

171
24



**UNIVERSIDAD NACIONAL AUTONOMA
DE MEXICO**

FACULTAD DE INGENIERIA

**DISEÑO DE UNA RED DE
COMUNICACIONES DISTRIBUIDAS Y
CONTROL DE ENERGIA ELECTRICA**

T E S I S

QUE PARA OBTENER EL TITULO DE:
INGENIERO MECANICO ELECTRICISTA
P R E S E N T A N :
VELAZQUEZ MONTERO JAVIER
VELAZQUEZ MONTERO RICARDO

DIRECTOR DE TESIS:

ING. EDUARDO RAMIREZ SANCHEZ

MEXICO, D. F.

1996

**TESIS CON
FALLA DE ORIGEN**

**TESIS CON
FALLA DE ORIGEN**



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

QUEREMOS AGRADECER A TODAS
AQUELLAS PERSONAS QUE NOS
BRINDARON SU COMPRENSIÓN Y
APOYO EN EL TRANCURSO DE
NUESTROS ESTUDIOS.

A NUESTROS PADRES, POR LA
ORIENTACIÓN BRINDADA DESDE EL
INICIO DE NUESTRAS VIDAS, Y POR
SUS ENSEÑANZAS, QUE SIRVIERON
DE BASE PARA NUESTRA FORMACIÓN.

A NUESTROS HERMANOS, POR SU
CONFIANZA Y APOYO.

A NUESTROS MAESTROS, POR
COMPARTIRNOS UNA PARTE DE SUS
CONOCIMIENTOS.

A NUESTRA ALMA MATER LA UNAM,
QUE FUE EL ESCENARIO DE
NUESTROS MEJORES TIEMPOS.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO I TECNOLOGÍA LONWORK	
1.1.- INTRODUCCIÓN	4
1.2.- LOS ELEMENTOS DE LONWORK	6
1.2.1 Los Circuitos Integrados NEURON	6
1.2.2 Transmisores/Receptores LONWORK	7
1.2.3 El Protocolo LONTALK	7
1.2.4 Banco de Desarrollo LONBUILDER	7
1.3.- EL SISTEMA LONBUILDER	8
1.3.1 Estación de Desarrollo LONBUILDER	9
1.3.2 Tarjetas Procesador LONBUILDER	11
1.3.2.1 Emulador NEURON LONBUILDER	13
1.3.2.1 Computadora en una tarjeta (SBC) LONBUILDER	13
1.3.2.3 Ruteador LONBUILDER	14
1.3.3 Tarjeta de Expansión LONBUILDER de Transmisión/Recepción	14
1.3.4 Tarjeta de Expansión LONBUILDER de Entrada/Salida	15
1.4.- SOFTWARE LONBUILDER	15
1.4.1 Medio Ambiente de Desarrollo Integrado LONBUILDER	16
1.4.2 Equipo de Desarrollo NEURON C	17
1.4.2.1 Compilador NEURON C	18
1.4.2.2 Depurador NEURON C	16
1.4.3 Herramientas para el Manejo de la Red	18
1.4.3.1 Manejador de la red	18
1.4.3.2 Analizador de protocolo	19
CAPÍTULO II CIRCUITOS INTEGRADOS NEURON	
2.1.- INTRODUCCIÓN	20
2.2.- RECURSOS DE HARDWARE DE LOS CIRCUITOS INTEGRADOS NEURON	23
2.2.1 Unidades del Procesador	23
2.2.2 Memorias	26
2.2.2.1 EEPROM	26
2.2.2.2 RAM estática	27
2.2.2.3 ROM programada	27
2.2.3 Interface de Memoria Externa	27
2.2.4 Entrada/Salida	31
2.2.4.1 Once terminales bidireccionales E/S	31
2.2.4.2 Dos temporizadores/contadores de 16 bits	31
2.2.5 Comunicación de la Red	32
2.2.6 Sistema de Reloj	32
2.2.6.1 Circuito la divisora de reloj	34
2.2.7 Funciones Adicionales	36
2.2.7.1 Modo dormido /despierto	36
2.2.7.2 Temporizador perro de guardia	36
2.2.8 Terminal de Servicio	37

