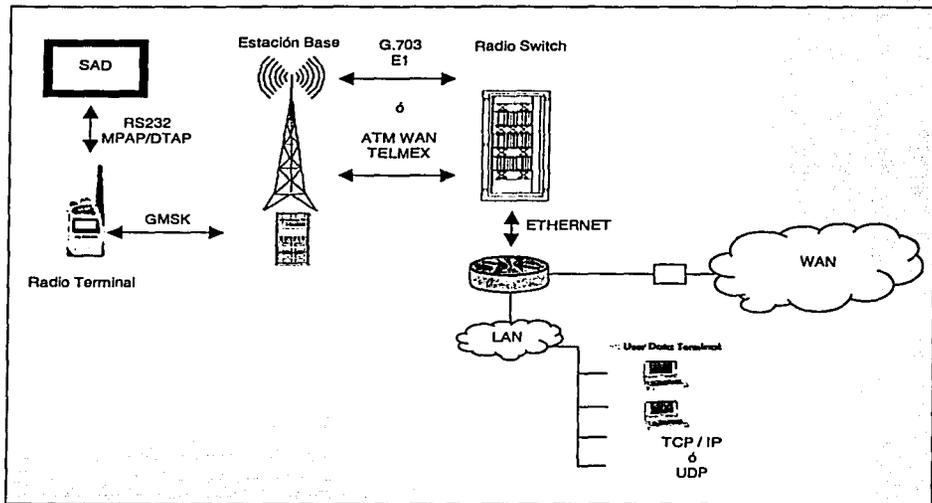


Fig. 7.3 Diagrama de interfaces.

CONCLUSIONES

Para el desarrollo del presente proyecto fue necesaria la integración de conocimientos de diversas áreas de la ingeniería como redes de computadoras, de radio comunicaciones, sistemas digitales y sistemas analógicos orientados a la instrumentación. Lo anterior permitió comprender la interrelación que hoy en día tienen todas estas áreas, y generó la oportunidad de acrecentar tanto conocimientos como experiencia en las mismas. Además, sentó las bases para desarrollar un sistema cuyos alcances no se limitan a transmitir el nivel de combustible de un generador y la temperatura de un sitio de repetición, por lo que el diseño que se llevó a cabo en este trabajo de tesis representa la base de futuros sistemas que deseen incorporar la transmisión inalámbrica de datos vía una red Tetrapol.

Como se mencionó previamente, el diseño del sistema se realizó considerando que la capacidad para utilizar los servicios de transmisión de datos de una red Tetrapol puede resultar muy útil en cualquier otro tipo de aplicación que requiera de la adquisición de datos y sobretodo de la transmisión de los mismos a una computadora remota para su análisis y/o almacenamiento, tales aplicaciones como la medición de flujo hidráulico en una presa, el monitoreo de las condiciones de diversas estaciones eléctricas y estaciones sísmicas, la medición de condiciones climatológicas en lugares adversos, o cualquier otra, es viable con las adecuadas modificaciones (como es el caso de la fuente de alimentación), y teniendo en cuenta que el sistema se debe ubicar dentro del área de cobertura de la red Tetrapol. Cabe mencionar, que dadas las características inalámbricas de comunicación, el sistema no se restringe a un ambiente fijo, por ello es posible emplearlo en aplicaciones desarrolladas para ambientes móviles, como es el monitoreo de variables en un vehículo.

Aunado a lo anterior, en caso de que la aplicación asignada al adquirente requiera, es posible utilizar los recursos de alta seguridad de las redes Tetrapol, para transmitir los datos confidencialmente. Por lo anterior, una de las características del sistema es su potencial crecimiento tanto en software como en hardware.

En el software, la programación estructurada utilizada en el μC , no sólo facilita la realización de pruebas, sino que simplifica su modificación, y permite la inclusión de rutinas específicas para una aplicación diferente. Por otro lado, dado que resulta más sencillo realizar cambios a la programación desarrollada en un lenguaje de alto nivel como Visual Basic que en un lenguaje de bajo nivel como ensamblador, los algoritmos

encaminados al procesamiento y despliegado de datos se llevan a cabo en el DAS. De esta manera se aprovechan las ventajas que ofrece una plataforma computacional (velocidad, procesamiento matemático, memoria, etc.), que explotadas por el software basado en un ambiente gráfico generan una interfaz muy amigable.

En el hardware, aunque el SAD cuenta con un ADC de ocho canales simples (cuatro diferenciales) sólo se utilizaron un canal en modo sencillo y uno en modo diferencial. Esto abre la posibilidad de monitorear dos variables adicionales en modo diferencial y una en modo sencillo, o cinco en modo sencillo. En caso de que alguna aplicación requiera la adquisición de un mayor número de variables es posible (con variaciones mínimas en la programación) añadir otro convertidor conectado en paralelo con el ADC existente, en un modo de operación mutuamente excluyente. Otra opción para incrementar el número de canales es implementando multiplexores.

Por otra parte, es posible aumentar la capacidad de almacenamiento, ya sea conectando hasta 8 circuitos integrados de memoria, o bien sustituyendo las memorias utilizadas (8 kbytes) por otras de igual funcionamiento, pero mayor capacidad (16, 32, 64 kbytes, 1 Mbyte), no utilizadas en este proyecto porque representaban un costo mayor, debido a que aún no se venden en México y porque las necesidades del proyecto no lo requirieron.

Otra ventaja que ofrece el sistema, es la adaptabilidad del software del μC a otros microcontroladores del mismo fabricante (AVR MEGA), con mayor capacidad (memoria, interrupciones, puertos, etc.), permitiendo así un sistema ampliamente escalable.

Finalmente, el desempeño del sistema al interior del sitio de repetición fue correcto. Las pruebas realizadas no mostraron errores cualitativos, sin embargo, dado que existe radiación constante incidente sobre el sistema, es probable que la interferencia capaz de alterar la secuencia de ejecución en el programa del Módulo Adquisidor de Datos pueda no ser evidente al analizar los resultados obtenidos. Esto debido a que el microcontrolador cuenta con un temporizador *WATCHDOG* que es un sistema de monitoreo de secuencia basado en un temporizador dedicado exclusivamente a reiniciar el programa cada determinado número de ciclos de reloj, por ende, si este temporizador no es reiniciado por medio de una instrucción antes de finalizar su conteo, el programa vuelve a ejecutarse desde el vector de reset, evitando así que la secuencia de ejecución quede confinada a un ciclo infinito.

En función de esta herramienta, la interferencia puede afectar la secuencia de ejecución sin ser detectada, a menos que la misma afecte la secuencia en un punto crítico como puede ser la comunicación o algún otro tipo de interacción con componentes externos al microcontrolador en la que sea notable un error. Es por ello que la siguiente etapa de este proyecto corresponde a la realización de un mayor número de pruebas dentro de un sitio de repetición para denotar los posibles errores aleatorios, realizar el diseño del circuito impreso con especificaciones altamente ruidosas y el uso de un contenedor de Faraday.

Finalmente, es posible concluir que el objetivo principal (desarrollar un sistema capaz de emplear los recursos de transmisión de datos de una red Tetrapol) se cumplió.

A continuación se presentan las características del sistema obtenido:

Módulo Adquisidor de Datos

- Escalabilidad:
 - 16 canales sencillos u 8 diferenciales
 - Capacidad de almacenamiento: 16, 32 64 kbytes, 1 Mbyte
- Facilidad de uso
- Rangos de medición
 - Temperatura 0 a 50 [°C]
 - Nivel de combustible 0 a 100%
- Resolución
 - Temperatura 0.48 [°C]
 - Nivel de combustible 0.1%

Software del DAS

- Confiabilidad
- Capacidad de comunicación punto multipunto
- Interfaz amigable
- Funcionamiento en sistemas operativos Microsoft (Windows 95, 98, XP, NT)
- Acceso controlado (manejo de contraseñas)

BIBLIOGRAFÍA

- Comer, Douglas; Redes de Computadoras, Internet e Interredes; PRENTICE HALL; 1ra edición; México, 1997; pp. 156-170
- Ford Merilee; Tecnologías de Interconectividad de redes; PRENTICE-HALL; México, 1998.
- Loshin Peter; TCP/IP Clearly explained; AP Professional; E.E.U.U.A. 1997; pp. 103-111.
- Gallo, Michael A.; Networking Explained; Digital Press, E.E.U.U.A. 1999.
- Stallings William; High-speed networks; PRENTICE-HALL; E.E.U.U.A. 1998.
- Bekkers, Rudi; Mobile Telecommunications: standars, regulation and applications; Boston, Artech House, 1999.
- Pérez Fontán Fernando; Ingeniería de Sistemas Trunking; Editorial Síntesis, España, 1998.
- Ziemer E. Rodger; Introduction to Digital Communications; PRENTICE HALL; 2da. Edición E.E.U.U.A. 1992.
- Morrison Ralph; Noise and Other Interfering Signals; Wiley – Interscience Publication, E.E.U.U.A. 1992.
- Hernando Rábanos, José Maria; Comunicaciones móviles; Editorial Centro de Estudios Ramón Areces, Madrid, 1997
- Microsoft Visual Basic 6.0 Reference Library / Microsoft Corporation
 - Language Reference
 - Controls Reference
 - Component Tools GuideMicrosoft Press, E.E.U.U.A., 1998

- www.tetrapol.com
- <http://www.eads-telecom.net>
- <http://www.num2002.ch/en/telekommunikation/forschung/tetrapol/>
- <http://www.vbwinsock.com>
- <http://www.atmel.com>
- <http://www.national.com>
- <http://www.microchip.com>
- <http://www.seguridadpublica.gob.mx>

APÉNDICE A

PROGRAMA DEL MÓDULO ADQUISIDOR DE DATOS

```

.include "C:\Program Files\Atmel\AVR Studio\Appnotes\8515DEF.INC"
.include "C:\Program Files\Atmel\AVR Studio\Appnotes\MOV_DATA.INC"
.include "C:\Program Files\Atmel\AVR Studio\Appnotes\CONV_MAC.INC"

.EQU      TMUESTREO = $60      ;TIEMPO DE MUESTREO
.EQU      BMUESTREO = $61      ;RESPALDO TIEMPO DE MUESTREO
.EQU      TPRESENCIA = $62     ;TIEMPO DE PRESENCIA A LA RT
.EQU      BPRESENCIA = $63     ;RESPALDO TIEMPO DE PRESENCIA A LA RT
.EQU      TRADIOACK = $64      ;TIEMPO DE RADIO ACKNOWLEDGE
.EQU      TIMEFLAG = $65       ;BANDERAS DEL TIMER1
.EQU      ACKTIME = $66        ;TIEMPO DE ACKNOWLEDGE
.EQU      CANALES = $67        ;CANALES
.EQU      ADDR_L = $68         ;DIRECCION BAJA DE EEPROM EXTERNA
.EQU      ADDR_H = $69         ;DIRECCION ALTA DE EEPROM ALTA
.EQU      FLAG = $6a           ;BANDERAS
.EQU      TRANSTYPE = $6b      ;TIPO DE TRANSMISION
.EQU      BANDERA = $6c        ;BANDERAS
.EQU      MSGFLAG = $6d        ;BANDERA DE MENSAJE
.EQU      NbrePEAT = $6e       ;RETRANSMISIONES
.EQU      XLOW = $6f           ;PARTE BAJA
.EQU      XHIGH = $70          ;PARTE ALTA
.EQU      IDLOW = $71          ;IDENTIFICADOR BAJA
.EQU      IDHIGH = $72         ;IDENTIFICADOR ALTA
.EQU      TIMEBKUPL = $73      ;RESPALDO TIEMPO BAJA
.EQU      TIMEBKUPH = $74      ;RESPALDO TIEMPO ALTA
.EQU      POLLTIMEL = $75      ;TIEMPO POLEO BAJA
.EQU      POLLTIMEH = $76      ;TIEMPO POLEO ALTA
.EQU      COUNTER1 = $77       ;Sequence Number backup
.EQU      COUNTER2 = $78       ;Expected Sequence Number backup
.EQU      ZLOW = $79           ;APUNTADOR BAJA
.EQU      ZHIGH = $7a          ;APUNTADOR ALTA
.EQU      MINUTERO1 = $7b       ;CONTADOR MINUTOS1
.EQU      MINUTERO2 = $7c       ;CONTADOR MINUTOS2
;*** DEFINICIONES DEL CONVERTIDOR A/D
.EQU      RD = 7               ;PD7
.EQU      WR = 6               ;PD6
.EQU      CS = 5               ;PD5
.EQU      CLK = 4              ;PD4
.EQU      INT = 2              ;PD2
.DEF      ADCLOW = R10          ;Z1 y Zh
.DEF      ADCHIGH = R11        ;REGISTROS DE CONVERSION
.DEF      MUX = R18            ;REGISTRO DE CONF. DE MULTIPLEXOR Y PARAMETROS
;*** DEFINICIONES DE LAS MEMORIAS
.EQU      SDATA = 0
.EQU      SCLP = 1
.EQU      WP = 2
.EQU      OFFON = 3
.EQU      ACKLED = 4
.EQU      DI = 7
.EQU      DO = 6
.EQU      SCLP = 1             ;SCL Pin numero (PuertoB)
.EQU      SDAP = 0            ;SDA Pinb número (PuertoB)

```

**TESIS CON
FALLA DE ORIGEN**

```

;*** REGISTROS DE TRABAJO
.DEF COUNT = R16
.DEF TXBUF = R17
.DEF BSLAVE = R18
.DEF BFLAG = R19
.DEF EPROM = R20
.DEF CSLAVE = R21
.DEF RXREG = R22
.DEF save_stack1 = r0
.DEF save_stack2 = r2
.DEF LO = R16
.DEF HI = R17
.DEF ID = R20
.DEF CONTADOR = R21
.DEF MSGTYPE = R22
.DEF AUXREG = R23
.DEF RXTMP = R24
.DEF XOR = R25
.DEF COUNTERn1 = R30
.DEF COUNTERn2 = R31
.DEF TIMELOW = R28
.DEF TIMEHIGH = R29
.DEF LENGHTl1sb = R30
.DEF LENGHTmsb = R31
;*** Igualdades y definiciones usadas por las rutina de aritmética de punto flotante
.DEF QUOT = R9
.DEF QUOTM = R10
.DEF QUOTH = R11
.DEF DEXP = R12
.DEF MANT1 = R13
.DEF MANT1M = R14
.DEF MANT1H = R15
.DEF EXP1 = R16
.DEF SIGN1 = R17
.DEF MANT2 = R18
.DEF MANT2M = R19
.DEF MANT2H = R20
.DEF EXP2 = R21
.DEF SIGN2 = R22
.DEF MANT1T = R23
.DEF COUNT = R24
.DEF TEMP = R25

.dseg
DECBUFF: .byte 10
MATHTMP: .byte 8 ;posicion dentro de la sram de los stack virtuales
.cseg

;*****
;* P R O G R A M A: VECTORES DE INICIO
;*****
.ORG $0000
rjmp Inicio
.ORG $0001
rjmp EXTINT0
.ORG $0002
rjmp EXTINT1
.ORG $0006
rjmp TIMER1
.org $0007
rjmp TIMER0
.org $009
rjmp uart_rx
;*****
;* Stack pointer: inicialización de localidades de pila
;*****
Inicio:
cli
ldi r16,high(RAMEND)
out sph,r16
ldi r16,low(RAMEND)

