

**UNIVERSIDAD NACIONAL DE INGENIERIA**  
FACULTAD DE INGENIERIA ELECTRICA Y ELECTRONICA



COMUNICACION ENTRE PROCESADORES  
DE UNA CENTRAL TELEFONICA DIGITAL  
UTILIZANDO UN CANAL PCM DE PRIMER ORDEN  
CON CONTROLADORES HDLC.

**TESIS**

PARA OPTAR EL TITULO PROFESIONAL DE:

***INGENIERO ELECTRONICO***

**Edmundo Alexander Ingar Armijo**

*Promoción 1979 - 1*

**LMA - PERU - 1994**

## SUMARIO

La operación interna de la Central Telefónica Digital con procesamiento distribuido, está basada en la comunicación interna entre sus procesadores; este sistema debe cumplir con requerimientos funcionales tales como operación, performance, confiabilidad y otros.

La tesis propone el desarrollo de un sistema de comunicaciones entre los procesadores de la central, basado en el uso de controladores HDLC teniendo como medio de comunicación las vías de conversación, que son los canales PCM de primer orden. La idea es diseñar una red interna de comunicaciones empleando el protocolo de comunicación HDLC (High Level Data Link Control), para lo cual se deben definir las especificaciones de los requerimientos.

El diseño del sistema se basa en los criterios de interconexión de sistemas abiertos y eligiendo filosofías de diseño estructurado para el hardware y software.

Las pruebas obtenidas en los laboratorios establecen que es factible la implementación de un sistema final y ser parte integrante del sistema de la Central Digital.

COMUNICACION ENTRE PROCESADORES  
DE UNA CENTRAL TELEFONICA DIGITAL  
UTILIZANDO UN CANAL PCM DE PRIMER ORDEN  
CON CONTROLADORES HDLC.

## RESUMEN

TITULO: COMUNICACION ENTRE PROCESADORES DE UNA CENTRAL TELEFONICA DIGITAL UTILIZANDO UN CANAL PCM DE PRIMER ORDEN CON CONTROLADORES HDLC

AUTOR Edmundo Alexander Ingar Armijo

GRADO A OPTAR : Ingeniero Electrónico

PROGRAMA : Ingeniería Electrónica

UNIVERSIDAD NACIONAL DE INGENIERIA

LIMA, Diciembre de 1993

La actual tecnología de las Centrales Telefónicas Digitales tienen una arquitectura típica basada en procesamientos distribuidos, gracias al avance de la microelectrónica que han desarrollado circuitos integrados potentes, permitiendo así el desarrollo de las telecomunicaciones.

Estas Centrales trabajan en ambientes estandarizados por los Organismos Internacionales tales como el CCITT, UIT, ISO etc. traduciéndose en Planes Técnicos Fundamentales de Telecomunicaciones, que adoptan cada país, dependiendo de sus necesidades. Sin embargo no se ha estandarizado la comunicación entre los procesadores internos de la central, dejando en libertad a los fabricantes para diseñar sus propios sistemas, usualmente son comunicaciones paralelas, seriales vía RS-232 o las

vias de comunicación sin emplear específicamente controladores HDLC (High Level Data Link Controller).

La tesis propone un sistema de comunicación entre los procesadores de la central, empleando controladores HDLC especializados, analizando filosofías y criterios de diseño para el sistema.

En el desarrollo de la tesis, inicialmente consideramos necesario una revisión general de los Sistemas PCM, asociándolos a las recomendaciones del CCITT y aspectos de la Telefonía Digital, para centrarnos en el objetivo principal de la tesis, proponer el Sistema de Comunicaciones entre los Procesadores de la Central.

La ingeniería del proyecto consiste en diseñar el sistema eligiendo objetivos, tales como la filosofía del diseño, análisis de los sistemas distribuidos (multiprocesadores) y elegir la arquitectura mas conveniente.

Una vez definida la filosofía y la elección de los objetivos pormenorizados se debe proceder a la formulación del Proyecto, determinando modelos teóricos y criterios de evaluación de arquitecturas propuestas.

En la formulación del Proyecto se establecen modelos teóricos, basados en los niveles de la Interconexión de Sistemas Abiertos (ISA) y adoptado por el CCITT en la Señalización Nro. 7, para la señalización entre centrales.

En cuanto al diseño del sistema se debe considerar los

aspectos del diseño Hardware y Software. En la etapa del diseño del sistema, se bosquejan todos los bloques funcionales y sus especificaciones unitarias. Para el diseño Hardware se hará un estudio minucioso de las facilidades técnicas que brindan los fabricantes de circuitos integrados especializados en telecomunicaciones ( específicamente, comunicación entre procesadores ) y que están disponibles en el mercado, a fin de elegir el mas apropiado y que cumplan con nuestra especificaciones. Se han evaluado tres controladores HDLC, previamente calificados: MT8952B (Mitel Semiconductor), serie HPCxx400 (National Semiconductor) y 8273 (Intel), resultando elegido el controlador de protocolo HDLC MT8952B, que se adapta a nuestra aplicación, Centrales Telefónicas Digitales de aplicaciones rurales (bajo tráfico).

Referente al diseño software, también se adopta la misma filosofía del diseño estructurado, proponiendo algoritmos básicos y técnicas de programación.

Las tarjetas prototipo se probarán en el Laboratorio comunicándose dos procesadores (iguales y de diferentes características) a fin de determinar la factibilidad de su aplicación en Sistemas de Conmutación PCM.

Finalmente hacemos un análisis de confiabilidad de la arquitectura a elegir analizando su compromiso con los costos del sistema.

Durante el desarrollo de la tesis se propone una

metodología del diseño de sistemas, que involucra desde la definición de las especificaciones del sistema hasta las pruebas de laboratorio y de campo, con la finalidad de obtener un producto técnicamente factible con posibilidades de industrializarlo.

## INDICE

	Pag.
PROLOGO	1
C A P I T U L O I	
DESCRIPCION DE LOS SISTEMAS PCM Y ASPECTOS DE TELEFONIA DIGITAL	3
1.1 Principio de los sistemas PCM	3
1.2 Sistemas de transmisión PCM	24
1.3 Sistemas de conmutación digital	30
C A P I T U L O II	
FILOSOFIAS DE DISEÑO Y SISTEMAS DISTRIBUIDOS	44
2.1 Filosofía del diseño estructurado.	44
2.2 Sistemas distribuidos - Multiprocesadores.	51
CAPITULO III	
FORMULACION DEL PROYECTO	65
3.1 Determinación de un modelo teórico.	65
3.2 Criterios de evaluación para la elección de un sistema.	79
CAPITULO IV	
DISEÑO DEL PROTOTIPO	101
4.1 Diseño del sistema	101

	Pag.
4.2 Diseño hardware	125
4.3 Diseño software	163
CAPITULO V	
PRUEBAS DE LABORATORIO	173
5.1 Descripción de las pruebas aisladas de hardware	174
5.2 Resultados de las pruebas	178
CAPITULO VI	
ANALISIS DE CONFIABILIDAD Y COSTOS	184
6.1 Análisis de confiabilidad	184
6.2 Análisis de costos	192
CONCLUSIONES Y RECOMENDACIONES	196
ANEXOS	199
BIBLIOGRAFIA	211

## PROLOGO

El desarrollo tecnológico en el Perú es incipiente, por su contexto en el ámbito tercer mundista, sin embargo existen esfuerzos aislados, en el desarrollo de equipos de telecomunicaciones y otras especialidades, demostrando que los profesionales peruanos tenemos la capacidad de desarrollar, equipos medianamente complejos.

Esto es posible gracias al desarrollo de la microelectrónica a nivel mundial, que nos facilitan circuitos integrados altamente especializados, compactos y disponibles en el mercado; requisitos indispensables para el desarrollo de las telecomunicaciones.

Uno de estos grandes esfuerzos ha efectuado ENTEL PERU S.A. en el desarrollo de una Central Telefónica Digital de aplicaciones rurales, Proyecto ANTARA, para instalar en los pueblos rurales, a fin de integrarlos y lograr su desarrollo.

El proyecto inicialmente se concibió en tres etapas, para desarrollar prototipos a ser transferidos a la industria privada para su fabricación en serie.

Actualmente, el proyecto ha desarrollado la segunda etapa, prototipo ANTARA B, que es un producto mejorado respecto a su antecesor ANTARA A, que estuvo operando en

la ciudad de Cañete, desde Junio 1990 y actualmente ha sido reemplazado por ANTARA B.

Una de las mejoras del nuevo prototipo es justamente la comunicación entre sus procesadores utilizando controladores HDLC, que fue un pequeño aporte, del autor de la presente tesis, como una solución a los problemas del primer prototipo ANTARA A. La idea fue complementada y desarrollada por los ingenieros del Proyecto y es parte integrante del sistema ANTARA B.

A Diciembre de 1993 se han instalado dos centrales ANTARA B en el Valle Sagrado de los Incas, con 4 unidades remotas sirviendo a las ciudades de: Urubamba, Calca, Pisac, Anta, Ollantaytambo y Aguas Calientes (Macchu Picchu), con aproximadamente 450 líneas y otra central en la ciudad de Cañete con una capacidad de 1024 líneas.

Mi agradecimiento especial a los técnicos e ingenieros que laboran en el proyecto y me brindaron su apoyo en el desarrollo del proyecto y en el presente trabajo ya sea en forma directa o indirecta; asimismo mi agradecimiento a nuestros jefes que confiaron en nuestra capacidad.

Tenemos el pleno conocimiento que el trabajo no es perfecto, pero sus problemas son superables y que con dedicación y apoyo podemos lograr un equipo altamente confiable, que podría industrializarse y sentar las bases de nuestro desarrollo tecnológico nacional.

Asimismo, mi agradecimiento al asesor de la presente tesis, Ing. Luís Sáez Cuyubamba, por su valiosa ayuda.

C A P I T U L O    I  
DESCRIPCION DE LOS SISTEMAS PCM Y ASPECTOS  
DE TELEFONIA DIGITAL

Considero necesario realizar un resumen de los principios básicos del sistema PCM y sobre todo asociándolo a las Recomendaciones del Comité Consultivo Internacional de Telefonía y Telegrafía CCITT, organismo internacional que recomienda los estándares en materia de telecomunicaciones. Así mismo una revisión general de los aspectos mas importantes de la Telefonía Digital para ubicarnos adecuadamente en el problema, que la presente tesis pretende resolver óptimamente.

**1.1 Principio de los sistemas PCM**

A fines de la década de los años 1930 en Francia se realizaron investigaciones de los métodos de modulación, para mejorar los sistemas de transmisión, cuyos problemas fundamentales eran el ruido y la distorsión de los sistemas análogos.

Uno de los mejores trabajos fue el propuesto por el grupo de investigadores liderados por ALEC REEVES que en 1937 presentó la Modulación por Impulsos Codificados (MIC) o comúnmente llamados SISTEMAS PCM ( Pulse Code

Modulation). La patente francesa se registró en 1938, tecnológicamente el invento aún no era factible llevarlo a la práctica, por las limitaciones técnicas de esa época, recién al inicio de la década de 1960 se emplea este sistema en las redes telefónicas y actualmente está en un permanente desarrollo con los últimos adelantos en electrónica y sistemas aplicados a las telecomunicaciones.

El CCITT ha recomendado dos diferentes sistemas de transmisión con PCM de primer orden uno sugerido por la CEPT y otro por la AT&T, los cuales se describirán mas adelante.

El sistema PCM puede describirse como un método para transferir la información analógica a la forma digital, la conversión de la señal analógica en una digital se basa en tres principios fundamentales:

- El Muestreo
- La Cuantificación
- La Codificación

#### **1.1.1 Muestreo**

Consiste en tomar muestras discretas de una señal que es continua en el tiempo, limitados en un ancho de banda (BW). Esto se hace a intervalos regulares de tiempo. Una muestra ideal en el instante  $t = \tau$  de una señal  $s(t)$  es el valor instantáneo  $s(\tau)$ .

El muestreo ideal no es físicamente realizable; en la práctica, una muestra es una medida del valor instantáneo

de una señal, evaluada durante un lapso de tiempo muy corto comparado con el intervalo entre muestras sucesivas; se sobreentiende que durante ese corto lapso de tiempo, la señal solo **varía** en una cantidad despreciable.

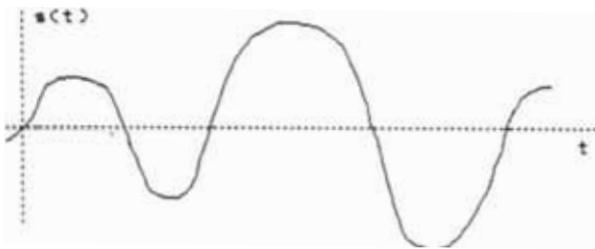
En los sistemas PCM, la señal real obtenida por muestreo consiste en un tren o secuencia de impulsos, cada uno de los cuales tiene una amplitud igual al valor de la señal en el instante del muestreo (Fig. 1.1). El muestreo debe efectuarse en forma periódica, con período  $T$ .

La regla básica que debe respetar el muestreo es el **TEOREMA DEL MUESTREO**, que se puede enunciar así:

"Si una señal contiene únicamente frecuencias inferiores a  $f_m$ , quedará completamente determinada por muestras (es decir valores instantáneos) tomadas a una frecuencia mayor o igual al doble de la frecuencia máxima de la señal original  $f_m$  ( $f_s > 2f_m$ )".

Según éste teorema, es posible reconstruir la señal original sin distorsión, interpolando entre los valores de las muestras. En la práctica, la reconstitución se obtiene haciendo pasar la señal muestreada por un filtro pasabajo que elimina todas las frecuencias iguales o superiores a  $f_m$ .

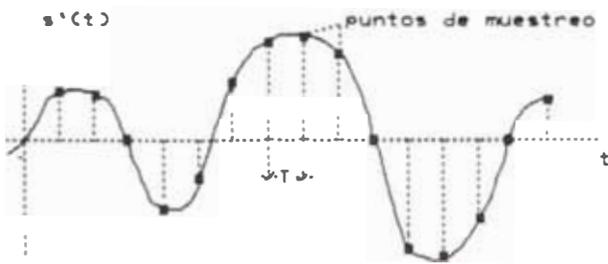
La Fig. 1.1-d muestra el espectro de una señal obtenida por muestreo. En forma muy similar a la traslación de frecuencias de los sistemas MDF ( Multiplex



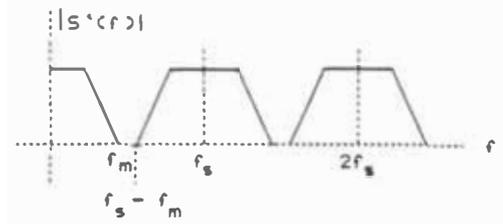
a) SENAL ANALOGA



b) EXPECTRO DE AMPLITUD DE LA SENAL ANALOGA



c) SEÑAL MUESTREADA



d) EXPECTRO DE AMPLITUD DE LA SENAL MUESTREADA

FIG 1.1  
TEOREMA DEL MUESTREO

de Frecuencia) el muestreo produce bandas laterales de la frecuencia de muestreo  $f_s$ ; si la frecuencia máxima comprendida en la banda de la señal original es  $f_m$ , la primera frecuencia del espectro transpuesto es  $f_s - f_m$ .

Para impedir que la banda de la señal original (0 Hz.- $f_m$ ) se superponga a la banda lateral inferior  $f_s - f_m$  (lo que produciría distorsión excesiva), entonces  $f_s - f_m$  debe ser mayor que  $f_m$ , siendo esta la condición del teorema del muestreo ( $f_m < f_s/2$ ).

El CCITT en la Recomendación G.711 ha elegido la frecuencia de muestreo  $f_s = 8000$  Hz. para el tratamiento de señales telefónicas con frecuencias vocales (BW = 300 a 3400 Hz.). Esto corresponde a un período de muestreo de  $T = 125 \mu s$ .

### 1.1.2 Cuantificación

Una muestra existe durante un instante discreto únicamente, pero el valor de su amplitud se obtiene a partir de una gama continua de valores. En otras palabras, es posible un número infinito de valores de muestra. Sin embargo se puede utilizar un número finito adecuado de valores discretos para representar en forma aproximada los valores de las muestras. Para obtener una medida discreta de una muestra se puede comparar ésta con una escala compuesta de un número finito de intervalos y especificar el valor de la amplitud de la muestra indicando el intervalo a que corresponde. Este proceso se

llama **CUANTIFICACION** y cada intervalo de la escala se llama **INTERVALO DE CUANTIFICACION**.

El problema que se plantea ahora es el del número de intervalos necesarios. Naturalmente cuando mayor sea el número de intervalos (dentro de una gama determinada) mejor y con mayor exactitud puede expresarse una muestra con su intervalo correspondiente. Sin embargo por razones prácticas, el número de intervalos no debe exceder una cifra razonable. Desde que este sistema está orientado a la digitalización, podemos expresar cada intervalo de cuantificación como una expresión binaria. Para la cuantificación de las señales de frecuencias vocales, el CCITT ha recomendado (Recomendación G.711) una gama de trabajo de 256 intervalos de cuantificación. Puesto que 256 es potencia de 2 ( $256 = 2^8$ ), cada número específico de 1 a 256, que identifica un intervalo específico, puede expresarse mediante una secuencia de 8 dígitos binarios (bits). Esta secuencia se transmite mediante un código de impulsos y se reconstruye un valor discreto cuando hay que volver a formar la señal original.

El proceso de cuantificación introduce necesariamente un error en los valores instantáneos de la señal. Si se supone una cuantificación uniforme, para hacer mínimo éste error, se iguala el valor reconstituido al valor central del intervalo de cuantificación. Dentro de una gama de funcionamiento, cada intervalo de cuantificación está limitado por dos valores de decisión. Los valores de

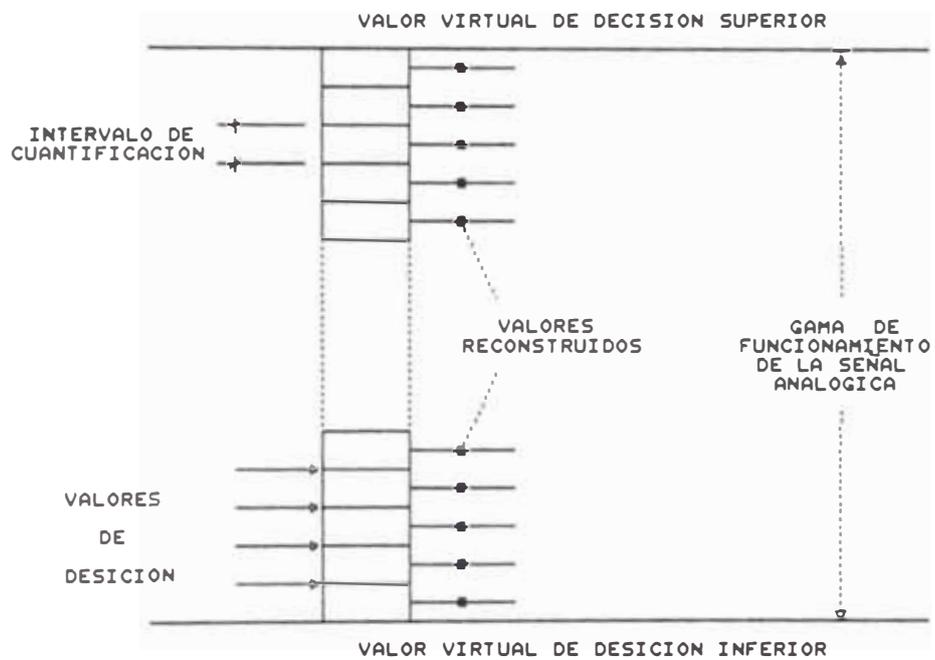


FIG 1.2  
CUANTIFICACION

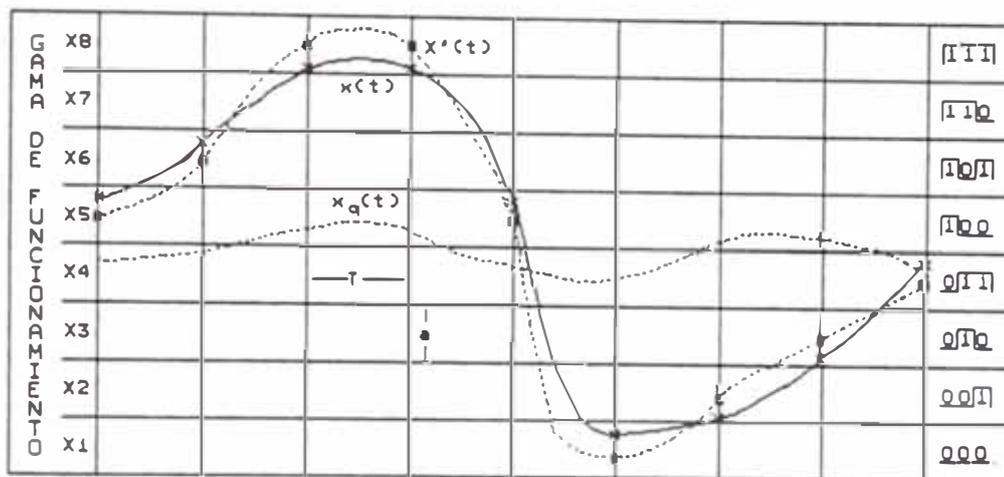
decisión situados a los extremos de la gama de funcionamiento son los valores virtuales de decisión; éstos no representan los valores de umbral del circuito de codificación, sino que indican los límites de la gama de valores de señal que pueden transmitirse sin los recortes de cresta.

El efecto de la cuantificación se muestra en la Fig. 1.3, donde el número de intervalos de cuantificación se ha limitado a 8 para mayor claridad. En el gráfico se representa la señal original continua  $x(t)$ , los valores de muestra reconstituidos (líneas punteadas), la señal de frecuencias vocales reconstituida  $x'(t)$ , obtenida por interpolación entre las muestras reconstituidas, y la diferencia  $x_q(t) = x'(t) - x(t)$ , que representa el efecto de la distorsión de cuantificación. Como los errores de cuantificación de las muestras sucesivas no están correlacionados, el efecto de la distorsión de cuantificación es similar al de un ruido aleatorio superpuesto, y no a una distorsión armónica.

Se ha supuesto hasta aquí que el tamaño de todos los intervalos de cuantificación es el mismo. En este caso, se demuestra que la potencia de la distorsión de cuantificación viene dada por

$$P_q = a^2/12 (w)$$

De esto se infiere que, cuando más pequeño es el valor del intervalo de cuantificación (en otras palabras, para una gama de funcionamiento determinada, cuando mayor es



$x(t)$  = Señal Original  
 $x'(t)$  = Señal Reconstituida  
 $x_q(t)$  = Distorsión de Cuantificación

$a$  = Intervalo de Cuantificación  
 $T$  = Periodo de Muestreo

FIG 1.3  
 CUANTIFICACION Y CODIFICACION

el número de intervalos de cuantificación), menor es la distorsión.

Las pruebas subjetivas realizadas revelan que ha de haber una relación mínima determinada entre la potencia de la señal  $P_s$  y la potencia de la distorsión de cuantificación  $P_q$ , donde  $10 \log(P_s/P_q) = S/N_q$  es la relación señal/ruido de cuantificación, de lo que se trata es lograr que esta distorsión sea despreciable para el oído humano. Esta relación mínima es de 24 dB. En consecuencia, la relación  $S/N_q$  debe mantenerse casi constante para la gama de niveles de señal que existe normalmente en telefonía. Esta gama puede estimarse en unos 30 dB.

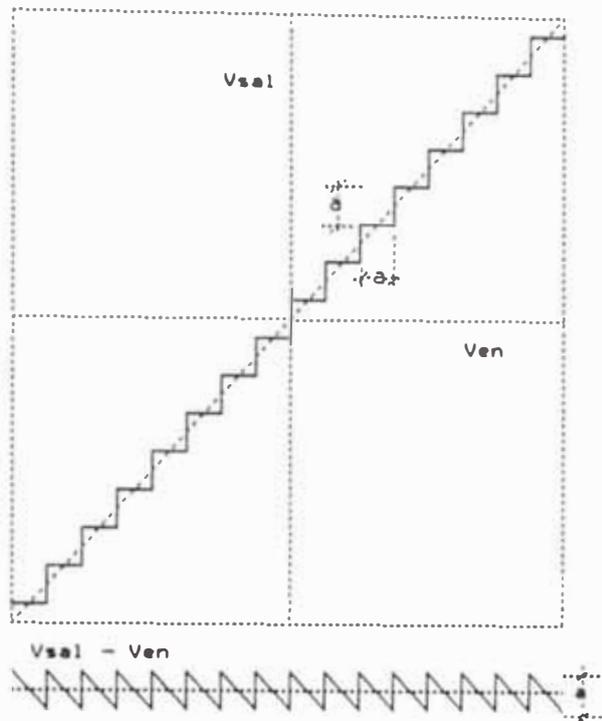
Con una cuantificación uniforme, esto es donde todos los intervalos de cuantificación tienen el mismo tamaño, no será posible obtener una relación  $S/N_q$  constante. Por esta razón, se utiliza la cuantificación no uniforme: el tamaño del intervalo,  $\Delta$ , se hace menor para valores pequeños de la señal que para los valores grandes de ésta. En principio cuando se parte del valor máximo de la señal, en que el tamaño del intervalo de cuantificación es  $\Delta_{max}$ , habría que reducir este tamaño a  $1/32$  al pasar también al  $1/32$  del valor máximo de la señal (es decir inferior en 30.1 dB al nivel máximo en el caso de las señales sinusoidales).

La Fig. 1.4 muestra la relación entre los valores analógicos a la entrada del cuantificador y los valores

reconstituidos a la salida del cuantificador, tanto para la cuantificación uniforme y no uniforme. La tensión de salida,  $v_{sal}$ , solo cambia cuando la tensión de entrada  $V_{en}$  pasa de un intervalo de cuantificación al intervalo adyacente. La diferencia  $V_{sal} - V_{en}$  es el error de cuantificación. Puede verse que asignando intervalos de cuantificación menores a los valores pequeños de  $V_{en}$ , se reduce proporcionalmente el error de cuantificación.

La asignación de los valores de la tensión de entrada  $V_{en}$  a los intervalos de cuantificación suele representarse gráficamente mediante la característica de cuantificación, llamada también ley de codificación. Las dos características especificadas por el CCITT Recomendación G.711 se ilustran en la Fig. 1.5, que representa la mitad (en este caso mitad positiva) de la característica total (la otra mitad es simétrica con respecto a cero). Ambas características están constituidas por segmentos de recta, en lugar de ser uniformes o continuas. Esto facilita sumamente la realización práctica de los circuitos de cuantificación y decodificación.

Estas dos características se denominan, respectivamente LEY A o de 13 segmentos y LEY  $\mu$  o de 15 segmentos. La primera se emplea en los equipos multiplex primarios PCM de 30 canales conforme a la Recomendación G.732 y la segunda, en los equipos multiplex primarios PCM conforme a la Recomendación G.733. La Fig. 1.5



CARACTERISTICAS DE LA CUANTIFICACION UNIFORME

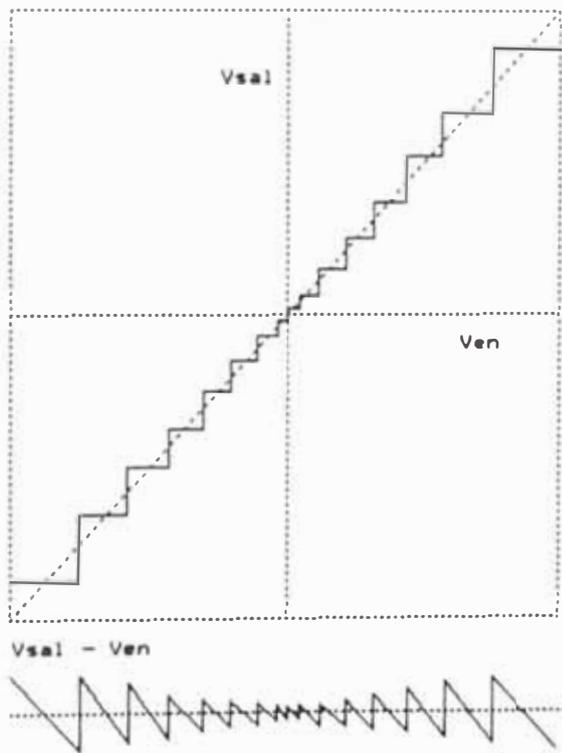


FIG. 1.4  
CARACTERISTICAS DE LA CUANTIFICACION NO UNIFORME

muestra las características con 7 y 8 segmentos, pero como los dos segmentos de las mitades positiva y negativa de cada característica que coinciden en cero tienen la misma pendiente, se pueden considerar como un solo segmento. Así pues, el número total de segmentos es solo de 13 y 15.

La Fig. 1.6 muestra los valores teóricos de  $S/N_q$ , relación señal a ruido de cuantificación, debido a la cuantificación de las dos leyes. Se indican también los límites de tolerancia especificados en la recomendación G.712, que permiten tener un cierto margen para tener en cuenta las imperfecciones de los codificadores y decodificadores prácticos, el envejecimiento, etc. Se observa que en la gama esencial de niveles de señal, la relación  $S/N_q$  es aproximadamente 33 dB superior para la cuantificación de 256 intervalos; que habida cuenta del valor apenas perceptible de unos 24 dB precedentemente mencionado, se ve que es posible conectar en tándem 10 circuitos de codificación y decodificación (códecs) sin que se reduzca indebidamente la calidad de la transmisión de la voz.