2.3.- ORGANIZACIÓN DEL PROCESADOR	38
2.3.1 Los Procesadores	38
2.3.2 Organización de las Memorias	39
2.3.2.1 Organización de la memoria MC143150	39
2.3.2.2 Organización de la memoria MC143120	40
2.4.- INTERFACE DE ENTRADA/SALIDA	41
2.4.1 Consideraciones de Hardware	42
2.4.2 Interface Digital	43
2.4.2.1 Bit E/S	43
2.4.2.2 Byte E/S	44
2.4.2.3 Detección de nivel	44
2.4.2.4 Nibble E/S	45
2.4.3 Funciones de Interface en Paralelo E/S	46
2.4.4 Interface Serial	47
2.4.4.1 Bit de corrimiento de E/S	47
2.4.4.2 Función NeuroAlambra:ta de E/S	47
2.4.4.3 Serial de E/S	47
2.4.5 Función de Interface Temporizador/Contador	47
2.4.5.1 Funciones de entrada del temporizador/contador	48
2.4.5.2 Funciones de salida del temporizador/contador	52

CAPÍTULO III PROTOCOLO LONTALK

3.1.- INTRODUCCIÓN	57
3.1.1 El Canal Físico	59
3.1.2 Nombrando, Direccionando y haciendo Rutas	61
3.1.2.1 Dominio	62
3.1.2.2 Subred	63
3.1.2.3 Nodo	66
3.1.2.4 Grupos	66
3.1.2.5 El ID único	67
3.1.2.6 Los formatos de dirección	67
3.1.2.7 Manejo de redes y generación de direcciones	67
3.1.2.8 Ruteadores y puentes	68
3.1.3 Servicio de Comunicación	71
3.1.3.1 Eficiencia, tiempo de respuesta, seguridad y confiabilidad	71
3.1.3.2 Seleccionando servicios de mensaje para la compatibilidad y eficiencia	71
3.1.3.3 Detección de colisión	72
3.1.3.4 Prioridad	72
3.1.3.5 Autenticidad	73
3.1.4 Temporizadores del Protocolo LONTALK	74
3.1.5 Paquetes Fluyendo al Circuito NEURON	74
3.1.6 Interpretación de los Datos	76
3.1.6.1 Variables de la red	76
3.1.7 Compatibilidad de la Aplicación	77
3.1.8 Servicio de Protocolo y Parámetros	77
3.1.9 Límites y Regiones	79
3.1.10 La Tabla de Dirección	80
3.2.- MEDICIÓN DEL TIEMPO DE RESPUESTA DEL LONTALK	80