```

```

.....
out      spl,r16
;.....
;* Definición de parámetros RS-232
;* xtal=4 Mhz , 4800 bauds, 8 bits de datos, 1 bit de stop, sin paridad
;.....
ldi      RXTMP, $33      ;4800 Bauds
out      UBRR, RXTMP
ldi      RXTMP, $9a
out      UCR, RXTMP     ;habilitación de Tx y Rx, y la interrupción de la uart
;.....
;* CONFIGURACIÓN DE PUERTOS Y TEMPORIZADORES
;* ESTADO INICIAL DE VARIABLES
;.....
ldi      RXTMP, $80
out      ACSR, RXTMP     ;comparador deshabilitado para ahorro de energía
ldi      RXTMP, 0b11000000 ;PD7,PD6, SENS2, DISPLAY, INT1, INTO, TXD, RXD
out      DDRD, RXTMP     ;Reg.de dir puerto D, configura puerto
ldi      RXTMP, 0b00010000 ;0 0 0 p z z 0 z
out      PORTD, RXTMP    ;Valor en el puerto D
ldi      RXTMP, 0b11111111 ;SDATA, SCLK, WP, OFFON, LED, LED
out      DDRA, RXTMP     ;Reg.de dir puerto B, configura puerto
ldi      RXTMP, 0b00000000 ;0 0 0 0 0 0 0 0
out      PORTA, RXTMP    ;Valor en el puerto B
ldi      RXTMP, 0b00000101
out      tcsr0, RXTMP    ;inicializa timer0, ck = 1024
out      tccr1b, RXTMP
ldi      RXTMP, $40
sts      ACKTIME, RXTMP  ;contador de 2seg.
ldi      RXTMP, $03
sts      NBREPEAT, RXTMP ;Contador de transmisiones por Frame
ldi      RXTMP, 0b00000000
out      gmsk, RXTMP     ;deshabilita la interrupción externa 0 (e int1)
ldi      RXTMP, $48
out      tcntlh, RXTMP   ;MSB
ldi      RXTMP, $e6
out      tcntll, RXTMP   ;LSB
ldi      RXTMP, $3c
out      TCNT0, RXTMP    ;recarga el timer0
                          ;199680*(1/4Mhz)=0.04992seg
ldi      RXTMP, $32
sts      TPRESENCIA, RXTMP ;50 x 12seg. = 10MIN
sts      BPRESENCIA, RXTMP ;inicia contador de 10 minutos
ldi      RXTMP, $19
sts      TRADIOACK, RXTMP ;50 x 12seg. = 300SEG
                          ;inicia contador de 5 minutos
ldi      RXTMP, $30
sts      TRANSTYPE, RXTMP ;"0"
ldi      RXTMP, 0b01000000
sts      BANDERA, RXTMP
ldi      RXTMP, 0b00000111 ;desactiva timers
cbr      RXTMP, 0b00000010 ;activa timer de PRESENCIA AL DAS
sts      TIMEFLAG, RXTMP
ldi      RXTMP, $03
sts      MINUTERO1, RXTMP
sts      MINUTERO2, RXTMP
sei
;.....
;habilita las interrupciones
MIENTRAS: lds      RXTMP, TIMEFLAG
sbrs    rjmp    MUESTREA
cbr     RXTMP, 0b10000000
sts     TIMEFLAG, RXTMP
rcall  DEQUIS
rcall  SEND_BUFFER
rcall  SEND_DLE_ETX_CS
rcall  SET_WAITACK ;establece bandera de estado WAIT_ACK
MUESTREA: lds     RXTMP, TIMEFLAG
sbrs    RXTMP, 6
rjmp   PRESENCIA
cbr     RXTMP, 0b01000000
sts     TIMEFLAG, RXTMP
rcall  CONVERTIDOR
rcall  MEMORIA
PRESENCIA: lds     RXTMP, TIMEFLAG

```

```

sbrs      RXTMP, 5
rjmp     RADIOACK
lds      RXTMP, TRANSTYPE
cpl      RXTMP, $30
brne    RADIOACK ;*0* cero, verifico si no estoy esperando un RADIO_T_ACK
lds      RXTMP, TIMEFLAG ;Aún no recibo el RADIO_T_ACK
cbr      RXTMP, 0b00100000
sts      TIMEFLAG, RXTMP
rcall    UTS_DAS_PRESENCIA ;Transmite mensaje al DAS, indicando presencia
RADIOACK:
lds      RXTMP, TIMEFLAG
sbrs    RXTMP, 4
rjmp    MIENTRAS
cbr     RXTMP, 0b00010000
sts     TIMEFLAG, RXTMP
lds     RXTMP, TRANSTYPE
cpl     RXTMP, $56 ;ES UNA *V*?
brne   SERAM
rcall  GETVAL
rjmp  MIENTRAS
SERAM:
cpl   RXTMP, $4d ;ES UNA *M*?
brne  SERAA
rcall CLEARZ
SERAA:
rjmp  MIENTRAS
EXTINT0: reti
EXTINT1: reti
;.....
;..... TIMER1 .....
;.....
TIMER1:
in     r0, sreg ;salva el registro de estados
push  RXTMP
ldi   RXTMP, $48
out   tcntlh, RXTMP ;MSB
ldi   RXTMP, $e6
out   tcntll, RXTMP ;LSB
lds   R2, TIMEFLAG
sbrs R2, 2
rjmp CMUESTREO ;Determina si es tiempo de muestrear
CPRESENCIA:
sbrs R2, 1
rjmp CPRESENCIA ;Es tiempo de transmitir un mensaje de presencia
CRADIOACK:
sbrs R2, 2
rjmp CHECK_RADIOACK ;Determina si se agotó el RADIO TRANSMISION ACK
rjmp TIMEDONE
CMUESTREO:
lds  RXTMP, MINUTERO1
dec  RXTMP
brne ONTIME2
lds  RXTMP, TMUESTREO ;contador de cada 12seg
dec  RXTMP
brne ONTIME1
lds  RXTMP, TIMEFLAG
sbr  RXTMP, 0b01000000
sts  TIMEFLAG, RXTMP
lds  RXTMP, BMUESTREO ;recarga registro con su respectivo respaldo en SRAM
ONTIME1:
sts  TMUESTREO, RXTMP
ldi  RXTMP, $05
ONTIME2:
sts  MINUTERO1, RXTMP
rjmp CPRESENCIA ;Menos del tiempo establecido
CPRESENCIA:
lds  RXTMP, MINUTERO2
dec  RXTMP
brne ONTIME4
lds  RXTMP, TPRESENCIA ;contador de cada 12seg
dec  RXTMP
brne ONTIME3
lds  RXTMP, TIMEFLAG
sbr  RXTMP, 0b00100000
sts  TIMEFLAG, RXTMP
lds  RXTMP, BPRESENCIA ;recarga registro con su respectivo respaldo en SRAM
ONTIME3:
sts  TPRESENCIA, RXTMP
ldi  RXTMP, $05
ONTIME4:
sts  MINUTERO2, RXTMP
rjmp CRADIOACK ;Menos del tiempo establecido

```

```

CHECK_RADIOACK:
    lds    RXTMP,TRADIOACK
    dec    RXTMP                ;decremento 12 segundos el radio ack timer
    breq   RADIOACK_OUT
    sts    TRADIOACK,RXTMP
    rjmp   TIMEDONE

RADIOACK_OUT:
    ldi    RXTMP,$19            ;50 x 12seg. = 300SEG
    sts    TRADIOACK,RXTMP     ;inicia contador de 5 minutos
    lds    RXTMP,TIMEFLAG
    abr    RXTMP,0b00010100
    sts    TIMEFLAG,RXTMP

;RETRANSMITIR 2 VECES
TIMEDONE:
    pop    RXTMP
    out    sreg,r0             ;recupera el registro de estados
    reti   ;regresa a esperar la siguiente interrupción
;.....
;.....          TIMER0
;.....

TIMER0:
    in     r0,sreg
    push  RXTMP
    ldi   RXTMP,$3c           ;recarga el timer0
    out   TCNT0,RXTMP        ;199680*(1/4Mhz)=0.04992seg
    lds   RXTMP,BANDERA
    sbrs RXTMP,2             ;Brinca si estamos en WAIT_ACK
    rjmp  CHECK_POLLTIME     ;si no, entonces chequeo si el POLLING TIME se agotó
                                ;Estamos en WAIT_ACK, detectado hasta 0.05segundos después de activado
                                ;NOTA: el timer de POLLING TIME, y el ACKTIME_OUT, son independientes, pero
                                ;para poder usar un solo timer del AVR para los dos, es necesario sincronizar
                                ;ambos timers, fijando el ACKTIME_OUT al POLLING TIME.
    lds   RXTMP,ACKTIME
    dec   RXTMP               ;PRECARGADO CON 40
    brne AUNFALTA            ;aún no son 2seg. ve a almacenar el ACKTIME a sram
                                ;Pasaron 2seg. Retransmitir en caso de ser posible
    rcall CLEAR_WAITACK
    lds   RXTMP,NbREPEAT
    dec   RXTMP               ;precargado con 3
    breq  MAX_Nb              ;decremento contador de número de retransmisiones
                                ;ya no puedo retransmitir, ve a MAX_Nb
    sts   NbREPEAT,RXTMP     ;aún puedo retransmitir
    lds   RXTMP,TIMEFLAG
    sbr   RXTMP,0b10000000
    sts   TIMEFLAG,RXTMP
    rjmp  RELOAD_ACKTIME     ;Otros 2seg más

MAX_Nb:
    ldi   RXTMP,$03
    sts   NbREPEAT,RXTMP     ;Recargo el contador de retransmisiones, para el próximo

FRAME
    lds   RXTMP,BANDERA
    cbr   RXTMP,0b00000010   ;establecer estado DISCONNECTED, close link level
    sts   BANDERA,RXTMP

RELOAD_ACKTIME:
    ldi   RXTMP,$40
    sts   ACKTIME,RXTMP

AUNFALTA:
CHECK_POLLTIME:
    push y1
    push yh
    lds   YL,POLLTIMEH
    lds   YH,POLLTIMEH
    sbiw  YL,$01
    brne FALTAUN            ;Aún no se agota el tiempo
                                ;Se agotó el tiempo
    lds   RXTMP,BANDERA
    cbr   RXTMP,0b00000010   ;establecer estado DISCONNECTED, close link level
    sts   BANDERA,RXTMP
    lds   YL,TIMEBKUPL
    lds   YH,TIMEBKUPH
                                ;recarga los contadores del Polling time con su backup
FALTAUN:
    sts   POLLTIMEH,YL
    sts   POLLTIMEH,YH
    pop   yh
    pop   y1
    pop   RXTMP
    out   sreg,r0
    
```

```

retl
;.....
;..... COMUNICACION UDT - ST .....
;.....
uart_rx:  in      r0,sreg
          push   RXTMP
          in     RXTMP,udr      ;r24
          cbi   ucr,rxcie     ;deshabilita la interrupcion de la UART
          cpi   RXTMP,$10     ;verifica si es un DLE
          brne rx            ;no es DLE
          lds   RXTMP,BANDERA ;es DLE, verifica si la bandera de comunicacion
          sbrs RXTMP,0        ;brinca si bandera de comunicacion establecida
          rjmp  exit_int     ;bandera de comunicacion no establecida
          sbrs RXTMP,1        ;brinca si estado es CONNECTED, MPAP CNX establecido
          rjmp  SET_CNK     ;Estado DISCONNECTED, ENTONCES VERIFICA SI ES CNX FRAME
          rjmp  CONNECT
;.....
;.....**Physical LEVEL**.....
;.....
rx:       cpi     RXTMP,$05     ;es la señal de conexión
          brne   noes          ;no, es otra cosa
data3:   sbis   usr,7          ;aguarda byte transmitido por la ST
          rjmp  data3
          in     RXTMP,udr
          cpi   RXTMP,$05     ;es la señal de conexión
          brne   noes          ;no, es otra cosa
send07:  ldi    RXTMP,$07     ;si es, contesta a la ST
          rcall mandal
          rcall Delay100us
          rcall Delay100us
          sbis   usr,7          ;aguarda byte transmitido por la ST
          rjmp  send07
          in     RXTMP,udr
          cpi   RXTMP,$13     ;es la señal de conexión
          brne   noes          ;Cancel ACK_TIME_OUT
          rcall CLEAR_WAITACK
          lds   RXTMP,BANDERA
          sbr   RXTMP,0b0000001 ;guarda el registro de bandera con el bit0 en alto
          sts   BANDERA,RXTMP ;es decir bandera de comunicacion establecida
noes:    rjmp  exit_int
;.....
;..... DISCONNECTED .....
;.....
SET_CNK:
          rcall FRAME_RECEPTION
          rcall CHECKSUM
          lds   RXTMP,BANDERA ;carga RXTMP con el registro de banderas de SRAM
          sbrs RXTMP,7        ;brinca si checksum no falló
          rjmp  DESCONECTADO
          lds   RXTMP,$0190
          cpi   RXTMP,$30
          brne DESCONECTADO
          rcall SEND_ACQ_CNK ;Manda ACQ_CNK, resetea contadores y activa timers
          rjmp  exit_int
DESCONECTADO:
          cli
          ldi   RXTMP,$80
          out  TMSK,RXTMP ;Desactiva interrupción OV_Timer0, TIMER1 ACTIVADO
          sei
          lds   RXTMP,BANDERA
          cbr   RXTMP,0b00000010 ;checksum falló, establecer estado DISCONNECTED
          sts   BANDERA,RXTMP
          lds   RXTMP,TIMEFLAG
          sbr   RXTMP,0b00000100 ;Desactiva timer de RADIOTRANS_ACK
          sbr   RXTMP,0b00000010 ;Desactiva timer de PRESENCIA al DAS
          sts   TIMEFLAG,RXTMP
          rjmp  exit_int
SEND_ACQ_CNK:
          ldi   RXTMP,$10
          rcall mandal
          ldi   RXTMP,$31

```

```

rcall    mandal
ldi      RXTMP,$10
rcall    mandal
ldi      RXTMP,$03
rcall    mandal
ldi      RXTMP,$32
rcall    mandal
ldi      COUNTERn1,$00      ;reset Sequence Number (r25)
sts      COUNTER1,COUNTERn1 ;r25 a $98
ldi      COUNTERn2,$00      ;reset expected Sequence Number (r26)
sts      COUNTER2,COUNTERn2 ;r26 a $99
lds      RXTMP,BANDERA
sbr      RXTMP,0b00000010    ;estado CONNECTED (bandera, bit 1 en alto)
sts      BANDERA,RXTMP
ldi      RXTMP,$03
sts      NREPEAT,RXTMP      ;Recargo el contador de retransmisiones
ret

;.....
;.....*CONNECT*.....
;.....
CONNECT:
rcall    FRAME_RECEPTION
rcall    CHECKSUM
lds      RXTMP,BANDERA
sbrc     RXTMP,7              ;brinca si checksum falló regresa a estado CONNECT
rjmp     FRAME_TYPE
rjmp     exit1                ;vuelve a estado CONNECT

FRAME_TYPE:
rcall    PUSHZ
lds      RXTMP,$0190         ;FRAMETYPE
cpi      RXTMP,$32          ;identificador de encabezados DTAP
breq     DATA              ;Ve a checar la secuencia de FRAMES
cpi      RXTMP,$30          ;identificador de CNX MPAP
breq     preSEND_ACQ_CNX    ;contesta con ACQ_CNX
cpi      RXTMP,$33          ;identificador de ACK MPAP
breq     preACK_RECEIVED    ;ACK recibido, cancela ACK_TIME OU:
rjmp     exit1              ;vuelve a estado CONNECT

preSEND_ACQ_CNX:
rcall    SEND_ACQ_CNX        ;manda ACQ_CNX, resetea contadores de FRAME y activa
timers 0 y 1
rcall    CLEAR_WAITACK      ;Cancela ACK_TIME_OUT, en caso de que estuviésemos en
WAIT_ACK previamente
rjmp     exit1              ;vuelve a estado CONNECT

preACK_RECEIVED:
lds      COUNTERn1,COUNTER1 ;saca Sequence Number (r30)
lds      RXTMP,$0191         ;RECEPTION SEQUENCE NUMBER
cp       RXTMP,COUNTERn1    ;determina si es el ACK FRAME esperado, u otro
brne     exit1              ;frame desconocido, salte
rcall    CLEAR_WAITACK      ;Cancel ACK_TIME_OUT
rjmp     exit1              ;vuelve a estado CONNECT

DATA:
lds      COUNTERn2,COUNTER2 ;saca expected Sequence Number (r26)
lds      RXTMP,$0191         ;RECEPTION SEQUENCE NUMBER
cp       RXTMP,COUNTERn2    ;determina si es el FRAME esperado, o un anterior
brne     PREVIOUS          ;Frame anterior
rcall    INCOUNTER          ;Frame esperado, contesta con un ACKFRAME
rcall    DTAP               ;Procesa protocolo DTAP
rjmp     exit1              ;vuelve a estado CONNECT

INCOUNTER:inc
COUNTern2
COUNTern2,COUNTern2       ;incrementa contadores de FRAME
ret
COUNTern2,COUNTern2       ;sube expected Sequence Number (r31)

PREVIOUS:dec
COUNTern2                  ;PREVIO = EXPECTED - 1
cp       RXTMP,COUNTern2    ;determina si es el FRAME anterior u otro
brne     exit1              ;frame desconocido, salte
inc      COUNTern2
rcall    SEND_ACKFRAME      ;FRAME previamente recibido, vuelve a enviar el ACK_FRAME
exit1: rcall POPZ
rjmp     exit_int           ;vuelve a estado CONNECT

;.....
;.....Salida de interrupción de UART.....
;.....