Para completar estas explicaciones, cabe señalar que el procedimiento de cuantificación no uniforme se denomina también **compresión y expansión**. Estos términos reflejan el hecho de que el procedimiento de cuantificación no uniforme puede realizarse, por ejemplo, sometiendo primero la señal analógica a una compresión

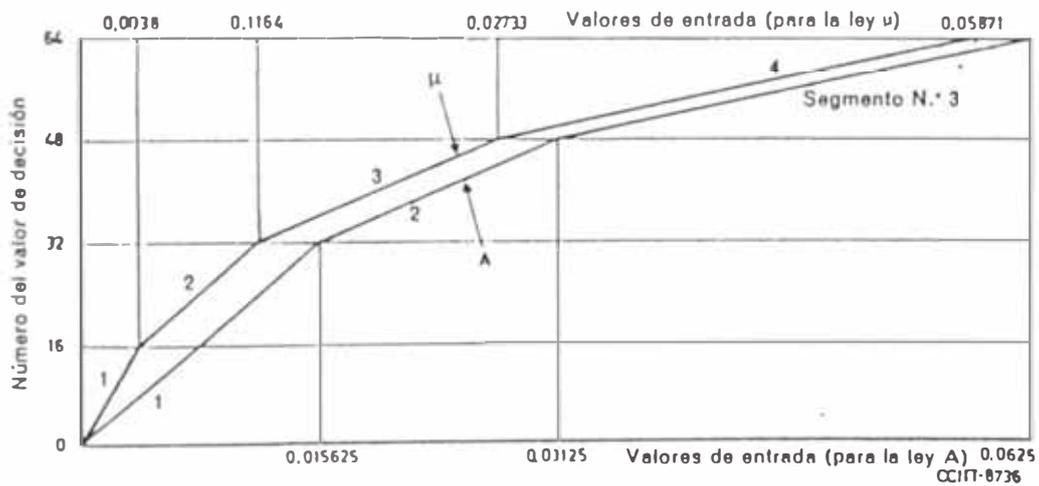
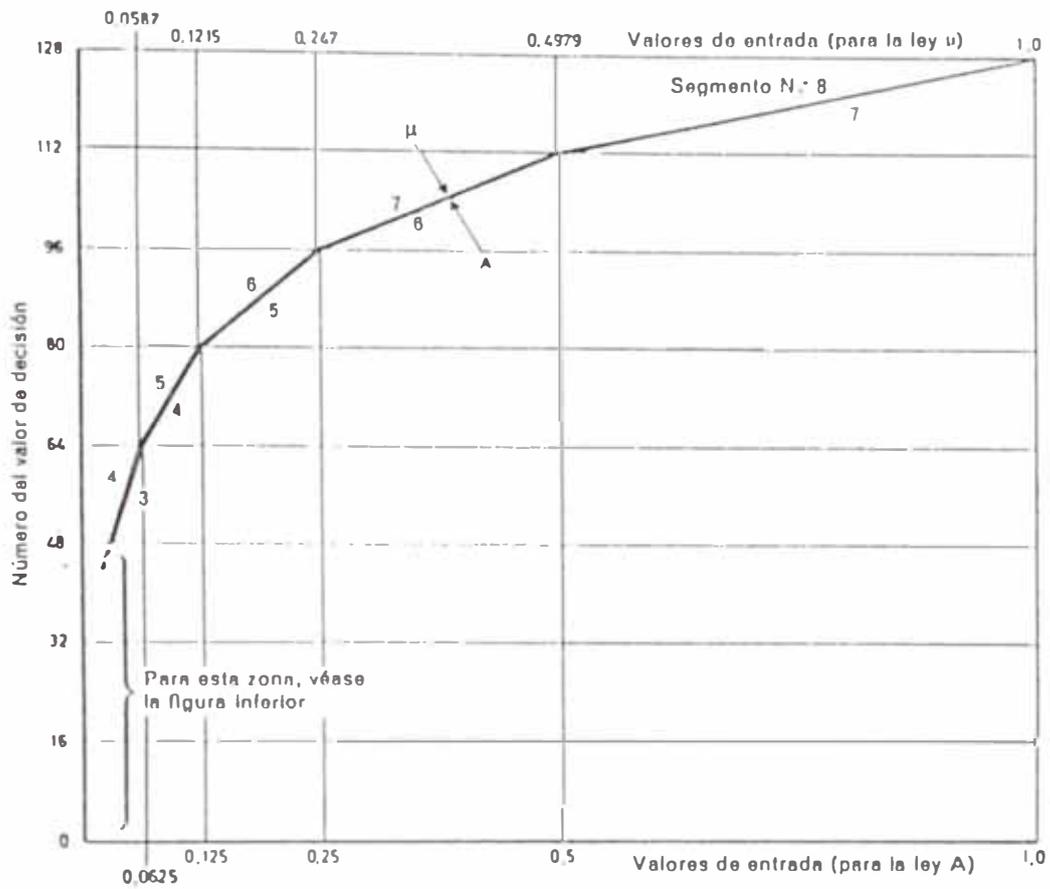
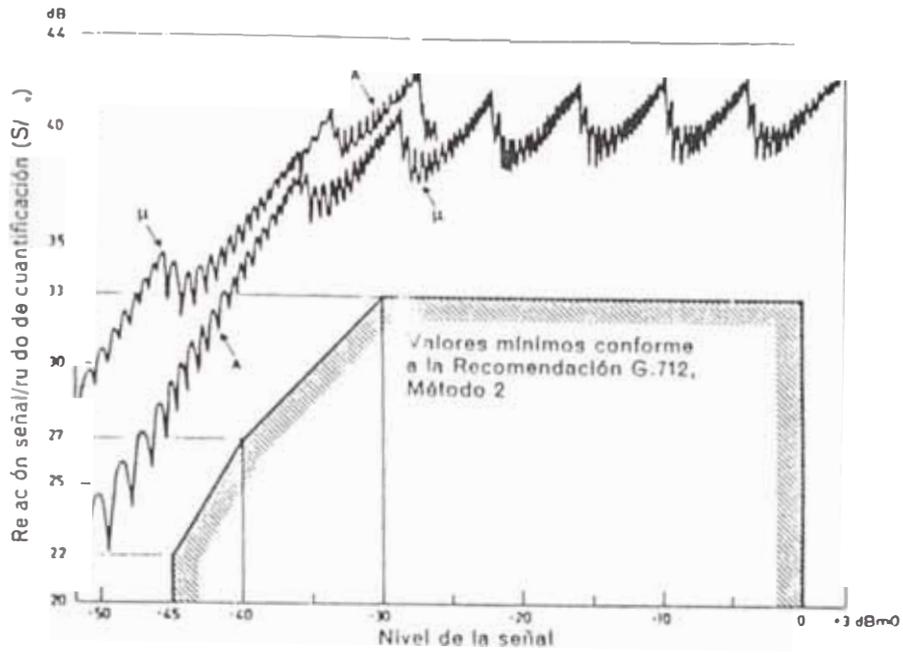


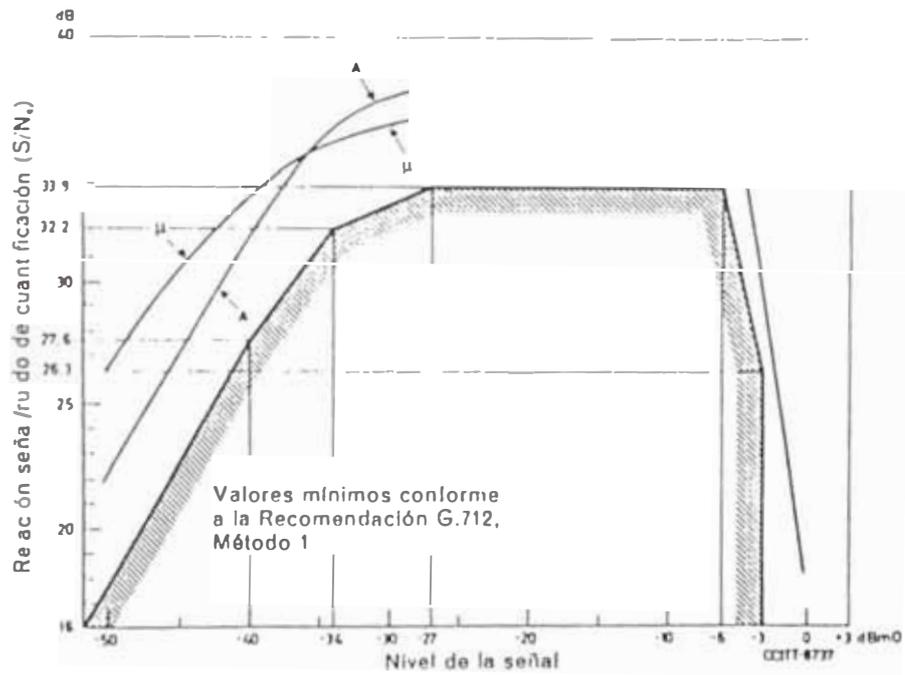
FIG 1.5

CARACTERISTICAS DE CUANTIFICACION

CCIT-8736



a) señal sinusoidal (ponderación psicofonética)



b) señal gaussiana o casi gaussiana de banda estrecha (no ponderada)

FIG 1.6

RELACION SEÑAL/RUIDO DE CUANTIFICACION

instantánea (reduciendo los valores elevados de señal en su proporción con respecto a los pequeños) y luego a una cuantificación uniforme; cuando la señal analógica se reconstituye, tiene lugar a una reconstitución uniforme y a continuación una expansión instantánea. El proceso de compresión y expansión puede efectuarse también en distinto orden, es decir, sometiendo la señal analógica a compresión uniforme y seguidamente a cuantificación no lineal. En la práctica se utilizan varios métodos.

La conversión entre las dos leyes de cuantificación (A y  $\mu$ ), necesaria en algunos casos (conexiones internacionales), es posible y da lugar a una relación S/Nq que es a lo sumo de 3 dB inferior a la de decodificación directa, según la ley A. Esta degradación no debe ser subjetivamente significativa.

En PCM, el nivel de sobrecarga tiene considerable importancia, analizando la Fig. 1.2, se define como el nivel (en dBm0) de una onda sinusoidal cuyos valores de cresta coinciden con los niveles virtuales de decisión. El valor del nivel de sobrecarga debe elegirse de modo que sólo se produzca raramente un corte de cresta para niveles vocales excepcionalmente elevados y jamás en el caso de los niveles normales de la voz o de transmisión de datos en la banda de frecuencias vocales, etc. (afortunadamente, la transmisión de la voz no es tan sensible a un corte moderado de crestas). Considerando los niveles que tienen las señales de frecuencias vocales

en la red telefónica real, el CCITT ha recomendado un nivel de sobrecarga de +3.14 dBm0 para la ley A y de 3.17 dBm0 para la ley  $\mu$ . Por razones prácticas, relativas al ajuste del codificador/decodificador, no es simplemente +3 dBm0 (mayor detalle referirse a la Recomendación G.711)

### 1.1.3 Codificación

Las muestras cuantificadas aún no son apropiadas para su tratamiento, porque sería muy difícil la construcción de circuitos regeneradores capaces de distinguir entre la gran gama de amplitudes de las muestras (256 intervalos). Sin embargo, hay gran flexibilidad en la codificación de estas amplitudes en formas eléctricas adecuadas para su tratamiento. En general la muestra cuantificada puede codificarse con cualquier sistema de numeración, pudiendo elegirse dos o más impulsos con menores niveles de amplitud por impulso; si partimos de: "un grupo de  $n$  impulsos, cada uno con  $b$  niveles de amplitud discreta posibles, puede representar  $b^n$  niveles de muestras cuantificadas", tendremos las siguientes alternativas:

$b^n$	$n$	$b$
256	1	256
256	2	16
256	4	4
256	8	2

TABLA 1.1

Como sabemos , los impulsos con dos niveles, es decir

los impulsos binarios son aptos para las aplicaciones eléctricas porque son fáciles de regenerar y tratamiento. Analizando la Tabla 1.1, se elige la última alternativa con lo cual un grupo de 8 bits constituyen una palabra PCM.

En la recomendación G.711 del CCITT se especifican códigos binarios simétricos (o replegados). En los mismos, el primer dígito binario indica el signo de la muestra (1, para las muestras positivas y 0 para las negativas) y los demás dígitos indican el número de intervalos de cuantificación a partir del centro de la gama de funcionamiento hacia adelante. Esta relación se muestra en la siguiente Tabla 1.2.

NUMERO DECIMAL DEL INTERVALO DE CUANTIFICACION	BIT DE SIGNO B <sub>7</sub>	PALABRA PCM BIT NUMERO							
		B <sub>8</sub>	B <sub>5</sub>	B <sub>4</sub>	B <sub>3</sub>	B <sub>2</sub>	B <sub>1</sub>	B <sub>0</sub>	
+128	1	1	1	1	1	1	1	1	
+127	1	1	1	1	1	1	1	0	
+126	1	1	1	1	1	1	0	0	
.	.	.	.	.	.	.	.	.	
+3	1	0	0	0	0	0	1	0	
+2	1	0	0	0	0	0	0	1	
+1	1	0	0	0	0	0	0	0	
-1	0	0	0	0	0	0	0	0	
-2	0	0	0	0	0	0	0	1	
-3	0	0	0	0	0	0	1	0	
.	.	.	.	.	.	.	.	.	
-126	0	1	1	1	1	1	0	0	
-127	0	1	1	1	1	1	1	0	
-128	0	1	1	1	1	1	1	1	

**TABLA 1.2**

Los fabricantes de los circuitos integrados generalmente clasifican sus componentes en digitales y

analógicos, sin embargo existe frecuentemente una combinación de ambos tipos. Por ejemplo la unión de señales analógica y digitales desarrolla el transductor activo análogo/digital comúnmente llamado CODEC. El cual recoge las recomendaciones del CCITT e inclusive mejora su rendimiento en cuanto a la inmunidad al ruido, tal es el caso del CODEC de Mitel Semiconductor, que como ejemplo mostramos su performance en la Fig. 1.7.

#### 1.1.4 Transmisión.

Las señales digitales dentro del terminal, usualmente se transmiten en la forma de tren de pulsos unipolares en el modo sin retorno a cero (NRZ). Esta forma de señal no es apropiada para la transmisión a largas distancias. Una forma mejor es una señal bipolar con retorno a cero (RZ), cuyas ventajas son:

No tiene potencia en las partes inferiores de su espectro, es decir, no tiene componentes de corriente continua; debido a la polaridad alternada de sus pulsos.

La interferencia entre símbolos está reducida por la característica de retorno a cero.

Por supuesto, también esta señal será atenuada y distorsionada durante la transmisión, y se le agregará ruido a la misma.

En algún punto de la línea de transmisión, la señal debe ser restaurada. Esto se efectúa introduciendo en la

línea un dispositivo que primero examina el tren de impulsos distorsionados para determinar los niveles lógicos y la ley de formación, para luego regenerarlos y volverlos a transmitir. A tales dispositivos se les denomina REPETIDOR REGENERATIVO.

A la vez que se le regenera la forma de los pulsos, se elimina el ruido agregado durante la transmisión, siempre y cuando la amplitud del ruido no sea alta y permita la regeneración sin errores. Normalmente, la señal de código regenerada es idéntica a la señal original transmitida. Aun después de una gran cantidad de repetidores regenerativos, la señal de código es prácticamente idéntica a la señal original. Esta es la razón de la alta calidad de la transmisión PCM.

#### 1.1.5 Demodulación.

Los procesos del receptor que convierten la señal entrante PCM a una señal analógica, nuevamente son: regenerados, decodificados y reconstruidas.

El proceso de regeneración tiene el mismo objetivo y se efectúa de la misma forma que en la línea de transmisión, es decir, los pulsos distorsionados son reemplazados por los regenerados, si la señal de entrada es bipolar, ésta se reconvierte en unipolar; luego entra al proceso de decodificación, los octetos de código se convierten en impulsos de amplitud, cuyas amplitudes son iguales a las de la muestra cuantificada.

La señal analógica es reconstruida en un filtro pasa

bajo con una frecuencia de corte apropiada que elimina todas las componentes superiores del espectro de frecuencia, quedando el espectro de la señal analógica deseada.

## **1.2 Sistemas de transmisión PCM.**

En las secciones precedentes se han explicado los principios fundamentales de la PCM. Nuestro siguiente propósito es utilizando estos fundamentos, describir la implementación de sistemas prácticos de transmisión y conmutación.

Ante todo, debemos conocer otro principio, que es el MULTIPLEX POR DIVISION DE TIEMPO, que complementado a lo anterior se implementan sistemas económicamente atractivos para la telefonía y otros servicios afines.

### **1.2.1 Multiplex por división de tiempo (TDM)**

Varias señales en forma de pulso pueden compartir un solo medio de transmisión común, si las señales tienen diferentes fases. En efecto si entrelazamos octetos codificados PCM, formaremos lo que en adelante denominaremos TRAMA (HIGH WAY) y el intervalo de tiempo que ocupa cada octeto se denomina INTERVALO DE TIEMPO (TIME SLOT) o CANAL PCM.

La frecuencia de repetición de la trama es igual a la del muestreo (8 KHz). En ésta trama aparte de las señales de voz codificadas es necesario insertar señales para la alineación (sincronización) de las tramas y para información de señalización (señalización por canal

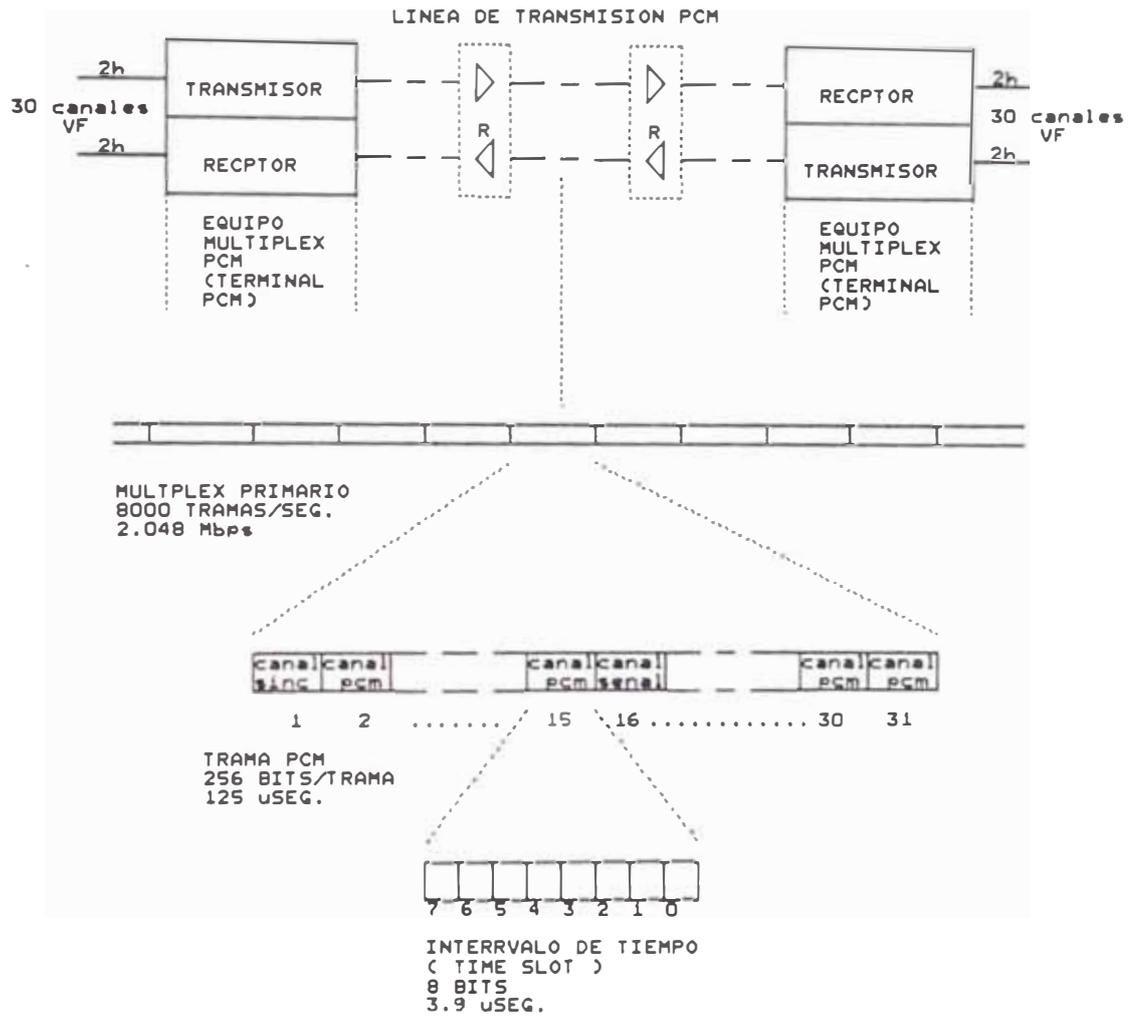


FIG 1.8

SISTEMA PCM DE PRIMER ORDEN Y LA ESTRUCTURA DE LA TRAMA

común o canal asociado). La Fig. 1.8, nos muestra la estructura de la trama multiplex de primer orden, Recomendación G.732.

### 1.2.2 Jerarquía de los sistemas PCM.

Para una mejor aplicación de éstos sistemas, es necesario establecer una jerarquía, que básicamente influirían los siguientes factores:

- La capacidad del medio de transmisión.
- La velocidad binaria de los terminales PCM.
- La eficacia y conveniencia del multiplaje.
- La necesidad conmutación por división de tiempo.

En la Fig. 1.9 se muestra la posible jerarquía basados en los sistemas PCM de primer orden.

### 1.2.3 Sistemas PCM de primer orden.

Como se mencionaron en las secciones anteriores, el CCITT recomienda dos sistemas PCM primarios, para usar en telefonía y están especificados bajo las Recomendación G.732 y G.733.

Es necesario distinguir entre un terminal PCM, un equipo multiplex PCM, y líneas de transmisión PCM. El terminal PCM es el que origina las señales digitales en un intervalo de tiempo a partir de la señal analógica correspondiente a un usuario, el equipo multiplex convierte una cantidad de señales analógicas (30 ó 24) en un tren de señales digitales en el lado de transmisión y efectúa las funciones inversas en el lado de la

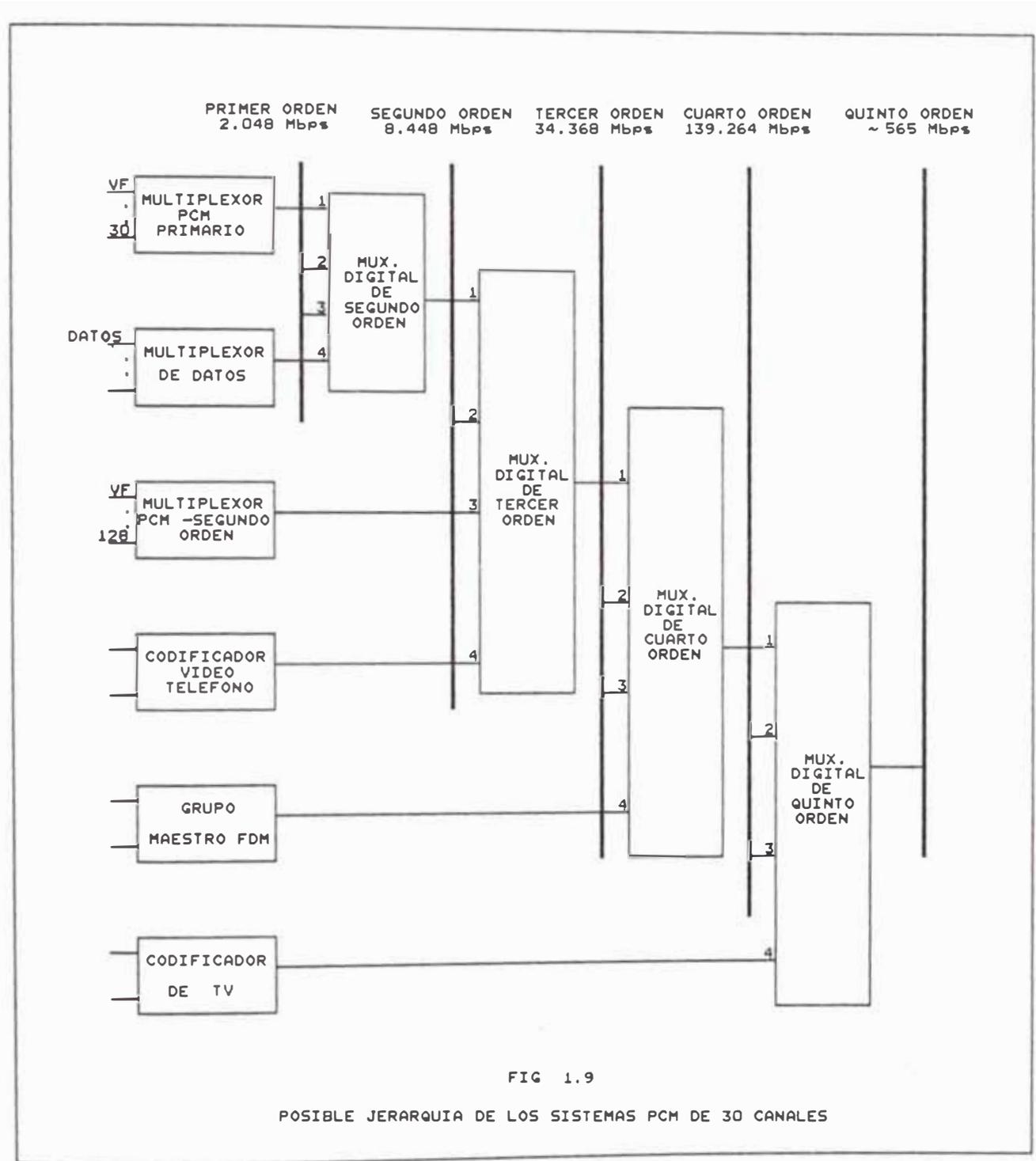


FIG 1.9

POSIBLE JERARQUIA DE LOS SISTEMAS PCM DE 30 CANALES

recepción, la línea de transmisión transporta las señales digitales entre los dos equipos multiplex.

Detallaremos a continuación el sistema de 30 canales (Recomendación G.732), que será el medio de transporte de la comunicación entre los procesadores de la central telefónica digital.

Los 30 canales de conversación analógica, junto con la señalización asociada, son convertidos en una señal digital por medio del equipo multiplex TDM-PCM. Esta señal digital se divide en tramas, con una velocidad de repetición de 8000 tramas/s. , esto es debido a la frecuencia de muestreo (8000 Hz.). Cada trama consiste en 32 intervalos de tiempo de 8 bits cada uno. De éstos, 30 se utilizan para canales PCM y los dos restantes para la sincronización y señalización.

Los canales PCM, transportan señales analógicas, dentro de la banda de frecuencias de 300 - 3400 Hz., codificadas de acuerdo a la ley A. El primer intervalo de tiempo de la trama corresponde al sincronismo, el propósito de estos 8 bits es formar una señal de reconocimiento para el receptor, a fin de mantener a éste sincronizado con el transmisor, de modo que cada canal PCM pueda ser correctamente identificado.

El intervalo de tiempo 16 es el de señalización, que tiene múltiples aplicaciones debido a la gran capacidad de información 64 Kbit/s, ofreciendo gran flexibilidad de aplicación. El CCITT ha recomendado el uso de este

intervalo en señalización por canal común, en este caso se utiliza un canal de señalización común a varios canales, para transmitir mensajes correspondientes a cualquiera de estos, según las necesidades. Esta es la señalización que se emplea en los sistemas de conmutación con control por programa almacenado (SPC) (señalización Nro. 7) y la señalización por canal asociado.

La aplicación de la señalización por canal asociado, actualmente nos permite introducir sistemas primarios PCM en las redes existentes. El esquema usa los intervalos de tiempo 16, en secuencias de 16 tramas, llamadas MULTITRAMAS. En la primera trama de la secuencia, el intervalo de tiempo 16 de la trama 0 transporta una palabra de MULTITRAMA, una señal que al receptor le indicará que se inicia una multitrama. El intervalo de tiempo 16 de la siguiente trama (trama 1) está dividido de modo que los primeros 4 bits llevan información de señalización asociada con el canal PCM 1 y los otros 4 bits información del canal PCM 17. En la trama 2, el intervalo de tiempo 16 lleva información para los canales 2 y 18, así sucesivamente hasta la trama 15.

De ésta manera 4 bits de señalización están asociados con cada canal PCM. Cada bit puede usarse para controlar algún dispositivo de control correspondiente a ese canal.

La TABLA 1.3 nos presenta los datos técnicos para los equipos multiplex TDM - PCM

Características	Sistema PCM Rec. G.732	Sistema PCM Rec. G.733
- Ancho de Banda (Hz.)	300 - 3400	300 - 3400
- Frecuencia de muestreo	8000 Hz.	8000 Hz.
Bits/muestra	8	8
- Intervalos de tiempo por trama	32	24
Canales PCM/trama	30	24
Velocidad de Tx Kbits/s	2048	1544
Ley de Codificación	A	$\mu$
Capacidad de señalización		
Canal Asociado	1-4 /canal	1-2 /canal
Canal Común	64 Kbits/s	4 Kbits/s

TABLA 1.3

### 1.3 Sistemas de conmutación digital

Dentro del concepto de conmutación de la telefonía clásica, la función de los equipos de conmutación telefónica, es la de recibir una señal analógica desde un usuario (abonado telefónico) que origina la comunicación y tiene por finalidad contactar con otro usuario (destino), durante este proceso hay intercambio de señales entre la central telefónica y los usuarios involucrados en la comunicación. Tradicionalmente los equipos de conmutación conectaban las señales de voz, las que eran transformadas en señales eléctricas analógicas por el aparato telefónico, conservando esta forma original eran conmutados a través de contactos (relés) estableciéndose un camino galvánico, ésta técnica se denomina CONMUTACION ANALOGA.

Con los últimos adelantos tecnológicos en los sistemas

de telecomunicaciones y la microelectrónica ha echo posible la implementación de equipos de CONMUTACION DIGITAL, los que tratan las señales en forma digital.

Aplicando los fundamentos y conceptos de los sistemas PCM, describiremos la técnica de la conmutación digital. La señal de voz enviada por el usuario llamante son digitalizadas mediante los procesos antes descritos como son el muestreo, la cuantificación y la codificación, finalmente el resultado es un dato de 8 bits.

Estas señales son entrelazadas con otras señales digitalizadas correspondientes a otros usuarios, lográndose formar una trama de 30 intervalos de tiempos (canales PCM) mas uno de sincronización y otro de señalización, tramas de primer orden Recomendación G.732 del CCITT. A cada usuario le corresponde un canal PCM que tiene un intervalo de tiempo, por lo que es fácilmente identificable. Estas tramas son procesadas por etapas de conmutación temporales o espaciales, basados en dispositivos semiconductores LSI o VLSI (memorias, interruptores o dispositivos especializados), en éste tipo de conmutación no existe un camino galvánico entre el usuario llamante y llamado.

Una vez efectuada la conmutación, las tramas salientes de las mismas características, son demultiplexadas para cada usuario, luego decodificadas regenerándose aproximadamente en su forma original, enviándose a su respectivo destino. Para garantizar la bidireccionalidad

de la comunicación es necesario establecer otra ruta de retorno.

### 1.3.1 Ventajas de la conmutación digital

Adicional a las ventajas expuestas en los fundamentos del sistema PCM, podemos mencionar las siguientes ventajas en la aplicación de éstos, en los sistemas de conmutación:

Ahorro en las redes de comunicación.- Los siguientes puntos nos muestran los méritos de los equipos de conmutación digital:

Los equipos de conmutación digital están disponibles para conectarse con los sistemas de transmisión digital, sin necesidad de la conversión análoga/digital, la función de sincronización en la central terminal es de fácil implementación.