```

```

exit_int: sbi      ucr,rxcie      ;habilita la interrupción de la uart
          pop
          out      sreg,r0
          reti
;.....
;.....SUBROUTINAS DE RECEPCIÓN DE FRAME Y CHECKSUM.....
;.....
FRAME_RECEPTION:
          ldi      zh,$01        ;a partir de la direccion $0190 (400d)
          ldi      zl,$90
          ldi      XOR,$10
data1:   sbis     usr,7          ;aguarda byte transmitido por la ST
          rjmp    data1
          in      RXTMP,udr
          st      z+,RXTMP      ;almacena en RAM y postincrementa
          eor     XOR,RXTMP     ;realiza or exclusiva y el resultado se almacena en XOR
          cpi    RXTMP,$10     ;verifica si es un 10 (delimitador final DLE)
          brne   data1        ;espera el siguiente byte, si lo anterior no se cumplió
otro10:  sbis     usr,7
          rjmp    otro10       ;Espera el siguiente byte para determinar si es otro 10
          in      RXTMP,udr
          eor     XOR,RXTMP     ;R25,R24, realiza or exclusiva
          cpi    RXTMP,$10     ;verifica si es otro 10, es decir el anterior no era DLE
          brne   sera03       ;se recibió el DLE, ahora espera el ETX ($03)
          rjmp    data1
sera03:  st      z+,RXTMP      ;almacena en RAM sea o no ETX
          cpi    RXTMP,$03     ;verifica si es el ETX
          breq   END_RECEPTION ;regresa de subrutina si efectivamente es el ETX
          rjmp   data1        ;no es ETX
END_RECEPTION:
CHECKSUM: sbis     usr,7          ;aguarda byte transmitido por la ST
          rjmp    CHECKSUM
          in      RXTMP,udr
          cp     XOR,RXTMP     ;compara nuestro CHECKSUM, con el recibido
          breq   CHECK_DONE    ;checksum correcto, determina FRAME TYPE
          lds    RXTMP,BANDERA
          cbr    RXTMP,0b10000000 ;checksum fallido (bandera, bit 7 en bajo)
          rjmp    UPBANDERA
CHECK_DONE:
          lds    RXTMP,BANDERA
          sbr    RXTMP,0b10000000 ;checksum correcto (bandera, bit 7 en alto)
UPBANDERA:
          sts    BANDERA,RXTMP
          ret
;.....
;..... SUBROUTINAS WAIT_ACK .....
;.....
SET_WAITACK:
          push   RXTMP
          lds    RXTMP,BANDERA
          sbr    RXTMP,0b00000100 ;BANDERA bit 2 en alto, estamos en WAIT_ACK
          sts    BANDERA,RXTMP
          pop
          ret
CLEAR_WAITACK:
          push   RXTMP
          lds    RXTMP,BANDERA
          cbr    RXTMP,0b00000100 ;BANDERA bit 2 en bajo, estamos en CONNECT
          sts    BANDERA,RXTMP
          pop
          ret
;.....
;..... SUBROUTINA DE PROTOCOLO DTAP .....
;.....
DTAP:    rcall   RESET_T_PRESENCE ;Resetea el TIMER de presencia
          lds    RXTMP,$0194
          cpi    RXTMP,$01
          brne   NOFUE
          lds    RXTMP,$0197     ;message type

```

```

        cpi      RXTMP,$00
        braeq   STU_CONNECT
        cpi      RXTMP,$01
        brne   SERA02
        rjmp    STU_NETWORK_INFO
SERA02: cpi      RXTMP,$02
        brne   SERA04
        rjmp    STU_ST_PRESENCE
SERA04: cpi      RXTMP,$04
        brne   SERA05
        rjmp    STU_RADIO_TRANS_ACK
SERA05: cpi      RXTMP,$05
        brne   NOFUE
        rjmp    STU_DATA_D
NOFUE:  ret
STU_CONNECT:
        rcall   PUSHXYZ
        push   CONTADOR           ;(R21)
        ldi    yh,$01             ;almacena DTAP interface version
        ldi    yl,$99             ;$0000 a $000b
        ldi    HI,$00
        ldi    LO,$10
        lds    CONTADOR,$0192
        dec    CONTADOR
        dec    CONTADOR
        dec    CONTADOR
        dec    CONTADOR
TRANSFER: ld      RXTMP,y+
        rcall   LO
        inc    CONTADOR
        dec    TRANSFER
        brne   TRANSFER
        clr    yh                 ;Limpia parte alta de yh
        clr    zh                 ;Limpia parte alta de zh
        lds    yl,$01a1           ;POLLING TIME
        lds    zl,$01a1           ;POLLING TIME
        ldi    RXTMP,$27
POR40:  clc                     ;limpia el carry
        add    yl,zl               ;Suma 40 veces el Polling time. PARTE BAJA
        adc    yh,zh               ;Suma parte ALTA
        dec    RXTMP
        brne   POR40             ;se multiplica por 40, la interrupción ocurre cada 0.05s
                                       ;así que si el POLLTIME es de 10s, 10x40=400 así 400x0.05=20s
                                       ;donde los 20s, son los 2xPOLLTIME

        sts    POLLTIMEL,YL
        sts    TIMEBKUPL,YL
        sts    POLLTIMEH,YH
        sts    TIMEBKUPH,YH
        pop
        rcall   POPXYZ
        lds    RXTMP,TIMEFLAG
        cbr    RXTMP,0b00000010    ;activa timer de PRESENCIA AL DAS
        sts    TIMEFLAG,RXTMP
        cli
        ldi    RXTMP,$48
        out    cnt1h,RXTMP         ;MSB
        ldi    RXTMP,$e6
        out    cnt1l,RXTMP        ;recarga el timer para 12s
        out    cnt1l,RXTMP        ;LSB
        ldi    RXTMP,$3c
        out    TCNT0,RXTMP        ;recarga el timer0
        ldi    RXTMP,$82
        out    TIMSK,RXTMP        ;activa interrupción OV_Timer0, TIMER1 TAMBIÉN ACTIVADO
        sei
        rjmp    UTS_CONNECT_ACK
UTS_CONNECT_ACK:
        push   MSGTYPE
        rcall   PUSHXYZ
        ldi    MSGTYPE,$50
        sts    MSGFLAG,MSGTYPE    ;UTS_CONNECT_ACK
        rcall   SETBUFFERA        ;Prepara apuntadores del BUFFER $89

```

```

ldi      RXTMP,$03          ;carga DTAP version
rcall   LOADBUFFER
clr     HI
ldi     LO,$16             ;carga COUNTRY CODE
rcall   ldse2b
rcall   LOADBUFFER
ldi     LO,$17             ;carga NETWORK CODE
rcall   ldse2b
rcall   LOADBUFFER
ldi     RXTMP,$00
rcall   LOADBUFFER
adw     YL,$04             ;4 bytes fijos dentro del header
rcall   UEQUIS
rcall   SETBUFFER          ;Prepara apuntadores del BUFFER $80
rcall   LOAD_FRAME_HEAD
rcall   DEQUIS
rcall   SEND_BUFFER
rcall   SEND_DLE_ETX_CS
rcall   POPXYZ
pop     MSGTYPE
;SET ACKTIME & ESTADO WAIT_ACK
rcall   SET_WAITACK        ;establece bandera de estado WAIT_ACK
ret

STU_NETWORK_INFO:
rcall   PUSHXYZ
push    CONTADOR          ;(R21)
ldi     YH,$01
ldi     YL,$99
ldi     HI,$00             ;almacena Network info
ldi     LO,$30             ;máximo 13 bytes
lds     CONTADOR,$0192
dec     CONTADOR
dec     CONTADOR
dec     CONTADOR
dec     CONTADOR
dec     CONTADOR          ;quito 5 bytes del DTAP header

TRANSFER2:ld
rcall   RXTMP,y+
rcall   stse2b
inc     LO
dec     CONTADOR
brne   TRANSFER2
ldi     HI,$00
ldi     LO,$30             ;Acceso a la dirección de NETWORK INFO
CHECKA: rcall   ldse2b
inc     LO
cpi     RXTMP,$01
breq   SYS_OP_MODE
cpi     SYS_OP_MODE
cpi     RXTMP,$FF
breq   REGRESA1
rcall   ldse2b
inc     LO
add    LO,RXTMP
rjmp   CHECKA

SYS_OP_MODE:
inc     LO
rcall   ldse2b
rcall   RXTMP,$00          ;PREGUNTO SI ES NORMAL MODE
breq   SET_NORMAL_MODE   ;SI ES, CONFIGURA REG. DE BANDERAS EN NORMAL MODE
lds     RXTMP,BANDERA
sbr    RXTMP,0b01000000    ;NO ES, CONFIGURA REG. DE BANDERAS EN FALL BACK MODE
rjmp   REGRESA1

SET_NORMAL_MODE:
lds     RXTMP,BANDERA
cbr    RXTMP,0b01000000   ;CONFIGURA REG. DE BANDERAS EN NORMAL MODE
REGRESA1: sts     BANDERA,RXTMP
pop     BANDADOR
rcall   POPXYZ
ret

```

```

STU_ST_PRESENCE:      ;En el STU_ST_PRESENCE, la UDT tiene que contestar con un UTS_ST_PRES_ACK
                    push   MSGTYPE
                    push   YL
                    push   YH
                    ldi    MSGTYPE,$55
                    sts    MSGFLAG,MSGTYPE
                    ldi    YL,$04          ;cuatro bytes fijos a enviar dentro de DATA
                    ldi    YH,$00
                    rcall  SETBUFFER      ;Prepara apuntadores del BUFFER $80
                    rcall  LOAD_FRAME_HEAD
                    rcall  UEQUIS
                    rcall  SEND_BUFFER
                    rcall  SEND_DLE_ETX_CS
                    pop    YH
                    pop    YL
                    pop    MSGTYPE
;SET ACKTIME & ESTADO WAIT_ACK
                    rcall  SET_WAITACK   ;establece bandera de estado WAIT_ACK
                    ret
STU_RADIO_TRANS_ACK:
                    lds    RXTMP,$0199
                    cp     RXTMP,R4
                    brne  NADA5
                    lds    RXTMP,$019a
                    cp     RXTMP,R5
                    brne  NADA5
                    lds    RXTMP,TIMEFLAG
                    sbr   RXTMP,0b00000100 ;desactiva timer de RADIO TRANS ACK
                    sts    TIMEFLAG,RXTMP
                    lds    RXTMP,$019b
                    cpi    RXTMP,$01
                    breq  TRANS_OK
                    cpi    RXTMP,$02
                    brne  NADA5
                    ret
TRANS_OK:          lds    RXTMP,TRANSTYPE
                    cpi    RXTMP,$4d      ;"M"
                    brne  NADA3
                    rcall  PUSHXYZ
                    rcall  NEXTBUFFER
NADA3:            ldi    RXTMP,$30
                    sts    TRANSTYPE,RXTMP
                    ret
NADA5:            ret
STU_DATA_D:      lds    RXTMP,$01ab      ;TIPO DE MSG
                    cpi    RXTMP,$56      ;"V"
                    breq  VA
                    cpi    RXTMP,$4d      ;"M"
                    breq  ME
                    cpi    RXTMP,$43      ;"C"
                    breq  CO1
                    rjmp  HECHO
CO1:             rjmp  CO
VA:             lds    RXTMP,$01ac
                    cpi    RXTMP,$41      ;"A"
                    breq  GETVAL
                    rjmp  HECHO
GETVAL:         rcall  r6,$01ad
                    rcall  CONVERTIDOR
                    ldi    MSGTYPE,$52
                    sts    MSGFLAG,MSGTYPE
                    rcall  SETBUFFERB     ;Prepara apuntadores del BUFFER $9b
                    ldi    RXTMP,$56      ;"V"
                    rcall  LOADBUFFER
                    ldi    RXTMP,$41      ;"A"
                    rcall  LOADBUFFER
                    mov    RXTMP,r6
                    rcall  LOADBUFFER

```

```

        ldi        z1,$9a                ;Apuntadores de datos del ADC almacenados temp. en RAM
        ldi        zh,$00
SENDVA:  ld         r1,RXTMP,z+          ;Carga TXBUF con el respectivo byte de datos del ADC
        rcall     LOADBUFFER
        dec       r1                    ;Decrementa contador de datos almacenados en RAM
        brne     SENDVA                 ;Si se agotaron los datos almacenados, envia STOP
        rcall     UEQUIS
        rcall     GETCOUNTERS
        rcall     SETBUFFER             ;Prepara apuntadores del BUFFER $80
        rcall     LOAD_FRAME_HEAD
        rcall     UPLINK_HEAD
        rcall     DEQUIS
        rcall     SEND_BUFFER
        rcall     SEND_DLE_ETX_CS
;SET ACKTIME & ESTADO WAIT_ACK
        ldi        RXTMP,$56           ;"V"
        sts       TRANSTYPE,RXTMP
        rcall     SET_WAITACK          ;establece bandera de estado WAIT_ACK
        lds       RXTMP,TIMEFLAG
        cbr      RXTMP,0b00000100     ;activa timer de RADIO TRANS ACK
        sts       TIMEFLAG,RXTMP
ME:      rjmp     HECHO
        rcall     PUSHXYZ
        lds       RXTMP,$01ac
        cpi      RXTMP,$45           ;"E"
        breq     CLEARZ
CLEARZ:  rjmp     HECHO
        clr      z1
        clr      zh
        clr      r9
        rcall     UZETA                ;Contador de paquetes de datos en memoria
        lds       r6,$01ad
        ldi      MSGTYPE,$52
        sts       MSGFLAG,MSGTYPE
        rcall     SETBUFFERB           ;Prepara apuntadores del BUFFER $9b
        ldi      RXTMP,$4d           ;"M"
        rcall     LOADBUFFER
        ldi      RXTMP,$45           ;"E"
        rcall     LOADBUFFER
        mov      RXTMP,r6
        rcall     LOADBUFFER
        mov      RXTMP,r9            ;r9 inicia en 0
        rcall     LOADBUFFER
        inc      r9
        ldi      HI,$00
        ldi      LO,$40
TRANSFER4:rcall  ldse2b                ;almacena FECHA Y HORA a partir de $001a y hasta $0032
        rcall     LOADBUFFER           ;24 bytes
        inc      LO
        cpi      LO,$58
        brne     TRANSFER4
        rjmp     NEXT7
NEXTBUFFER:
        ldi      MSGTYPE,$52
        sts       MSGFLAG,MSGTYPE
        rcall     SETBUFFERB           ;Prepara apuntadores del BUFFER $9b
        ldi      RXTMP,$4d           ;"M"
        rcall     LOADBUFFER
        ldi      RXTMP,$45           ;"E"
        rcall     LOADBUFFER
        mov      RXTMP,r6
        rcall     LOADBUFFER
        mov      RXTMP,r9            ;r9 inicia en 0
        rcall     LOADBUFFER
        inc      r9
NEXT7:  ldi      RXTMP,$4d           ;"M"
        sts       TRANSTYPE,RXTMP
        rcall     READ
        rcall     UEQUIS
        rcall     GETCOUNTERS

```

```

rcall    SETBUFFER           ;Prepara apuntadores del BUFFER $80
rcall    LOAD_FRAME_HEAD
rcall    UPLINK_HEAD
rcall    DEQUIS
rcall    SEND_BUFFER
rcall    SEND_DLE_ETX_CS
;SET ACKTIME & ESTADO WAIT_ACK
rcall    SET_WAITACK        ;establece bandera de estado WAIT_ACK
lds     RXTMP, TIMEFLAG
cbr     RXTMP, 0b00000100   ;activa timer de RADIO TRANS ACK
sts     TIMEFLAG, RXTMP
CO:      rjmp    HECHO
rcall    PUSHXYZ
lds     RXTMP, $01ac
cpl     RXTMP, $4f         ;"0"
brne    HECHO
ldi     yh, $01
ldi     yl, $ad
ldi     HI, $00           ;almacena FECHA Y HORA a partir de $0040
ldi     LO, $40          ;24 bytes
TRANSFER3:ld  RXTMP, y+
rcall    stse2b
inc     LO
cpl     yl, $c5
brne    TRANSFER3
lds     RXTMP, $01c5
sts     TMUESTREO, RXTMP
sts     BMUESTREO, RXTMP   ;contador de cada 12seg
lds     RXTMP, $01c6
ldi     LO, $0f
and     RXTMP, LO
sts     CANALES, RXTMP
lds     RXTMP, TIMEFLAG
cbr     RXTMP, 0b00000001   ;activa timer de MUESTREO
sts     TIMEFLAG, RXTMP
lds     RXTMP, $01c7
sts     TPRESENCIA, RXTMP
sts     BPRESENCIA, RXTMP ;Tiempo de presencia al DAS
HECHO:   rcall    POPXYZ
ret
UTS_DAS_PRESENCE:
rcall    PUSHXYZ
ldi     MSGTYPE, $52
sts     MSGFLAG, MSGTYPE
rcall    SETBUFFERB
ldi     RXTMP, $50        ;"P"
rcall    LOADBUFFER
ldi     RXTMP, $52        ;"R"
rcall    LOADBUFFER
ldi     RXTMP, $45        ;"E"
rcall    LOADBUFFER
ldi     RXTMP, $53        ;"S"
rcall    LOADBUFFER
ldi     RXTMP, $53        ;"S"
rcall    LOADBUFFER
ldi     RXTMP, $41        ;"A"
rcall    LOADBUFFER
ldi     RXTMP, $44        ;"D"
rcall    LOADBUFFER
ldi     RXTMP, $30        ;"0" CERO
rcall    LOADBUFFER
ldi     RXTMP, $31        ;"1" UNO
rcall    LOADBUFFER
ldi     RXTMP, $50        ;"P"
sts     TRANSTYPE, RXTMP
rcall    UEQUIS
rcall    GETCOUNTERS
rcall    SETBUFFER       ;Prepara apuntadores del BUFFER $80
rcall    LOAD_FRAME_HEAD
rcall    UPLINK_HEAD
rcall    DEQUIS

```

```

        rcall    SEND_BUFFER
        rcall    SEND_DLE_ETX_CS
;SET ACKTIME & ESTADO WAIT_ACK
        rcall    SET_WAITACK           ;establece bandera de estado WAIT_ACK
        lds     RXTMP,TIMEFLAG
        cbr     RXTMP,0b00000100      ;activa timer de RADIO TRANS ACK
        sts     TIMEFLAG,RXTMP
        rcall    POPXYZ
        ret

;.....
;.....*SUBROUTINAS DE TRANSMISIÓN*.....
;.....
SETBUFFERA:
        clr     yh
        clr     yl
        ldi    xh,$00                ;almacena BUFFER a partir de Dirección $008a
        ldi    xl,$89                ;$0089 a $025F
        ret

SETBUFFERB:
        clr     yh
        clr     yl
        ldi    xh,$00                ;almacena BUFFER a partir de Dirección $007c
        ldi    xl,$9b                ;$009b a $025F
        ret

SETBUFFER:ldi    xh,$00                ;almacena BUFFER a partir de Dirección $0080
        ldi    xl,$80                ;$0080 a $025F
        ret

LOADBUFFER:
        st      x+,RXTMP
        adiw   yl,$01                ;Contador de bytes a transmitir
        cpi    RXTMP,$10             ;En el buffer no se almacenan los 10's repetidos
        brne  BUFDONE                ;pero si se cuentan.
        adiw   yl,$01                ;Este contador incluye los 10's repetidos
BUFDONE: ret
        manda2: eor     XOR,RXTMP           ;R25,R24
                out    $0c,RXTMP
                sbis   $0b,6
        aguantaa2: sbis   aguantaa2
                rjmp  $0b,6
                cpi    RXTMP,$10         ;Limpia el bit de TXC (Transmit Complete)del registro USR
                brsq  mandal
                ret
        mandal: eor     XOR,RXTMP           ;R25,R24
                out    $0c,RXTMP
        aguantaa: sbis   aguantaa
                rjmp  $0b,6
                sbi    $0b,6
                ret
SEND_ACKFRAME:
        ldi    RXTMP,$10
        rcall  mandal
        ldi    XOR,$10                ;10 XOR 33
        ldi    RXTMP,$33
        rcall  mandal
        mov    RXTMP,COUNTERn2
        rcall  manda2
        rcall  SEND_DLE_ETX_CS
        ret
GETCOUNTERS:
        mov    R3,yl                ;contador de DATOS del buffer incluyendo 10's rep.
        mov    R7,yh
        cpi    yl,$10
        brne  CLR1
        adiw   yl,$01
        CLR1: adiw   yl,$0e            ;Agrego 14 bytes de encabezado fijo
        mov    z1,yl                ;Z es el siguiente CONTADOR
        mov    zh,yh
        cpi    z1,$10
        brne  CLR2
        adiw   yl,$01
    
```

```

CLR2:    adiw    y1,$08                ;Agrego 8 bytes de encabezado fijo
         rcall  GETID                ;Obtengo identificador interno
         ret

GETID:   rcall  PUSHXYZ
         lds    RXTMP,IDLW
         mov    XL,RXTMP
         mov    R4,RXTMP
         lds    RXTMP,IDHIGH
         mov    XH,RXTMP
         mov    R5,RXTMP
         adiw   XL,1                  ;Incrementa identificador
         sts    IDLOW,XL
         sts    IDHIGH,XH
         sbiw   XL,$01
         cpi    XL,$10
         brne   LOWCLR
CKHIGH:  cpi    XH,$10
         breq   AMBOS
         rjmp   UNO
LOWCLR:  cpi    XH,$10
         breq   UNO
         rcall  POPXYZ
         ret
AMBOS:   rcall  POPXYZ
         adiw   YL,$02
         ret
UNO:     rcall  POPXYZ
         adiw   YL,$01
         ret

LOAD_FRAME_HEAD:
         rcall  PUSHZ
         lds    COUNTERn1,COUNTER1
         ldi    RXTMP,$10
         st     x+,RXTMP
         ldi    RXTMP,$32
         st     x+,RXTMP
         mov    RXTMP,COUNTERn1      ;ENVIA SEQUENCE NUMBER
         st     x+,RXTMP
         mov    RXTMP,YL
         st     x+,RXTMP
         mov    RXTMP,YH
         st     x+,RXTMP
         ldi    RXTMP,$01
         st     x+,RXTMP
         ldi    RXTMP,$00
         st     x+,RXTMP
         mov    RXTMP,MSGTYPE
         st     x+,RXTMP
         ldi    RXTMP,$00
         st     x+,RXTMP
         inc    COUNTERn1            ;Incrementa contador de secuencia
         sts    COUNTERn1,COUNTERn1
         rcall  POPZ
         ret

UPLINK_HEAD:
         mov    RXTMP,R4              ;Manda Identificador interno
         st     x+,RXTMP
         mov    RXTMP,R5
         st     x+,RXTMP
         mov    RXTMP,z1              ;Contador medio
         st     x+,RXTMP
         mov    RXTMP,zh
         st     x+,RXTMP
         ldi    RXTMP,$8a            ;Identificador de radioterminal
         st     x+,RXTMP
         ldi    RXTMP,$38
         st     x+,RXTMP
         ldi    RXTMP,$71
         st     x+,RXTMP
         ldi    RXTMP,$d2            ;Dirección IP
         st     x+,RXTMP

```

```

ldi    RXTMP,$00
st     x+,RXTMP
ldi    RXTMP,$01
st     x+,RXTMP
ldi    RXTMP,$01
st     x+,RXTMP
ldi    RXTMP,$81      ;Puertos UDP
st     x+,RXTMP
ldi    RXTMP,$00
st     x+,RXTMP
ldi    RXTMP,$81
st     x+,RXTMP
ldi    RXTMP,$00
st     x+,RXTMP
ldi    RXTMP,$00
st     x+,RXTMP
mov    RXTMP,R7      ;Contador de datos
st     x+,RXTMP
mov    RXTMP,R3
st     x+,RXTMP
ret

SEND_BUFFER:
ldi    zh,$00
ldi    zl,$80
ldi    XOR,$00      ;10 XOR 0 = 10
MORE:  ld    RXTMP,z+
       rcall mandal
       cpse x1,z1
       rjmp MORE
       cpse xh,zh
       rjmp MORE
       ret

SEND_DLE_ETX_CS:
ldi    RXTMP,$10
rcall  mandal
ldi    RXTMP,$03
rcall  mandal
mov    RXTMP,XOR      ;envia checksum
rcall  mandal
ret

RESET_T_PRESENCE:
lds   RXTMP,TIMEBKUPL ;extrae de SRAM el backup
sts   POLLTIMEL,RXTMP
lds   RXTMP,TIMEBKUPH ;extrae de SRAM el backup
sts   POLLTIMEH,RXTMP ;recarga parte alta
ret

;.....
;..... RUTINA del CONVERTIDOR .....
;.....
CONVERTIDOR:
push  RXTMP      ;Una instrucción dura 250ns
push  r23
push  r18      ;MUX
push  y1
push  yh
push  ADCLOW   ;= R30 Z1 y Zh
push  ADCHIGH  ;= R31 REGISTROS DE CONVERSION
clr   r1      ;Limpia contador de datos del ADC a almacenar en RAM
ldi   yh,$00   ;almacena DATOS DEL ADC a partir de Dirección $009A
ldi   y1,$9a   ;máximo 8 bytes
lds   RXTMP,CANALES
cpi   RXTMP,$00
breq  APAGADO
ldi   r23,0b00000001
QUECANAL: lds   RXTMP,CANALES
cpi   r23,0b00010000 ;Determina si hay que comenzar de nuevo con el canal 1
breq  APAGADO        ;brinca, si se han muestreado los cuatro canales
and   RXTMP,r23      ;Determina si el canal en cuestión esta activado
breq  NEXTCH        ;Si no esta activado brinca a NEXTCH
cpi   r23,0b00000001
brne  CANAL2

```

```

ldi      MUX,0b00000000      ;Carga MUX con la configuración del CANAL 1
rjmp    MUXDONE
CANAL2:  cpi      r23,0b00000010
brne    CANAL3
ldi      MUX,0b00010000      ;Carga MUX con la configuración del CANAL 2
rjmp    MUXDONE
CANAL3:  cpi      r23,0b00000100
brne    CANAL4
ldi      MUX,0b00100000      ;Carga MUX con la configuración del CANAL 3
rjmp    MUXDONE
CANAL4:  ldi      MUX,0b00110000 ;Carga MUX con la configuración del CANAL 4
rjmp    MUXDONE
APAGADO: pop      ADCHIGH      ;= R31 REGISTROS DE CONVERSION
pop      ADCLOW      ;= R30 Zl y Zh
pop      yh
pop      y1
pop      r18
pop      r23
pop      RXTMP
ret
MUXDONE: sbi      PORTD,CS      ;CS en alto deshabilita RD y WR
sbi      PORTD,RD      ;RD debe estar en alto
out      PORTC,MUX      ;Configuración de multiplexor, etc. al buffer de salida
swab    MUX            ;Invierto nibbles
cbi      PORTD,CS      ;CS y WR en bajo para habilitar buffer de entrada del adc
cbi      PORTD,WR      ;ADC adquiere información del puerto C, pulso de 50ns min
nop
sbi      PORTD,WR      ;Termina pulso en bajo, inicia conversión
sbi      PORTD,CS      ;La conversión dura 22 ciclos de reloj
CLOCKGEN: sbi      PORTD,CLK     ;Reloj en alto
nop      ;Compensación de tiempo
cbi      PORTD,CLK     ;Reloj en bajo
sbi      PINB,PB0      ;Determina si la conversión finalizó
rjmp    CLOCKGEN      ;No. no ha terminado, genera otro ciclo de reloj
cbi      PORTD,CS      ;CS y RD en bajo para habilitar buffer de salida del adc
cbi      PORTD,RD      ;ADC habilita su buffer de salida, byte1
nop      ;Tiempo extra para Output Data valid
in      ADCLOW,PINC    ;Lee el puerto c
sbi      PORTD,RD      ;Genera segundo pulso de RD para habilitar byte 2
sbi      PORTD,CS      ;Retardo entre alto bajo de RD=50ns min
cbi      PORTD,CS      ;ADC habilita su buffer de salida, byte2
nop      ;Output Data valid
in      ADCHIGH,PINC   ;Lee el puerto c
or      ADCHIGH,MUX    ;Coloca bandera de canal en el 2 byte, bits 4 a 7
st      y*,ADCLOW      ;Datos almacenados a partir de la dirección $9a
st      y*,ADCHIGH
inc     r1
inc     r1
sbi      PORTD,RD      ;RD y CS en alto para la siguiente conversión
sbi      PORTD,CS      ;CS en alto deshabilita RD y WR
NEXTCH:  lsl      r23      ;Si no esta activado, recorre la mascara
rjmp    QUECANAL      ;Continua con el muestreo del siguiente canal
;.....
;ESCRITURA DE MEMORIAS
;.....
MEMORIA: ;escritura en memorias externas
clr     RXTMP
cp      RXTMP,r1      ;Determina si hay datos a almacenar en EEPROM
brac    ADIOS        ;No hay datos, salte
push   TXBUF         ;= R17
push   BSLAVE        ;= R18
push   BFLAG         ;= R19
push   EEPROM        ;= R20
push   r23           ;= R28
push   VL            ;= R29
push   YH            ;= R30
push   z1            ;= R31
push   zh            ;= R31
;Memorias de 8192 bytes o 64kbits