Se cumple las expectativas del procesamiento del tráfico con la gran capacidad que se puede obtener con la combinación de etapas de conmutación.

Facilita la conmutación de grupos de circuitos que pueden usarse mas eficientemente.

Mejoramiento de la calidad de transmisión.- Una conversión análoga/digital produce ruido de cuantización. En las redes de conmutación digital la conversión A/D es eliminada a excepción de las

entradas del usuario a la red. Pero una vez digitalizada la señal es tratada en esta forma y es regenerada digitalmente cuando es necesaria, de manera tal que no influye en la señal análoga.

Miniaturización, Ahorro y Alta Rentabilidad en Equipamiento.- Las etapas de conmutación son diferentes a los sistemas análogos convencionales de contactos mecánicos, en esta nueva técnica se emplean circuitos lógicos basados en compuertas y memorias de estado sólido, así como componentes especializados empleados en procesadores que controlan las centrales (centrales SPC), consiguiéndose así una gran reducción de espacio y costos de componentes y mantenimiento del equipo.

Funcionalidad en el servicio de las comunicaciones.- La voz es transmitida por códigos PCM a 64 kbits/s. Esta capacidad es bastante alta que permite establecer una línea de datos a 64 kbits/s, en las que actualmente se están desarrollando nuevas formas de información para futuros servicios, tales como datos e imágenes y en el futuro el ISDN.

### 1.3.2 Estructura básica de un equipo de conmutación digital.

La Fig. 1.10 muestra el modelo funcional de una Central Telefónica Digital, a partir de la cual se desarrollan sistemas prácticos lográndose una estructura básica de un equipo de conmutación digital, tal como se

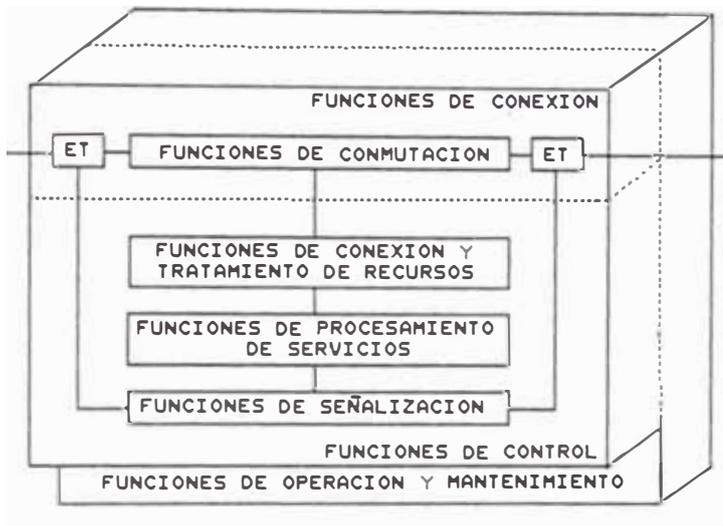


FIG 1.10  
 MODELO FUNCIONAL DE UNA CENTRAL DIGITAL  
 RECOMENDACION Q.521 DEL CCITT

muestra en la Fig. 1.11, que es el diagrama de bloques del sistema ANTARA y describimos brevemente las funciones de cada módulo:

Módulos de Abonados.- Las señales analógicas de los usuarios son convertidas a señales digitales, funciones del SLIC y éstas señales son concentradas y conmutadas en el modulo regional, para ser interconectadas a la Matriz Principal de la Central. El circuito SLIC provee la corriente hacia el equipo del usuario, protección contra sobre voltaje, supervisión, conversión A/D, la conversión de 2 a 4 hilos, pruebas de abonados y transmisión de la señal.

Módulos de Troncales.- La función básico de este módulo es la interconexión con otras centrales o unidades remotas de la central. Con el hardware y software apropiado puede operar troncales analógicas, digitales e interfases digitales.

Módulo de Señalizadores.- Este módulo esta asociado directamente al de troncales, procesando las señales de multifrecuencia MFC-R2, que integran el protocolo de interconexión de Centrales pudiendo ser analógicas o digitales.

Módulo de Procesador Central.- El control de las funciones del equipo de conmutación, tales como el procesamiento de las llamadas, operación y mantenimiento y detección de fallas es controlado

por el procesador central. En el presente sistema tiene redundancia para una mejor confiabilidad.

Módulo de Conmutación Central.- Constituida por la matriz digital del tipo cross point, con la capacidad de conmutar todos los canales principales (high ways) de los módulos periféricos: abonados, troncales, señalizadores y otros.

Módulo de Sincronización.- Este módulo es la fuente de la sincronización de la red digital que involucra los sistemas de transmisión y conmutación, se basa en un reloj maestro con una alta estabilidad.

Como se puede apreciar el sistema ANTARA tiene una configuración jerárquica-modular y con procesamiento distribuido basado en procesadores centrales y regionales definiendo plenamente sus funciones.

### **1.3.3 Procesos orientados a la conmutación digital.**

Por lo general en los sistemas de conmutación el factor tiempo está restringido. Por ejemplo los pulsos de discado o la señal DTMF del abonado deberán ser monitoreados en intervalos de tiempo regulares y razonables, así como el establecimiento de la comunicación. La central siempre debe estar alerta para responder en forma rápida la demanda de los usuarios que producen eventos en forma aleatoria y ocurren asincrónicamente en el tiempo por lo que el sistema

telefónico debe responder oportunamente y dentro de un plazo razonable y esto obliga a trabajar con Procesos en Tiempo Real y en Sistemas Distribuidos por los múltiples procesadores que controlan la Central Telefónica.

Se entiende por procesamiento en tiempo real aquellos procesos que ejecutan y controlan tareas de un procesador en tiempos determinados, para lo cual se requiere un Sistema Operativo adecuado basado en interrupciones, para fijar los tiempos en el que deben realizarse los procesos (Sistemas Operativos de Tiempo Real).

El ciclo básico de éstos sistemas es un intervalo de tiempo fijado por interrupciones de reloj del procesador. La determinación del ciclo básico depende de los requerimientos de señalización y del intercambio entre la lógica del hardware y software.

Una de las aplicaciones típicas del proceso a tiempo real, son las centrales telefónicas, donde el factor tiempo está restringido ante los eventos producidos por los usuarios en forma directa o indirecta, los cuales se presentan al azar. El procesador encargado del módulo de abonados detecta, explorando periódicamente los puntos apropiados de la central, los eventos (cambios de estados) y determina el orden de procesamiento (tareas) necesarios para cada demanda, de acuerdo a las prioridades establecidas. Las cuales se respetan para su ejecución, de tal manera que durante el ciclo básico, primero se ejecuten las tareas mas importantes,

pudiéndose ejecutar todas, en caso contrario se suspende la tarea en proceso y espera el próximo ciclo respetando las prioridades.

En la central se producirán gran cantidad de eventos las cuales necesitan ser procesadas, existiendo un compromiso entre la velocidad de exploración y la velocidad de procesamiento, estas diferencias se superan mediante el tratamiento de colas de software. Así, cuando se detecta un evento, este es analizado y puesto en una cola apropiada de acuerdo al tipo de procesamiento que requiere. La cola es el intermedio en el cual los eventos son apilados en el orden que llegan. En la Fig. 1.12 se bosquejan algunos de los eventos que se producen en la comunicación entre dos usuarios.

Otro de los aspectos importantes en los procesos de conmutación digital son los sistemas de interrupciones

Una interrupción es un control externo de tipo hardware o software que permite avisar al procesador de la ocurrencia de un suceso que precisa su atención y tratamiento.

Para asegurar el trabajo apropiado de la central, las diferentes tareas que el procesador debe llevar a cabo son clasificadas y ejecutadas en varios grados de urgencia. Para garantizar que ninguna información externa se pierda y las tareas sean llevadas en un orden correcto de prioridades, los programas son agrupados en varios niveles de interrupción, con programas de alto nivel a

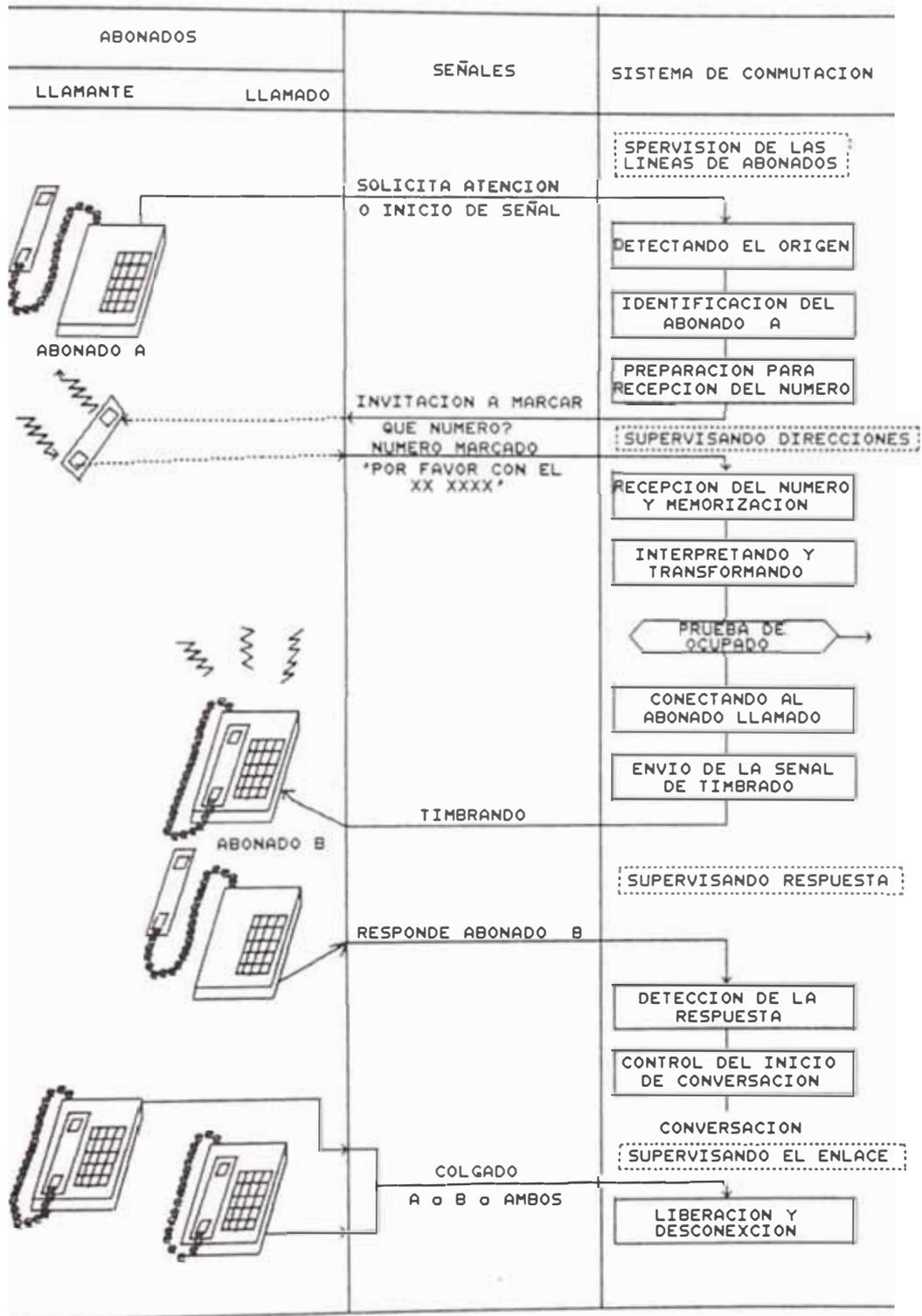


FIG 1.12

PRINCIPIOS DE LA ADMINISTRACION DE EVENTOS  
PRODUCIDOS POR LAS LLAMADAS TELEFONICAS

los cuales se les permite interrumpir la ejecución de un programa de menor nivel. Cuando ocurre esto el programa en ejecución es detenido, y los contenidos de los registros usados en el programa son salvados, luego se ejecuta el programa que interrumpió y el control es retornado al programa interrumpido, restaurando sus registros.

Las causas que pueden generar una interrupción son diversas, entre las principales se encuentran:

Interrupciones por errores o averías: tales como falla de alimentación, errores de paridad de memoria, fallas del sistema, etc.

Interrupciones de reloj: en los sistemas operativos de tiempo real.

Interrupciones externas: aviso de modificaciones importantes en los periféricos.

Interrupciones por causa de programas: recepción de mensajes, direccionamientos incorrectos, desbordamiento, intento de escritura en zonas prohibidas, etc.

Interrupciones del sistema de entrada/salida: fin de temporizaciones, fin de transferencias de datos, condiciones de periféricos libres o dispuestos, etc.

Para el control y gestión del sistema de interrupciones deberá considerarse los siguientes aspectos:

Garantizar el desarrollo de la tarea en proceso, cuando se produce una interrupción.

Permitir diversas interrupciones.

Establecer un sistema jerárquico de interrupciones.

Dentro de la organización del software de la central, consideramos la siguiente jerarquía:

Nivel de falla.- Es el mas alto dentro del sistema. Cuando una falla se detecta por software o hardware , el programa de procesamiento de fallas se activa de inmediato para reconfigurar al sistema de la condición de falla, aun cuando se ejecuten programas de otro nivel.

Nivel de reloj.- Se especifican en el nivel de reloj aquellas tareas que requieren un procesamiento de tiempo real muy alto o de ejecución periódica. Este nivel esta controlado por un programa unidad que se activa por la interrupción de reloj cuya duración del ciclo básico es de 8ms a 10ms, para nuestro caso es de 10 ms.

Nivel de funciones críticas en el tiempo.- Estas tareas tienen un nivel de prioridad intermedio, debido a sus funciones que están restringidas por temporizadores internos y externos.

Niveles de funciones no críticas.- En este nivel se ejecutan las tareas simples que están en cola

y no es muy urgente su atención.

La Fig. 1.13 nos muestra un ejemplo de la ejecución de las tareas en un ciclo básico.

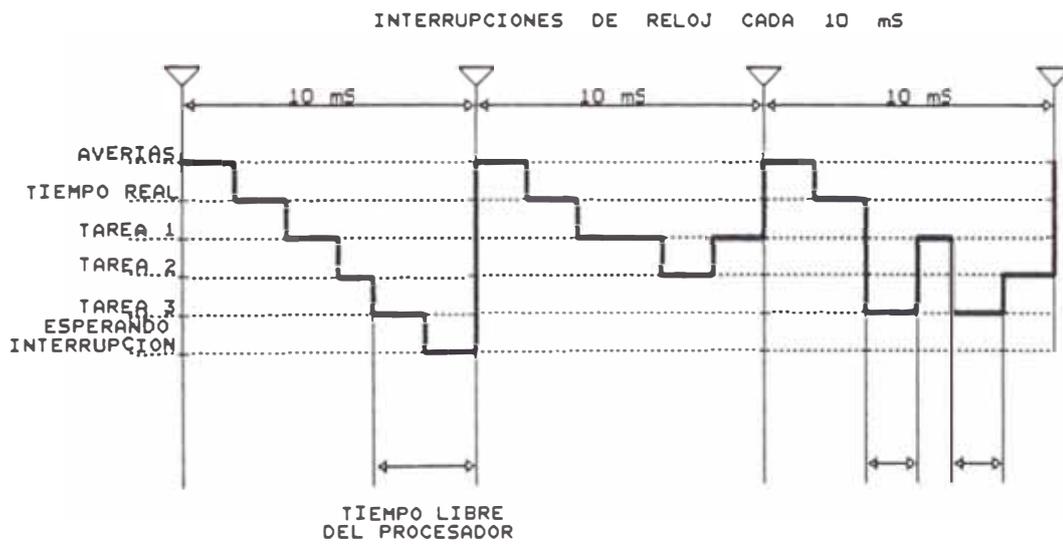


FIG 1.13

EJECUCION DE TAREAS EN UN INTERVALO DE TIEMPO REAL

## C A P I T U L O      I I

### FILOSOFIAS DE DISEÑO Y SISTEMAS DISTRIBUIDOS

Como determináramos en el capítulo anterior, los sistemas de conmutación digital, están basados en un sistema distribuidos ( multiprocesadores).

El objetivo principal de la tesis, es bosquejar el diseño del sistema de comunicación entre los procesadores de la central, empleando los canales PCM de primer orden; para lo cual debemos adoptar una filosofía de diseño y aplicar ésta en los sistemas de multiprocesadores, específicamente en la comunicación entre ellos.

En cuanto a la filosofía de diseño que actualmente es muy usado, es el DISEÑO ESTRUCTURADO, que será uno de los pilares fundamentales para la elección de nuestro objetivo.

Asimismo, en el presente capítulo se tratará los aspectos importantes de los sistemas distribuidos.

#### **2.1 Filosofía del diseño estructurado.**

Basándonos en el criterio de DISEÑO ESTRUCTURADO cuya metodología es ampliamente practicado y aplicado en la actualidad.

Los aspectos comunes para la variedad de las técnicas

de diseño estructurado incluyen los siguientes principios:

- Especificación total del sistema.
- Arquitectura jerárquica y modular.
- Personalización de cada módulo funcional.
- Definición precisa de los módulos de interfase.

La aplicación de éstos principios de diseño estructurado, determinan un bajo costo en el desarrollo e implementación del Software y Hardware del Sistema.

Un buen desarrollo del Sistema (especialmente en el software) aplicando los principios del diseño estructurado, puede reducir grandemente el costo de los otros componentes del sistema, esto se aprecia porque es mas fácil modificar el software que el hardware.

Es necesario recalcar, que aún se apliquen las técnicas mas sofisticados en el desarrollo del software, si no está asociado a un eficiente desarrollo del hardware (lo que implica el uso de los circuitos integrados mas convenientes), no podría esperarse buenos resultados.

Se deben definir módulos hardware discretos e independientes; analizando rigurosamente sus entradas, salidas y su funcionalidad, similarmente los módulos software serán definidos de igual forma.

La filosofía del diseño del software estructurado producen módulos los cuales son altamente reusables, sobre un número y variedad de aplicaciones en un sistema

o sistemas. En los sistemas modulares jerárquicos, el módulo de menor nivel es normalmente reusado con mayor frecuencia, ellos proveen núcleos específicos de funcionalidad requeridos por un gran porcentaje de los módulos de nivel superior, esto es particularmente válido para módulos software, el cual soportan interacción con el hardware, y es llamado comúnmente software de acoplamiento hardware.

### **2.1.1 Filosofía del diseño de software estructurado.**

En ésta sección revisaremos íntegramente, lo que sería una aproximación para el diseño modular del software, poniendo énfasis en la aplicación de esta técnica, con el objetivo inicial de diseñar el acoplamiento del software y hardware.

Cuando hablamos del diseño de un software modular, nos referimos a la generalidad aceptada en el desarrollo del software metodológicamente considerando su integridad y el aprovechamiento del diseño iterativo.

Estas técnicas están resumidas, generalmente, en algoritmos, que aparentemente son simples, pero tienen complejos procesos de diseño, que a continuación trataremos de explicar.

Un módulo es una abstracción de una función, está definido por las entradas, las salidas que se generarán y la función, lo que se ejecutará. Un módulo puede por sí solo pertenecer a uno o más módulos subordinados, cada módulo subordinado tiene sus propias especificaciones de

entradas y salidas, alguna de las cuales pueden ser definidas en los módulos superiores.

La jerarquía de los módulos subordinados es relativa en virtud de sus entradas, salidas y funciones. Directamente o indirectamente esos atributos contribuyen para la definición operativa del módulo superior. Uno o más módulos pueden ser considerados equivalentes si la especificación de sus entradas, salidas y funciones son idénticas, aún que tengan variaciones en los últimos detalles de implementación.

Para empezar el proceso del diseño, un solo módulo del sistema es especificado, llamado **PROGRAMA PRINCIPAL**. El proceso involucra especificar sus entradas, salidas y funciones. El proceso de definir la función, el cual el módulo ejecutará, implica perfeccionar el módulo, pudiendo llegar a ser un pequeño subordinado. Si es así, cada módulo generado nuevamente en su turno es perfeccionado. Como podría ser esperado, el módulo de nivel mas bajo es en el diseño del sistema jerárquico, lo mas específico y singular en su función. El algoritmo termina, cuando cada módulo ha sido perfeccionado, hasta que no requiera futuras creaciones de subordinados.

Una parte integral de la estrategia del diseño estructurado, es la de cercar la estructura de un dato lógico con un módulo de software el cual ejecuta operaciones en esa estructura. Otro módulo no puede, legítimamente operar en la estructura del dato, si no a

través del cercamiento modular.

Esta práctica es llamada, TECNICAS DE INFORMACION ESCONDIDA, en adelante diremos que el módulo que esconde la información acerca de la estructura está ACOPLADO a esa estructura.

Información escondida está basada en la premisa que el efecto de un cambio para la estructura de un dato fundamental está limitado para los módulos en el cual está operando.

Para nuestros propósitos el gran beneficio de ésta estrategia es la creación de la generalización y una funcionalidad altamente reusable. Si cada módulo software el cual opera en una estructura, realiza un objetivo a través de la interfase proveída por el módulo de acoplamiento, entonces ésta funcionalidad es reusable en cada instancia. Por ejemplo, consideremos una pila de datos estructurados. Una pila es la última entrada, la primera salida de datos estructurada esta normalmente acoplada por un módulo, el cual provee las siguientes operaciones: PUSH, POP, IS\_EMPTY, IS\_FULL , en la pila. Otros módulos pueden operar en la pila, pero solamente a través de esos módulos de pila. Un elemento es tomado (PUSHed) o puesto (POPed) del o en el extremo superior de la pila; una llamada para el módulo IS\_EMPTY o IS\_FULL retorna un valor para indicar si todos o ningún espacio de la pila está disponible respectivamente.

Cualquier módulo llamante no tiene conocimiento del

formato de la pila. consecuentemente, ellos no pueden crear funcionalidades para operar en ellos. En su lugar reusan la funcionalidad provista por los módulos de las pilas.

El objetivo del acoplamiento estratégico no está limitado a estructuras lógicas de datos, si no que también es aplicable asociándolos con dispositivos hardware.

Así como se procedió en la estructura de datos, cercaremos el hardware con preguntas mediante un módulo de software, el cual ejecutará operaciones en él. Cualquier otro módulo que opera en ese hardware , lo hará solo a través del módulo cercado. Como consecuencia de esto, el módulo de acoplamiento hardware es altamente reusable.

Resumiendo, en ésta sección, podemos reconocer algunos beneficios, tales como:

Si el objetivo es una estructura de datos software, solamente la capa de acoplamiento inmediata del software necesita alteración para cambiar la estructura del dato de acuerdo a demanda.

Si numerosas aplicaciones requieren acceso al objetivo, cada uno de esas aplicaciones pueden recurrir y confiar en el software de acoplamiento. Como resultado el software para acceder al objetivo puede ser reusado, como una

opción mejor que rediseñar el software para todas esas aplicaciones.

La misma técnica y beneficios se aplican cuando acoplamos un módulo software a un dispositivo físico (hardware). Desde que un dispositivo hardware está diseñado para ejecutar tareas especificadas en una o mas funciones y cualquier software diseñado para interactuar directamente con el dispositivo efectuará la performance de sus funciones, las que serán usadas por las diversas aplicaciones que la requieran.

### **2.1.2 Filosofía del diseño de hardware estructurado.**

Como se mencionó anteriormente el hardware para cualquier sistema debe diseñarse empleando componentes especializados en funciones determinadas, para aplicar eficientemente las ventajas del diseño estructurado. Con el desarrollo de la microelectrónica se han originado los circuitos integrados potentes y económicos, tecnología adecuada que requiere nuestro sistema.

El uso de éstos dispositivos permitirá lograr el objetivo de interactuar adecuadamente con el software, diseñado con esta misma filosofía.

El objetivo en lograr un módulo de interfase soporte hardware que interactúe con los otros módulos, este modulo es totalmente útil para otros módulos del sistema. En efecto todos los módulos primero requieren de una iteración de una forma u otra con varios elementos del

sistema hardware. El hardware invariablemente demanda el mismo protocolo exacto, que alrededor de él se crea un módulo independiente y altamente reusable para servir a todos los módulos del sistema que requieren de él. Haciendo esto, todos los módulos del sistema tienen un acceso transparente a cualquier elemento hardware y el módulo Interfase Soporte Hardware (ISH) es creado una sola vez.

La transparencia es una consideración clave. Esto significa que los módulos del sistema pueden solamente comunicarse con el hardware a través de mensajes parametrados de y hacia el módulo ISH. Lo intrincado de la interacción directa del hardware, son completamente escondidas tras el software del módulo ISH. Este concepto es crucial para nuestra definición de acoplamiento software y hardware.

Referente a los detalles de la estructura hardware del sistema se revisarán los aspectos generales en la siguiente sección, sistemas de multiprocesadores y en los siguientes capítulos el detalle del sistema hardware.

## **2.2 Sistemas distribuidos - Multiprocesadores.**

El término de sistemas distribuidos se usa comúnmente para describir a diferentes clases de sistemas con múltiples procesadores, en los que la potencia de tratamiento de la información se encuentra repartida en el espacio, los que aparecieron como una alternativa a los sistemas clásicos en los que la potencia del

tratamiento de la información se concentraba en un único procesador.

El esquema básico consiste en un conjunto de procesadores conectados mediante un mecanismo de comunicación e interconexión; formando así redes y sistemas de comunicaciones.

La realización de equipos potentes para el tratamiento de procesos e información, tal es el caso de la Central Telefónica Digital, está basada en la cooperación sistemática y ordenada de elementos de menor potencia, funcionando en paralelo y en forma asíncrona o sincrónica.

En la actualidad, con el desarrollo tecnológico de la electrónica, nos permite la realización de equipos secuenciales potentes y rápidos, los que resultan atractivos para soportar los sistemas de tratamiento de señales en tiempo real.

Resumiendo diremos que esta tecnología está basada en el uso de múltiples procesadores, no necesariamente de las mismas características, entre los que se distribuye las funciones y carga del sistema de conmutación telefónica digital, disponiendo de un medio de comunicaciones entre ellos bajo una estructura jerárquica modular.

### **2.2.1 Componentes de un sistema distribuido.**

Estos sistemas están constituidos por un conjunto de elementos hardware y software, capaces de realizar

conjuntamente, una determinada función orientada a la resolución del problema objeto de la aplicación del sistema.

Dichos elementos serán tareas (programas), recursos (periféricos, memorias, etc.), entre los que podrán establecerse determinadas relaciones. Por ejemplo comunicaciones entre tareas con el objeto de intercambiar información para la cooperación en la resolución de una determinada función.

Es evidente que el mecanismo que permita materializar dichas relaciones, deberá estar igualmente distribuido entre los diferentes componentes del sistema. Dicho mecanismo estará formado básicamente por un conjunto de programas software residentes en los procesadores.

La materialización de las relaciones entre los elementos dentro de un sistema distribuido implica el establecimiento de enlaces entre ellos, con el objeto de permitir el intercambio de información.

La realización de un enlace implica la utilización de una vía de comunicación, que comúnmente se denomina camino lógico. Un camino lógico es pues, una vía de comunicación bidireccional y que se caracterizará por las prestaciones que debe satisfacer, entre el que cabe destacar es el volumen de la información que transportará, que se mide en bps. La utilización de éstos caminos deben obedecer a un control y gestión.

Por otra parte los caminos lógicos deben ser

soportados por los caminos físicos existentes. Un camino físico es una vía de comunicación, realizada sobre un soporte material capaz de permitir la transmisión de la información, mediante la utilización de alguno de sus parámetros físicos de dicho medio. En nuestro caso el medio ya está definido Sistema PCM de primer orden (Recomendación G.732 del CCITT).

### **2.2.2 Estructura de los caminos lógicos y físicos.**

La estructura de los caminos físicos, que son utilizados por los caminos lógicos, se le denomina topología del sistema. Entre las soluciones más utilizadas para establecer las topologías están:

**Interconexión total tipo malla.-** Resulta evidente que la mejor solución consiste en interconectar directamente todos los procesadores del sistema mediante un conjunto de caminos físicos que los enlace dos a dos. En éste caso todos los caminos lógicos entre cada par de procesadores, compartirán el mismo camino físico directo entre ellos. Las limitaciones de la aplicación de ésta solución vienen dadas por el costo que representan realizar esta interconexión total, por lo que quedaría limitado a casos en los que existan pocos procesadores y las distancias entre ellos reducidas.

**Interconexión parcial tipo estrella.-** Cuando las distancias entre los procesadores son grandes, se

hace necesario reducir los caminos físicos estableciendo nodos de interconexión que permitan establecer, en forma indirecta los diferentes caminos lógicos necesarios en el sistema. La implementación de esta topología es complicada debido fundamentalmente a la dificultad de incorporar en dichos modelos los parámetros reales del sistema lo que conduce a utilizar modelos basados en suposiciones de comportamiento muy simples, no llegándose a optimizar.

**Interconexión tipo bus.-** Esta interconexión es la mas utilizada, el principio de funcionamiento es establecer los caminos lógicos a través de un único camino físico pudiendo ser bidireccional. Existen dos variantes de interconexión tipo bus:

**Interconexión Serial.-** La información se transfiere bit a bit, independientemente del código, tipo de transmisión, velocidad, etc; haciendo uso de un único camino físico. Es la forma normal de transmitir información a largas distancias.

**Interconexión en Paralelo.-** En este caso se transmiten simultáneamente todos los bits de un carácter lo que implica un medio físico de transmisión con tantos elementos físicos (conductores) como bits contenidos en el carácter o palabra a transferir. Ello

conlleva una mayor complejidad del medio y redonda en una mayor velocidad de transmisión, comúnmente es usado por sistemas integrados, es decir dentro de un propio centro de procesamiento.

**Interconexión tipo árbol.-** La topología tipo árbol puede ser conceptualizada como una interconexión tipo bus expandido, de ésta forma las ventajas y desventajas encontradas al bus son aplicables al árbol.

**Interconexión en anillo.-** Con el mismo objetivo de simplificar la estructura del sistema de interconexión aparecieron las topologías en anillo, en éste caso los diferentes caminos lógicos se realizan sobre un mismo camino físico que interconectan a los diferentes elementos del sistema, cerrándose sobre sí mismo. El anillo está constituido por un conjunto de nodos, enlazados mediante conexiones punto a punto. Las funciones que realizan los diferentes nodos pueden revestir ciertos grados de complejidad según el tipo de sistema que se implemente.