```

```

rcall    DADDR
sbrc    YH,7 ;Verifica si ya se usaron todas las mem.
rjmp    FULL ;Memorias llenas no puedo hacer nada.
lds     BFLAG,FLAG;R19
cbi     PORTA,OFFON ;habilito memorias
rcall   delay100us
cbi     PORTA,WP ;desactivo WP
rcall   delay100us ;retardo para estabilizar memorias
cir     EPROM
rcall   SETSLAVE ;Prepara BSLAVE en función de YH, y también R23
ldi     z1,$9a ;Apuntadores de datos del ADC almacenados temp. en RAM
ldi     zh,$00

BYTE:
rcall   BSTART ;SLAVE se configura como 1 0 1 0 A2 A1 A0 R/W
mov     TXBUF,BSLAVE ;envia STARTBIT
rcall   TX ;carga el TXBUF con la direccion de esclavo
mov     TXBUF,YH ;carga el TXBUF con la direccion MSB
rcall   TX
mov     TXBUF,YL ;carga el TXBUF con la dirección LSB
rcall   TX

STORE:
ld      TXBUF,z+ ;Carga TXBUF con el respectivo byte de datos del ADC
rcall   TX
ldiw   YL,$01 ;Incrementa el apuntador de direccion de EPROM
ldi     RXTMP,0b11100000 ;Máscara para extraer esclavo al que se escribirá
and    RXTMP,YH ;Aplico máscara
cpse   r23,RXTMP
rjmp   NEXTMEM ;determina si aún hay memorias disponibles
dec    r1 ;Decrementa contador de datos almacenados en RAM
breq   DOSTOP ;Si se agotaron los datos almacenados, envía STOP
rjmp   STORE

NEXTMEM:
rcall   SENDSTOP ;Finaliza la escritura en la memoria actual
sbrc   YH,7 ;Verifica si ya se usaron todas las mem.
rjmp   BYTEDONE ;Memorias llenas
dec    r1 ;Decrementa contador de datos almacenados en RAM
breq   BYTEDONE ;Si se agotaron los datos a almacenar, salte
rcall   SETSLAVE ;Prepara el esclavo memoria
rjmp   BYTE

DOSTOP:
rcall   SENDSTOP
BYTEDONE:
sbi     PORTA,WP ;activa WP
nop
sbi     PORTA,OFFON ;DESACTIVA MEMORIAS
sbr     BFLAG,$0b00000010 ;Activo bandera de por lo menos una escritura, bit1
rcall   UADDR
sts     zh,FLAG,BFLAG
FULL:
pop     zh ;= R31
pop     z1 ;= R30
pop     YH ;= R29
pop     YL ;= R28
pop     r23
pop     EPROM ;= R20
pop     BFLAG ;= R19
pop     BSLAVE ;= R18
pop     TXBUF ;= R17

ADIOS:
ret
;.....
; * BAJAR_DATOS_PC
; * Envía a la PC los datos de las lecturas almacenadas en la eeprom externa
;.....
READ:
lds     BFLAG,FLAG;R19
sbrs   BFLAG,1 ;verifico bandera de por lo menos una escritura
ret
cbi     PORTA,OFFON ;habilito memorias
rcall   delay100us
cir     EPROM ;r20 = 0
rcall   DADDR2 ;contadores de lectura iniciados con los ultimos
;valores de los contadores de escritura
rcall   DZETA
    
```

```

READ1:   rcall   SETCSLAVE      ;CSLAVE en función de Z
         dec     CSLAVE        ;Iniciado como 0b10100001 - 1
         rcall   BSTART
         mov     TXBUF,CSLAVE
         rcall   TX
         clr     TXBUF
         rcall   TX
         clr     TXBUF
         rcall   TX           ;primera dirección de lectura $0000
READ2:   rcall   BSTART
         inc     CSLAVE        ;CSLAVE reinicia en mem 000 modo lectura
         mov     TXBUF,CSLAVE
         rcall   TX
         ser     RXTMP
         mov     r3,RXTMP
         rjmp    RBYTE
SENDACK: cbr     EEPROM,0b01000000 ;DO = 0
         rcall   BITOUT        ;envia ack
RBYTE:   rcall   RX1           ;RECIBO BYTE DE LA MEM.
         rcall   LOADBUFFER
         adiw    z1,1         ;Incrementa contador de direcciones
CHKLENGHT: cpi    x1,$38       ;Compara con $38
         brsh   CHKMSB        ;Brinca si es mayor o igual a $1c
CHKMSB:  cpi    xh,$02
         breq  CHKPAR
CHKMEM:  cpse   z1,r7         ;Envío 471 a 472 datos en cada FRAME DTAP
         rjmp  NADA          ;Determino si fue el último byte almacenado
         cpse   zh,r8         ;parte baja
         rjmp  NADA          ;parte alta
NADA:    ldi    RXTMP,0b11000000 ;Efectivamente, fue el último byte almacenado
         and   RXTMP,zh       ;Máscara para extraer esclavo
         cpse   r23,RXTMP    ;Aplico máscara
         rjmp  MEMNEXT      ;esclavo al que se desea leer es el usado actualmente
MEMNEXT: rcall   BSTART      ;No es el mismo esclavo
         rjmp  SENDACK      ;Envia ack a la mem. y lee el sig. byte
CHKPAR:  com    r3
         brne  EOFFRAME
MEMDONE: sbr    BFLAG,0b10000000 ;Activo bandera de memorias vacias, bit 7
         ldi   RXTMP,$45 ;"E"
         sts   TRANSTYPE,RXTMP
         rcall BSTOP
         ldi   RXTMP,$31 ;"1" UNO
         rcall LOADBUFFER
         rjmp  PORFIN
EOFFRAME: cpse   z1,r7         ;Determino si fue el último byte almacenado
         rjmp  NADA2        ;parte baja
         cpse   zh,r8         ;parte alta
         rjmp  NADA2
NADA2:   rcall   UZETA        ;GUARDA Z para no alterar los apuntadores al regresar
         rcall   BSTOP
         ldi    RXTMP,$30
         rcall   LOADBUFFER ;"0" CERO
         rjmp   PORFIN
PORFIN:  sbi    PORTA,PFON    ;desactivo memorias
         sts    FLAG,BFLAG
         ret
;.....SUBRUTINAS.....
;.....
SETSLAVE: ldi    BSLAVE,0b11000000
         and   BSLAVE,YH
         mov   r23,BSLAVE
         swap BSLAVE
         ldi  RXTMP,$a0
         add  BSLAVE,RXTMP
         ret

```

```

SENDSTOP:rcall    BSTOP
              ldi    RXTMP,$02           ;contador de intentos de recepción de ACK
NOACK:          rcall delay5ms          ;retardo para permitir escritura en EEPROM
              rcall delay100us
              rcall BSTART             ;envia byte de prueba a la EEPROM
              mov   TXBUF,BSLAVE        ;carga el TXBUF con la direccion de esclavo
              rcall TX                  ;
              sbrs  EEPROM,DI           ;verifico ACK de la memoria, brinca si no lo hay
              ret   ;escritura completada
              dec   RXTMP               ;no se recibió ACK
              brne NOACK               ;inténtalo una vez más

BSTART:        sbi    PORTA,SCLK        ;asegura niveles altos
              sbi    PORTA,SDATA        ;bus not busy
              rcall delay20us
              sbi    ddrA,SDATA         ;línea de datos como salida
              cbi    PORTA,SCLK         ;reloj en nivel bajo
              rcall delay20us
              sbi    PORTA,SCLK         ;reloj en nivel alto
              rcall delay20us
              cbi    PORTA,SDATA        ;cambio de SDATA de alto a bajo
              rcall delay20us
              cbi    PORTA,SCLK         ;reloj en bajo fin de pulso
              ret

BSTOP:         sbi    DDRA,SDATA        ;línea de datos como salida
              nop
              nop
              cbi    PORTA,SDATA        ;SDATA en bajo
              rcall delay20us
              sbi    PORTA,SCLK         ;reloj en alto
              rcall delay20us
              sbi    PORTA,SDATA        ;cambio de SDATA de bajo a alto
              rcall delay20us
              cbi    PORTA,SCLK         ;reloj en bajo fin de pulso
              ret

BITOUT:        sbi    DDRA,SDATA        ;línea de datos como salida
              rcall delay20us
              sbrs  EEPROM,DO           ;brinca si DO = 1
              rjmp  BITLOW              ;el bit a enviar es 0
              sbi    PORTA,SDATA        ;el bit a enviar es 1
              rjmp  CLKOUT              ;genera pulso de reloj para validar dato
              cbi    PORTA,SDATA        ;bit = 0

BITLOW:        rcall delay20us
CLKOUT:        sbi    PORTA,SCLK        ;reloj en alto
              rcall delay20us
              cbi    PORTA,SCLK        ;reloj en bajo
              ret

BITIN:         sbr   EEPROM,0b10000000 ;DI = 1
              sbi    PORTA,SDATA        ;condición de "Bus not busy"
              rcall delay20us
              cbi    DDRA,SDATA         ;línea de datos como entrada
              cbi    PORTA,SDATA        ;SDATA SIN PULL-UP
              rcall delay20us
              sbi    PORTA,SCLK         ;reloj en alto, genera pulso para detectar bit o ack
              rcall delay20us
              rcall delay20us
              sbis  PINA,SDATA          ;brinca si SDATA en alto
              cbr   EEPROM,0b10000000 ;DI = 0, ACK recibido, o bit = 0
              rcall delay20us
              cbi    PORTA,SCLK         ;reloj en bajo, fin de pulso
              ret

TX:            ldi    COUNT,$08         ;contador de bits (r16)
TXLP:         cbr   EEPROM,0b01000000 ;DO = 0
              sbrc  TXBUF,7            ;brinca si el bit a enviar es 0
              sbr   EEPROM,0b01000000 ;DO = 1, el bit a enviar es 1
              rcall BITOUT             ;enviar bit
              rol   TXBUF              ;recorre a la izquierda un bit el byte de transmisión
              dec   COUNT               ;decrementa el contador de bits a enviar
              brne TXLP                ;faltan bits?
              rcall BITIN              ;ya se enviaron todos los bits, verifica ACK
              sbrc  EEPROM,DI           ;brinca si se recibió el ACK, DI = 0
    
```

```

        cbi          PORTA,PA4          ;led de fallo de ACK encendido
        ret
RX1:    clr          RXTMP
        ldi          COUNT,$08        ;R16
        cbc
RXLP:   rol          RXTMP
        rcall       BITIN
        sbrc        EEPROM,DI
        sbr         RXTMP,$01
        dec         COUNT
        brne        RXLP
        ret
SETCSLAVE: ldi       CSLAVE,0b11100000
        and         CSLAVE,$h
        mov         R23,CSLAVE
        swap
        ldi         RXTMP,$a1
        add         CSLAVE,RXTMP
        ret
;*****
;* FUNCION
;* Rutinas de retardo
*
;*****
Delay20us:push    r21
        ldi         r21,$06
delay20u: dec     r21
        brne       delay20u
        pop        r21
        ret
Delay50us:push    r21
        ldi         r21,$41
retacha1: dec     r21
        brne       retacha1
        pop        r21
        ret
Delay100us:push   r21
        ldi         r21,$88
retacha2: dec     r21
        brne       retacha2
        pop        r21
        ret
Delay5ms: push     r21
        push        r22
        ldi         r22,$1c
retacha3: ldi     r21,$00
retacha4: dec     r21
        brne       retacha4
        dec         r22
        brne       retacha3
        pop        r22
        pop        r21
        ret
;SUBROUTINAS PARA ESCRIBIR EN LA EEPROM
;EL DATO A ESCRIBIR ESTA EN UN REGISTRO
STSE2b: push       r22
        out         EEDR,RXTMP      ;PREPARA EL REGISTRO DE DATOS
        out         EARL,LO
        out         EARH,HI
        ldi         R22,$04
        out         EECR,R22        ;ACTIVA EL BIT EEMWE
        ldi         R22,$06
        out         EECR,R22        ;ACTIVA EL BIT EEW
        sbic        EECR,EWE        ;ESPERA A QUE EEW=0
E2_1:   rjmp        E2_1
        pop        r22
        ret
;*****
;SUBROUTINAS PARA LEER LA EEPROM
;LA DIRECCION SE ESPECIFICA DE FORMA DIRECTA
LDSE2b: push       r22

```

```

out      EEARL, LO
out      EEARH, HI
ldi     R22, $01
out      EECR, R22      ;ACTIVA EL BIT EERE
in      RXTMP, EEDR
pop     r22
ret
PUSHXYZ: mov     r10, x1
         mov     r11, xh
         mov     r12, y1
         mov     r13, yh
PUSHZ:  mov     r14, z1
         mov     r15, zh
         ret
POPXYZ:  mov     x1, r10
         mov     xh, r11
         mov     y1, r12
         mov     yh, r13
POPZ:   mov     z1, r14
         mov     zh, r15
         ret
DEQUIS: lds     XL, XLOW
         lds     XH, XHIGH
         ret
UEQUIS: sts     XLOW, XL
         sts     XHIGH, XH
         ret
DZETA:  lds     ZL, ZLOW
         lds     ZH, ZHIGH
         ret
UZETA:  sts     ZLOW, ZL
         sts     ZHIGH, ZH
         ret
DADDR2: lds     R7, ADDR_L
         lds     R8, ADDR_H
         ret
DADDR:  lds     YL, ADDR_L
         lds     YH, ADDR_H
         ret
UADDR:  sts     ADDR_L, YL
         sts     ADDR_H, YH
         ret

```

APÉNDICE B

PROGRAMA DEL SERVIDOR DE APLICACIÓN DE DATOS

VARIABLES Y FUNCIONES GLOBALES

```
Global Const KEY_RETURN = &HD
Global renglon As Integer
Global tiempo1 As Variant
Global datos As Object
Global presence(35) As Date
Global maxrow As Integer
Global row As Integer
Global puerto As Long
Global newseg As String
Global oldseg As String
Global remotedata(35) As String
Global totalbytes(35) As Integer
Global newdisp As Integer
Global lastseg As String
Global msgexpected As String
Global lapso As Integer
Global tmuestreo As String
Global tpomuestreo As String
Global canal As String
Global chanel As Integer
Global configura As Integer
Global seg As Integer
Global maxseg As Integer
Global indicador As Integer
Global msgind As Integer
Global indicador2 As Integer
Global remover(35) As String
Global timepres As Integer
Global presencetime(35) As Integer

Private Function FindFirstFreeSocket() As Long
    Dim i As Integer
    FindFirstFreeSocket = -1
    For i = 0 To Winsockarray.UBound
        If Winsockarray(i).Tag = "free" Then
            FindFirstFreeSocket = i
            Exit Function
        End If
    Next i
End Function
```


MENÚ PRINCIPAL

Código que establece parámetros de inicio y despliega la ventana de menú principal

```

Private Sub Form_Load()
    Winsock1.Protocol = sckUDPProtocol
    Winsock1.LocalPort = 30024
    Winsock1.Bind (30024)
    Winsock1.RemotePort = 30024
    Winsock2.Protocol = sckUDPProtocol
    Winsock2.LocalPort = 30025
    Winsock2.Bind (30025)
    Winsock2.RemotePort = 30024
    Winsock3.Protocol = sckUDPProtocol
    Winsock3.LocalPort = 30026
    Winsock3.Bind (30026)
    Winsock3.RemotePort = 30024
    Timer1.Interval = 60000 'interrupción cada minuto
    Timer2.Interval = 60000 'interrupción cada 10 minutos
    Timer1.Enabled = True
    Set datos = CreateObject("Excel.application")
    datos.workbooks.opentext FileName:="c:\datos1.xls"
    lapso = 0
    newdisp = 0
    lastseg = "0"
    msgexpected = "!"
    canal = Chr(0)
    configura = 0
    presence(1) = Now
    presence(2) = Now
    For i = 0 To 34
        remover(i) = "no"
    Next i
    For i = 0 To 34
        presencetime(i) = 10
    Next i
End Sub

```

Código que despliega la información del dispositivo seleccionado

```

Private Sub combo_activos_Click()
    Dim i As Integer
    C_configurar.Enabled = True
    C_verinfo.Enabled = True
    C_bajarinfo.Enabled = True
    Label1.Caption = combo_activos.Text
    i = 1
    Do While datos.cells(i, 1).Formula <> ""
        i = i + 1
    Loop
    maxrow = i - 1

```

```

For i = 1 To maxrow
  If combo_activos.Text = datos.cells(i, 1).Formula Then
    renglon = i
  End If
Next i
For i = 0 To 3
  Label_data(i).Enabled = True
  Text1(i).Enabled = True
  Text1(i).Text = datos.cells(renglon, i + 2).Formula
Next i
C_modificar.Enabled = True
Winsock1.RemoteHost = datos.cells(renglon, 4).Formula
End Sub

```

Opción "CONFIGURAR EQUIPO"

```

Private Sub C_configurar_Click()
  Form2.Show
  Form1.Hide
End Sub

```

Opción "OBTENER INFORMACIÓN DE LAS MEMORIAS"

```

Private Sub C_bajarinfo_Click()
  Dim indice As Long
  indice = FindFirstFreeSocket      'Investiga si hay un puerto libre dentro del arreglo de Winsocks
  If indice = -1 Then               'Si no hay ningún puerto libre añade otro puerto
    indice = Winsockarray.UBound + 1
    Load Winsockarray(indice)
    puerto= Winsockarray(indice).LocalPort
  End If

  Winsock1.SendData "ME" + msgexpected + Puerto
  Winsockarray(indice).tag="busy"
  Form1.Enabled = False
  F_espera.Show
  Timer2(indice).Enabled = True    'Activa interrupción cada 10 minutos
End Sub

```

Opción "VALORES ACTUALES"

```

Private Sub C_verinfo_Click()
  Dim indice As Long
  indice = FindFirstFreeSocket      'Investiga si hay un puerto libre dentro del arreglo de Winsocks
  If indice = -1 Then               'Si no hay ningún puerto libre añade otro puerto
    indice = Winsockarray.UBound + 1
    Load Winsockarray(indice)
    puerto= Winsockarray(indice).LocalPort
  End If
  Winsock1.SendData "VA" + msgexpected + puerto
  Form1.Enabled = False
  F_espera.Show
  Timer2(indice).Enabled = True    'Activa interrupción de 10 minutos
End Sub

```

Opción "MODIFICAR"

```
Private Sub C_modificar_Click()
    Form1.Visible = False
    F_pass.Visible = True
End Sub
```

Opción "SALIR"

```
Private Sub C_SALIR_Click()
    datos.run ("macro2") 'activa macro de excel que restablece la protección
    datos.run ("guardar") 'activa macro de excel que guarda los cambios realizados
    datos.application.quit
    Set datos = Nothing
End
End Sub
```

.....

RUTINA DE MODIFICACIÓN

.....

VENTANA DE CONTRASEÑA**Opción "ACEPTAR"**

```
Private Sub C_pass_aceptar_Click()
    Dim nusuario, password As Variant
    nusuario = "tesis"
    password = "tesis"
    If Text1.Text = nusuario And Text2.Text = password Then
        Text1.Text = ""
        Text2.Text = ""
        F_pass.Visible = False
        Form1.Visible = True
        Form1.Frame1.Enabled = True
        Form1.C_aceptar.Visible = True
        Form1.C_cancelar.Visible = True
        Form1.c_añadir.Visible = True
        Form1.C_modificar.Visible = False
        Form1.C_bajarinfo.Visible = False
        Form1.C_verinfo.Visible = False
        Form1.C_configurar.Visible = False
        Form1.C_SALIR.Visible = False
    Else
        MsgBox "PASSWORD INCORRECTO"
    End If
End Sub
```

Opción "CANCELAR"

```
Private Sub C_pass_cancelar_Click()
    Text1.Text = ""
    Text2.Text = ""
    F_pass.Visible = False
    Form1.Visible = True
End Sub
```

VENTANA DE MODIFICACIÓN**Opción "ACEPTAR"**

```
Private Sub C_aceptar_Click()  
  datos.run ("nomclave")  
  If newdisp = 1 Then  
    datos.cells(maxrow + 1, 1).Formula = T_nombre.Text  
  End If  
  For i = 0 To 15  
    datos.cells(maxrow + 1, i + 2).Formula = Text1(i).Text  
  Next i  
  C_aceptar.Visible = False  
  C_cancelar.Visible = False  
  c_añadir.Visible = False  
  C_modificar.Visible = True  
  Frame1.Enabled = False  
  C_bajarinfo.Visible = True  
  C_verinfo.Visible = True  
  C_configurar.Visible = True  
  C_SALIR.Visible = True  
  combo_activos.Visible = True  
  L_edactivos.Visible = True  
  L_nombre.Visible = False  
  T_nombre.Visible = False  
End Sub
```

Opción "CANCELAR"

```
Private Sub C_CANCELAR_Click()  
  Frame1.Enabled = False  
  C_bajarinfo.Visible = True  
  C_verinfo.Visible = True  
  C_configurar.Visible = True  
  C_SALIR.Visible = True  
  C_cancelar.Visible = False  
  C_aceptar.Visible = False  
  c_añadir.Visible = False  
  C_modificar.Visible = True  
  combo_activos.Visible = True  
  L_edactivos.Visible = True  
  L_nombre.Visible = False  
  T_nombre.Visible = False  
End Sub
```

Opción "AÑADIR"

```
Private Sub c_añadir_Click()  
  L_nombre.Visible = True  
  T_nombre.Visible = True  
  c_añadir.Visible = False  
  combo_activos.Visible = False  
  L_edactivos.Visible = False  
  newdisp = 1  
End Sub
```

.....
RUTINA DE CONFIGURACIÓN
.....

VENTANA DE CONTRASEÑA

Opción "ACEPTAR"

```
Private Sub C_aceptar_Click()  
Dim nusuario, password As Variant  
nusuario = "tesis"  
password = "tesis"  
If Text1.Text = nusuario And Text2.Text = password Then  
    Text1.Text = ""  
    Text2.Text = ""  
    F_config.Show  
    Form2.Hide  
Else  
    MsgBox "PASSWORD INCORRECTO"  
End If  
End Sub
```

Opción "CANCELAR"

```
Private Sub C_cancel_Click()  
Text1.Text = ""  
Text2.Text = ""  
Form1.Show  
Form2.Hide  
End Sub
```

VENTANA DE CONFIGURACIÓN

```
Private Sub Form_Load()  
For i = 0 To 3  
    Check1(i).Value = 0  
Next i  
End Sub
```

Opción "ACEPTAR"

```
Private Sub Command1_Click()  
Dim element As Variant  
canal = Chr(0)  
For i = 0 To 7  
    If Check1(i).Value = 1 Then  
        canal = Chr(Asc(canal) + 2 ^ i)  
    End If  
Next i  
On Error GoTo rutinaerror  
If CInt(Text1.Text) < 1 Or CInt(Text1.Text) > 255 Then  
    MsgBox "INTRODÚZCA UN NÚMERO ENTRE 1 Y 255 POR FAVOR"  
Exit Sub  
Else  
    tmuestreo = Chr(CInt(Text1.Text))  
End If
```

```

If CInt(Text2.Text) < 1 Or CInt(Text2.Text) > 255 Then
    MsgBox "INTRODÚZCA UN NÚMERO ENTRE 5 Y 255 POR FAVOR"
Exit Sub
Else
    timepres = Chr(CInt(Text2.Text))
End If
If canal = Chr(0) Then
    MsgBox "DEBE SELECCIONAR AL MENOS UN CANAL"
Else
    Form1.Show
    F_config.Hide
End If
configura = 1 'Activa bandera para enviar mensaje de configuración
i = 1 'Obtiene el número total de líneas de la base de datos
Do While datos.cells(i, 1).Formula <> ""
    i = i + 1
Loop
maxrow = i - 1
For i = 1 To maxrow 'Obtiene la línea en la que se encuentra el dispositivo
    If Form1.combo_activos.Text = datos.cells(i, 1).Formula Then
        row = i
    End If
Next i
presencetime(row) = CInt(Text2.Text) 'guarda el tiempo de presencia del dispositivo elegido
Exit Sub
rutinaerror:
    If Err = 13 Then
        MsgBox "INTRODÚZCA UN NÚMERO ENTRE 1 Y 255 POR FAVOR"
    Exit Sub
    End If

```

End Sub

Código del envío del mensaje de configuración

```

Private Sub Form_Activate()
    If configura = 1 Then
        Winsock1.SendData "CO" & Now & tmuestreo & canal & timepres
        configura = 0
        canal = Chr(0)
    End If
End Sub

```

Opción "CANCELAR"

```

Private Sub Command2_Click()
    Form1.Show
    F_config.Hide
End Sub

```



```

Do While k < totalbytes(j)
  valor = Asc(Mid(remotedata(j), k, 1))
  valor = valor + (Asc(Mid(remotedata(j), k + 1, 1)) And 3) * 2 ^ 9
  Select Case Asc(Mid(remotedata(j), k + 1, 1)) And 48
    Case 0
      valor=valor * 10
      hexcel.cells(row, 1).Formula = valor
      If chanel And 14 = 0 Then
        row = row + 1
      End If
    Case 16
      valor=valor * 20
      hexcel.cells(row, 2).Formula = valor
      If chanel And 12 = 0 Then
        row = row + 1
      End If
  End Select
  k = k + 2
Loop
Next j
hexcel.Visible = True
For k = 0 To maxseg 'Limpia todas las localidades donde se guardaron los segmentos
  remotedata(k) = ""
Next k
lastseg = "0" 'Limpia la variable que indica si se trata del último segmento
If Asc(msgexpected) < 122 Then 'Incrementa en uno msgexpected
  perm1 = Asc(msgexpected) + 1
  msgexpected = Chr(perm1)
Else
  perm1 = 33
  msgexpected = Chr(perm1)
End If
End If
Next i
End If
End Sub

```

RUTINA DE MONITOREO DE VALORES ACTUALES

```

Private Sub Winsockarray_DataArrival(Index As Integer, ByVal bytesTotal As Long)
  Dim strdata, añadir, typemsg, msg, perm As String
  Dim i, j, totalbyte, perm1 As Integer
  Dim valor As Integer
  Winsockarray(Index).GetData strdata
  RaiseEvent IncomingData(Index, Data)
  typemsg = Mid(strdata, 1, 2)
  msg = Mid(strdata, 3, 1)
  If typemsg="VA" && msg = msgexpected Then
    presence(renglon - 2) = Now
    Form1.Hide
    Form3.Show
    Form3.Label2.Caption = "VALORES ACTUALES DEL " & combo_activos.Text
    Form3.Text1(0).Text = Text1(0)
    Form3.Text1(1).Text = Text1(1)
  End If
End Sub

```

```

totalbyte = WS1.BytesReceived - 3
i = 4
Do While i < totalbyte
    valor = Asc(Mid(strdata, i, 1))
    perm = Mid(strdata, i + 1, 1)
    valor = valor + (Asc(perm) And 3) * 2 ^ 9
    Select Case Asc(Mid(strdata, i + 1, 1)) And 48
        Case 0
            valor=valor*10
            Form3.Text1(2).Text = valor
        Case 16
            valor=valor*20
            Form3.Text1(3).Text = valor
    End Select
    i = i + 2
Loop
If Asc(msgexpected) < 122 Then
    perm1 = Asc(msgexpected) + 1
    msgexpected = Chr(perm1)
Else
    perm1 = 33
    msgexpected = Chr(perm1)
End If
EndIf

```

.....

RUTINA DEL TEMPORIZADOR 1

.....

```

Private Sub Timer1_Timer()
    Dim i, limite As Integer
    limite = combo_activos.ListCount - 1
    For i = 0 To limite
        If DateDiff("n", presence(combo_activos.ItemData(i)), Now) >
presencetime(combo_activos.ItemData(i)) Then
            remover(combo_activos.ItemData(i)) = "si"
        End If
    Next i
    i = 0
    While i <= limite
        If remover(combo_activos.ItemData(i)) = "si" Then
            remover(combo_activos.ItemData(i)) = "no"
            combo_activos.RemoveItem (i)
            limite = combo_activos.ListCount - 1
        Else
            i = i + 1
        End If
    Wend
End Sub

```

RUTINA DEL TEMPORIZADOR 2

```

Private Sub Timer2(index)_Timer()
    Dim perm1 As Integer
    If lapso < 10 Then
        lapso = lapso + 1
    Else
        F_espera.Hide
        Form1.Show
        Form1.Enabled = True
        MsgBox ("FALLA EN LA TRANSMISIÓN")
        Winsockarray(index).tag="free"
        Timer2.Enabled = False
        lapso = 0
        If Asc(msgexpected) < 122 Then 'Incrementa en uno msgexpected
            perm1 = Asc(msgexpected) + 1
            msgexpected = Chr(perm1)
        Else
            perm1 = 33
            msgexpected = Chr(perm1)
        End If
    End If
End Sub

```

RUTINA DE RECEPCIÓN DEL PUERTO WINSOCK2

```

Private Sub Winsock2_DataArrival(ByVal bytesTotal As Long)
    Dim strdata, añadir, name As String
    Dim IPdir As Variant
    Dim row As Integer
    row = 0
    WS2.GetData strdata
    IPdir = WS2.RemoteHostIP
    name = Mid(strdata, 5, 5)
    i = 1 'Obtiene el número total de líneas de la base de datos
    Do While datos.cells(i, 1).Formula <> ""
        i = i + 1
    Loop
    maxrow = i - 1
    For i = 1 To maxrow 'Obtiene la línea en la que se encuentra el dispositivo
        If name = datos.cells(i, 1).Formula Then
            row = i
        End If
    Next i
    If row <> 0 Then 'Si el dispositivo está en la base de datos
        datos.run ("nomclave") 'corre la macro de excel que quita la protección de escritura
        datos.cells(row, 4).Formula = IPdir 'actualiza la dir IP del dispositivo
        presence(row - 2) = Now 'registra el tiempo de este mensaje
        For i = 0 To combo_activos.ListCount - 1
            If combo_activos.List(i) = name Then
                añadir = "no"
            Else

```

```

        añadir = "si"
    End If
Next i
If añadir = "si" Then
    combo_activos.AddItem name
    combo_activos.ItemData(combo_activos.NewIndex) = row - 2
End If
End If
End Sub

```

.....

RUTINA DE RECEPCIÓN DEL PUERTO WINSOCK3

.....

```

Private Sub Winsock3_DataArrival(ByVal bytesTotal As Long)
    Dim strdata, añadir, name As String
    Dim IPdir As Variant
    Dim row As Integer
    Dim valor As Integer
    row = 0
    WS3.GetData strdata
    IPdir = WS3.RemoteHostIP
    name = Mid(strdata, 3, 5)
    i = 1 'Obtiene el número total de líneas de la base de datos
    Do While datos.cells(i, 1).Formula <> ""
        i = i + 1
    Loop
    maxrow = i - 1
    For i = 1 To maxrow 'Obtiene la línea en la que se encuentra el dispositivo
        If name = datos.cells(i, 1).Formula Then
            row = i
        End If
    Next i
    If row <> 0 Then 'Si el dispositivo está en la base de datos
        datos.run ("nomclave") 'activa macro de excel que quita la protección de escritura
        datos.cells(row, 4).Formula = IPdir 'actualiza la dir IP del dispositivo
        presence(row - 2) = Now 'registra el tiempo de este mensaje
        Form1.Hide
        Form4.Show
        Form4.Label1.Caption = "ALARMA DEL " & name
        Form4.Text1.Text = datos.cells(row, 2).Formula
        Form4.Text2.Text = datos.cells(row, 3).Formula
        valor = Asc(Mid(strdata, 8, 1))
        valor = valor + (Asc(Mid(strdata, 9, 1)) And 3) * 2 ^ 9
        Form4.Text3.Text = datos.cells(row, 2).Formula
        Select Case Asc(Mid(strdata, 9, 1)) And 48
            Case 0
                valor=valor * 10
                Form4.Label4 = "Temperatura ambiente"
                Form4.Text3.Text = valor
            Case 16
                Valor=valor * 20
                Form4.Label4 = "Nivel de combustible del generador"
                Form4.Text3.Text = valor
        End Select
    End If
End Sub

```

APÉNDICE C

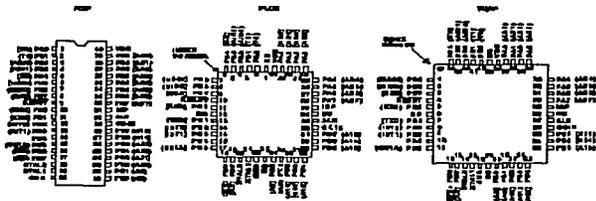
HOJAS DE ESPECIFICACIONES DE COMPONENTES UTILIZADOS

- Microcontrolador 8 bits, ATMEL AT90S8515 C-2
- Convertidor Analógico Digital, National ADC10158 C-6
- Memoria serial EEPROM, Microchip 24LC64 C-9
- Amplificador de instrumentación, Analog Devices AD620 C-11
- Display LCD, AND491 C-13
- Sensor de temperatura, National LM35 C-15

Features

- Utilizes the AVR[®] RISC Architecture
- AVR - High-performance and Low-power RISC Architecture
 - 118 Powerful Instructions - Most Single Clock Cycle Execution
 - 32 x 8 General Purpose Working Registers
 - Up to 8 MIPS Throughput at 8 MHz
- Data and Nonvolatile Program Memory
 - 4K/8K Bytes of In-System Programmable Flash
 - Endurance: 1,000 Write/Erase Cycles
 - 256/512 Bytes of SRAM
 - 256/512 Bytes of In-System Programmable EEPROM
 - Endurance: 100,000 Write/Erase Cycles
 - Programming Lock for Flash Program and EEPROM Data Security
- Peripheral Features
 - One 8-bit Timer/Counter with Separate Prescaler
 - One 16-bit Timer/Counter with Separate Prescaler
 - Compare, Capture Modes and Dual 8-, 9- or 10-bit PWM
 - On-chip Analog Comparator
 - Programmable Watchdog Timer with On-chip Oscillator
 - Programmable Serial UART
 - Master/Slave SPI Serial Interface
- Special Microcontroller Features
 - Low-power Idle and Power Down Modes
 - External and Internal Interrupt Sources
- Specifications
 - Low-power, High-speed CMOS Process Technology
 - Fully Static Operation
- Power Consumption at 4 MHz, 3V, 25°C
 - Active: 3.0 mA
 - Idle Mode: 1.0 mA
 - Power Down Mode: <1 µA
- I/O and Packages
 - 32 Programmable I/O Lines
 - 40-pin PDIP, 44-pin PLCC and TQFP
- Operating Voltages
 - 2.7 - 6.0V (AT90S4414-4 and AT90S8515-4)
 - 4.0 - 6.0V (AT90S4414-8 and AT90S8515-8)
- Speed Grades
 - 0 - 4 MHz (AT90S4414-4 and AT90S8515-4)
 - 0 - 8 MHz (AT90S4414-8 and AT90S8515-8)

Pin Configurations



**8-bit AVR[®]
Microcontroller
with 4K/8K
bytes In-System
Programmable
Flash**

**AT90S4414
AT90S8515**

Rev. 0841E-04/99



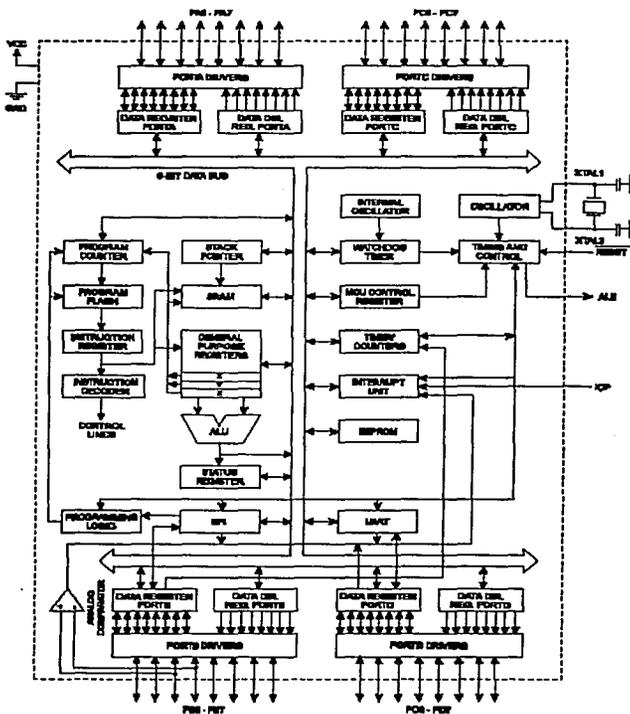


Description

The AT90S4414/AT90S8515 is a low-power CMOS 8-bit microcontroller based on the AVR RISC architecture. By executing powerful instructions in a single clock cycle, the AT90S4414/8515 achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed.