### **2.2.3 Transmisión de la información.**

Para establecer la comunicación entre los procesadores, es necesario definir los elementos básicos de la información a transmitir, y estos son los códigos asociados a las letras del alfabeto u otros símbolos como

dígitos y signos, que comúnmente son denominados caracteres. El código elegido es el ASCII (American Standard Code for Information Interchange), conocido también como el Código CCITT No 5, actualmente es el más usado por su estandarización. En la transmisión sincrónica los datos fluyen del emisor al receptor con una cadencia fija y constante, marcado por una base de tiempo común para todos los elementos que intervienen en la transmisión.

En el extremo receptor se reconstruye la señal de reloj a partir de la señal recibida en línea, realizándose en él, el sincronismo de bit y entrega al procesador la información y el reloj ya sincronizadas para proceder a su identificación con la ayuda de algunos caracteres especiales, dependiendo del protocolo de comunicaciones empleados.

#### **2.2.4 Modos de explotación de la transmisión.**

Los circuitos pueden emplearse de tres modos básicos:

- **SIMPLEX:** La transmisión se realiza solamente en un sentido, sin posibilidad de hacerlo en sentido contrario. Comúnmente es empleado en telemetría.
  
- **SEMI-DUPLEX:** La transmisión se lleva a cabo alternativamente en uno y otro sentido, exigiendo un cierto tiempo para cada inversión que reduce la eficiencia del sistema. Es el modo

mas usado en la transferencia de información entre procesadores.

- DUPLEX Consiste en la transmisión simultanea e independiente en ambos sentidos. Este modo de explotación, si bien soporta una gran eficiencia de línea, exige un hardware muy complicado y costoso.

### 2.2.5 Detección de errores.

Cómo las características de los caminos físicos no son ideales, existe la posibilidad de producirse errores en la transmisión de la información.

Simultáneamente con las técnicas de transmisión han ido desarrollándose métodos orientados hacia la solución de este problema y que permiten detectar un amplio subconjunto de los errores que pueden producirse en la transmisión de un bloque de información.

En el caso de la transmisión síncrona, suele ser mas habitual el uso de métodos de control de error denominados detección longitudinal o cíclica (CRC); en este caso, la información utilizada para la detección de los errores (habitualmente de 16 bits) se genera a partir de los coeficientes del polinomio resto obtenido de la división de un polinomio (de grado  $m-1$ ) cuyos coeficientes binarios son los  $m$  bits que constituyen la información, por un polinomio cociente de referencia.

### 2.2.6 Control de flujo de la información.

La comunicación de dos elementos a través de un camino

lógico se realizará mediante un diálogo; que consiste en un intercambio de mensajes entre el origen y destino de la información.

Generalmente, cada mensaje transmitido exige una confirmación de su receptor correcta o incorrecta, pudiéndose evaluar también la evolución de los métodos empleados para el control del flujo.

En los inicios de estos sistemas, se puede apreciar cómo para cada mensaje de información, se requería otro mensaje de confirmación en sentido contrario, la ausencia de ésta indicaba error.

La utilización de este tipo de diálogo implica que el número total de mensajes que circula por el camino lógico sea el doble del número de mensajes efectivos, lo cual conduce a una baja del rendimiento.

Una manera de disminuir el volumen de información transmitida, consiste en eliminar la necesidad de responder la transmisión de cada mensaje mediante un mensaje específico; esto se consigue incorporando la confirmación en mensajes de información transmitidas en sentido contrario, reservando para casos excepcionales las confirmaciones positivas o negativas, mediante mensajes especiales.

El volumen de la información a transferirse, depende de las tareas y aplicaciones de cada elemento (procesador) del sistema, pudiendo ser no simétrica.

Se puede enviar paquetes de información sin previa

confirmación de los anteriores, en estos casos es necesario fijar el número máximo de mensajes que pueden transmitirse sin haber recibido confirmación, comúnmente denominado **ventana** de mensajes sin confirmar.

De acuerdo a los acápites anteriores se ha observado que a parte de los mensajes que transmiten la información conteniendo los datos, es necesario el intercambio de otros tipos de mensajes en el diálogo para realizar las funciones complementarias, tales como inicio y fin de una comunicación, confirmación de la recepción de mensajes, supervisión y otras. Estos tipos de mensajes se codifican y se asocian como parte de la información a transferir.

#### **2.2.7 Protocolo de comunicación.**

Al conjunto de reglas y procedimientos que regulan el intercambio de información entre los procesadores que cooperan se les denomina protocolo de comunicaciones.

En un sistema distribuido un protocolo permite fundamentalmente iniciar, mantener y terminar un diálogo entre los elementos del sistema, asimismo un protocolo regula la forma en que deben generarse e interpretarse los elementos orientados al control de errores y la forma de recuperar las informaciones recibidas erróneamente, igualmente estará prevista la forma de identificar el camino lógico que se utiliza para el intercambio de la información y la identificación del tipo de mensajes.

Como se aprecia el elemento básico del protocolo es el mensaje, dentro del cual además de los datos, existirán

otras informaciones destinadas a permitir: las detecciones de errores, la identificación del camino, el control de flujo de las informaciones y la identificación del tipo de mensajes que se trate. Todas estas informaciones se materializarán en bloques con una determinada estructura que constituirá el formato del protocolo.

De acuerdo a las características de control y acceso se pueden tener hasta 4 protocolos

PROTOCOLO	CONTROLADO	POR DEMANDA
CENTRALIZADO	SONDEO	INTERRUPCION
DISTRIBUIDO	TOKEN-PASSING	CSMA/CD

**Sondeo.**- Normalmente relacionado con topologías tipo estrella y bus, requiere un controlador en la estación central el cual interroga a cada estación secundaria por su necesidad de acceder a la red. Su desventaja es la pérdida de recursos en la red.

**Interrupciones.**- Un protocolo basado en interrupciones, también se implementa en topologías estrella y bus, está basado en el momento que una estación secundaria interrumpe al controlador de la estación primaria planteándole su requerimiento.

**Token Passing.**- Este protocolo, usado en topologías anillo y bus, distribuye el control entre las estaciones de la red, sin embargo el

acceso a ella debe obedecer una orden. Ninguna estación debe acceder a la red hasta no recibir el token. El token es un conjunto de bits que es pasado de estación a estación. Cuando una estación tiene el token, puede enviar su mensaje, cuando termina de hacerlo, el token es pasado a la siguiente estación y así sucesivamente. Si una de las estaciones no tiene nada que transmitir, simplemente pasa el token a la siguiente.

**CSMA/CD (carrier sense multiple access with collision detection).**- Este protocolo es usado en topologías tipo bus y árbol distribuye el control en las estaciones de la red, el acceso a ella es ejecutada a demanda. Con CSMA/CD, todas las estaciones tienen la capacidad de "escuchar" si el medio de comunicaciones está siendo usado por otra. Cuando una estación requiere enviar su mensaje, verifica que nadie lo esté haciendo, de ser así envía su información. Existe la posibilidad que otra estación, en esos momentos coincidan en la transmisión, lo que provocarían una colisión de datos en el medio; las estaciones están preparados para detectar esa colisión y reintentar el envío del mensaje. Como se puede observar en este protocolo, la parte crítica es la recuperación después de la colisión. Es en éste aspecto que la respuesta y rendimiento de

este protocolo sea probabilístico y no determinístico y por otra parte su implementación es muy compleja.

Los análisis antes planteados nos ha permitido fijar el objetivo del diseño de nuestro sistema: jerárquico modular con la filosofía del diseño estructurado y con los conceptos de los sistemas distribuidos multiprocesadores; planteando la arquitectura mostrada en la Fig. 2.1 y que en los siguientes capítulos se detallarán y analizaremos con mayor profundidad.

## C A P I T U L O    I I I

### FORMULACION DEL PROYECTO.

En el presente capítulo se discutirá y analizará un modelo teórico del sistema de la comunicación entre los procesadores de la central, basados en un modelo referencial estandarizado para sistemas abiertos, propuesto por la Organización Internacional de Estandarización (International Standards Organization - ISO); así como una metodología para la evaluación de sistemas similares al nuestro.

#### **3.1 Determinación de un modelo teórico.**

Un sistema de Comunicaciones entre Procesadores es un componente crítico en cualquier sistema de Procesamiento Distribuido y las fallas producidas en este componente, resultará catastrófico para el sistema. Adicionalmente su performance debe ser muy alta, para nuestro caso debe garantizar la transferencia adecuada de datos y control dentro de la Central Telefónica Digital y periféricos. Por eso no debemos perder de vista, en nuestro diseño, las siguientes características básicas: Alta performance, alta confiabilidad y configuración flexible.

##### **3.1.1 Modelo de interconexión de sistemas abiertos.**

La Organización de Estándares Internacionales (

International Standards Organization - ISO ), ha definido un modelo referencial de siete niveles para redes, los cuales son de mucha ayuda para el diseño de redes de comunicaciones, en especial para sistemas abiertos.

Para determinar nuestro modelo teórico recurriremos al modelo referencial OSI (Open Systems Interconnections) o en español ISA (Interconexión de Sistemas Abiertos). El modelo ISA es una referencia estandarizada internacionalmente, y que el CCITT lo acoge en su Recomendación X.200, cuyo propósito es proporcionar una estructura bien definida para modelos de interconexión e intercambio de información entre usuarios (estaciones) de un sistema de comunicaciones, con aplicaciones ilimitadas no solamente en sistemas abiertos, sino en comunicaciones entre procesadores de un sistema cerrado, con posibilidades de integrarse en un sistema abierto. En la actualidad, ISA solo considera protocolo de conexión orientada; esto es, protocolo con el cual se establece una conexión lógica antes de transferir datos. Una aplicación clara y cercana al objetivo de ésta tesis es la Señalización de Canal Común, es decir, la Señalización Nro. 7 del CCITT Recomendación Q.700.

El modelo ISA es un concepto en virtud al cual las relaciones entre las estaciones de una red de comunicaciones, están representados por una jerarquía de niveles de protocolo. Cada nivel tiene una o mas funciones, empleando los servicios proporcionados de los

niveles adyacentes, especialmente las inferiores.

Una primera aproximación de nuestro modelo la haremos, tomando como referencia el modelo ISA y contemplará los siete niveles previstos.

Desde el punto de vista particular de un nivel, los niveles inferiores proveen un "servicio de transferencia" con características específicas. La forma en la cual los niveles bajos son realizados es intrascendente para los niveles superiores. Correspondientemente, a los niveles inferiores no le concierne el significado de la información proveniente de los niveles altos o las razones de su transferencia.

Las características funcionales de cada nivel son resumidas a continuación:

**Nivel 1 Físico.**- En este nivel se definen y se materializan las características mecánicas, eléctricas, funcionales y de procedimiento para establecer, mantener y terminar la interconexión física entre los elementos de la red. El nivel físico efectúa la transmisión transparente de un tren de bits sobre un circuito soportado en algún medio físico de comunicaciones.

**Nivel 2 Enlace de datos.**- El objetivo de éste nivel, es proporcionar los elementos necesarios para establecer, mantener y terminar interconexiones de enlaces de datos; proporcionando una confiable transferencia de información a través

del medio físico.

**Nivel 3 Red.-** Proporciona a los niveles superiores la independencia de las tecnologías de transmisión, ruteo y retransmisión entre los usuarios finales. La comunicación entre usuarios finales queda regulado mediante un protocolo de red.

**Nivel 4 : Transporte.-** El nivel 4, proporciona un servicio de transporte de la información a nivel usuarios finales (para efectuar sesiones); entre los servicios proporcionado tenemos:

Servicios orientados al establecimiento de una conexión.

Servicios orientados hacia la realización de una transacción o intercambio de datos

. Servicios orientados hacia la difusión de informaciones a múltiples destinatarios.

Una de las razones que justifica la existencia de éste nivel, es la optimización de los recursos, con el objeto de minimizar los costos de dichos intercambios de informaciones.

**Nivel 5 : Sesión.-** El nivel de Sesión proporciona la estructura de control para la comunicación entre aplicaciones. Cumple con las funciones de: establecer, administrar y terminar las comunicaciones a nivel de sesión entre las aplicaciones cooperantes.

**Nivel 6 : Presentación.-** El objetivo de éste nivel es proporcionar un conjunto de servicios a las unidades funcionales de nivel superior. Estos servicios están orientados a la interpretación de la estructura de las informaciones intercambiadas por los procesos de aplicación. Las siguientes podrían ser alguna de las funciones, encomendadas a este nivel:

En lo que se refiere a protocolo de terminales virtuales: selección del tipo de terminal y la gestión de los formatos de presentación de los datos.

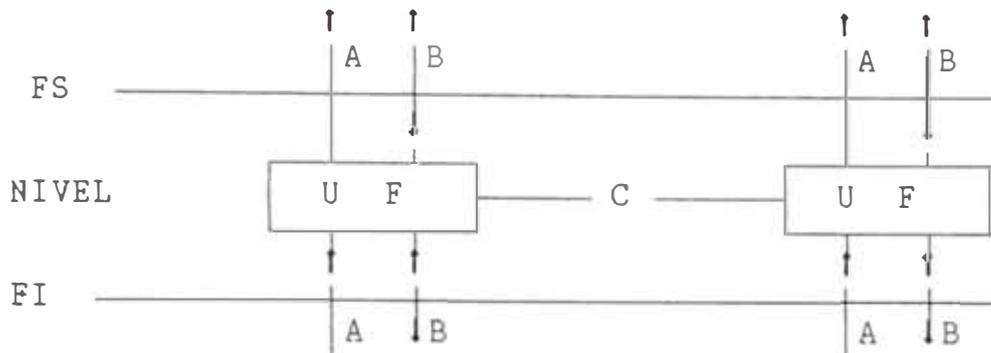
En lo referente al manejo de archivos virtuales: ordenes de manejo, formateo de archivos y conversión de código de los datos.

En lo referente a la transferencia de la información y a la manipulación de las tareas: formateo de los datos y ordenes de control, así como el control de la forma de transferir informaciones.

**Nivel 7 : Aplicación.-** Se trata del nivel superior del modelo de referencia y en él se llevan a cabo las funciones específicas de comunicación entre los diferentes procesos de aplicación que constituyen el sistema. Las comunicaciones entre los procesos se realiza mediante un determinado protocolo.

Para una comprensión mas efectiva de los niveles del

modelo ISA de referencia explicaremos los servicios primitivos de cada nivel. En la Fig. 3.1 se muestra los servicios, su unidad funcional y la relación entre ellas.



Donde:

- A = Servicio de nivel
- B = Servicio primitivo (interfase)
- C = Protocolo
- U F = Unidad Funcional
- F S - Frontera lógica superior
- F I = Frontera lógica inferior

Fig. 3.1

### TIPOS DE SERVICIOS PRIMITIVOS

Describiendo el gráfico anterior diremos:

- Unidad Funcional U F Es la materialización de un conjunto de funciones y/o servicios.
- Fronteras FS y FI Son delimitadores lógicos de una o varias unidades funcionales, entre los niveles adyacentes.
- Servicio de nivel A Es el servicio ofrecido a un nivel superior en una interfase del nivel, es el resultado de uno o varios procesos lógicos

- Interfase de nivel B Llamado también servicios primitivos, es la que involucra la especificación de la interconexión lógica entre los niveles adyacentes.
- Protocolo C Es el conjunto de reglas y formatos en virtud del cual se pueden intercambiar información de control y datos de usuarios entre pares de unidades funcionales.

El uso de servicios primitivos no excluye cualquier implementación específica de un servicio, en términos de interfases primitivas. Un servicio primitivo consiste de un nombre y uno o más parámetros, el cual son pasados en la dirección de un servicio primitivo.

El nombre de los servicios primitivos contiene tres elementos, definidos en la Recomendación X.210 del CCITT:

**Petición.**- Un primitivo requerimiento por un servicio usuario, para invocar un elemento del servicio.

**Indicación.**- Primitiva indicación de un servicio proveedor, para advertir que el elemento del servicio ha sido invocado por el servicio usuario, en el punto de acceso del servicio o unidad funcional.

**Respuesta.**- Una respuesta primitiva por el servicio usuario, para completar a un particular punto de

acceso del servicio; el cual previamente ha indicado el elemento del servicio, para ese punto de acceso.

**Confirmación.**- Es una primitiva confirmación de un servicio proveedor para completar a un particular punto de acceso del servicio, en el cual el elemento del servicio ha sido previamente invocado por una petición de la unidad funcional.

No todos los tipos de servicios anteriormente mencionados pueden ser asociados en todos los servicios del nivel. Además de lo anterior, en el nombre se debe especificar la acción a ser ejecutada. Así como especificar sub-niveles si son necesarios.

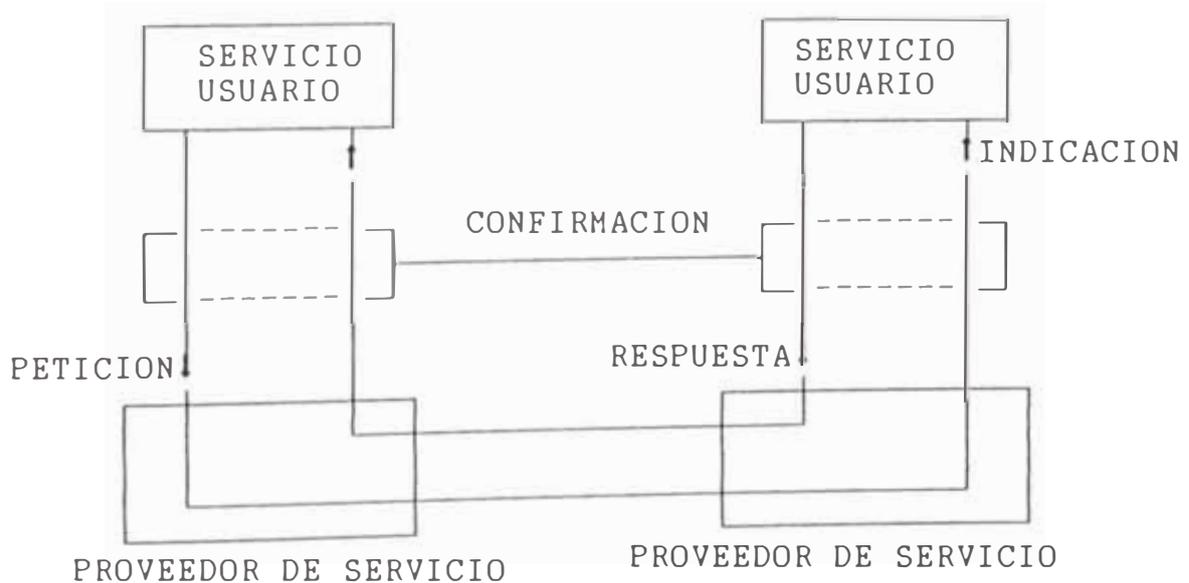


Fig 3.2

FLUJO BASICO DE LOS SERVICIOS PRIMITIVOS

La Fig 3.2 nos muestra en forma resumida lo que podría ser un flujo de servicios primitivos dentro de un nivel del ISA.

En la siguiente Fig. 3.3 nos muestra los siete niveles referenciales ISA, y que la presente tesis tomará como referencia para diseñar el modelo del sistema.

### **3.1.2 Especificaciones de nuestro modelo teórico empleando los niveles ISA.**

Un primer intento del diseño de nuestro sistema sera formular, definir y especificar las unidades funcionales en cada nivel ISA. Este diseño se irá refinando a medida que avancemos su implementación.

#### **Nivel 1 : Físico**

**Análisis.-** Los procesadores de la central estarán ubicados localmente y a distancias relativamente cortas. La tesis propone el empleo de sistemas PCM de primer orden para la comunicación entre los procesadores y las centrales digitales basan su diseño empleando las técnicas de PCM.

**Especificaciones.-** Debido a las características del sistema PCM la comunicación entre los procesadores será serial. Los medios físicos disponibles serán: cable de pares twisteados, cables coaxiales, fibras ópticas, tarjetas madre (back plane) y otros. Estos medios permiten la interconexión en banda base, es decir, no necesitan modulación, donde los bits son representados por niveles de voltaje. Se propone

que el canal PCM de 64 Kbps sea asumido como parte del nivel 1. Se definirá las características mecánicas y electrónicas, así como las funcionales del medio de comunicación. En ellos están involucrados los niveles de voltaje, corriente, impedancia, asignación de pines y conectores físicos.

### **Nivel 2 : Enlace de datos**

- **Análisis.**- Existen limitaciones inherentes en la implementación física del medio de comunicación antes elegido, el cual puede inducir errores. Los procesadores de la central intercambiarán información ( datos y control ), las que deben ser plenamente garantizadas respetando su integridad y libre de errores.
- **Especificaciones.**- En este nivel se debe emplear un protocolo que permita la detección de errores de transmisión y recupero de la información. La tesis propone el uso de controladores HDLC ( High Data Link Control) cuyo protocolo incorpora códigos de detección de errores CRC (Cyclic Redundance Check). Así se puede enmascarar las deficiencias del medio de comunicación. Mas adelante se explicará con mas detalle esta técnica.

### **Nivel 3 : Red**

- **Análisis.**- Como se apreció en los capítulos

anteriores el diseño de la Central Digital está basado en una arquitectura jerárquica modular conformada por procesadores centrales y regionales, dependiendo de sus funciones. Para la comunicación entre ellos se deben establecer caminos físicos y/o virtuales para transferir la información, las que deben ser controlados por un procesador central, además se tiene disponible los recursos de transmisión y conmutación internos de la central, los mismos que pueden ser aprovechados por nuestro sistema. En este nivel se debe definir los modos de operación y topologías

- **Especificaciones.**- El control de red será efectuado por el procesador central y usará los recursos internos de la central para el enrutamiento de las comunicaciones y gestionar los caminos lógicos necesarios. El modo de operación es del tipo Maestro (procesador central) Esclavo (procesadores regionales) multipunto. Optando una topología tipo bus.

#### **Nivel 4 : Transporte**

- **Análisis.**- Los niveles anteriores han definido el bloque de transmisión. En este nivel ya se define la comunicación entre los procesadores cooperantes (usuarios finales), los que efectuarán sesiones, intercambiando datos y/o

transacciones, pudiendo ser destinos unitarios o múltiples.

- **Especificaciones.**- El nivel de transporte siempre operará entre procesadores cooperantes (end to end). Controlará el establecimiento de las comunicaciones, el control de flujo (destinos particulares o múltiples) y control de errores a nivel de usuarios finales (end to end). En el control de flujo también prever el reconocimiento del enlace mediante preguntas y confirmaciones, así como la asignación de un número determinado de mensajes sin previa confirmación, llamadas "ventanas" de mensajes.

#### **Nivel 5 : Sesión**

- **Análisis.**- Las coordinaciones a nivel de sesión entre los procesadores interconectados por los niveles anteriores, deben ser definidas, para lograr el entendimiento entre los procesos. Como se puede apreciar, aquí la comunicación ya es a nivel de procesos y se debe definir el modo de operación semiduplex o full duplex y otros aspectos.
- **Especificaciones.**- En este nivel se define la comunicación a nivel de procesos de la aplicación. El diálogo entre los procesadores inicialmente debe ser duplex con confirmación, a fin de facilitar el protocolo de este nivel y

ayudar a controlar mejor el flujo de tráfico especificado en el nivel anterior.

#### **Nivel 6 : Presentación**

- **Análisis.**- Las recomendaciones para elegir los procesadores en el diseño de sistemas de conmutación digital, es que éstos sean homogéneos, ayudando así a la reducción de los costos de implementación software y hardware. Considerando esta recomendación, la Central está conformado por procesadores centrales y regionales homogéneos. Esto facilitará grandemente la transferencia de datos y control entre los procesadores.
- **Especificaciones.**- Debido a que los procesadores son homogéneos no se necesita la conversión de códigos. En este nivel se definen los accesos a las bases de datos de la central, transacciones, transferencia y actualización de archivos. Así como la priorización de las tareas que deben ser definidas conjuntamente con el Sistema Operativo de la Central.

#### **Nivel 7 : Aplicación**

- **Análisis.**- Recordando el Capítulo 2 se definieron las aplicaciones de la central a través de las Interfases Soporte Hardware ISH, cuyas funciones serán regionalizadas apropiadamente, para ser gobernados por procesadores centrales y

regionales. Entre los datos importantes están control de dispositivos, datos de conmutación y mensajes entre procesadores.

- **Especificaciones.**- En éste nivel se definen las funciones específicas de comunicación entre los diferentes procesos para cada aplicación que constituyen el sistema.

### **3.2 Criterios de evaluación para la elección de un sistema.**

Describiremos aquí un procedimiento de evaluación desarrollada por Henson, Cleaver y Kaisler, para evaluar sistemas y su arquitectura. En este proceso ocho categorías de criterios de evaluación son identificados. Cada categoría está compuesto por varios sub-factores, aplicado para ambos componentes: hardware y software.

El método de análisis y evaluación es íntegro. A cada criterio se le ha asignado un peso relativo con los otros. Este peso es dividido entre componentes hardware y software. Cada arquitectura recibe una figura de mérito basado en la suma de sus propios pesos calificados para todos los criterios. La arquitectura con las más altas figuras de mérito es estimado como la mejor.

Las ocho categorías son: **simplicidad, factibilidad, modularidad/flexibilidad, mantenimiento, expansión, confiabilidad, costos y performance.** Estas son definidas en ambos términos de hardware y software en las siguientes secciones. Desde que muchos de los criterios

son subjetivos, se recomienda que las evaluaciones sean hechas por un equipo de profesionales competentes, con un resultado final promediado.

La aproximación usada para evaluar arquitecturas candidatas consiste de tres pasos.

**Paso 1:** El equipo de evaluación determina los factores relativos de peso para los 8 criterios de selección. El peso para cada criterio de selección será subdividido en pesos para los factores de hardware y software. La suma total de los criterios de hardware y software deben ser 100 puntos. Un ejemplo de posibles pesos para cada criterio y para los factores de hardware y software se muestran en la Tabla 3.1.

CRITERIOS	HARDWARE	SOFTWARE	TOTAL
SIMPLICIDAD	40	60	15
FACTIBILIDAD	75	25	15
MODULARIDAD	35	65	15
MANTENIMIENTO	30	70	5
EXPANSION	40	60	10
CONFIABILIDAD	40	60	10
COSTOS	25	75	20
PERFOMANCE	60	40	10
TOTAL			100

TABLA 3.1

**Paso 2:** En este paso, cada miembro del equipo de evaluación evalúa las arquitecturas candidatos usando los criterios de selección. Una calificación individual es dada para ambos factores hardware y software, para cada criterio. Las calificaciones individuales son promediados para cada factor y transferidas en una tabla de evaluación.

**Paso 3:** Los factores de hardware y software para cada criterio son sumados y transferidos en una tabla sumaria. La Figura de Mérito (FDM) es calculada por la siguiente fórmula:

$$FDM_j = \left( \sum_{i=1}^8 C_{ij} * W_i \right) * 100\%$$

Donde:

$c_{ij}$  = Valor del i-ésimo criterio para la j-ésima arquitectura  
 $w_i$  = Valor del i-ésimo peso  
 $FDM_j$  = Valor de la FDM para la j-ésima arquitectura.

La arquitectura seleccionada será aquella que tiene la más alta Figura de Mérito.

### 3.2.1 Definición de los criterios de selección.

Los ítemes individuales para la evaluación mostrados en la Tabla 3.1, tienen las siguientes definiciones:

#### **A.- Simplicidad**

Un sistema innecesariamente complejo hace mas difícil su construcción, operación y mantenimiento. La simplicidad, es por eso un criterio importante en la evaluación de las arquitecturas de sistemas alternativos. La simplicidad está determinada por:

**A.1 Número de componentes hardware.-** Con una medición simple, el numero total de componentes hardware debe ser considerada. Una alta calificación se asigna a la arquitectura que tenga menos componentes totales. El número de diferentes tipos de componentes deberían también ser considerados con una calificación alta para menores tipos diferentes de componentes. Considerar dos sistemas, uno de los cuales tiene 10 procesadores idénticos, el otro tiene 4 procesadores , cada uno de diferente tipo. El sistema con los 10 idénticos procesadores recibe una calificación baja por el numero total de componentes pero una alta calificación por su menor numero de componentes diferentes.

**A.2 Complejidad estructural.-** La complejidad estructural es una función de la forma en el cual los componentes del sistema son interconectados. En una evaluación de un sistema multiprocesador, esto es importante

para considerar ambos: el número de caminos de interconexión y la complejidad de la estructura. A más caminos requeridos (por ejemplo interconexión entre dispositivos), menor puntaje.

- A.3 Redundancia.-** Algunas arquitecturas de sistemas pueden acomodar adicionalmente procesadores redundantes mas fácilmente que otros. Esto debe tener mejor puntaje.
- A.4 Número de componentes software.-** El número de componentes software tales como procesos, programas y bases de datos, deben ser considerados. Un alto puntaje es asignado para menor número de componentes software.
- A.5 Complejidad del software.-** La complejidad del software se relaciona con el número de caminos de comunicaciones entre los componentes del software y la estructura formada por los caminos de comunicación. Una alta calificación es dada para menores caminos de comunicación. También se da buena calificación a la estructura menos compleja.
- A.6 Redundancia de software.-** Base de datos duplicados (backup) y redundancia de software deben ser a menudo proveídas para reunir requerimientos de alta disponibilidad. Esto adiciona complejidad al sistema software.

Arquitecturas que pueden soportar la necesidad para alta disponibilidad de una manera íntegra debería recibir alta calificación.

## **B. Factibilidad**

La factibilidad de una arquitectura está relacionado con su adaptabilidad y capacidad de proveer un nivel dado de performance entre la tecnología actual y sistemas cercanamente futuros.

**B.1 Disponibilidad de existencias normalizadas (off the shelf).**- Este factor determina la disponibilidad de unidades tales como el procesador central, unidades de comunicación entre procesadores, terminales y unidades de discos. Una alta calificación es dada para una arquitectura que contiene más componentes de fabricación normalizada en existencia.

**B.2 Factibilidad tecnológica.**- Este factor es importante para imponer la factibilidad tecnológica del hardware que no puede ser reemplazado por componentes normalizados. Una baja calificación debe darse a una arquitectura que requiere solución de problemas tecnológicos significantes.

**B.3 Requerimientos de interfase.**- Este factor se relaciona con la facilidad de interfasear con las diferentes unidades. Una baja

calificación se asigna si es necesario desarrollar una interfase especial o si consideraciones especiales son requeridos para manejar la interfase.

**B.4 Disponibilidad de software estándar.-** Este factor se refiere a la disponibilidad de sistemas software estándar el cual puede satisfacer la performance y los requerimientos de soporte. Una alta calificación se asigna por la disponibilidad de sistemas software para procesadores comerciales. Están incluidos en este grupo el sistema operativo, el sistema de administrador de archivos, e implementación de lenguajes para sistemas de orden superior (lenguajes de alto nivel).