Block Diagram

Figure 1. The AT90S4414/8515 Block Diagram



TESIS CON
FALLA DE ORIGEN

AT90S4414/8515

The AVR core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers.

The AT90S4414/8515 provides the following features: 4K/8K bytes of In-System Programmable Flash, 256/512 bytes EEPROM, 256/512 bytes SRAM, 32 general purpose I/O lines, 32 general purpose working registers, flexible timer/counters with compare modes, internal and external interrupts, a programmable serial UART, programmable Watchdog Timer with internal oscillator, an SPI serial port and two software selectable power saving modes. The Idle Mode stops the CPU while allowing the SRAM, timer/counters, SPI port and interrupt system to continue functioning. The power down mode saves the register contents but freezes the oscillator, disabling all other chip functions until the next external interrupt or hardware reset.

The device is manufactured using Atmel's high density nonvolatile memory technology. The on-chip in-system programmable Flash allows the program memory to be reprogrammed in-system through an SPI serial interface or by a conventional nonvolatile memory programmer. By combining an enhanced RISC 8-bit CPU with In-System Programmable Flash on a monolithic chip, the Atmel AT90S4414/8515 is a powerful microcontroller that provides a highly flexible and cost effective solution to many embedded control applications.

The AT90S4414/8515 AVR is supported with a full suite of program and system development tools including: C compilers, macro assemblers, program debugger/simulators, in-circuit emulators, and evaluation kits.

Comparison Between AT90S4414 and AT90S8515

The AT90S4414 has 4K bytes of In-System Programmable Flash, 256 bytes of EEPROM and 256 bytes of Internal SRAM. The AT90S8515 has 8K bytes of In-System Programmable Flash, 512 bytes of EEPROM and 512 bytes of Internal SRAM. Table 1 summarizes the different memory sizes for the two devices.

Table 1. Memory Size Summary

Part	Flash	EEPROM	SRAM
AT90S4414	4K bytes	256 bytes	256 bytes
AT90S8515	8K bytes	512 bytes	512 bytes

Pin Descriptions

VCC

Supply voltage

GND

Ground

Port A (PA7..PA0)

Port A is an 8-bit bidirectional I/O port. Port pins can provide internal pull-up resistors (selected for each bit). The Port A output buffers can sink 20mA and can drive LED displays directly. When pins PA0 to PA7 are used as inputs and are externally pulled low, they will source current if the internal pull-up resistors are activated. The Port A pins are tri-stated when a reset condition becomes active, even if the clock is not active.

Port A serves as Multiplexed Address/Data input/output when using external SRAM.



**Port B (PB7..PB0)**

Port B is an 8-bit bidirectional I/O port with internal pull-up resistors. The Port B output buffers can sink 20 mA. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not active.

Port B also serves the functions of various special features of the AT90S4414/8515 as listed on page 58.

Port C (PC7..PC0)

Port C is an 8-bit bidirectional I/O port with internal pull-up resistors. The Port C output buffers can sink 20 mA. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not active.

Port C also serves as Address output when using external SRAM.

Port D (PD7..PD0)

Port D is an 8-bit bidirectional I/O port with internal pull-up resistors. The Port D output buffers can sink 20 mA. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not active.

Port D also serves the functions of various special features of the AT90S4414/8515 as listed on page 64.

RESET

Reset input. A low level on this pin for more than 50 ns will generate a reset, even if the clock is not running. Shorter pulses are not guaranteed to generate a reset.

XTAL1

Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

XTAL2

Output from the inverting oscillator amplifier

ICP

ICP is the input pin for the Timer/Counter1 Input Capture function.

OC1B

OC1B is the output pin for the Timer/Counter1 Output CompareB function

ALE

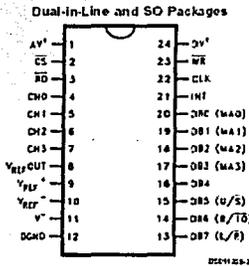
ALE is the Address Latch Enable used when the External Memory is enabled. The ALE strobe is used to latch the low-order address (8 bits) into an address latch during the first access cycle, and the AD0-7 pins are used for data during the second access cycle.

Crystal Oscillator

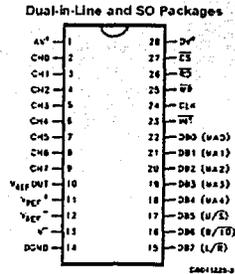
XTAL1 and XTAL2 are input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator, as shown in Figure 2. Either a quartz crystal or a ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven as shown in Figure 3.

ADC10154/ADC10158

Connection Diagrams



Top View
Order Number ADC10154
NS Package Number M24B



Top View
Order Number ADC10158
NS Package Numbers
M26B or N26B

Pin Descriptions

- AV+** This is the positive analog supply. This pin should be bypassed with a 0.1 μ F ceramic capacitor and a 10 μ F tantalum capacitor to the system analog ground.
- DV+** This is the positive digital supply. This supply pin also needs to be bypassed with 0.1 μ F ceramic and 10 μ F tantalum capacitors to the system digital ground. AV+ and DV+ should be bypassed separately and tied to same power supply.
- DGND** This is the digital ground. All logic levels are referred to this ground.
- V-** This is the negative analog supply. For unipolar operation this pin may be tied to the system analog ground or to a negative supply source. It should not go above DGND by more than 50 mV. When bipolar operation is required, the voltage on this pin will limit the analog input's negative voltage level. In bipolar operation this supply pin needs to be bypassed with 0.1 μ F ceramic and 10 μ F tantalum capacitors to the system analog ground.
- VREF+, VREF-** These are the positive and negative reference inputs. The voltage difference between VREF+ and VREF- will set the analog input voltage span.
- VREFOut** This is the internal band-gap voltage reference output. For bipolar operation of the voltage reference, this pin needs to be bypassed with a 330 μ F tantalum or electrolytic capacitor.
- CS** This is the chip select input. When a logic low is applied to this pin the WR and RD pins are enabled.

- RD** This is the read control input. When a logic low is applied to this pin the digital outputs are enabled and the INT output is reset high.
- WR** This is the write control input. The rising edge of the signal applied to this pin selects the multiplexer channel and initiates a conversion.
- INT** This is the interrupt output. A logic low at this output indicates the completion of a conversion.
- CLK** This is the clock input. The clock frequency directly controls the duration of the conversion time (for example, in the 10-bit bipolar mode $t_c = 22/t_{CLK}$ and the acquisition time ($t_A = 6/t_{CLK}$)).
- DB0(MA0) - DB7 (L/R)** These are the digital data inputs/outputs. DB0 is the least significant bit of the digital output word; DB7 is the most significant bit in the digital output word (see the Output Data Configuration table). MA0 through MA4 are the digital inputs for the multiplexer channel selection (see the Multiplexer Addressing tables), U/S (Unsigned/Signed), 8/TD, (8/10-bit resolution) and L/R (Left/Right justification) are the digital input bits that set the A/D's output word format and resolution (see the Output Data Configuration table). The conversion time is modified by the chosen resolution (see Electrical Characteristics table). The lower the resolution, the faster the conversion will be.
- CH0-CH7** These are the analog input multiplexer channels. They can be configured as single-ended inputs, differential input pairs, or pseudo-differential inputs (see the Multiplexer Addressing tables for the input polarity assignments).

ADC10154/ADC10158

Absolute Maximum Ratings (Notes 1, 3)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Positive Supply Voltage ($V^+ = AV^+ = DV^+$)	6.5V
Negative Supply Voltage (V^-)	-6.5V
Total Supply Voltage ($V^+ - V^-$)	13V
Total Reference Voltage ($V_{REF}^+ - V_{REF}^-$)	6.6V
Voltage at Inputs and Outputs	$V^- - 0.2V$ to $V^+ + 0.3V$
Input Current at Any Pin (Note 4)	±5 mA
Package Input Current (Note 4)	±20 mA
Package Dissipation at $T_A = 25^\circ\text{C}$ (Note 5)	500 mW
ESD Susceptibility (Note 6)	2000V
Soldering Information	
N Packages (10 Sec)	260 C
J Packages (10 Sec)	300 C
SO Package (Note 7):	
Vapor Phase (60 Sec)	215 C
Infrared (15 Sec)	220 C

Storage Temperature	-65°C to +150°C
Ceramic DIP Packages	-40°C to +150°C
Plastic DIP and SO Packages	

Operating Ratings (Notes 2, 3)

Temperature Range	$T_{MIN} \leq T_A \leq T_{MAX}$
ADC10154CIWM, ADC10158CIW, ADC10158CIWM	-40°C $\leq T_A \leq$ +85°C
Positive Supply Voltage ($V^+ = AV^+ = DV^+$)	4.5 V_{DC} to 5.5 V_{DC}
Unipolar Negative Supply Voltage (V^-)	DGND
Bipolar Negative Supply Voltage (V^-)	-4.5V to -5.5V
$V^+ - V^-$	11V
V_{REF}^+	$AV^+ + 0.05 V_{DC}$ to $V^- - 0.05 V_{DC}$
V_{REF}^-	$AV^+ + 0.05 V_{DC}$ to $V^- - 0.05 V_{DC}$
V_{REF} ($V_{REF}^+ - V_{REF}^-$)	0.5 V_{DC} to V^+

Electrical Characteristics

The following specifications apply for $V^+ = AV^+ = DV^+ = +5.0 V_{DC}$, $V_{REF}^+ = 5.000 V_{DC}$, $V_{REF}^- = GND$, $V^- = GND$ for unipolar operation or $V^+ = -5.0 V_{DC}$ for bipolar operation, and $f_{CLK} = 5.0$ MHz unless otherwise specified. Boldface limits apply for $T_A = T_J = T_{MIN}$ to T_{MAX} ; all other limits $T_A = T_J = 25^\circ\text{C}$. (Notes 8, 9, 12)

Symbol	Parameter	Conditions	Typical (Note 10)	CIN and CIWM Suffixes	Units (Limit)
				Limits (Note 11)	
UNIPOLAR CONVERTER AND MULTIPLEXER STATIC CHARACTERISTICS					
	Resolution			10 + Sign	Bits
	Unipolar Integral Linearity Error	$V_{REF}^+ = 2.5V$ $V_{REF}^+ = 5.0V$	±0.5	±1	LSB LSB (Max)
	Unipolar Full-Scale Error	$V_{REF}^+ = 2.5V$ $V_{REF}^+ = 5.0V$	±0.5	±1.5	LSB LSB (Max)
	Unipolar Offset Error	$V_{REF}^+ = 2.5V$ $V_{REF}^+ = 5.0V$	±1	±2	LSB LSB (Max)
	Unipolar Total Unadjusted Error (Note 13)	$V_{REF}^+ = 2.5V$ $V_{REF}^+ = 5.0V$	±1.5	±2.5	LSB LSB (Max)
	Unipolar Power Supply Sensitivity	$V^+ = +5V \pm 10\%$ $V_{REF}^+ = 4.5V$			
	Offset Error		±0.25	±1	LSB (Max)
	Full-Scale Error		±0.25	±1	LSB (Max)
	Integral Linearity Error		±0.25		LSB
BIPOLAR CONVERTER AND MULTIPLEXER STATIC CHARACTERISTICS					
	Resolution			10 + Sign	Bits
	Bipolar Integral Linearity Error	$V_{REF}^+ = 5.0V$		±1	LSB (Max)
	Bipolar Full-Scale Error	$V_{REF}^+ = 5.0V$		±1.25	LSB (Max)



24AA64/24LC64

64K I²C™ CMOS Serial EEPROM

DEVICE SELECTION TABLE

Part Number	Vcc Range	Max Clock Frequency	Temp Ranges
24AA64	1.8-5.5V	400 kHz ¹	C
24LC64	2.5-5.5V	400 kHz	I, E

¹ 100 kHz for Vcc < 2.5V

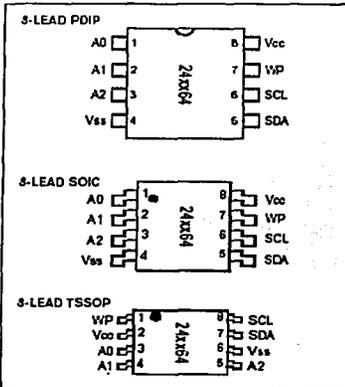
FEATURES

- Low power CMOS technology
 - Maximum write current 3 mA at 5.5V
 - Maximum read current 400 µA at 5.5V
 - Standby current 100 nA typical at 5.5V
- 2-wire serial interface bus, I²C compatible
- Cascadable for up to eight devices
- Self-timed ERASE/WRITE cycle
- 32-byte page or byte write modes available
- 5 ms max write cycle time
- Hardware write protect for entire array
- Output slope control to eliminate ground bounce
- Schmitt trigger inputs for noise suppression
- 1,000,000 erase/write cycles guaranteed
- Electrostatic discharge protection > 4000V
- Data retention > 200 years
- 8-pin PDIP, SOIC (150 and 208 mil) and TSSOP packages
- Temperature ranges:
 - Commercial (C): 0° to 70°C
 - Industrial (I): -40°C to +85°C
 - Automotive (E): -40°C to +125°C

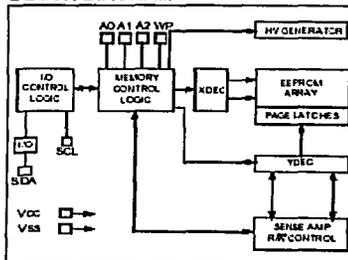
DESCRIPTION

The Microchip Technology Inc. 24AA64/24LC64 (24xx64¹) is a 8K x 8 (64K bit) Serial Electrically Erasable PROM capable of operation across a broad voltage range (1.8V to 5.5V). It has been developed for advanced, low power applications such as personal communications or data acquisition. This device also has a page-write capability of up to 32 bytes of data. This device is capable of both random and sequential reads up to the 64K boundary. Functional address lines allow up to eight devices on the same bus, for up to 512 Kbits address space. This device is available in the standard 8-pin plastic DIP, 8-pin SOIC (150 and 208 mil), and 8-pin TSSOP.

PACKAGE TYPE



BLOCK DIAGRAM



¹I²C is a trademark of Philips Corporation.

²24xx64 is used in this document as a generic part number for the 24AA64/24LC64 devices.

24AA64/24LC64

1.0 ELECTRICAL CHARACTERISTICS

1.1 Maximum Ratings*

V_{CC} 1.0V
 All inputs and outputs w.r.t. V_{SS} 0.6V to V_{CC} + 1.0V
 Storage temp range -55°C to +150°C
 Junction temp. with power applied -40°C to +125°C
 Soldering temperature of leads (10 seconds) +300°C
 ESD protection on all pins ±4 kV

*Notice: Stresses above those listed under "Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational listings of this specification is not implied. Exposure to maximum rating conditions for extended periods may affect device reliability.