**B.5 Factibilidad del software.-** Si el sistema software no es normalizado y no está disponible, la factibilidad tecnológica del sistema software requerido deberá ser evaluado. Si es nuevo, conceptos de software no demostrado deben usarse para permitir que el sistema funcione, una arquitectura que requiere tales conceptos deben tener baja calificación.

**B.6 Desarrollo del software Tiempos de implementación.-** Este factor está basado en la estimación del tiempo requerido para

desarrollar e implementar las aplicaciones software. Está relacionada con la complejidad estructural de la arquitectura del sistema. Una alta calificación es dada para bajos tiempos de estimación para el desarrollo e implementación del software.

### **C.- Modularidad y flexibilidad**

Como parte de la estructura, la flexibilidad es la habilidad de un sistema para manejar diferentes situaciones de carga o lógicas. La modularidad implica una división de un sistema en un número de módulos que pueden ser como componentes separados. Esos módulos tienen interfases limitados para otros módulos.

**C.1 Capacidad en exceso.-** El exceso de capacidad permite al sistema manejar variedades y posibles incrementos de carga sin modificación. Una alta calificación se asigna al exceso de capacidad, el cual puede resultar de gran utilización sin afectar la performance.

**C.2 Adaptabilidad de interfase.-** Este factor evalúa el grado por el cual las interfases son diseñados para permitir futuros cambios con relativa facilidad. Una alta calificación para un alto grado de adaptabilidad.

- C.3 Generalidad de la arquitectura hardware.-**  
Este factor está relacionado con el grado por el cual un sistema es aplicable en diferentes ambientes. Se da una alta calificación para una arquitectura hardware que tiene un alto grado de aplicación sobre una variedad de sistemas característicos.
- C.4 Parametrización.-** La parametrización del software contribuye significativamente en la habilidad del sistema para trabajar en condiciones cambiantes. Un sistema con alto grado de parametrización tiene un alto puntaje.
- C.5 Ubicación de funciones.-** El grado de flexibilidad provista por la arquitectura en la asignación de funciones es importante en la creación de sistemas para una variedad de ambientes. La flexibilidad también es necesaria para permitir que el sistema sea reconfigurado fácilmente en caso de falla de una porción del sistema. Una baja calificación debe ser asignada para una arquitectura que limita la asignación de funciones necesarias.
- C.6 Evolución del desarrollo del proyecto.-** Este factor se refiere al grado con el cual la aplicación del software puede ser

desarrollada en una característica incremental. Un sistema flexible es uno que puede evolucionar a la máxima configuración operacional sin requerir un software masivo o modificaciones estructurales. Un alto puntaje se asigna para un sistema con alto grado de desarrollo evolutivo.

#### **C.7 Generalidades de la arquitectura software.-**

Este factor considera el grado de aplicación que tiene el software para diferentes ambientes, si son aplicables. Una calificación alta es dada para una arquitectura que reduce el número de módulos software (comerciales).

#### **D.- Mantenimiento**

El mantenimiento del sistema es una medida de la facilidad de realizar el mantenimiento preventivo y correctivo para ambos componentes hardware y software. Asociado a este concepto están las ideas de reparación y servicio.

**D.1 Diseño adecuado del hardware.-** Este factor está relacionado con el grado de facilidad en el mantenimiento hardware que ha sido incorporado en su diseño.

**D.2 Estandarización.-** Este factor relaciona la reparación del equipamiento con componentes similares. Si los componentes son

estandarizados el inventario de los repuestos es reducido, así como el número de diagnósticos requeridos para detectar problemas. Una alta calificación es asignada para un alto grado de estandarización.

- D.3 Diseño del software adecuado.-** Este factor considera el uso de buenas técnicas de diseño y administración práctica que asegura un sistema comprensible y estructurado. Una alta calificación para un sistema que está sujeto a todas esas técnicas.
- D.4 Diagnostico software.-** Se asigna una alta calificación para un sistema que proporcione más detalles de los diagnósticos y detección de errores obvios, así como facilidades de recuperación.
- D.5 Personal de soporte.-** Una alta calificación para un sistema que no requiere un personal altamente calificado para el mantenimiento.
- D.6 Soporte de facilidades.-** Se asigna una alta calificación para sistemas que disponen facilidades de pruebas e instalación de soluciones del problema. Estas facilidades pueden incluir simuladores y dispositivos de medición de performance.
- D.7 Detección y corrección de errores.-** Este factor considera la habilidad del sistema

para mantener su integridad en el caso de afrontar fallas de software o base de datos. Procedimientos de corrección pueden ser automáticos o con la asistencia de personal. Se da una mayor calificación para una arquitectura que posea la capacidad inherente para soportar un alto grado de procedimientos de detección y corrección de errores.

#### **E.- Expansión**

La expansión de una arquitectura está relacionada con su capacidad de crecimiento en performance y capacidad de almacenar datos y su habilidad para agregar nuevas funciones o características.

##### **E.1 Esfuerzo requerido para sumar su capacidad.-**

Este factor se relaciona con la facilidad y costos con el cual una arquitectura puede ser expandida. Una alta calificación es dada para una alta estimación de la facilidad con la cual una arquitectura se acomoda al cambio.

##### **E.2 Esfuerzo para reemplazar componentes hardware con cambios de tecnología.-**

Este factor considera la extensión del tiempo de vida de la arquitectura por modificación apropiada usando los adelantos tecnológicos. Se asigna una alta calificación para una arquitectura que puede fácilmente acomodarse a los cambios

debido a los adelantos tecnológicos por reemplazo o mejora de los componentes hardware.

**E.3 Adición o mejora de procesadores.-** Este factor se relaciona con la facilidad por el cual la performance del sistema puede ser mejorado agregando procesadores o escalar a uno mejorado y más veloces dentro de la misma familia de procesadores. Una alta calificación para una arquitectura que puede fácilmente acomodar procesadores adicionales o más potentes.

**E.4 Capacidad del software** Este factor determina la habilidad del software para poder soportar un incremento de la carga de trabajo. Se asigna una alta calificación para una arquitectura que minimiza los ajustes en la memoria/base de datos por los cambios debidos al incremento.

**E.5 Esfuerzo requerido para agregar o retirar módulos software.-** Este factor se relaciona con la adaptabilidad de la arquitectura de referencia (básica) para lograr su adaptación a un ambiente particular. Tiene mayor calificación una arquitectura que minimiza los esfuerzos necesarios para agregar o restar funciones o características referente

a las capacidades básicas del sistema.

## **F.- Confiabilidad**

La confiabilidad del sistema es una compleja medición de la probabilidad que un sistema rendirá satisfactoriamente en un intervalo de tiempo dado, cuando es usado bajo condiciones estables. El concepto de confiabilidad incluye tiempos de operación, tiempos de caídas y efectividad del sistema. Los últimos conceptos también involucran tareas de confiabilidad, disposición del sistema y diseño adecuado del sistema. La confiabilidad del software involucrará veracidad y viabilidad. Veracidad es el grado por el cual el software representa un mundo real. Viabilidad es la adecuación y precisión con la que el software continúa de acuerdo a los requerimientos del sistema en situaciones inusuales o en el caso de errores por falla.

**F.1 Disponibilidad.-** Este factor considera la disponibilidad instantánea del sistema acordes satisfactoriamente con los requerimientos operacionales. La disponibilidad básica está definida para ser la relación del tiempo de operación con el tiempo total. Una alta calificación es dada para una arquitectura que asegure una alta disponibilidad.

- F.2 Tiempos de recuperación.-** Este factor relaciona la habilidad de la arquitectura para seleccionar una ruta alternativa en recuperación por un mal funcionamiento hardware. Una alta calificación para un menor tiempo de recuperación.
- F.3 Grado de redundancia.-** La disponibilidad de un sistema depende en su habilidad para trabajar alrededor de funciones defectuosas o pérdida de módulos. Con la presencia de elementos redundantes, la arquitectura incrementa la probabilidad de rutas disponibles. Una alta calificación para una arquitectura que posee un alto grado de redundancia en todos los módulos críticos y con lo que se evita situaciones de falla en un solo punto.
- F.4 Degradación.-** Este factor se relaciona con la habilidad del sistema para formular sus tareas ante el evento de una o más funciones defectuosas del hardware. Como las funciones defectuosas se acumulan, la performance del sistema puede deteriorarse a un punto donde la velocidad de ejecución de alguna tarea operacional puede ser reducida. Una alta calificación es asignada para una arquitectura que puede degradarse grandemente

y mantener sus funciones esenciales en caso de múltiples fallas.

**F.5 Detección y corrección de fallas.-** Este factor se relaciona con la habilidad del sistema para preservar su disponibilidad después de experimentar cierta clase de fallas. Esas fallas pueden ser detectadas y corregidas a nivel de un módulo local sin requerir un cambio en el control del sistema. Un alto puntaje para las arquitecturas que enfatizan grandemente la detección y corrección de fallas.

**F.6 Verificación y validación (v and v).-** Estos factores se relacionan con la habilidad para probar y garantizar que el software está operativo y se formulará satisfactoriamente dentro de la carga de trabajo esperada. Una alta puntuación para arquitecturas que faciliten el uso fuerte de procedimientos del V and V en el diseño e implementación del software.

**F.7 Tiempos de recuperación.-** Este factor relaciona la habilidad del sistema software para desviar el control del sistema y el flujo de datos ante un mal funcionamiento de un módulo. La recuperación es una función de un tiempo de carga para instalar la capacidad

en el módulo de reserva o redefinir las tablas del sistema. Una alta calificación para una arquitectura que minimiza los tiempos de recuperación.

**F.8 Efectividad bajo condiciones de carga inusuales.**- Este factor es complementario para alguno de los anteriores en el cual se relaciona la habilidad del sistema software para distribuir tareas durante condiciones de cargas inusuales en orden para mantener un nivel satisfactorio de performance. Una alta calificación se asigna a una arquitectura que está bien implementado para operar bajo condiciones de carga inusuales.

**F.9 Detección y corrección de errores.**- Este factor se relaciona con la habilidad de detectar y corregir fallas locales sin cambiar el control del sistema o flujo de datos. Una alta puntuación para una arquitectura que da énfasis a esta característica.

#### **G.- Costos**

Un sistema de costos provee una común referencia para comparar sistemas. Los costos resultan de un número de factores y un intento que deben ser hecho para considerarlos todos relevantes en costos de ciclos de vida en la evaluación de una

arquitectura.

- G.1 Costos de desarrollo.- Estos son costos gastados en el análisis de requerimientos, diseños y desarrollo del sistema hardware y software.
- G.2 Costos de implementación.- Estos costos consideran la codificación y pruebas del software en conjunto con las facilidades de pruebas.
- G.3 Costos de instalación.- Son los costos que involucran la integración del software y hardware en el área operativa, incluyendo el sistema completo de chequeo y pruebas de aceptación.
- G.4 Costos de mantenimiento (contrato en planta).- Estos son los costos de mantenimiento del hardware y software del sistema.
- G.5 Inventario partes de reserva.- Estos son los costos para desarrollar un suministro de partes comúnmente usados y fácilmente reemplazables en cada instalación. La disponibilidad de partes de reserva significa rápida recuperación y reparación ante el evento de una falla de hardware. El número de partes de reserva requerido dependerá del tamaño de la instalación, la arquitectura

seleccionada y la relativa necesidad de una reparación rápida.

**G.6 Conversión.-** Este factor costo considera las modificaciones de cualquier suministrador hardware o software necesario durante el desarrollo del sistema.

**G.7 Vida útil (depreciación y amortización).-** Este factor costo es usado para los criterios de vida útil esperados de la arquitectura versus futuros costos beneficios.

**G.8 Costos de entrenamiento.-** Este factor considera los costos de entrenamiento a los operadores en el sistema software y hardware combinados.

**G.9 Costos en cambios requeridos.-** Este factor considera el costo de la modificación del sistema y mejoramiento como requerimientos del cambio del sistema

**G.10 Costos de documentación.-** Este factor considera el costo de desarrollo y mantenimiento del sistema de documentación.

**G.11 Estudios para la generación del sistema.-** Algunos sistemas requieren una generación extensiva en el área de aplicación sitio por sitio, ejemplo unidades remotas de la Central.

**G.12 Costos de operación.-** Este factor considera

los costos de operación del sistema. como por ejemplo energía, facilidades del ambiente, etc.

**G.13 Personal.-** Involucra los costos del personal para la instalación.

## **H.- Perfomance**

La perfomance del sistema es la habilidad para complementar la carga de trabajo dada entre un tiempo designado obligatoriamente. La perfomance puede ser evaluada por componentes individuales así como para todo el sistema.

**H.1 Factores hardware.-** Los componentes hardware de la arquitectura deben ser evaluados en el contexto del sistema global. En la definición de la arquitectura, factores individuales tales como los siguiente deben ser tomados en consideración:

Respuesta para las interrupciones

Velocidad de la unidad del procesador central.

Velocidad del disco

Ancho de banda del bus de interconexión

Memoria principal requerida y la máxima disponible

Almacenamiento masivo

Utilización de subsistemas

Esos factores deben ser pesados conjuntamente

para estimar su impacto en toda la arquitectura del sistema. La transacción entre el tiempo de procesamiento y la memoria debe ser cuidadosamente considerado en el contexto de la utilización del subsistema. El impacto de la carga pico en la performance del sistema es crucial para la determinación de excesos de capacidad y su ubicación en la arquitectura. En resumen, un balance debe llevarse a cabo entre los factores de performance entre un arquitectura candidato que no hay incrementos de costos significativos.

**H.2 Factores de software.**- De manera similar que el hardware, un balance debe efectuarse entre los diferentes factores de performance para el software que no incrementa significativamente el costo del sistema. La evaluación de esos factores es muy subjetiva porque los detalles de diseño todavía no han sido formulados. Entre los factores a ser evaluados están:

Tiempos de respuesta

Procesamientos de sobrecarga

Recuperación de la base de datos

Carga de las comunicaciones

Procesamiento de fondo tiempo/carga

Capacidad en exceso

Tiempo de la transformación del dato

Eficiencia del sistema

Fácil operabilidad

## CAPITULO IV

### DISEÑO DEL PROTOTIPO

En los capítulos anteriores se formularon los lineamientos del diseño del Sistema de la Comunicación entre los Procesadores de la Central, basándonos en el modelo ISA, la cual nos permitió definir funciones básicas y especificaciones preliminares; en el presente capítulo se diseñará un prototipo y constará de tres partes:

- Diseño del Sistema.
- Diseño Hardware.
- Diseño Software.

#### **4.1 Diseño del sistema.**

El ciclo de vida de un sistema ha sido definido para tener cuatro fases: definición del problema, diseño del sistema, implementación del sistema y operación del sistema. Debemos seguir una metodología apropiada para diseñar sistemas con múltiples procesadores y su comunicación. Esta metodología presupone una buena definición del problema y especificaciones, el cual llamaremos requerimientos de especificaciones.

El modelo debe considerar el diseño de sistemas

distribuidos. Cuya arquitectura es evaluada para varios modelos de tráfico representando sus características propias, definidas primitivamente en el capítulo anterior.

En el proceso del diseño del sistema se trataran los siguientes aspectos:

Especificaciones Generales

Modelo de Tráfico

Algoritmos

Pruebas de integración software y hardware

#### **4.1.1 Especificación de requerimientos funcionales.-**

Quizás la porción más ignorada en el desarrollo de sistemas de multiprocesadores, es la especificación de los requerimientos. Muchos sistemas desarrollados son caracterizados por su pobre definición de requerimientos.

Para las especificaciones, tomamos como base del diseño del sistema el modelo del sistema de Señalización de Canal Común, recomendados por el CCITT (Q.700) y las consideraciones de interconexión de sistemas abiertos ya especificados en el capítulo III sección 3.1.2; nuestro análisis se centrará, específicamente en los niveles 3 y 2, debido a que el nivel 1 está plenamente identificado y especificado, con respecto a los niveles superiores, dependerá del diseño particular y final del sistema.

La Fig. 4.1 ilustra las funciones de tratamiento de los mensajes entre los procesadores, que en el CCITT se conoce como Parte de Transferencia de Mensajes (PTM) y

abarca los niveles 1, 2 y 3 del ISA; como se puede apreciar solamente estamos analizando el sistema para un solo procesador, la cual será repetitiva en los otros procesadores sin importar, en estos niveles, su jerarquía. Se definen los bloques funcionales generales pertenecientes a los niveles 1, 2 y 3 ISA.

#### **4.1.1.1 Especificaciones del nivel 3.**

Cada mensaje debe contener una etiqueta de direccionamiento, que será usado para el enrutamiento del mensaje, que debe incluir:

Indicaciones explícitas del procesador destino y procesador origen del mensaje.

Un código adicional que permita identificar usuarios internos en cada procesador que vienen a ser procesos.

El sistema debe establecer un plan de codificación, a través de etiquetas, para el enrutamiento adecuado de los mensajes.

**A.- Tratamiento de los mensajes.-** Tiene como bloques funcionales

**A.1 Encaminamiento de mensajes.-** Es el proceso de selección por el cual el mensaje es enviado al procesador destino (usuario). Estos mensajes pueden venir del nivel 4 o del Discriminador de Mensajes.

**A.2 Distribución de mensajes.-** Es el proceso mediante el cual si un mensaje es recibido se

distribuye a la parte usuario correspondiente, esta distribución la efectúa analizando la etiqueta de encaminamiento.

**A.3 Discriminación de mensajes.-** Este proceso permite discriminar si el mensaje recibido es para este procesador o para otro, esta decisión está basada en el análisis de la etiqueta de encaminamiento. Si pertenece al procesador, el mensaje es derivado al bloque Distribución de Mensajes, de lo contrario va al bloque de Encaminamiento de Mensajes para su retransmisión.

**B.- Gestión de la red.-** Se especifican los siguientes bloques funcionales:

**B.1 Gestión del tráfico.-** Tiene como tareas la del control del Encaminamiento de Mensajes, control del flujo, esto es posible gracias a tipos de mensajes recibidos que informan del estado de la red y también generan información del estado del tráfico.

**B.2 Gestión de enlaces.-** Permite conocer el estado de las rutas e informa a los otros módulos sobre la disponibilidad de rutas o estados anormales.

**B.3 Gestión de rutas.-** El sistema puede considerar rutas alternativas de la comunicación entre los procesadores, para lo cual este módulo puede

gestionar rutas emergentes si se requieren.

- C.- **Pruebas y mantenimiento.**- El sistema debe incluir algunas funciones de pruebas y mantenimiento haciendo uso de los mensajes del nivel 3, para garantizar la operación del sistema.

Otra especificación importante del PTM es **la capacidad de transferir mensajes**, en la cual se especifica los métodos para establecer una transferencia de mensajes.

Los requerimientos para la transferencia tiene que ser determinado primeramente por los requerimientos del control de llamadas (call processing) de la central telefónica. Sin embargo el PTM puede servir para transportar otros tipos de mensajes, como información de servicios y otros. Toda la información a ser transferida debe ser ensamblada en formato de mensaje, que debe incluir lo siguiente:

**Transparencia de código.**- La información con cualquier combinación de código que genere el usuario debe ser transparente para la PTM.

**Información de servicio.**- Cada mensaje debe contener un código de información de servicio.

**Etiqueta del mensaje.**- Como lo mencionáramos anteriormente cada mensaje debe contener una etiqueta para funciones de ruteo y encaminamiento.

**Longitud del mensaje.**- La información debe tener un número entero de octetos (bytes). La cantidad de información transferible en un mensaje está limitado

por algunos parámetros del sistema. De acuerdo a un análisis previo, generalmente un mensaje entre procesadores de la central, para el procesamiento de llamadas; está en el rango de 16 a 32 octetos.

#### 4.1.1.2 Especificaciones del nivel 2.

Los mensajes entregados por los niveles jerárquicos superiores son transferidos por el enlace mediante unidades de mensajes de longitud variable. Las unidades de mensaje contienen además de la información entre procesadores, información de control de transferencia, para asegurar el funcionamiento del enlace.

En la Fig. 4.2 se muestran las funciones de la parte control del estado del enlace y cuyas funciones son:

- A. **Delimitación y alineamiento de mensajes.**- El inicio y fin de una unidad de mensaje está indicado por un único octeto llamado BANDERA (flag), debe asegurarse que este octeto no sea imitado dentro del mensaje. Las pérdidas de alineamiento del enlace puede ocurrir durante la operación, se deben establecer procedimientos para estos eventos.
- B. **Detección de errores.**- La función detección de errores es efectuada por 16 bits de chequeo provista al final de cada unidad de mensaje. Los bits de chequeo son generados por el enlace transmisor empleando un algoritmo especificado que involucra todos los bits de información

precedentes. En el lado receptor también se efectúa la misma operación, aplicando el mismo algoritmo, con los octetos de información obteniendo los 16 bits de chequeo, que son comparados con los que se recibieron, si hay consistencia, los octetos de información están libres de errores.

- C. **Corrección de errores.**- La corrección de errores básica se efectuará mediante el método no obligado de confirmación. Esta corrección de errores se efectúa mediante el control secuencial de número de mensajes transmitidos y recibidos, cuando el mensaje ha sido transmitido correctamente el colateral dará una confirmación positiva del número de mensaje recibido; de lo contrario enviará una confirmación negativa, que obligará a la retransmisión.

Otro método es la retransmisión preventiva de los mensajes. Cada unidad de mensaje debe ser retenido en el terminal del enlace hasta que llegue un positivo reconocimiento. En este procedimiento también se analizan los numeros secuenciales de mensajes. Ambos métodos tienen un protocolo de procedimientos establecidos.

- D. **Alineación inicial.**- Se deben establecer procedimientos para la inicialización y alineamiento del sistema, cuando se inicia su

operación. El procedimiento es basado en un intercambio obligado de estados del enlace y procesadores asociados.

**E. Supervisión de errores del enlace.**- Se debe estar contabilizando los problemas del enlace, tales como pérdida de alineamiento, excesivos datos errados, repetición de mensajes; para efectuar operaciones de mantenimiento oportuno.

**F. Control de flujo.**- El control de flujo es iniciado cuando es detectado la congestión del enlace, esta situación es notificada al lado transmisor, para que se tomen las precauciones del caso.

#### 4.1.1.3 **Unidades de mensajes.**

Serán necesarios tres tipos de mensajes para el intercambio de información entre los procesadores, a saber:

**Mensajes de información**

**Mensajes del estado del enlace**

**Mensajes de relleno**

En la Fig. 4 3 se muestran los formatos de las tres unidades de mensajes y obedecen a la siguiente descripción:

**A. Bandera (BAN).**- La bandera de apertura indica el comienzo de la unidad del mensaje y la bandera al final del mensaje indica su terminación. La configuración de bits es 01111110.

- B. **Indicador de longitud (IL).**- Sirve para indicar el número de octetos que siguen al presente y preceden a los bits de chequeo.
- C. **Octeto de información de servicio (OIS).**- Este octeto se divide en indicador de servicio y campo de subservicio. El campo indicador de servicio servirá para asociar la información con un determinado usuario (proceso).
- D. **Números secuenciales directo e inverso (NSD y NSI).**- El NSD es el número secuencial del mensaje que está siendo enviado y el NSI es el número secuencial de mensaje que ha sido confirmado.
- E. **Bits indicadores directo e inverso (BID y BII).**- Los BID y BII conjuntamente con los NSD y NSI son usados para el método de control de errores básicos para efectuar los controles de la secuencia de mensajes y las funciones de reconocimientos.
- F. **Campo de información de mensajes (CIM).**- Consiste en un número entero de octetos, que contienen información de los usuarios; como mínimo 2 octetos y menor o igual a 32. Este último valor puede ser extendido de acuerdo a necesidades.
- G. **Campo de estados del enlace (CEE).**- Empleado en la unidad de mensaje de estado del enlace, estas informaciones empleadas para iniciar la operación del sistema o recuperarlo de una falla, se

definen los siguientes estados:

- Estado "O" Fuera de alineamiento
- Estado "N" Alineamiento Normal
- Estado "E" Alineamiento de Emergencia
- Estado "OS" - Fuera de servicio.

H. **Bits de chequeo de errores (BCE).**- Son los 16 bits (2 octetos) anteriormente descritos, como CRC.

#### 4.1.2 Modelo de tráfico.

En los acápites anteriores se han introducido conceptos de gestión de tráfico y rutas y son aplicados al destino y origen de los mensajes. Ahora nuestro propósito es establecer un modelo de tráfico considerado a nivel de sistema. La Fig. 4.4 nos ilustra un modelo de tráfico general que considera las limitaciones de los aspectos físicos de la construcción del sistema. El tráfico espurio es todo tráfico producido por ocupaciones inadmisibles en los equipos, debido a la limitación del propio sistema, mala calidad de la transmisión, congestión de tráfico y otros factores.

El tráfico ofrecido perdido, se refiere a la posibilidad que puedan faltar órganos de comunicación, en las horas pico de tráfico y el cursado no efectivo a posibles cambios de direcciones, congestión en las rutas u otras causas. Todas estas causas generan un tráfico no ofrecido por el sistema y son reintentos que deben ser contemplados.

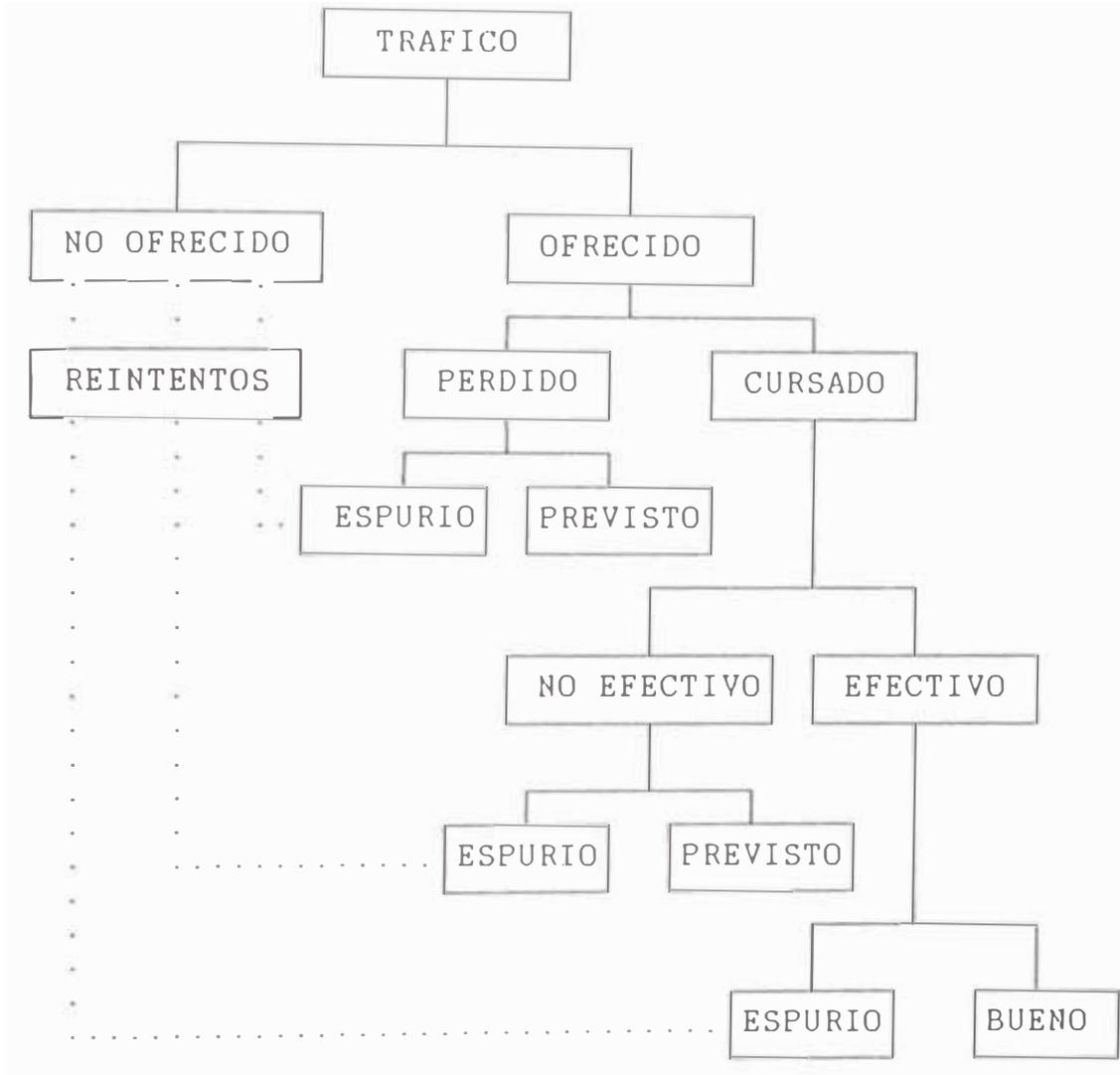


Fig 4.4

MODELO DE TRAFICO

### 4.1.3 Algoritmos básicos.

Ahora bosquejaremos los algoritmos básicos para cada nivel anteriormente analizado. Para un diseño más efectivo, es preferible analizar diagramas que describan las funciones, considerando temporizaciones.