TABLE 1-1 PIN FUNCTION TABLE

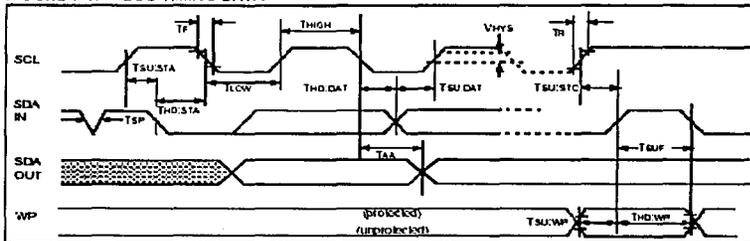
Name	Function
A0,A1,A2	User Configurable Chip Selects
V _{SS}	Ground
SDA	Serial Data
SCL	Serial Clbck
WP	Write Protect Input
V _{CC}	+1.8 to 5.5V (24AA64) +2.5 to 5.5V (24LC64)

TABLE 1-2 DC CHARACTERISTICS

Parameter	Symbol	Commercial (C): V _{CC} = +1.8V to 5.5V		Units	Conditions
		Min	Max		
All parameters apply across the recommended operating ranges unless otherwise noted.					
Commercial (C): V _{CC} = +1.8V to 5.5V Tamb = 0°C to +70°C Industrial (I): V _{CC} = +2.5V to 5.5V Tamb = -40°C to +85°C Automotive (E): V _{CC} = +2.5V to 5.5V Tamb = -40°C to 125°C					
A0, A1, A2, SCL, SDA, and WP pins: High level input voltage	V _{IH}	0.7 V _{CC}	—	V	V _{CC} ≥ 2.5V V _{CC} < 2.5V V _{CC} > 2.5V (Note)
Low level input voltage	V _{IL}	—	0.3 V _{CC} 0.2 V _{CC}	V	
Hysteresis of Schmitt Trigger inputs (SDA, SCL pins)	V _{IHS}	0.05 V _{CC}	—	V	
Low level output voltage	V _{OL}	—	0.40	V	I _{OL} = 3.0 mA @ V _{CC} = 4.5V I _{OL} = 2.1 mA @ V _{CC} = 2.5V
Input leakage current	I _I	-10	10	µA	V _{IN} = V _{SS} to V _{CC} , WP = V _{SS} V _{IN} = V _{SS} or V _{CC} , WP = V _{CC}
Output leakage current	I _{IO}	-10	10	µA	V _{OUT} = V _{SS} to V _{CC}
Pin capacitance (all inputs/outputs)	C _{IN} , C _{OUT}	—	10	pF	V _{CC} = 5.0V (Note) Tamb = 25°C, f _{TEST} = 1 MHz
Operating current	I _{CC Write}	—	3	mA	V _{CC} = 5.5V
	I _{CC Read}	—	400	µA	V _{CC} = 5.5V, SCL = 400 kHz
Standby current	I _{CCS}	—	1	µA	SCL = SDA = V _{CC} = 5.5V A0, A1, A2, WP = V _{SS}

Note: This parameter is periodically sampled and not 100% tested.

FIGURE 1-1: BUS TIMING DATA





Low Cost, Low Power Instrumentation Amplifier

AD620

FEATURES

EASY TO USE

Gain Set with One External Resistor
(Gain Range 1 to 1000)

Wide Power Supply Range (± 2.3 V to ± 18 V)
Higher Performance than Three Op Amp IA Designs
Available in 8-Lead DIP and SOIC Packaging
Low Power, 1.3 mA max Supply Current

EXCELLENT DC PERFORMANCE ("B GRADE")

50 μ V max. Input Offset Voltage
0.6 μ V/C max. Input Offset Drift
1.0 nA max. Input Bias Current
100 dB min Common-Mode Rejection Ratio (G = 10)

LOW NOISE

9 nV/ $\sqrt{\text{Hz}}$ @ 1 kHz. Input Voltage Noise
0.28 μ V p-p Noise (0.1 Hz to 10 Hz)

EXCELLENT AC SPECIFICATIONS

120 kHz Bandwidth (G = 100)
15 μ s Settling Time to 0.01%

APPLICATIONS

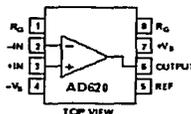
Weigh Scales
ECG and Medical Instrumentation
Transducer Interface
Data Acquisition Systems
Industrial Process Controls
Battery Powered and Portable Equipment

PRODUCT DESCRIPTION

The AD620 is a low cost, high accuracy instrumentation amplifier that requires only one external resistor to set gains of 1 to

CONNECTION DIAGRAM

8-Lead Plastic Mini-DIP (N), Cerdip (Q)
and SOIC (H) Packages



1000. Furthermore, the AD620 features 8-lead SOIC and DIP packaging that is smaller than discrete designs, and offers lower power (only 1.3 mA max supply current), making it a good fit for battery powered, portable (or remote) applications.

The AD620, with its high accuracy of 40 ppm maximum nonlinearity, low offset voltage of 50 μ V max and offset drift of 0.6 μ V/C max, is ideal for use in precision data acquisition systems, such as weigh scales and transducer interfaces. Furthermore, the low noise, low input bias current, and low power of the AD620 make it well suited for medical applications such as ECG and noninvasive blood pressure monitors.

The low input bias current of 1.0 nA max is made possible with the use of Superbeta processing in the input stage. The AD620 works well as a preamplifier due to its low input voltage noise of 9 nV/ $\sqrt{\text{Hz}}$ at 1 kHz, 0.28 μ V p-p in the 0.1 Hz to 10 Hz band, 0.1 pA/ $\sqrt{\text{Hz}}$ input current noise. Also, the AD620 is well suited for multiplexed applications with its settling time of 15 μ s to 0.01% and its cost is low enough to enable designs with one in-amp per channel.

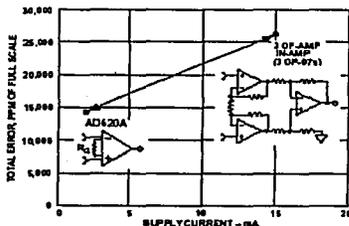


Figure 1. Three Op Amp IA Designs vs. AD620

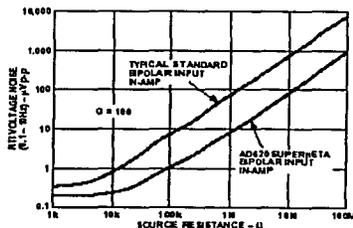


Figure 2. Total Voltage Noise vs. Source Resistance

REV. E

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices.

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781/329-4700 World Wide Web Site: <http://www.analog.com>
Fax: 781/326-8763 © Analog Devices, Inc., 1999



Intelligent Alphanumeric Application Notes

Intelligent Alphanumeric Displays

Interface Specifications

Specifications for AND Intelligent Alphanumeric displays are given in this section. The LCD controller used is the HD44780 or equivalent.

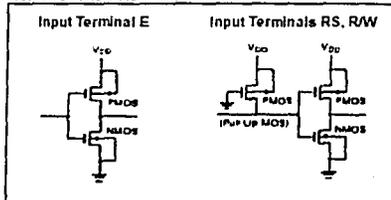
- AND671GST/GST-LED 16 characters x 1 line
- AND471GST/GST-LED 16 characters x 2 lines
- AND481GST/GST-LED 16 characters x 2 lines
- AND401GST/GST-LED 16 characters x 2 lines
- AND501GST/GST-LED 20 characters x 2 lines
- AND731GST/GST-LED 16 characters x 4 lines
- AND771GST/GST-LED 24 characters x 2 lines
- AND721GST/GST-LED 20 characters x 4 lines
- AND591GST/GST-LED 40 characters x 2 lines
- AND791GST/GST-LED 40 characters x 4 lines

Connector Pin Assignments (except for AND791GST)

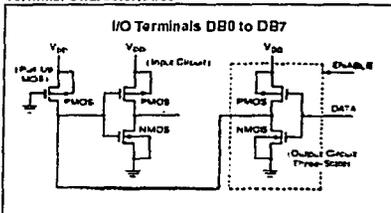
Pin No.	Signal	Function
1	GND	Ground
2	V _{DD}	+5 Power Supply
3	V _O	LCD Drive Voltage
4	RS	"H" Data Input "L" Command Input
5	R/W	Read/Write
6	E	Enable
7	DB0	Data Bus DB0-DB7 is for 8 bit operation
8	DB1	
9	DB2	
10	DB3	
11	DB4	
12	DB5	
13	DB6	
14	DB7	
15*	LED	LED Anode (for LED option)
16*	LED	LED Cathode (for LED option)

* See specific mechanical drawing for details.

Terminal Characteristics



Terminal Characteristics



When Input is at the intermediate level with CMOS, excessive current flows through the input circuit to the power supply. To avoid this, input level must be fixed at high or low.



Intelligent Alphanumeric Application Notes

Character Position and Character Address

For each device, the relationship between character position and character address is straightforward. A hexadecimal code for each character position in each device is given in the following charts. Character positions are numbered from left to right beginning in the top left corner as you view the device from the front.

AND671

Character Position:

1	2	3	4	5	6	7	8	9	10	11	12	13	14
---	---	---	---	---	---	---	---	---	----	----	----	----	----

Hex Address:

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D
----	----	----	----	----	----	----	----	----	----	----	----	----	----

AND491, 481, 471

Character Position:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----

Hex Address:

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Character Position:

17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

AND501

Character Position:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Character Position:

21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

AND731

Character Position:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----

Hex Address:

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Character Position:

17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Character Position:

17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Character Position:

33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

AND771

Character Position:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Character Position:

25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23	24	25	26	27
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

AND591

Character Position:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Character Position:

41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

AND721

Character Position:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Character Position:

21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Character Position:

41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F	30	31	32	33
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Character Position:

61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F	40	41	42	43
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

AND791

Character Position:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F	30	31	32	33	34	35	36	37	38	39	40
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Character Position:

41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Character Position:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F	20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F	30	31	32	33	34	35	36	37	38	39	40
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

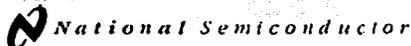
Character Position:

41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Hex Address:

30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	50	51	52	53	54	55	56	57	58	59	5A	5B	5C	5D	5E	5F	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Note: Address locations for Lines 1 & 2 are controlled by E1, and lines 3 & 4 are controlled by E2.



November 2000

LM35 Precision Centigrade Temperature Sensors

General Description

The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature. The LM35 thus has an advantage over linear temperature sensors calibrated in ° Kelvin, as the user is not required to subtract a large constant voltage from its output to obtain convenient Centigrade scaling. The LM35 does not require any external calibration or trimming to provide typical accuracies of $\pm 1/4^\circ\text{C}$ at room temperature and $\pm 1/2^\circ\text{C}$ over a full -55 to $+150^\circ\text{C}$ temperature range. Low cost is assured by trimming and calibration at the wafer level. The LM35's low output impedance, linear output, and precise inherent calibration make interfacing to readout or control circuitry especially easy. It can be used with single power supplies, or with plus and minus supplies. As it draws only 60 μA from its supply, it has very low self-heating, less than 0.1 C in still air. The LM35 is rated to operate over a -55 to $+150^\circ\text{C}$ temperature range, while the LM35C is rated for a -40 to $+110^\circ\text{C}$ range ($\pm 10^\circ$ with improved accuracy). The LM35 series is available pack-

aged in hermetic TO-46 transistor packages, while the LM35C, LM35CA, and LM35D are also available in the plastic TO-92 transistor package. The LM35D is also available in an 8-lead surface mount small outline package and a plastic TO-220 package.

Features

- Calibrated directly in ° Celsius (Centigrade)
- Linear ± 10.0 mV/°C scale factor
- 0.5°C accuracy guaranteeable (at $+25^\circ\text{C}$)
- Rated for full -55 to $+150^\circ\text{C}$ range
- Suitable for remote applications
- Low cost due to wafer-level trimming
- Operates from 4 to 30 volts
- Less than 60 μA current drain
- Low self-heating, 0.08 C in still air
- Nonlinearity only $\pm 1/4^\circ\text{C}$ typical
- Low impedance output, 0.1 Ω for 1 mA load

LM35 Precision Centigrade Temperature Sensors

Typical Applications

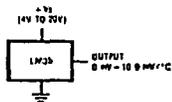
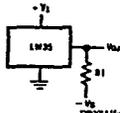


FIGURE 1. Basic Centigrade Temperature Sensor
($\pm 2^\circ\text{C}$ to $+150^\circ\text{C}$)



Choose $R_1 = -V_{OUT}/60 \mu\text{A}$
 $V_{OUT} = 1,000$ mV at $+150^\circ\text{C}$
 $= 1250$ mV at $+25^\circ\text{C}$
 $= -300$ mV at -55°C

FIGURE 2. Full-Range Centigrade Temperature Sensor

LM335

Absolute Maximum Ratings (Note 10)

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications.

Supply Voltage	+35V to -0.2V
Output Voltage	+6V to -1.0V
Output Current	10 mA
Storage Temp.:	
TO-46 Package.	-60°C to +180°C
TO-92 Package.	-60°C to +150°C
SO-8 Package.	-65°C to +150°C
TO-220 Package.	-65°C to +150°C
Lead Temp.:	
TO-46 Package.	300°C
(Soldering, 10 seconds)	

TO-92 and TO-220 Package. (Soldering, 10 seconds)	260°C
SO Package (Note 12)	
Vapor Phase (60 seconds)	215°C
Infrared (15 seconds)	220°C
ESD Susceptibility (Note 11)	2500V
Specified Operating Temperature Range: T_{MIN} to T_{MAX} (Note 2)	
LM135, LM35A	-55°C to +150°C
LM135C, LM35CA	-40°C to +110°C
LM35D	0°C to +100°C

Electrical Characteristics

(Notes 1, 6)

Parameter	Conditions	LM35A			LM35CA			Units (Max.)
		Typical	Tested Limit (Note 4)	Design Limit (Note 5)	Typical	Tested Limit (Note 4)	Design Limit (Note 5)	
Accuracy (Note 7)	$T_A = +25^\circ\text{C}$	± 0.2	± 0.5		± 0.2	± 0.5		$^\circ\text{C}$
	$T_A = -10^\circ\text{C}$	± 0.3			± 0.3		± 1.0	$^\circ\text{C}$
	$T_A = T_{MAX}$	± 0.4	± 1.0		± 0.4	± 1.0		$^\circ\text{C}$
	$T_A = T_{MIN}$	± 0.4	± 1.0		± 0.4	± 1.5		$^\circ\text{C}$
Nonlinearity (Note 8)	$T_{MIN} \leq T_A \leq T_{MAX}$	± 0.18		± 0.35	± 0.15		± 0.3	$^\circ\text{C}$
Sensor Gain (Average Slope)	$T_{MIN} \leq T_A \leq T_{MAX}$	+10.0	+9.9, +10.1		+10.0		+9.9, +10.1	mV/°C
Load Regulation (Note 3) $0.5I_L \leq 1\text{ mA}$	$T_A = +25^\circ\text{C}$	± 0.4	± 1.0		± 0.4	± 1.0		mV/mA
	$T_{MIN} \leq T_A \leq T_{MAX}$	± 0.5		± 3.0	± 0.5		± 3.0	mV/mA
Line Regulation (Note 3)	$T_A = +25^\circ\text{C}$	± 0.01	± 0.05		± 0.01	± 0.05		mVV
	$4V_S \leq V_S \leq 30V$	± 0.02		± 0.1	± 0.02		± 0.1	mVV
Quiescent Current (Note 9)	$V_S = +5V, +25^\circ\text{C}$	56	67		56	67		μA
	$V_S = +5V$	105		131	91		114	μA
	$V_S = +30V, +25^\circ\text{C}$	56.2	68		56.2	68		μA
	$V_S = +30V$	105.5		133	91.5		116	μA
Change of Quiescent Current (Note 3)	$4V_S \leq V_S \leq 30V, +25^\circ\text{C}$	0.2	1.0		0.2	1.0		μA
	$4V_S \leq V_S \leq 30V$	0.5		2.0	0.5		2.0	μA
Temperature Coefficient of Quiescent Current		+0.39		+0.5	+0.39		+0.5	$\mu\text{A}/^\circ\text{C}$
Minimum Temperature for Rated Accuracy	In circuit of Figure 1, $I_L = 0$	+1.5		+2.0	+1.5		+2.0	$^\circ\text{C}$
Long Term Stability	$T_A = T_{MAX}$, for 1000 hours	± 0.08			± 0.08			$^\circ\text{C}$