Antes de bosquejar, previamente describiremos los símbolos empleados en los diagramas de flujo de los algoritmos:

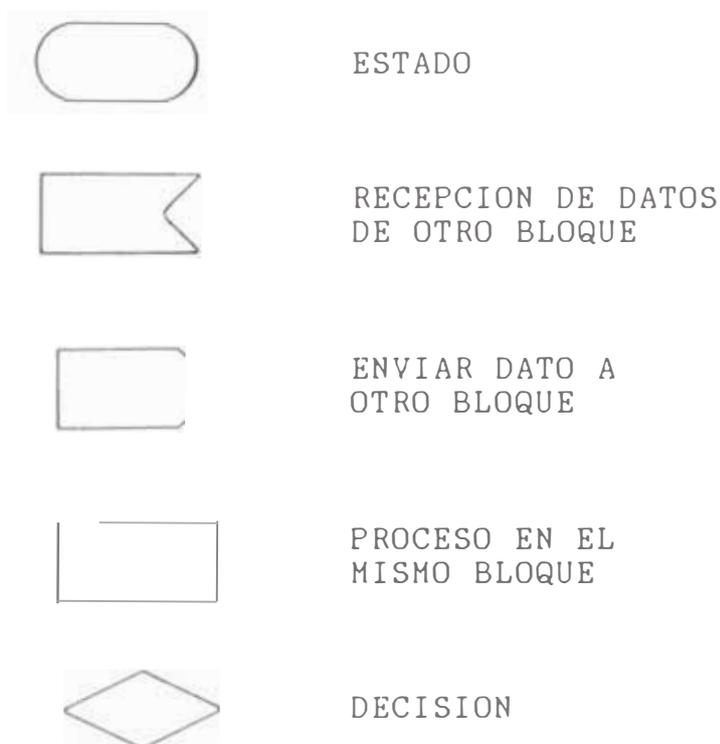


FIG. 4.5  
DESCRIPCION DE SIMBOLOS

#### 4.1.3.1 Algoritmos del nivel 2.

Los procesos ejecutados por cada bloque funcional de este nivel son:

- A.- Control de alineamiento inicial.-** Controla el alineamiento inicial. Este procedimiento es aplicable para ambos casos de inicialización: después del power-on y el recupero del sistema después de una falla.
- B.- Delimitación, alineamiento y detección de error.-** Es el bloque de interfase entre el nivel 1 y el 2, recibe el mensaje del bloque Parte Control de Transmisión para el procesador remoto. Después de efectuar la inserción de ceros, adjuntar los bits de chequeo y banderas, es transmitido al nivel 1. De manera similar para la parte de recepción la unidad de mensaje recibida de otro procesador, es chequeada para su validación (después de retirar los ceros insertados, verificación con los bits de chequeo y banderas), si es válida se envía al nivel 3.
- C.- Parte control de transmisión.-** Tiene las facilidades de control de transmisión. Determina el tipo de mensaje. Si la UM (unidad de mensaje general) es un UDM (unidad de mensaje entre procesadores y no del sistema) lo tomará del buffer de transmisión, al mismo tiempo, este mensaje, se debe almacenar en el buffer de retransmisión.
- D.- Parte control de recepción.-** Este bloque recibe la UM , ya libre de errores y procede a chequear los números secuenciales inverso y directo, así como los

bits indicadores (NSI, NSD, BID, BII), si es correcto el mensaje es pasado al nivel 3, caso contrario es descartado avisando el rechazo.

**E.- Control del estado del enlace.-** En los diagramas se muestran diferentes situaciones de control de enlaces, desde caídas de procesadores locales y remotos hasta congestiones de red y alineamientos.

En las figuras 4.6 al 4.11, se ilustran los diagramas de flujo básicos de los bloques funcionales descritos.

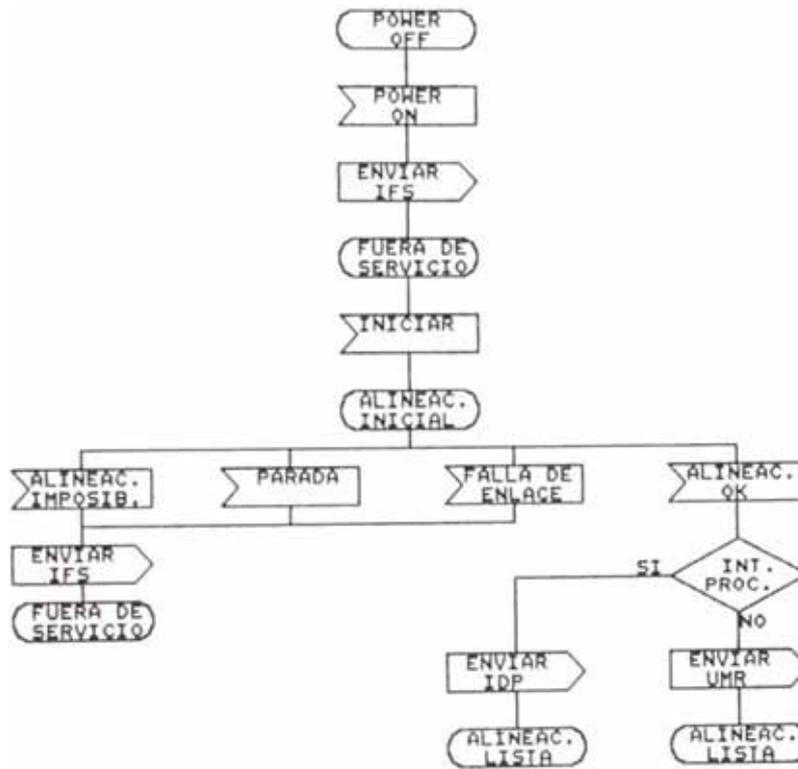
#### 4.1.3.2 Algoritmos del nivel 3.

Las funciones del nivel 3 incluyen el tratamiento de los mensajes, la administración de la red y facilidades de prueba y mantenimiento. En la Fig. 4.12 se ilustra los diagramas de flujo básicos de las funciones principales, en referencia a las otras requieren de un análisis similar, que en la presente tesis no la tocaremos en forma profunda.

**A.- Tratamiento de mensajes.-** Es dividido en dos importantes funciones:

**Discriminación y distribución de mensajes.-** Estas a su vez tienen las siguientes funciones: Examinar las etiquetas de encaminamiento para determinar si la UDM (unidad de mensaje) es para el procesador o no. Examina el octeto Servicio de Información para identificar la parte usuaria y entregar su mensaje.

**Encaminamiento de mensajes.-** Tiene las funciones siguientes: Determinar el posible enlace para llegar



IFS : INDICACION FUERA DE SERVICIO  
 IDP : INDICACION DE ESTADO 'INTERRUPCION DEL PROCESADOR'  
 UDM : UNIDAD DE MENSAJE  
 UMR : UNIDAD DE MENSAJE DE RELLENO  
 IEE : INDICACION DE ESTADO 'EMERGENCIA'  
 IEN : INDICACION DE ESTADO 'ALINEACION NORMAL'  
 IEF : INDICACION DE ESTADO 'FUERA DE SERVICIO'

FIG. 4.6  
 DIAGRAMA DE FLUJO DEL CONTROL DE ESTADO DEL ENLACE  
 ALINEACION INICIAL

al destino, chequear el enlace seleccionado, transferir el mensaje al buffer del nivel 2.

#### **4.2 Diseño hardware.**

Dada la existencia de las especificaciones, se aprecia que el diseño del hardware se centra en los niveles 1 y 2; la metodología elegida usa una convencional aproximación para diseñar la solución de un único procesador y sus interfases para el cumplimiento de los requerimientos funcionales para estos niveles. El resultado de este esfuerzo será la base para entender a los otros procesadores de las diferentes jerarquías que necesiten comunicación.

Resumiendo, los bloques funcionales en sus respectivos niveles 1 y 2 tendremos:

**Nivel 1      Enlace físico de datos      Define las**  
características físicas, eléctricas y funcionales del  
enlace de datos y su acceso. Este nivel proporciona  
los servicios de portadora a la comunicación entre los  
procesadores.

##### **Funciones específicas:**

- \* En el ambiente de Centrales Digitales el medio de transporte usado es el canal PCM de sistema de primer orden 64 Kbits/s.
- \* El acceso se realizará a través de funciones de conmutación temporal.
- \* El uso de las funciones de conmutación permitirá

reconfiguraciones del sistema.

- **Nivel 2 - Enlace de datos.**- Define las funciones y procedimientos relacionados a la transferencia de mensajes entre los procesadores , sobre un enlace físico individual. Las funciones del nivel 2 conjuntamente con las del nivel 1 proporcionan una portadora para la transferencia de mensajes confiables entre dos procesadores.

**Funciones específicas:**

- \* Delimitación de las unidades de mensajes mediante banderas.
- \* Prevenir la imitación de las banderas usando la técnica del inserción de bits (bit stuffing).
- \* Detección de errores chequeando los bits de cada unidad de mensajes.
- \* Corrección de errores mediante la retransmisión de unidad de mensajes.
- \* Control de la secuencia de mensajes y su respectivo reconocimiento por el destino.
- \* Detección de las condiciones de falla del enlace y su monitoreo.

En el Diagrama de Bloques de la Fig. 2.1 (capítulo II), se muestra la arquitectura de una central digital indicando todos sus módulos funcionales las cuales están controlados por procesadores y que la presente tesis propone el uso de controladores HDLC para la comunicación entre ellos, para el cumplimiento corporativo de las

funciones del sistema.

La base fundamental del presente diseño es elegir una adecuada arquitectura en la cual se debe incluir un controlador de protocolo HDLC, para lo cual se ha realizado una investigación entre los principales fabricantes de estos circuitos integrados especializados, con posibilidades de disponer en el mercado nacional:

Intel Corporation

Mitel Semiconductor

National Semiconductor Corporation

Texas Instrument

Motorola y otros

Después de realizar una previa evaluación considerando los criterios de selección, mencionados en el Capítulo III y a las consideraciones técnicas de nuestras especificaciones para este diseño particular, precalificaron los siguientes componentes:

Controladores de protocolo HDLC MT8952B

Fabricante : Mitel Semiconductor

Controladores programables de HDLC/SDLC 8273

Fabricante : Intel Corporation

Microcontroladores de alta performance con controladores HDLC HPC16400/HPC36400/HPC 46400

Fabricante : National Semiconductor Corporation

De acuerdo a la definición de los criterios de selección mencionados anteriormente se obtiene la Tabla

4.1 para las arquitecturas basadas con los controladores precalificados.

Del análisis de la Tabla 4.1 se determina las Figuras de Mérito (FDM) de acuerdo a las fórmulas planteadas en el Capítulo III, la arquitectura ganadora, para la comunicación entre los procesadores, es la basada en el controlador de protocolo HDLC MT8089B de Mitel Semiconductor.

Antes de estudiar minuciosamente el controlador seleccionado, detallaremos el diseño del nivel 1, para un procesador. En la Fig. 4.13 se muestra la interconexión de las interfases paralelas entre el procesador, la matriz de conmutación y el controlador, de la misma manera las interconexiones con las interfases seriales PCM entre la matriz y el controlador. Esta interconexión es versátil para cada procesador porque permite flexibilidad en la configuración del sistema y el controlador puede utilizar cualquier canal PCM disponible, no necesariamente el canal 16 de la trama, y cualquier canal principal (high way), para comunicarse con otros procesadores a través de las vías de comunicación. Este detalle se muestra en la Fig 4.14. Con referencia a la matriz de conmutación puede ser cualquier tipo de cross point digital, que permita la conmutación digital de canales PCM de primer orden, en nuestro caso nuestra implementación se basará en la matriz MT8980 de Mitel Semiconductor, cuya información técnica se adjunta

CRITERIOS	MTEL MT8952			INTEL 8373			NACIONAL HPC16400		
	HM	SM	PARCIAL	HM	SM	PARCIAL	HM	SM	PARCIAL
SIMPLICIDAD	35	55	13.50	40	60	15.00	20	40	8.00
FACTIBILIDAD	70	25	14.25	50	20	10.50	40	15	8.25
MODULARIDAD	30	50	12.00	25	50	11.25	35	60	14.25
MANTENIMIENTO	20	50	3.50	25	65	4.50	30	70	5.00
EXPANSION	30	50	8.00	20	40	6.00	40	60	10.00
CONFIABILIDAD	35	55	9.00	35	55	9.00	25	50	7.50
COSTOS	25	70	19.00	20	60	16.00	10	40	10.00
PERFORMANCE	50	30	8.00	50	30	8.00	55	35	9.00
FIGURA DE MERTO			87.25			80.25			73.00

TABLA 4.1

CUADRO COMPARATIVO DE COMPONENTES

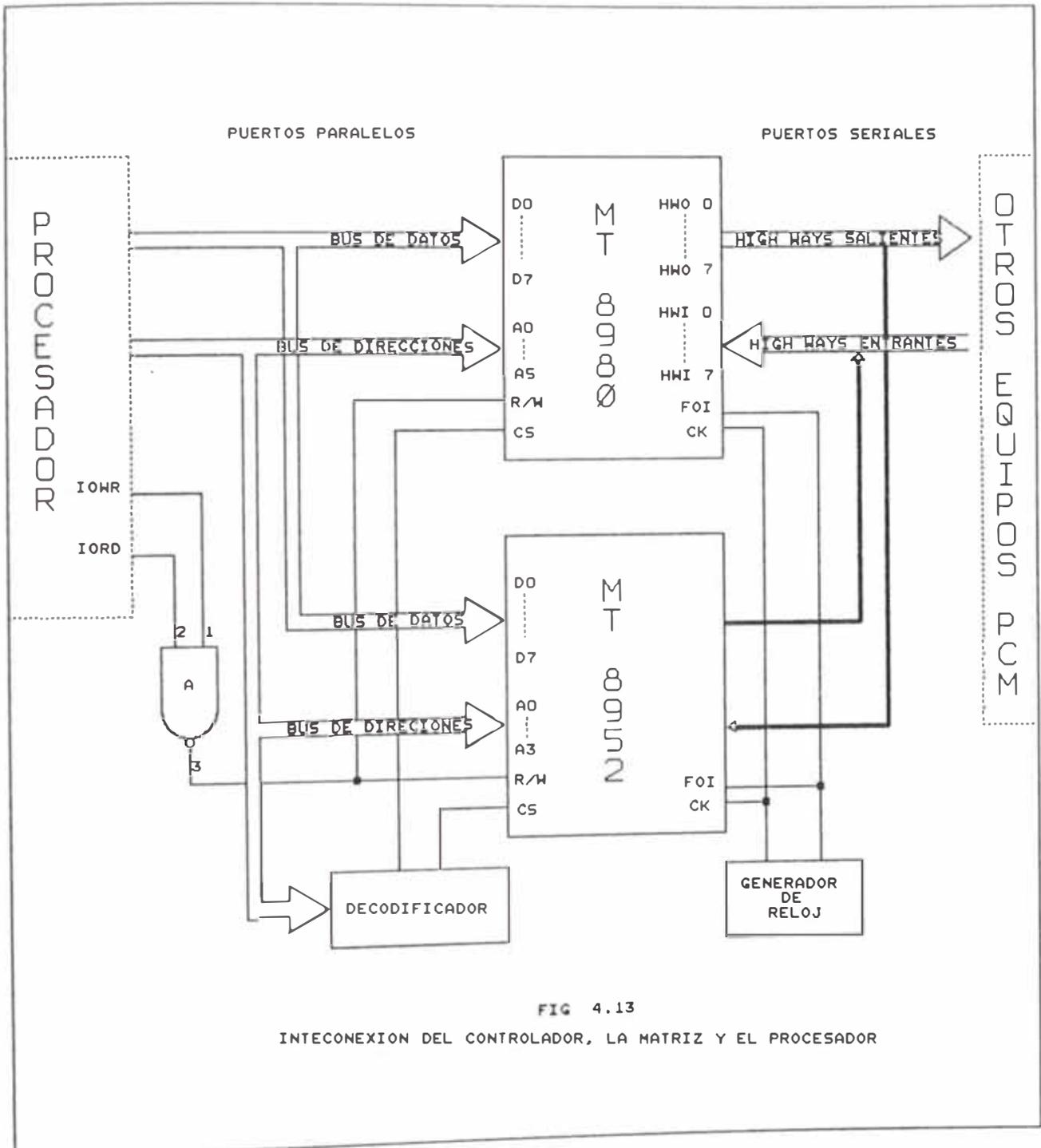


FIG 4.13

INTECONEXION DEL CONTROLADOR, LA MATRIZ Y EL PROCESADOR

en los Anexos.

#### 4.2.1 Descripción del controlador de protocolo HDLC MT8952B.

En esta sección describiremos en detalle las características hardware y software del controlador, analizando si cumplen con las especificaciones de los niveles 1 y 2 de nuestro sistema en proceso de diseño.

##### 4.2.1.1 Introducción

El controlador de protocolo HDLC MT8952B formatea paquetes de datos de acuerdo a la recomendación X.25 del CCITT. Tiene habilitadores separados para las secciones de transmisión y recepción y capacidad de detección de dirección de un solo byte, el controlador de protocolo puede enviar y recibir los paquetes selectivamente.

El controlador de protocolo HDLC MT8952B manipula una estructura de protocolo orientado a bit y formatea los datos según el protocolo de conmutación por paquetes definido en la recomendación X.25 (nivel 2) del CCITT. Este transmite y recibe el paquete de datos (información y control) serialmente en el siguiente formato:

FLAG	CAMPO DE DATOS	FCS	FLAG
1 BYTE	n BYTES (n>2)	2 BYTES	1 BYTE

Provee transparencia de datos insertando y borrando ceros. Genera y detecta los flags, varios estados del canal de enlace y la secuencia de aborto. Además, provee

chequeo de redundancia cíclica (CRC) del paquete de datos usando el polinomio definido por el CCITT. Puede generar y detectar la secuencia de Go Ahead y reconocer una dirección de 1 byte en la trama recepcionada. También es posible inhabilitar las funciones de protocolo y proveer acceso transparente al bus serial a través del puerto paralelo.

**Formato de trama.**- Todas las tramas empiezan con una bandera de apertura y terminan con una bandera de cierre. Entre estas dos banderas, una trama contiene el dato y la secuencia de chequeo de trama (FCS).

\* **Bandera.**- La bandera es un patrón único de 8 bits (01111110) definiendo los límites de trama. La sección de transmisión genera las banderas y los adiciona automáticamente a la trama a ser transmitida. La sección de recepción busca banderas en los paquetes entrantes bit por bit y establece la sincronización de trama.

\* **Datos.**- El campo de datos se refiere a los campos de dirección, control e información definidos en la recomendación del CCITT. Una trama válida debe tener un campo de datos de al menos 16 bits. El primer byte en el campo de datos es la dirección de la trama.

\* **Secuencia de chequeo de trama (FCS).**- Los 16 bits siguientes al campo de datos son los bits de secuencia de chequeo de trama. El polinomio

generador es:

$$G(x) = x^{16} + x^{12} + x^5 + 1$$

El transmisor calcula el FCS de todos los bits del campo de datos y los transmite después del campo de datos y antes de la bandera final. El receptor realiza un cómputo similar de todos los bits recepcionados en los campos de dato y FCS y lo compara con FOB8H. Si son iguales, se asume que el dato recibido esta libre de errores.

- \* **Inserción y borrado de ceros.**- El controlador de protocolo, mientras envía datos desde el FIFO o los 16 bits FCS, chequea la transmisión bit a bit e inserta un CERO después de cada secuencia de cinco UNOs continuos (incluidos los cinco últimos bits del FCS) para asegurar que la secuencia de bandera no es simulada. Igualmente el receptor examina el contenido de la trama entrante y descarta algún CERO siguiendo a cinco UNOs continuos.

**Particularidades de operación del controlador de protocolo.**- Estas particularidades la podemos aprovechar en las funciones control del enlace.

- \* **Aborto.**- El transmisor aborta una trama enviando ocho UNOs consecutivos. El bit FA en el Registro de Control junto con una operación de escritura

al Registro de Dato a Transmitir habilita la transmisión de la secuencia de aborto en vez del byte escrito al registro. En el lado receptor, un aborto de trama es definido como siete o más UNOs continuos que ocurren después de la bandera de inicio y antes de la bandera final del paquete. Una interrupción puede ser generada con la recepción de una secuencia de aborto usando el bit FA del Registro de Interrupciones.

- \* **Llenado del tiempo entre tramas y estados del canal de enlace.**- Cuando el controlador de protocolo HDLC no está enviando paquetes, el transmisor puede estar en cualquiera de los tres estados mencionados abajo dependiendo de la condición de los bits IFTFO y IFTF1 en el Registro de Control. Estos bits son usados también para inhabilitar la función de protocolo y proveer el acceso transparente al bus serial a través del puerto del microprocesador.
- \* **Estado inactivo.**- El estado inactivo es definido como 15 o mas UNOs consecutivos. Cuando el controlador de protocolo HDLC está observando esta condición en el canal receptor, el bit de inactivo en el Registro General de Estados es puesto en ALTO. En el lado transmisor, el controlador de protocolo termina el estado inactivo cuando un dato es cargado en el FIFO de

transmisión.

- \* **Estado de llenado del tiempo entre tramas.-** El controlador de protocolo **envía** banderas consecutivos (7EH) en el estado de llenado del tiempo entre tramas y termina este estado cuando un dato es cargado en el FIFO de transmisión.
- \* **Estado go ahead.-** Go Ahead es definido por la secuencia de 9 bits 011111110 (7FH seguido por un CERO), y por lo tanto 7F's consecutivos aparecen como Go Aheads. Una vez que el transmisor está en el estado 'Go Ahead' continuará así aun después que un dato es cargado en el FIFO. Este estado sólo puede cambiarse seteando los bits IFTF en el Registro de Control a algún estado diferente a 'Go Ahead'. La recepción de esta secuencia es indicada por el bit GA del Registro General de Estados y el controlador de protocolo puede generar una interrupción si se habilita el bit GA del Registro Habilitador de Interrupciones.
- \* **Estado de transferencia transparente de datos.-** El controlador de protocolo, en este estado, inhabilita las funciones de protocolo definidas anteriormente y provee acceso bidireccional al bus serial a través del puerto paralelo. Tal como los otros estados, este puede ser seleccionado en ambos modos de temporización.

- \* **Tramas inválidas.**- Alguna trama de menos de 32 bits entre banderas de apertura y cierre (correspondiente a 16 bits de dato y 16 bits de FCS) es considerada inválida. El controlador de protocolo ignora la trama sólo si la longitud de trama entre banderas es menor que 24 bits. Para tramas de longitud entre 24 y 32 bits, transfiere el campo de datos al FIFO e indica FCS errado en el Registro de Estado del FIFO.

#### 4.2.1.2 Descripción funcional.

El diagrama funcional de bloques del controlador de protocolo HDLC se muestra en la Fig. 4.15. Este tiene dos puertos. El puerto serial transmite y recibe paquetes de datos formateados y el puerto paralelo provee una interfase al microprocesador para acceder a los registros del controlador de protocolo.

El puerto serial puede ser configurado para operar en dos modos dependiendo del bit IC en el Registro de Control de Temporización. Este puede transmitir/recibir paquetes en los canales seleccionados en Formato-PCM o puede, usando las señales de habilitación (TxCEN y RxCEN), transmitir/recibir paquetes a una velocidad igual a la del reloj de entrada CKi.

El puerto del microprocesador permite transferencia de datos paralela entre el controlador de protocolo y el bus del sistema. Esta interfase consiste del bus de datos (D0-D7), bus de direcciones (A0-A3), reloj E, chip select

(CS) y control R/W.

El microprocesador puede leer y escribir todos los registros del controlador de protocolo. La dirección de estos registros son:

BITS DE DIRECCION				REGISTROS	
A3	A2	A1	A0	LECTURA	ESCRITURA
0	0	0	0	Estado del FIFO	-
0	0	0	1	Dato recibido	Dato a transmitir
0	0	1	0	Control	Control
0	0	1	1	Dirección de recepción	Dirección de recepción
0	1	0	0	Control canal C (Tx)	Control canal C (Tx)
0	1	0	1	Control temporización	Control temporización
0	1	1	0	Flag de interrupción	Temporizador Watch-Dog
0	1	1	1	Habilita interrupción	Habilita interrupción
	0	0	0	General de Estado	-
1	0	0	1	Estado canal C (Rx)	-

La salida IRQ es de drenador abierto, activo BAJO indicando una petición de interrupción al CPU. El monitoreo y control de las interrupciones que puede originar el controlador de protocolo es implementado por el Registro Flag de Interrupciones (IFR) y el Registro Habilitador de Interrupciones (IER). Eventos específicos ponen en ALTO un bit en el Registro Flag de Interrupciones. Tal evento no necesariamente interrumpe al CPU. Para emitir una interrupción (llevar la salida IRQ a BAJO) el bit en IER que coincide con el Registro Flag de Interrupciones debe estar en ALTO. El bit IRQ del Registro General de Estados es el complemento del estado del pin IRQ. Si una interrupción es emitida, este bit se pondrá en ALTO de lo contrario será BAJO.

**Salidas TEOP y REOP.**— El controlador de protocolo HDLC provee dos señales separadas TEOP y REOP que indican el final del paquete transmitido y recibido respectivamente. TEOP es un pulso ALTO de una duración de un bit emitido durante el último bit de la bandera final o de una secuencia de aborto del paquete transmitido. REOP es también un pulso ALTO de una duración de un bit emitido cuando una bandera final es recibido o un paquete entrante es abortado o un paquete inválido de 24 o más bits es detectado. Sin embargo, REOP no es generado para paquetes inválidos de longitud menor que 24 bits. Estas señales de 'fin de paquete' son útiles para multiplexar varios enlaces de datos a un único controlador de protocolo HDLC.

**Modos de temporización.**— El controlador de protocolo tiene dos modos de temporización. Estos modos de temporización se refieren sólo a la configuración del puerto serial y no están relacionados con el puerto paralelo del microprocesador.

\* **Modo de temporización interna.**— El modo de temporización interna es utilizado para aplicaciones con varios productos que usan la arquitectura formato-PCM. Los datos/paquetes son desplazados dentro/fuera serialmente en formato-PCM usando las señales de temporización FOi y C2i/C4i. Además el controlador de protocolo

reserva un canal (canal-1) del Formato-PCM para información de control (canal-C) y este canal no puede ser usado por el dato empaquetado. Mientras el controlador de protocolo está en el modo de temporización interna, la entrada de reloj CKi puede ser igual a la velocidad de Tx/Rx o 2 veces la velocidad de Tx/Rx dependiendo del bit BRCK del Registro de Control de Temporización como se muestra a continuación:

Bit BRCK	Entrada CKi	Velocidad de Tx/Rx
0	4.096 Mhz-C4i 2.048 Mhz-C2i	2.048 Mbps 2.048 Mbps

El controlador de protocolo usa las señales de temporización del formato-PCM FOi y C2i/C4i, y habilita las secciones de transmisión y recepción en los canales apropiados según la programación de los bits TCO-TC3 del Registro de Control de Temporización. Las entradas TxCEN y RxCEN son ignoradas en este modo.

- \* **Modo de temporización externa.**- En el modo de temporización externa, las secciones de transmisión y recepción son habilitados independientemente por las entradas de control TxCEN y RxCEN y los paquetes de datos formateados son desplazados dentro/fuera serialmente a una velocidad igual a la frecuencia del reloj en CKi.

La salida es transmitida en el flanco de subida y el receptor muestrea en el flanco de bajada del reloj. Los controles TxCEN y RxCEN son independientes y asíncronos y tienen efecto solo después que el bit actual en el paquete es transmitido/recibido.

Mientras que el controlador de protocolo provee el paquete de datos en un número limitado de canales en el formato-PCM cuando opera en el modo de temporización interna, este puede enviar el paquete de datos en cualquiera o en todos los canales del formato-PCM si opera en el modo de temporización externa con señales de habilitación apropiadas en TxCEN y RxCEN.

**Transferencia de datos transparente.**- Seteando los bits IFTF apropiadamente en el Registro de Control, las funciones de protocolo pueden ser inhabilitadas. Esto provee un acceso bi-direccional al puerto serial a través de la interfase al microprocesador, con 19 bytes de FIFO en cada dirección. La transferencia de datos transparente facilita funciones en formatos de byte ancho y es disponible en ambos modos de temporización excepto cuando los bits de control de temporización son seteados para 1 bit/trama durante el modo de temporización interna. El dato a transmitir es desplazado fuera serialmente en CDSTo y sólo los bits menos significativos del

byte cargado son transmitidos si los bits de control de temporización son seteados para seleccionar 2,6 ó 7 bits/trama. Cuando el FIFO de transmisión está vacío, el último byte o la porción del último byte, escrito al FIFO es transmitido repetidamente. Similarmente el dato serial en CDSTi es desplazado dentro y convertido al formato de byte ancho. En el caso que el canal seleccionado sea de 2, 6 o 7 bits/trama, se recepciona sólo los bits mas significativos de cada byte. Puede notarse que ninguno de los estados relacionados al protocolo o bits de interrupción son aplicables en este estado. Sin embargo, los estados relacionados al FIFO y sus bits de interrupción sí son aplicables.

**Temporizador watchdog.** - Este es un contador binario de 11 estados con FOi como entrada y WD como salida. Este contador puede ser reseteado de dos maneras, por la entrada externa (RST) o escribiendo XXX01010 en el Registro del Temporizador Watchdog. La salida WD es normalmente ALTO y si el Registro del Temporizador Watchdog no es escrito dentro de  $2^{10}$  ciclos de la entrada FOi después que fue reseteado, la salida WD se irá a BAJO por un período de  $2^{10}$  ciclos de FOi. Aun si la entrada FOi no es requerida para formateo de datos en el modo de temporización externo, es necesaria para la operación del temporizador watchdog.

Orden de los bits en transmisión y recepción.- El bit menos significativo (LSB) correspondiente a D0 en el bus de datos es transmitido primero en la salida serial (CDSTo). En el lado receptor, el primer byte recibido en la entrada serial (CDSTi) es considerado como el LSB y colocado en el D0 del bus de datos.

#### 4.2.1.3 Descripción de registros.

Hay varios registros en el controlador de protocolo HDLC accesibles al microprocesador asociado vía el bus de datos. La dirección de estos registros fueron dados anteriormente y sus detalles funcionales son dados a continuación.

- A. Registro de estados del FIFO (lectura).- Este registro indica el estado de los FIFOs de transmisión y recepción y del byte recibido según se indica

D7	D6	D5	D4	D3	D2	D1	D0
ESTADO BYTE Rx		ESTADO FIFO Rx		ESTADO FIFO Tx		LOW	LOW

Estado del byte de Rx.- Estos dos bits (D7 y D6) indican el estado del byte recibido listo a ser leído del FIFO de recepción. El estado es

codificado como sigue:

D7	D6	ESTADO
0	0	Byte del paquete
0	1	Primer byte
1	0	Ultimo byte (FCS bueno)
1	1	Ultimo byte (FCS errado)

**Estado del FIFO de Rx.** - Estos bits (D5 y D4) indican el estado del FIFO de recepción.

D5	D4	ESTADO
0	0	FIFO de Rx vacío
0	1	Menor que o igual a 14 bytes
1	0	FIFO de Rx lleno
1	1	Mayor que o igual a 15 bytes

**ESTADO DEL FIFO DE Tx.** - Estos bits (D3 y D2) indican el estado del FIFO de transmisión.

D3	D2	ESTADO
0	0	FIFO de Tx lleno
0	1	Mayor que o igual a 5 bytes
1	0	FIFO de Tx vacío
1	1	Menor que o igual a 4 bytes

**B. Registro de dato recibido (lectura).** - Leyendo este registro, pone el primer byte del FIFO de recepción en el bus de datos. El primer bit del dato recibido en la entrada serial CDSTi es considerado el LSB y está disponible en D0 del

bus de datos.

D7	D6	D5	D4	D3	D2	D1	D0
RD7	RD6	RD5	RD4	RD3	RD2	RD1	RDO

**C. Registro de dato a transmitir (escritura).-**

Escribiendo en este registro pone el dato presente en el bus de datos, en el FIFO de transmisión. El LSB (D0) es transmitido primero.

D7	D6	D5	D4	D3	D2	D1	D0
TD7	TD6	TD5	TD4	TD3	TD2	TD1	TDO

**D. Registro de control (lectura/escritura).-** Este registro es usado para control de propósitos generales del controlador de protocolo.

D7	D6	D5	D4	D3	D2	D1	D0
TxEN	RxEN	RxAD	RA6/7	IFTF1	IFTFO	FA	EOP

- **TxEN-Habilitación de transmisión.**- Cuando es ALTO, este bit habilita al transmisor y cuando es bajo lo inhabilita poniendo la salida serial CDSTo en alta impedancia. Si el transmisor es inhabilitado durante la transmisión de un paquete usando este bit, el controlador de protocolo esperara hasta que el paquete completo y la bandera de cierre es transmitido o el paquete es abortado antes de poner la

salida CDSTo en alta impedancia. Así el bit TxEN controla la transmisión paquete por paquete a diferencia que la entrada TxCEN (pin 1) el cual la controla bit por bit. Sin embargo, si el controlador de protocolo esta en el estado de transferencia de datos transparente, la transmisión parará dentro de dos períodos de bit (máximo) y pone la salida en alta impedancia.

- **RxEN-Habilitación de recepción.**- Este bit habilita la recepción cuando es ALTO y lo inhabilita cuando es BAJO. Si este bit va a BAJO durante la recepción de un paquete, el receptor se inhabilitará sólo cuando el paquete completo y su bandera de cierre es recibido o es detectado un aborto. Así el bit RxEN controla la sección de recepción paquete por paquete a diferencia de la entrada RxCEN (pin 2) la cual controla bit por bit. Sin embargo, si el controlador de protocolo está en el estado de transferencia de datos transparente, el receptor se inhabilitará inmediatamente.
- **RxAD-Detecta dirección de recepción.**- Este bit cuando es ALTO, habilita la detección de dirección para los paquetes recibidos. Esto fuerza al receptor a reconocer sólo los paquetes que tienen una dirección igual a la

programada en el Registro de Dirección de Recepción o si el byte de dirección es la dirección All-Call (todos UNOs). La comparación de dirección es hecha sólo con los 7 bits superiores y un All-Call es definida como los 7 bits superiores a UNO en el campo de dirección recibido. Si RxAD es BAJO, la detección de dirección es inhabilitada y todos los paquetes válidos son reconocidos.

- **RA6/7-Dirección de recepción de 6/7 bits.-** Este bit, cuando es alto limita la detección de dirección a solo los seis bits superiores del byte de dirección recibido y cuando es BAJO permite la comparación de dirección de 7 bits. RA6/7 es ignorado si RxAD es BAJO.
- **IFTFO Y IFTF1-Llenado del tiempo entre tramas.-** Estos bits permiten al transmisor estar en uno de los estados activo o inactivo o permite al controlador de protocolo a estar en el estado de transferencia de datos transparente.

IFTF1	IFTFO	RESULTADO
0	0	Estado inactivo (todos UNOs)
0	1	Flags consecutivos
1	0	Transferencia de datos transparente
1	1	Estado Go Ahead (7FH consecutivos)

- **FA-Aborto de trama.-** Cuando este bit es ALTO, 'marca' el siguiente byte escrito en el FIFO de

transmisión y hace que una secuencia de aborto (ocho UNOs) sea transmitida. La secuencia de aborto será transmitida en vez del byte que fue marcado. El bit FA es puesto a CERO cuando se escribe el dato al FIFO de transmisión. Por lo tanto, una 'lectura' de este registro no refleja el último dato escrito a éste.

- EOP-Fin de paquete.- Escribiendo un UNO a este bit 'marca' al siguiente byte escrito al FIFO de transmisión para indicar que éste es el último byte de dato del paquete. Este bit es puesto a CERO escribiendo el dato al FIFO de transmisión.

**E. Registro de dirección de recepción (lectura/escritura):**

D7	D6	D5	D4	D3	D2	D1	D0
RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0

El dato en este registro define la única dirección del controlador de protocolo HDLC. Si el reconocimiento de dirección es habilitado usando los bits RxAD y RA6/7 en el Registro de Control, un paquete entrante es reconocido solo si su byte de dirección (siete o seis bits más significativos) es igual a los bits correspondientes en este registro o si su dirección es un 'All-Call'. El LSB del Registro

de Dirección de Recepción es BAJO permanentemente y la comparación de dirección es hecha sólo con los bits restantes del registro.

**F. Registro de control de temporización (lectura/escritura).**- El Registro de Control de Temporización controla el modo de temporización y otras operaciones relacionadas y provee un reset por software al controlador de protocolo.

D7	D6	D5	D4	D3	D2	D1	D0
RST	IC	C1EN	BRCK	TC3	TC2	TC1	TC0

- **RST-Reset.**- Cuando este bit es puesto ALTO, todos los registros en el controlador de protocolo HDLC son reseteados y los datos de los FIFO's son perdidos. Esto es equivalente al reset externo con la excepción que no afecta el temporizador watch-dog y la salida WD.
- **IC-Control interno.**- Cuando este bit está en CERO el controlador de protocolo está en el modo de temporización externo. Las secciones de transmisión y recepción son habilitadas por las entradas TxCEN y RxCEN respectivamente y FOi es usado sólo para la operación del temporizador watch-dog. Cuando este bit es un UNO el controlador de protocolo está en el modo de temporización interna. Las secciones de transmisión y recepción son habilitadas por las

temporizaciones generadas internamente derivadas desde las entradas CKi y FOi. La entrada FOi define el comienzo de una trama y las secciones de transmisión y recepción son habilitadas en los canales determinados por los bits TCO-TC3. Las entradas TxCEN y RxCEN son ignoradas en este modo.

- **C1EN-Habilitador del canal-1.**- Cuando es ALTO, habilita la transmisión de la información del canal-C en el canal-1 del Formato-PCM saliente (CDSTo) y cuando es BAJO, pone a CDSTo en alta impedancia durante este período. Sin embargo, la información del canal-C es recibida independientemente y el Registro de Estado del Canal-C es actualizado continuamente. Note que C1EN es relevante sólo en el modo de temporización interna.
- **BRCK-Relacion del reloj de transferencia.**- Este bit es usado durante el modo de temporización interna y es ignorado en el modo de temporización externa. Este deberá ser ALTO si el reloj de entrada CKi es igual a la velocidad de transferencia y deberá ser ALTO si el reloj es 2 veces la velocidad de transferencia.
- **TC0-TC3-Bits de control de temporización.**- En el modo de temporización interna las secciones de transmisión y recepción son habilitados

durante los tiempos definidos por los bits TC0-TC3. Esto se aplica sólo a los canales 0,2,3 y 4 del formato-PCM para transferir paquetes de datos transparentes (canal-1 transfiere información del canal-C). La salida CDSTo es puesta en alta impedancia durante los restantes intervalos de tiempo no habilitados por estos bits.

TC3	TC2	TC1	TC0	NUMERO DE CANAL	BITS/TRAMA
X	0	0	0	0	1
X	0	0	1	0	2
0	0	1	0	0	6
1	0	1	0	0	7
X	0	1	1	2	8
X	1	0	0	3	8
X	1	0	1	4	8
X	1	1	0	2 y 3	16
X	1	1	1	2,3 y 4	24

#### G. Registro flag de interrupciones (lectura).-

Leyendo el Registro Flag de Interrupciones pone los bits de estado de interrupciones en el bus de datos. Este registro es reseteado cuando es leído y un bit particular no será seteado hasta que su condición particular ocurra otra vez. El detalle funcional de cada bit se da a continuación.

D7	D6	D5	D4	D3	D2	D1	D0
GA	EOPD	Tx DONE	FA	Tx 4/19 FULL	Tx URUN	Rx 15/1 FULL	Rx OFLW

- **GA-Go ahead.**- Este bit en ALTO indica la detección de una secuencia 'Go Ahead' en el dato entrante.
- **EOPD-Detección de fin de paquete.**- Un UNO en este bit confirma la recepción de una bandera 'fin de paquete', una secuencia de aborto o un paquete inválido de 24 o más bits en el dato entrante.
- **TxDONE-Transmisión realizada.**- Este bit en ALTO indica que la transmisión del paquete está completa y el FIFO de transmisión está vacío. El flanco de bajada de la salida TEOP hace que este bit sea ALTO si el FIFO está vacío.
- **FA-Aborto de trama.**- Este bit es ALTO para indicar que un aborto de trama fue detectado en el dato entrante.
- **Tx 4/19 FULL-FIFO de transmisión completo 4/19.**- Este bit en ALTO indica que el FIFO de transmisión sólo tiene 4 bytes y otros 15 bytes pueden ser cargados. Este bit tiene relevancia solo cuando el FIFO de transmisión está siendo descargado y no cuando está siendo cargado.
- **TxURUN-FIFO de transmisión vacío.**- Este bit en ALTO indica que el FIFO de transmisión está vacío sin que se halla dado al controlador de protocolo la indicación de 'fin de paquete'. En este caso el controlador de protocolo

transmitirá una secuencia de aborto automáticamente.

- Rx 15/19 FULL-FIFO de recepción completo 15/19.- Este bit en ALTO confirma que el FIFO de recepción tiene 15 bytes dentro y puede recepcionar 4 bytes más.
- RxOFLW-FIFO de recepción lleno.- Este bit en ALTO indica que el FIFO de recepción está lleno. El byte que causa esto y los siguientes bytes recepcionados, mientras el FIFO está en este estado, son perdidos.

#### H. Registro del temporizador watchdog (escritura).-

El Registro del Temporizador Watchdog opera en conjunto con el temporizador watchdog y la salida WD. Escribiendo el código XXX01010 en el registro resetea el temporizador watchdog. Si el registro no se vuelve a escribir dentro de  $2^{10}$  ciclos de FOi después de resetear el temporizador, la salida WD irá a BAJO. Este registro tiene el único propósito de resetear el temporizador y es relevante sólo si se escribe con el dato dado anteriormente.

#### I Registro de habilitación de interrupciones

(lectura/escritura).- Este registro habilita/inhabilita las interrupciones especificadas en el Registro Flag de Interrupciones (IFR). Poniendo en ALTO los bits

apropiados en este registro (IER) habilita la fuente de interrupción asociada. Sin embargo, los bits enmascarados en el IFR son aún válidos pero no hacen que la salida IRQ vaya a BAJO.

- J. **Registro general de estados (lectura).**- Este registro contiene información de los estados del controlador de protocolo.

D7	D6	D5	D4	D3	D2	D1	D0
Rx OFLW	Tx URUN	GA	ABRT	IRQ	IDLE	LOW	HIGH

**Rx OFLW-FIFO de recepción rebazado.**- Este bit en ALTO indica que el FIFO de recepción fue rebazado. El byte que causa esto y los siguientes bytes recibidos mientras el FIFO está en este estado serán perdidos. Note que éste es el mismo que el bit RxOFLW del Registro Flag de Interrupciones (IFR) y sólo puede ser aclarado si IFR es leído.

- **Tx URUN-FIFO de transmisión vacío.**- Este bit en ALTO indica que el FIFO de transmisión está vacío. Bajo esta condición el paquete que está transmitiéndose es abortado. Este bit es el mismo que el bit TxURUN del IFR y sólo puede ser aclarado leyendo este registro.
- **GA-Go ahead.**- Este bit es ALTO si una secuencia 'go ahead' es recibida en el dato entrante y es aclarado cuando el IFR es leído.

Este bit es el mismo que el bit GA en IFR.

- **ABRT-Aborto.**- La recepción de ocho UNOs consecutivos en el dato entrante, pone a ALTO este bit y una lectura al Registro General de Estados, lo aclara.
- **IRQ-Petición de interrupción.**- Este bit se refiere al estado de la salida IRQ del controlador de protocolo. Si es ALTO, indica que la salida IRQ es BAJO y viceversa.
- **IDLE-Canal inactivo.**- Este bit en ALTO indica que el receptor está detectando un canal inactivo en su entrada (mínimo 15 UNOs).
- **Reset.**- Cuando el controlador es reseteado por un flanco de bajada de un pulso en el pin RST o seteando un '1' en el bit RST en el Registro de Control de Temporización, el dispositivo se pone en los siguientes estados:

Todos los bits de los siguientes registros son borrados Registro de Control de Temporización, Registro de Habilitación de Interrupciones, Registro de Control y el Registro de Interrupciones.

Todos los bits en el Registro General de Estados son borrados excepto los dos bits menos significantes.

Los Registros de Transmisión y Recepción son borrados y el Registro Estado de Fifos

refleja sus estados correspondientes.

La salida WD cambia de estado a '0'.

Son deshabilitados el transmisor y receptor.

#### 4.2.1.4 Operación de transmisión.

Después del reset del power on la sección de transmisión es deshabilitada. Seguidamente se deben de programar adecuadamente los Registros de Control de Temporizaciones (velocidad del reloj, modo de control interno o externo, selección del canal PCM en modo interno) y el Registro de Control (habilitación de las direcciones, parte de transmisión y recepción). En la transmisión de PAQUETES NORMALES, el dato es escrito en el FIFO de transmisión empezando por la dirección. Cuando el dato es detectado en el FIFO de transmisión el controlador procederá de la siguiente manera:

- Si el transmisor está en estado libre, el octeto de 8 unos que está siendo transmitido es completado y seguidamente empieza con una bandera y seguido de los datos del FIFO de transmisión.
- Si el transmisor está en el estado de Llenado de Tiempo entre Tramas, la bandera que en estos instantes es transmitido, es completada y empieza con otra bandera y seguido de los datos del FIFO de transmisión.
- Si el transmisor está en el estado de Go Ahead, continuará en ese estado hasta que el dato es cargado en el FIFO de transmisión y solamente cuando cambien

los estados de los bits IFTF del Registro de Control podrá transmitirse el dato.

Para indicar que un particular octeto es el último del paquete, el bit EOP en el Registro de control debe ser seteado, antes que el último octeto sea escrito en el FIFO de transmisión. El bit EOP del Registro de Control es borrado automáticamente cuando el último dato es escrito en el FIFO. Después de la transmisión del último octeto del paquete, el FCS es enviado seguido de la bandera de cierre. Si existen datos adicionales en el FIFO de transmisión, entonces se generará otra bandera y se transmitirá seguido del nuevo paquete. En caso que no hay octetos por transmitir, el transmisor asume el estado del canal seleccionado.

Durante la transmisión de cualquier dato o FCS el controlador chequea la información transmitida bit por bit e insertando ceros después de cada 5 unos consecutivos.

La **transmisión underrun**, ocurre cuando el último octeto cargado en el FIFO no fue marcado con el EOP bit y no hay más octetos. En tal situación el controlador transmite la secuencia de Aborto (8 unos) y vuelve al estado de canal seleccionado. También se puede abortar una trama que está siendo cargada en el FIFO, basta con setear el bit FA en el Registro de Control antes de escribir otro octeto; este bit es borrado automáticamente cuando el octeto marcado es escrito en el FIFO. Durante

el proceso de transmisión cuando le toca al octeto marcado una secuencia de Trama de aborto es enviada.

De manera similar seteando los bits adecuados se puede programar el controlador en modo Go Ahead.

#### 4.2.1.5 Operación de recepción.

Después del power on la sección de recepción es deshabilitada. Se programan los mismos registros que en el caso de la transmisión y adicionalmente el registro de Direcciones. Para habilitar la detección de dirección el bit RXAD del Registro de Control debe ser seteado y cargar la dirección deseada en el Registro de Direcciones.

Para la recepción de PAQUETES NORMALES, la información serial empieza a ser sincronizada y chequea las banderas y canales en estado libre. Si un canal libre es detectado el bit IDLE en el Registro General de Estados es seteado. Una vez que la bandera es detectada la recepción es sincronizada automáticamente para almacenar los octetos entrantes. El paquete entrante es analizado bit por bit retirando los ceros insertados, el FCS es calculado y la información es escrita en el FIFO de recepción. El FCS y los otros caracteres de control nunca son escritos en el FIFO de recepción.

está habilitado la detección de dirección, el primer octeto siguiente de la bandera es comparado con el octeto del Registro de Direcciones, si no son iguales el

paquete entrante es ignorado de lo contrario es recibido en forma normal. Todos los octetos son escritos en el FIFO de recepción e indicados con sus dos bits de estados en el Registro de Estados del FIFO, en el cual se puede saber si es el primero, el último, si llegó correctamente con el FCS correcto o incorrecto. También se pueden detectar todos los estados de transmisión, tales como Tramas de Aborto, Go Ahead y paquetes inválidos por fallas del enlace u otras causas.

En los Anexos se encuentran la información técnica del controlador de protocolos. El diseño del circuito básico del controlador HDLC MT8952 conjuntamente con la matriz de conmutación MT 8980 es mostrada en el Plano de la Fig. 4.16.

#### **4.2.2 PROPUESTA DE UNA TARJETA DE COMUNICACIONES ISA.**

Como se puede apreciar el controlador de protocolo HDLC MT8952B de Mitel, cumple con los requerimientos de nuestras especificaciones. De todo el análisis anteriormente realizado, estamos en la capacidad de proponer el hardware de una tarjeta que nos permita las comunicaciones entre procesadores basada en la interconexión de sistemas abiertos (ISA), Fig 4.19 (Anexo). Las características principales son:

- En la tarjeta está incluida una matriz de conmutación digital que tienen 6 canales principales (high ways) disponibles de los 8, para aplicaciones diversas en el ámbito de las telecomunicaciones, concentradores,

- telemetría, centrales de alarma, etc.
- La tarjeta, basada en el procesador 8088 de Intel, dispone de los periféricos necesarios para trabajar con sistemas con procesamiento en tiempo real. Además tienen disponibles periféricos para reportar señalizaciones y una interfase serial RS-232 para Operación y Mantenimiento.
  - Por las características del controlador HDLC MT8952, en la programación de temporización en modo interno y externo, posibilita el empleo de otras temporizaciones que no sean del formato-PCM (telecomunicaciones) sino relojes externos, para otras aplicaciones como en la industria CAM (Computing Aid Manufacturing), donde se necesiten una gran confiabilidad en las comunicaciones empleando sistemas de transmisión inmune al ruido eléctrico, por ejemplo fibras ópticas u otros.
  - Su aplicación no solamente estaría destinado a los sistemas de conmutación Digital, se ampliaría a otros sistemas que manejen un tráfico de mensajes moderado entre procesadores. Esto por la forma de acceder a los FIFOs de recepción y transmisión.
  - Se podría mejorar la capacidad de transferencia de mensajes usando dispositivos de transferencia rápida de memoria (DMA), con lo que se lograría un sistema potente para la interconexión de sistemas abiertos.
- En el Anexo 1 se muestra el plano de esta tarjeta.

### 4.3 Diseño del software.

En esta sección se esbozarán las interfases soporte hardware y software, sugiriendo recomendaciones para el diseño final del software para nuestro sistema.

La expresión "comunicación entre procesadores" es muy general y vaga, sin embargo si definimos que cada procesador ejecuta tareas, llamados **procesos** que necesitan transferir datos o control a otros procesos, entonces se identifica a que realmente los usuarios de la mensajería entre los procesadores son los procesos. Estos pueden estar en el mismo procesador o en diferentes procesadores. En la Fig. 4.17 se muestran los métodos de transferencia de control y datos. En la transferencia de control existen dos métodos con retorno y sin retorno. similarmente en la transferencia de datos existen transferencia directa e indirecta. Estos métodos básicos se ilustran en la Fig 4.18 cuando se trata de transferencias entre bloques funcionales de diferentes procesadores.

Los diagramas de flujo analizados en la sección del Diseño del Sistema son los fundamentales para la transferencia de los mensajes entre los procesadores (procesos) y seran a través de los controladores HDLC, anteriormente estudiado.

La tesis no pretende obtener el programa completo y funcional del software del sistema. Solamente en esta sección se darán recomendaciones para el diseño del

software estructurado, como se determinó en el capítulo II.

Ahora nuestro objetivo es lograr un módulo de acoplamiento hardware y software, aplicando la filosofía del diseño estructurado.

Teniendo caracterizado los aspectos físicos y funcionales del controlador, podemos bosquejar el módulo ISH (Módulo Interface Soporte de Hardware). El formato recomendado de las especificaciones de diseño estará basado en la siguiente plantilla:

```

FUNCION Módulo_Nombre (CP,Parámetros_Lista)
Descripción

```

Los elementos de la plantilla son definidos de la siguiente manera:

- **Función.**- Es una clave opcional el cual indica que el nombre del módulo es una función. Una función asume un valor cuando retorna después de ejecutarse.
- **Módulo\_Nombre.**- Es el nombre del módulo que debe ser invocado por el software.
- **CP.**- Es el parámetro enviado al módulo describiendo la función del CP (Controlador de Protocolo) en él. Esta información es mandatorio desde que en el sistema puede estar concebido para contener varios controladores de protocolos.
- **(Parámetros\_Lista).**- Es una lista desde 0 a varios parámetros el cual define la interfase del software

para el módulo. Los parámetros son individualmente separados por comas. Cada parámetro puede o no mantener válido un valor previa a su invocación o retorno.

- **Descripción.**- Es la descripción de la función que el módulo ejecuta.

#### 4.3.1 Módulo de la Interfase Soporte de Hardware (ISH).

Las siguientes rutinas están basados en las características del hardware del controlador HDLC MT8952B. Como tal ellos proveen un soporte de interacción directa con el hardware del controlador:

Registro Control de Temporización

Registro de Control

Registro de Datos Recibidos

Registro de Datos Transmitidos

Registro de Dirección de Recepción

Registros de Estados.

##### **A. Módulos del registro de control de temporización:**

**FUNCION EnModoInt (CP)**

Pregunta si el CP está programado en modo interno de temporización. La función retorna V (verdadero) si lo está y F (falso) caso contrario.

**Set EnModoInt (CP,siono)**

Programa el registro en modo de temporización interna de acuerdo a los valores requeridos por el registro.

**FUNCION EnModoExt (CP)**

Pregunta si el CP está programado en modo externo de temporización. La función retorna V (verdadero) si lo está y F (falso) caso contrario.

Set EnModoExt (CP,siono)  
 Programa el registro en modo de temporización externa de acuerdo a los valores requeridos por el registro.

Set Clk2Mega (CP,siono)  
 Programa el registro para trabajar con 2.048 Mhz de acuerdo a los parámetros requeridos por el registro. V si se programa a 2 Mhz y F a 4.096 Mhz. Notar que esta programación sólo es válida para modo de temporización interna.

SeleccionCanalTx (CP,canal)  
 En modo de temporización interna selecciona el canal PCM por el cual se transmitirá de acuerdo al valor del parámetro canal.

## B. Módulos del registro de control

Set FinPaquete (CP, siono)  
 Programa al registro para indicar que el siguiente octeto es el último del paquete. V si es cierto y F caso contrario.

Set Aborto (CP, siono)  
 Programa al registro para indicar que la trama que va a ser enviada abortará. V si es cierto y F caso contrario.

Set EstadoLibre (CP, siono)  
 Programa al registro para indicar que el controlador está en el estado Libre todos unos. V verdadero y F caso contrario.

Set EstadoBan (CP, siono)  
 Programa al registro para indicar que el controlador está en el estado llenado de tiempo entre tramas, transmite puras banderas. V si es cierto y F caso contrario.

Set EstadoGoAhead (CP, siono)  
 Programa al registro para indicar que el controlador está en el estado Go Ahead transmite continuos 7Fh. V verdadero y F caso contrario.

FUNCION RxDireccion (CP)  
 Pregunta si el CP está programado en modo recepción con dirección. La función retorna V (verdadero) si lo está y F (falso) caso contrario.

Set RxDirHab (CP,siono, dir)  
 Programa el registro para habilitar la recepción con dirección de acuerdo a los valores de los parámetros 6 ó 7 bits (dir).

Set RxHab (CP,siono)  
 Programa el registro para habilitar la sección de recepción. Vuelve valores V si lo está y F caso contrario.

Set TxHab (CP,siono)  
 Programa el registro para habilitar la sección de transmisión. Vuelve valores V si lo está y F caso contrario.

#### C. Módulo del registro de datos recibidos

LeerFifoRx (CP,num,data,bufrx)  
 Permite la lectura del FIFO de recepción, como salida tiene el controlador, número de octetos y los datos leídos por la interfase paralela del procesador y los almacena en el buffer direccionado por bufrx.

#### D. Módulo del registro de dato a transmitir

EscribirFifoTx (CP,num,data,buftx)  
 Permite la escritura en el FIFO de transmisión; como entradas tiene el controlador, número de octetos y los datos a ser transmitidos leídos del buffer cuya dirección es buftx.

EscribirUltimoFifoTx (CP,data,buftx)  
 Permite la escritura del último octeto del paquete en el FIFO de transmisión; como entradas tiene el controlador, número de octetos y los datos a ser transmitidos leídos del buffer cuya dirección es buftx.

Así podemos seguir especificando todas las funciones y rutinas del controlador que formaría la Interfase Soporte Hardware (ISH), estos módulos deben ser lo mas elaborado posible, debido a que serán altamente reusables por los módulos de las estructuras superiores.

#### 4.3.2 Módulo de la Interfase Soporte Software (ISS).

Seguidamente se define la Interfase Soporte Software (ISS) que a diferencia de los anteriores no interactúan directamente con el hardware. Estos módulos pertenecen a un nivel jerárquico superior, no menos importante que la ISH, y de acuerdo a su caracterización se pueden definir los siguientes grupos funcionales:

- Transmisión de Mensajes
- Recepción de Mensajes
- Estados de Transmisión
- Detección de Errores

Como ejemplo, se propone el siguiente diseño de las especificaciones para las funciones de transmisión y recepción, observese que estos módulos ya no interactúan directamente con el dispositivo hardware:

##### A. Transmisión de mensajes.

TxMensaje (CP,origen,destino,num,mensaje)  
Envía el mensaje originado en el procesador 'origen' con destino al procesador 'destino', con una longitud 'num' del mensaje.

##### B. Recepción de mensajes.

RxMensaje (CP,origen,destino,num,mensaje)  
Recibe el mensaje originado en el procesador 'origen' con destino al procesador 'destino', con una longitud 'num' del mensaje.

Un ejemplo de la pseudo codificación para la transmisión de mensajes sería:

```

TxMensaje(CP,origen,destino,num,mensaje)
{
    .

    SeleccionCanalTx (CP,canal)
    EscribirFifoTx (CP,num,data,buftx)
    Set FinPaquete (CP, siono)
    EscribirUltimoFifoTx (CP,data,buftx)
    Set TxHab (CP,siono)
}

```

Siguiendo los mismos criterios podemos desarrollar el software del sistema hacia los niveles superiores, de acuerdo a las especificaciones definidas en el diseño del sistema.

Para la codificación de los programas se pueden emplear preferentemente lenguajes de alto nivel, si se cuenta con buenos compiladores e infraestructura para realizar las pruebas en emuladores u otras facilidades que permita detectar problemas en el desarrollo del software. Para desarrollo de pequeños sistemas es atractivo los lenguajes de bajo nivel, el Ensamblador, cuyos compiladores son de gran difusión y baratos existiendo herramientas de depuración.

Finalmente, en este capítulo debemos mencionar un aspecto importante, que es la documentación técnica, que debe desarrollarse en forma concurrente con los avances del desarrollo del sistema y actualizarse constantemente.

La documentación debe tener una estructura adecuada,

aquí mencionamos algunas partes importantes:

Especificaciones Técnicas.

Descripción del Sistema.

Descripción del desarrollo hardware.

Descripción del desarrollo software.

Manual de Operación y Mantenimiento.

## CAPITULO V

### PRUEBAS DE LABORATORIO

En este capítulo, antes de describir las pruebas realizadas en los Laboratorios, deseo resumir las etapas fundamentales que se requieren para desarrollar un sistema I+D (Investigación y Desarrollo); de acuerdo a las experiencias adquiridas y los estudios realizados para la presente tesis:

#### **Etapa 1.- Desarrollo del sistema**

Definir las especificaciones generales.

Definir los modelos teóricos (deben incluir modelos de tráfico y fiabilidad)

Definir los algoritmos básicos.

Desarrollo del sistema

Pruebas de integración hardware-software en el laboratorio.

Pruebas de campo del sistema integrado.

#### **Etapa 2.- Desarrollo del hardware**

Especificaciones de los bloques funcionales.

Determinación de un modelo físico.

Diseño de circuitos.

Cableado de tarjetas prototipo (wire up)

Pruebas aisladas en el laboratorio (wire up)

Diseño de CADs

Pruebas aisladas en laboratorio de los PCBs (printed circuit board).

Pruebas de integración hardware-software.

### **Etapa 3.- Desarrollo del software**

Especificaciones de los módulos funcionales.

Definición del proyecto básico y modelo lógico.

Proyecto detallado.

Codificación de los programas.

Pruebas con emuladores en el laboratorio

Pruebas de integración software

Pruebas de integración software-hardware.

Una vez integrado y probado el prototipo en el laboratorio es necesario realizar las pruebas de campo, en el ambiente real, donde el sistema trabajará y se efectuará las pruebas con tráfico real y otros factores que no pudieron simularse en el laboratorio.

En este capítulo describiremos las pruebas de hardware aisladas que se efectuaron en los laboratorios, haciendo uso de las tarjetas cableadas (Wire Up) y computadores personales PCs.

#### **5.1 Descripción de las pruebas aisladas de hardware**

Para probar la bondades de las características funcionales del Controlador de Protocolo HDLC de Mitel Semiconductor MT8952B, se diseñó el circuito que se muestra en el Plano 2 Controlador HDLC - Versión Mitel (Fig. 5.1), en la cual se aprecia la interfase del

puerto paralelo, conformado por el bus de datos y el bus de control. La sincronización es gobernada por un reloj externo (CK), para posibilitar la prueba con diferentes frecuencias y en los modos de temporización interna y externa; para el caso de temporización modo interno la señal de sincronismo (FOI), es proporcionada de la misma fuente del CK; que se supone que es el Módulo de Sincronización de la Central. Para lograr una confiabilidad en el medio de transmisión, a la salida y entrada del controlador se han implementado Line Drivers (en recepción AM26LS32 y en transmisión AM26LS31), que permiten la conversión de las señales de NO-BALANCEADAS a BALANCEADAS, que son más inmune al ruido y permiten un gran FAN OUT, cubriendo grandes distancias (a 4 Mhz aprox. 100 mts). Esta misma característica nos permitirá interconectar a los sistemas de transmisión digital, para efectuar pruebas con unidades remotas.

Los procesadores que emplearemos serán Computadores personales XT y AT, para lo cual se ha diseñado e implementado interfases que permiten transferir los controles de entrada y salida, así como la transferencia de datos en forma bidireccional. En el Plano 3 Interfase PC (Fig. 5.2), podemos apreciar, el diagrama circuitual de los decodificadores para los puertos de entrada/salida, los buffer para las señales de control y direcciones y los transceiver para el bus de datos. Esta interfase se instalará en un SLOT libre de la PC que

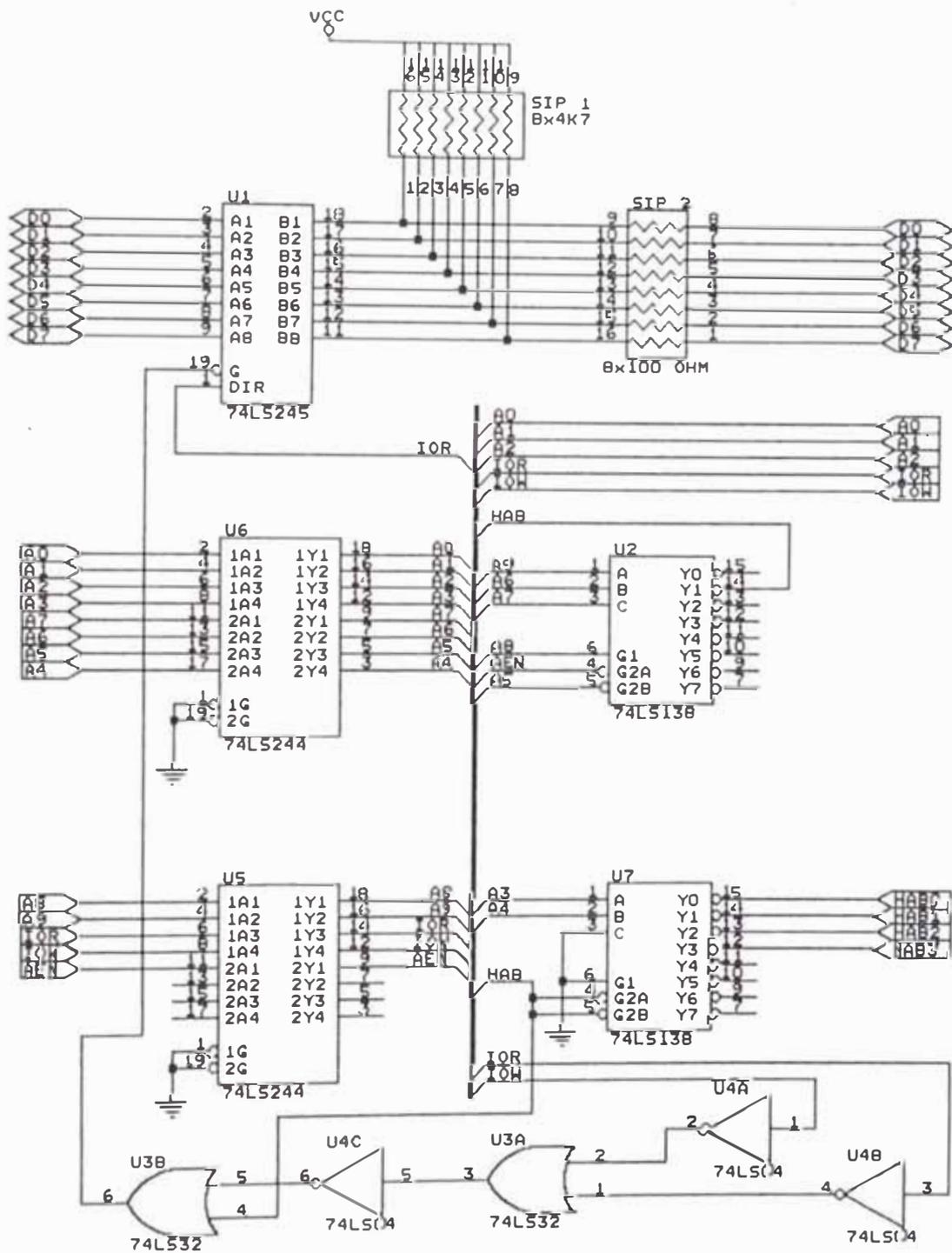


FIG 5.2

TESIS DE GRADO  
 Autor: EDMUNDO A.INGAR A.

Title		INTERFASE PC-XT
Size	Document Number	REV
A	1	1.0
Date:	December 8, 1993	Sheet 1 of 1

viene previsto, para propósitos generales (como el nuestro) empleando los puertos y direcciones recomendados por los fabricantes de las PCs.

Efectuando la siguiente interconexión mostrada en la Fig. 5.3; se procedieron a realizar las pruebas y verificación de temporización de las señales importantes en los controles de escritura y lectura, cuyos resultados se muestran en el Diagrama de Tiempos Fig. 5.4.

Empleando los programas de prueba PC\_DARTX.ASM y PC\_DARRX.ASM, codificados en Assembler, cuyos listados se muestran en el Anexo, se probaron los accesos a los diferentes registros del controlador escribiendo y leyendo datos diferentes, con la ayuda del programa de depuración DEBUG. Estas pruebas se desarrollaron en el ambiente DOS 3.20.

## **5.2 Resultados de las pruebas**

Se efectuaron diferentes pruebas, entre las más importantes podemos mencionar:

**A.- Pruebas con un controlador y una PC.-** Además de las pruebas mencionadas en la sección anterior (medición de tiempos) se efectuó la transmisión y recepción de un paquete, en el mismo controlador, haciendo un lazo galvánico entre su transmisión y recepción. El mensaje fue recibido correctamente. Esta prueba puede ser incluido como una rutina de Operación y Mantenimiento,

para que el mismo sistema autoverifique su operación con la ayuda de un buffer con salida controlada, reemplazando al lazo galvánico.

**B.- Pruebas con dos controladores y dos PCs.-** Esta configuración, mostrada en la Fig. 5.5, permite efectuar lo siguiente:

**B.1 Pruebas del Registro de Direcciones.-** Se habilitó el Registro de Direcciones programándolo con diferentes direcciones y se enviaron mensajes con direcciones correctas y falsas, respondiendo en forma adecuada el controlador.

**B.2 Pruebas con dos Procesadores Diferentes.-** Esta prueba se efectuó empleando una AT-286-25Mhz y una XT-8088-4.77Mhz. Los dos procesadores anteriormente mencionados son homogéneos, pero diferentes en la especificación de su hardware. Son homogéneos porque el mismo programa ejecutado en la XT puede correr en la AT sin modificación alguna. Como explicáramos en el Capítulo III, la central esta conformado por procesadores homogéneos pero de diferentes características en cuanto a su performance, pudiendo establecer una jerarquía entre ellos; y estos pueden ser comunicados por el mismo Sistema de Comunicación entre Procesadores, haciendo uso del controlador de protocolo HDLC

TARJETA DE SINCRONISMO Y RELOJ

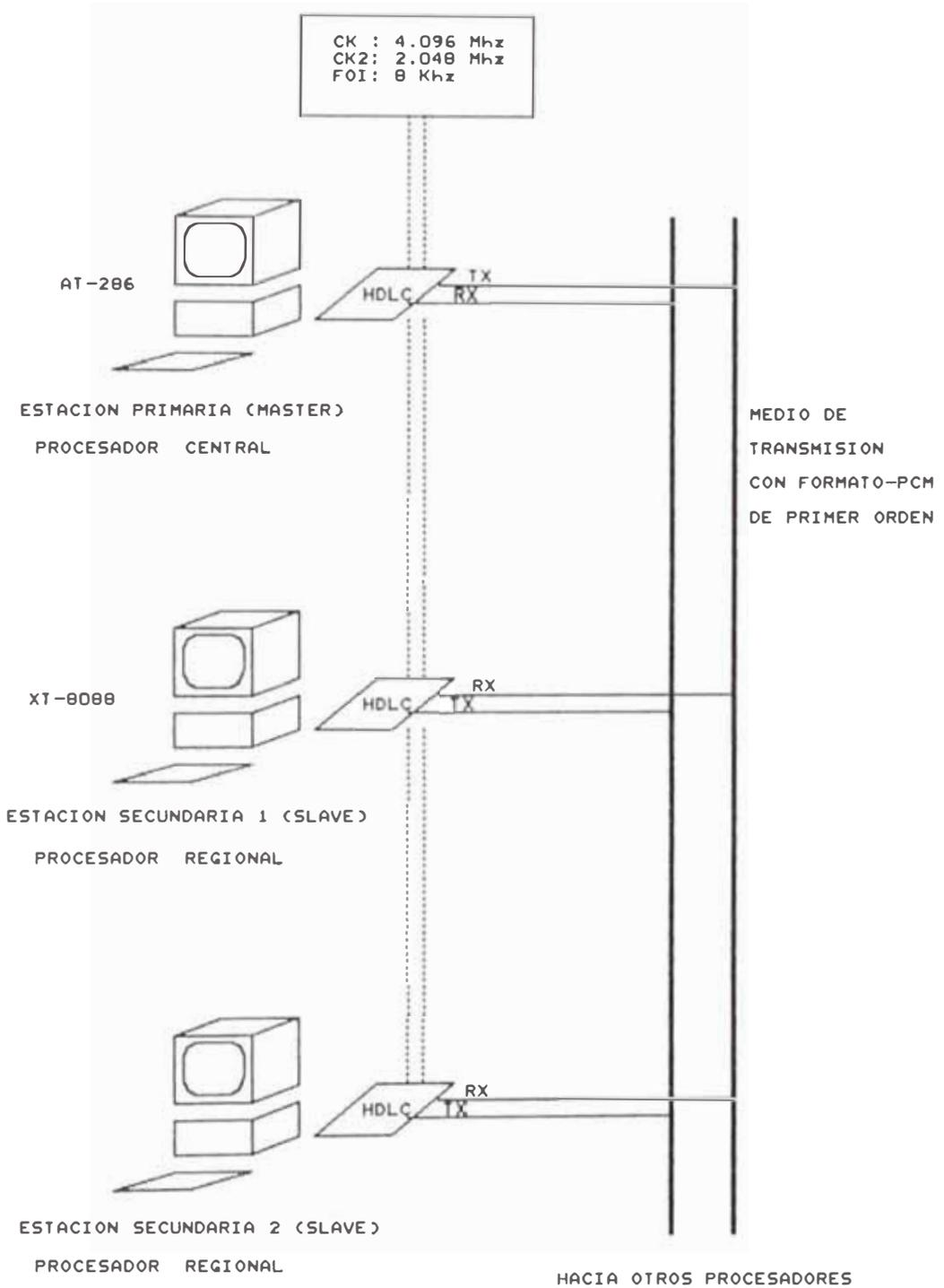


FIG 5.5

PRUEBAS DEL CONTROLADOR CON VARIOS PROCESADORES

MT8952 en modo de temporización interna.

**B.3 Pruebas de los Diferentes Estados de Transmisión.**- Se probaron con éxito los siguientes estados:

Estado Libre (todos unos)

Estado de llenado de tiempo entre Tramas  
(se transmitían banderas continuas)

Estado de Go Ahead (transmitía continuos  
7Fh)

Modo de Transferencia de Datos  
Transparente. Se transmitían los datos  
sin importar la temporización del sistema  
PCM; para otras posibles aplicaciones.

## C A P I T U L O    V I

### ANALISIS DE CONFIABILIDAD Y COSTOS

En la formulación del proyecto, capítulo III, se mencionaron los criterios de confiabilidad y costos para elegir el sistema mas adecuado; en este capítulo discutiremos algunos aspectos para analizar la confiabilidad de un sistema y al final algunas sugerencias para las consideraciones de costos.

#### **6.1 Análisis de confiabilidad.**

Primero definiremos algunos términos. La palabra **confiabilidad** puede ser usado de dos maneras. Primero, un termino genérico para describir la longevidad del sistema operando exitosamente y el segundo, como una definición matemática; esto es la probabilidad que un sistema estará funcionando por un período específico bajo ciertas condiciones. Los cálculos de la confiabilidad matemática para un específico sistema puede ser muy compleja y en algunos casos puede ser imposible.

Sin embargo, la performance operacional del sistema puede ser expresado por otras mediciones, alguno de los cuales con cálculos simples. Una de esas medidas es la **DISPONIBILIDAD**. Como nuestro sistema trabajará por largos períodos de tiempo, entonces la disponibilidad puede ser

calculada para el sistema si conocemos el Tiempo Medio Entre Fallas, comúnmente conocido como MTTF (Mean Time To Failure) y el Tiempo Medio de Reparaciones, conocido como MTTR (Mean Time To Repair) del sistema.

El MTTF es el tiempo medio que el sistema operará antes que falle. Esto depende de cómo esté diseñado el sistema, construido y en qué ambiente físico operará. En un sistema donde la reparación es posible, esto también, puede ser medido y es conocido como Tiempo Medio Entre Fallas, en inglés MTBF (Mean Time Between Failure). El MTTR depende, también de este factor, pero es afectado de otros factores tales como el tiempo que toma el técnico en reparar la avería en el lugar, el suministro de repuestos y la habilidad que tiene el técnico en reparar el sistema.

Es posible obtener estimados o mediciones del MTTF y MTTR de los componentes electrónicos del sistema; pueden ser obtenidos mediante análisis, observación o proporcionados por los fabricantes. Si nosotros conocemos MTTF y MTTR de cada componente, podemos computar fácilmente la disponibilidad del componente, usando la siguiente fórmula:

$$a = \frac{MTTF}{MTTF + MTTR} \quad 6.1$$

Conociendo los MTTFs y MTTRs individuales de cada componente, podemos, bajo ciertas restricciones, calcular

el MTTF y MTTR del sistema completo. Las fórmulas matemáticas usadas en este proceso nos permitirá calcular el numero de tarjetas ISA y procesadores de reserva requeridos para el sistema propuesto por la tesis para conseguir una determinada especificación de Confiabilidad.

Si hacemos ciertas presunciones acerca de la distribución de tiempos de falla (tiempo de vida) para cada uno de los componentes de nuestro sistema y además asumimos que el tiempo de vida de los componentes es una distribución exponencial, podemos obtener la Función Densidad de Probabilidad (FDP) de cualquier variable como:

$$f(x) = \frac{1}{M} e^{-x/M} \quad -\infty \leq x \leq \infty \quad 6.2$$

Donde M es el tiempo de vida útil del componente. La Fig. 6.1 muestra la función FDP para M = 1.

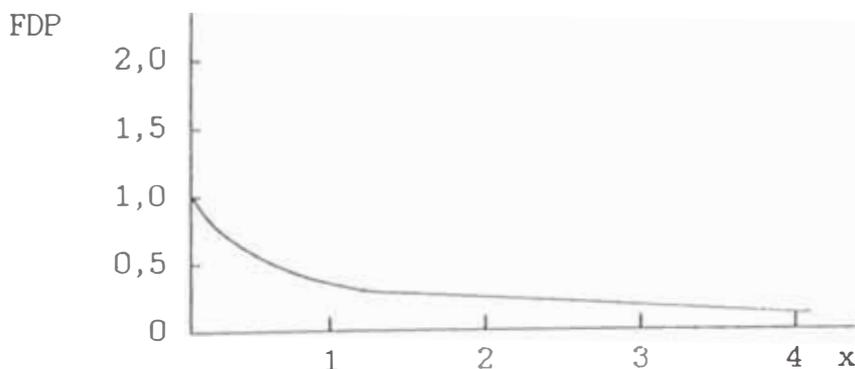


Fig. 6.1

FDP para una Distribución Exponencial

Intuitivamente el área bajo la curva entre dos puntos  $(x_1, f(x_1))$  y  $(x_2, f(x_2))$ ,  $0 < x_1 < x_2 < \infty$ , representa la probabilidad que el tiempo de vida caiga entre  $x_1$  y  $x_2$ . La distribución exponencial es frecuentemente usado para cálculos de confiabilidad ya que no depende del tiempo. Esto es, si un componente no ha fallado durante un tiempo  $t$ , la probabilidad que pueda fallar en un intervalo de tiempo mayor que  $t$  es independiente de  $t$ .

Alguna veces es mas útil la Función Distribución Acumulativa FDA, el cual da valores de la probabilidad que un resultado será menor o igual al resultado esperado. La FDA para un componente con distribución exponencial de tiempo de vida es:

$$F(t) = \int_{-\infty}^{\infty} f(x) dx = 1 - e^{-t/M} \quad 6.3$$

Nuevamente  $M$  es el tiempo de vida del componente. La Fig. 6.2 muestra la FDA para el sistema con distribución de tiempo exponencial, para  $M = 1$ . Note como la probabilidad que el sistema tiende a fallar se incrementa con el tiempo.

En muchos casos de diseño de especificaciones de confiabilidad la FDA es muy útil, sobre todo en sistemas que trabajarán por largos períodos ininterrumpidamente, como en el caso de las centrales telefónicas digitales, que sus fabricantes garantizan un tiempo de vida superior

a los 10 años.

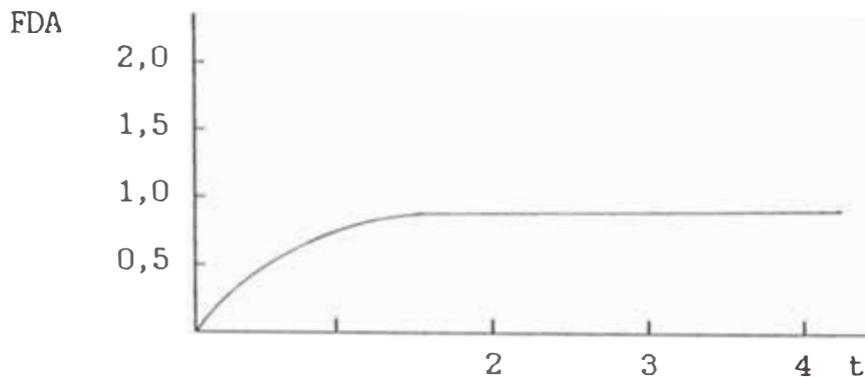


Fig. 6.2

### FDA para una Distribución Exponencial

#### 6.1.1 Cálculo de la disponibilidad de un sistema

La disponibilidad es una medida adecuada de la confiabilidad del sistema. Podemos calcular la disponibilidad de las partes del sistema por aplicación simple de la Teoría de Probabilidades.

Decimos que dos elementos independientes están en serie cuando los dos tienen que operar para el funcionamiento del sistema. Por independiente se entiende que la falla de un elemento no tiene efecto en la operación o falla del otro elemento. La disponibilidad para un sistema configurado en serie con dos elementos independientes con disponibilidades  $a_1$  y  $a_2$  es:

$$a = a_1 \times a_2 \qquad 6.4$$

Decimos que dos elementos están en paralelo cuando el sistema puede operar si le falta un elemento. En este

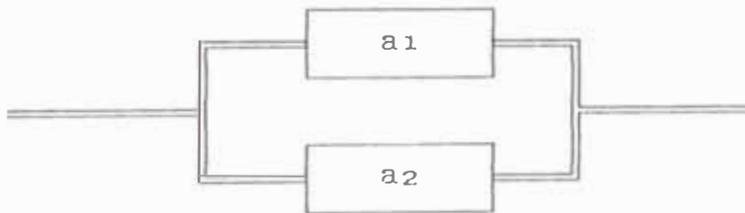
sentido la palabra paralelo es sinónimo de 'redundante'. La disponibilidad para un sistema de dos elementos con disponibilidades  $a_1$  y  $a_2$ , configurado en paralelo es:

$$a = 1 - (1 - a_1)(1 - a_2) \quad 6.5$$

A menudo estas configuraciones se representan gráficamente como las mostradas en la Fig. 6.3.



**Sistema serial con dos elementos**



**Sistema paralelo con dos elementos**

**FIG. 6.3**

### **Configuraciones de confiabilidad de los sistemas**

De esta manera es posible simplificar el modelo de confiabilidad del sistema a un elemento. la Fig. 6.4 nos muestra los pasos de simplificación de un sistema de cinco elementos independientes, el tratamiento es similar a reducir una red de resistencias, sólo que en este caso se aplican las fórmulas 6.4 y 6.5, según sea el caso reducción serial o paralela.

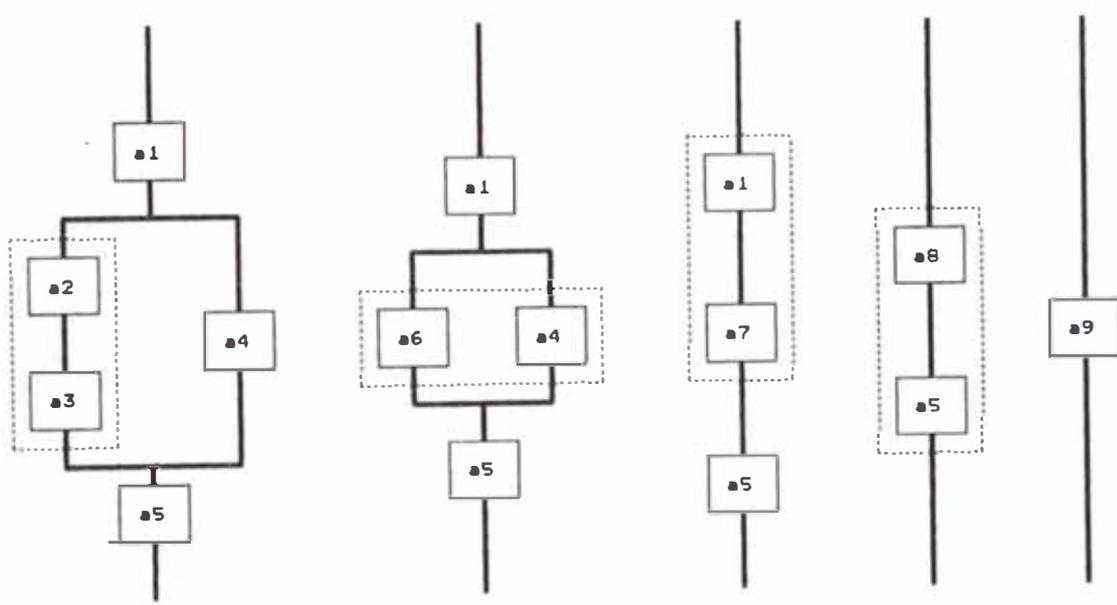


FIG 6.4

PASOS DE REDUCCION DE LA DISPONIBILIDAD DE UN SISTEMA COMPLEJO

Para mayor ilustración, examinemos algunos cálculos para los sistemas seriales y paralelo con un ejemplo:

Asumamos que el sistema está constituido por dos procesadores (elementos)

Proc 1 MTTF = 1000 hrs.                      MTTR    5 hrs.

Proc 2 MTTF = 2000 hrs.                      MTTR - 5 hrs.

Aplicando la fórmula 6.1 tenemos

$$a_1 = 0,99502$$

$$a_2 = 0,99751$$

La disponibilidad para una configuración serial se obtiene aplicando la fórmula 6.4, resultando:

$$a_{\text{serial}} = (0.99502)(0.99751) = 0.99254$$

y para una configuración paralela, aplicamos la fórmula 6.5, obteniendo:

$$a_{\text{paralelo}} = 1 - (1 - 0.99502)(1 - 0.99751) = 0.99999$$

Como se presume, la disponibilidad del sistema paralelo tiene una confiabilidad superior a la serial, porque se trata de un sistema redundante.

Aplicando estos conceptos a nuestro sistema de Comunicación entre Procesadores, específicamente en nuestra tarjeta ISA y agrupando convenientemente los módulos funcionales tenemos la Tabla 6.1 donde figuran los MTTF y MTTR, obtenidos de algunas referencias bibliográficas, análisis y observaciones:

ELEMENTOS	MTTF hrs	MTTR hrs
PCB	50,000	5
CPU	10,000	3
MEMORIA	20,000	2
HDLC	20,000	2
MATRIZ	20,000	2

TABLA 6.1

Usando la fórmula 6.1 tenemos las siguientes disponibilidades individuales:

$$a_{PCB} = 0,9999$$

$$a_{CPU} = 0,9997$$

$$a_{MEM} = 0,9999$$

$$a_{HDLC} = 0,9999$$

$$a_{MAT} = 0,9999$$

La disponibilidad del sistema, que debe tener una configuración serial, es de 0.9996; esto significa que en 10 000 hrs. de funcionamiento se espera 4 horas de falla.

Si la especificación de confiabilidad es más exigente se podría disponer de un sistema redundante, en otras palabras duplicar la tarjeta ISA. Ahora la configuración del sistema sería paralelo y aplicando las fórmulas tenemos:

$$a_{sistema} = 1 - (1 - 0,9996)^2 = 0,9999999$$

Teniendo prácticamente un sistema de muy alta confiabilidad.

## 6.2 Análisis de costos

En referencia a los costos se deben analizar los

criterios mencionados en el capítulo III, haciendo tablas comparativas eligiendo la mejor opción.

Muchos proyectos se truncan por motivos de altos costos, por lo tanto es importante los planteamientos del diseño y especificar modelos de confiabilidad factibles; como un ejemplo de la sección anterior para obtener una alta confiabilidad prácticamente se duplican los costos y no solamente en cuanto al hardware, el sistema se complica aun mas porque se tiene que desarrollar un software adicional para manejar sistemas redundantes.

A continuación resumimos los criterios que deben evaluarse y fueron explicados en el capítulo III:

- Costos de desarrollo
- Costos de implementación
- Costos de instalación
- Costos de mantenimiento
- Costos de inventarios de partes de reserva
- Costos de conversión
- Costos de vida útil (depreciación y amortización)
- Costos de entrenamiento
- Costos de posibles cambios requeridos
- Costos de documentación
- Costos del estudio para generación del sistema
- Costos de operación
- Costos de personal técnico

El costo referencial del hardware de la tarjeta ISA que se propone en la tesis, se muestra en la Tabla 6.2

Suponiendo, que la tarjeta se implemente en un tamaño estandard, doble eurocard, el costo total de la tarjeta se aproximaría a lo siguiente:

- Componentes (Tabla 6.2)	\$ 75.85
- Fabricación de la Tarjeta (doble cara, hueco metalizado) PCB	\$ 70.00
- Ensamblaje (mano de obra)	\$ 50.00
Ferretería y accesorios	\$ 40.00

Haciendo un total de \$ 235.85, considerando el costo de los componentes a precios FOB. Esta tarjeta quedaría lista para las pruebas de laboratorio.

## CONCLUSIONES Y RECOMENDACIONES

La propuesta de la presente tesis ha sido implementada en el prototipo de la Central ANTARA B del proyecto ANTARA, desarrollado por ENTEL PERU S.A. Actualmente está pasando las pruebas de campo en las ciudades del Valle Sagrado de los Incas y en Cañete.

De la experiencia del sistema implementado, nos permite obtener las siguientes conclusiones y recomendaciones:

La técnica del uso de las vías de comunicación, en los sistemas de conmutación análogos y digitales, para la comunicación entre sus procesadores, no es nueva; la novedad de las centrales ANTARA B es el uso de controladores de protocolo HDLC, en los diferentes procesadores, que simplifica bastante el sistema, puesto que el controlador ya tiene implementado los niveles 1 y 2 del modelo ISA (Interconexión de Sistemas Abiertos) y las recomendaciones del CCITT y que las pruebas de campo demuestran su validez.

Esta implementación ha permitido, una gran flexibilidad en la configuración de redes de telecomunicaciones basadas en las centrales ANTARA B; tal es el caso del Valle Sagrado de los Incas. La central madre se instaló en la ciudad de Urubamba atendiendo a las unidades remotas de Pisac, Calca,

Anta y Ollantaytambo, interconectados a través de enlaces digitales radiales (Nokia). Las otras centrales se han instalado en las ciudades de Cañete y Aguas Calientes (Macchu Picchu), sin unidades remotas, teniendo las posibilidades de atender unidades remotas si se requiere, sin mayores modificaciones.

Para desarrollar un sistema, es importante definir claramente las especificaciones de los requerimientos, definiendo los límites lógicos y físicos; con la finalidad de prever su integración con otros módulos o sistemas.

La filosofía del diseño estructurado jerárquico modular, ofrece grandes facilidades para la implementación de sistemas, existiendo una relación casi biunívoca entre el hardware y software.

El requisito indispensable, para una buena aplicación de la filosofía del diseño estructurado en hardware y software, es el uso de componentes electrónicos especializados y estandarizados, que ayudan a definir los bloques funcionales del sistema.

Los equipos desarrollados con la filosofía del diseño estructurado, ofrecen grandes ventajas como: la gran flexibilidad para integrar nuevos módulos y ampliar su cobertura, equipos compactos con bajo consumo de energía, facilidades de operación y mantenimiento, alta confiabilidad posibilitando la implementación de

estructuras redundantes, facilidades para escalar a nuevos productos mas potentes y otros.

La definición de los modelos teóricos nos ayudan a visualizar a nivel macro el ámbito donde trabajará el sistema, de allí su importancia de definirlos adecuadamente.

Seguir la metodología de diseño de proyectos, esbozada en la presente tesis permitirá desarrollar proyectos para implementar sistemas, desde la concepción de la idea hasta su implementación y operación, en forma organizada y segura disminuyendo costos y tiempos de implementación.

Es importante la programación de pruebas de laboratorio y de campo en las etapas de diseño hardware, software y el sistema integrado hardware-software, para detectar a tiempo posibles problemas de diseño.

La documentación técnica debe implementarse paralelamente con el desarrollo del proyecto, haciendo continuas actualizaciones, hasta el documento final.

En la implementación del software se debe considerar herramientas de diseño y de depuración, pudiendo ser plataformas comerciales o desarrollos propios.

Similarmente, para el diseño hardware se debe contar con el instrumental adecuado.

**Multiple Processor System for Real-time Applications**

Burt H. Liebowitz - John H. Carson

Prentice-Hall, Inc. - 1985

**Real-Time Software**

Robert L. Glass

Prentice-Hall, Inc. - 1983

**Digital Hardware Design**

John B. Peatman

McGraw-Hill - Book Company - 1980

**Volume VI - Fascicle VI.7****Specifications of Signalling System No. 7**

CCITT - Blue Book 1989

**Sistemas de Transmisión - Tomo 1**

UIT 1976

**Study Group XI - Report R-189, R-190, R-191**

CCITT - White Book 1989-1992

**Manual Mitel 1987 - 1990****Telecommunications Databook**

National Semiconductor Corporations 1987

**Microcommunications Handbook**

Intel 1987

**Telephone Switching Engineering**

Nippon Telegraph &amp; Telephone Corporations - JICA