CAPÍTULO IV	TRANSMISORES/RECEPTORES LONWORK	
4.1.-	INTRODUCCIÓN	82
4.2.-	COMUNICACIÓN DE LA RED	84
4.2.1	Modo Directo	85
4.2.1.1	Modo directo de terminación simple	86
4.2.1.2	Modo directo diferencial (conexión directa)	90
4.2.1.3	Modo directo diferencial (transformador acoplado)	95
4.2.1.4	Transmisor/receptor RS-485	97
4.2.2	Modo de Propósito Especial	98
4.2.2.1	E/S física del modo de propósito especial	100
4.2.2.2	Estado de los bits	101
4.3.-	TRANSMISORES/RECEPTORES LONBUILDER	103
4.3.1	Transmisor/Receptor LONBUILDER de Par Trenzado	103
4.3.2	Transmisor/Receptor LONBUILDER TP-RS485	103
4.3.3	Transmisor/Receptor LONBUILDER RF	104
CAPÍTULO V	DISEÑO DE LA RED USANDO LONBUILDER	
5.1.-	INTRODUCCIÓN	105
5.1.1	Definición del Proyecto	105
5.1.1.2	Transmisión de los datos del HC16 al nodo LONWORK	106
5.1.1.3	Red LONWORK	106
5.1.1.4	Visualización de los datos	107
5.1.2	Desarrollo del Proyecto	107
5.1.2.1	Identificación de los nodos y asignación de su función	107
5.1.2.2	Definición de la interface interna de cada nodo	108
5.1.2.3	Los programas de la aplicación de cada nodo	108
5.1.2.4	Depuración y prueba individual de cada nodo	114
5.1.2.5	Instalación de los nodos sobre la red	114
5.1.3	Transmisor/Receptor utilizado	115
CAPÍTULO VI	INTERFACE ENTRE EL CIRCUITO INTEGRADO NEURON Y MCU'S	
6.1.-	INTRODUCCIÓN	117
6.1.1	Interface en Paralelo	117
6.1.1.1	Modo maestro/esclavo A	117
6.1.1.2	Modo maestro/esclavo B	118
6.1.1.3	Paso de testigo	119
6.1.1.4	Protocolo de saludo	120
6.1.2	Bit de Corrimiento de Entrada/Salida	120
6.1.3	Función NeuroAlambrada de Entrada/Salida	121
6.1.3.1	Modo maestro NeuroAlambrado	121
6.1.3.2	Modo esclavo NeuroAlambrado	121
6.1.4	Interface Serial	121
6.2.-	ELECCIÓN DE LA INTERFACE DE LA RED LONWORK	122
CONCLUSIONES		127
APÉNDICE		A-1
BIBLIOGRAFÍA		

INTRODUCCIÓN

La tecnología ha jugado un papel más que relevante y estratégico, indispensable para dar continuidad a nuestro entorno. La ingeniería, como creadora de la infraestructura tecnológica, ha tenido que avanzar a pasos agigantados, tratando de resolver los problemas que las necesidades de la sociedad han ido planteando.

Dentro del marco de la apertura económica, con el Tratado de Libre Comercio de Norteamérica, se observa claramente que las empresas mexicanas deben ser lo suficientemente competitivas para no perecer ante compañías extranjeras y nacionales. El alto nivel de tecnología de empresas extranjeras, es la base fundamental para lograr niveles de calidad elevados en sus productos, reducción de gastos de operación, administración, así como un mayor aprovechamiento de las materias primas.

Es necesario, por lo tanto, que en México se apliquen tecnologías de punta para poder competir con los mercados internacionales. El papel de los ingenieros en este momento es crucial para el ámbito mexicano, ya que debemos ser capaces de aprovechar la infraestructura actual, y optimizar la utilización de los recursos disponibles, apoyados de la nueva tecnología que nos llega de los diferentes países, primordialmente de Estados Unidos.

Los microprocesadores han hecho posible muchos avances científicos, industriales y comerciales que no se hubiesen podido lograr por otros medios. A lo largo de los años hemos presenciado cómo se ha ido asumiendo, dirigiendo y absorbiendo el fenómeno microprocesador, y el hecho es que hoy es imposible prescindir de él en la realización de un diseño digital no trivial.

Por otro lado, en los últimos años la industria de las redes de gran cobertura ha madurado y es hoy por hoy un sector estable. Las redes locales, por el contrario, constituyen un campo relativamente nuevo. La tecnología en la que se basan, empezó a adquirir interés a mediados de los setenta, y es en la actualidad uno de los sectores de más rápido crecimiento dentro de la Industria de comunicaciones.

La automatización de edificios, fábricas, máquinas, vehículos, etc. han sido de gran importancia en la vida moderna. La tecnología LON (redes de operación local) ofrece poderoso medio para implementar una variedad de sistemas distribuidos, los cuales sensan, monitorean y controlan, además de otras aplicaciones. LONWORK es una colección de herramientas y componentes, los cuales pueden ser utilizados para aplicaciones de construcción específica LON.

En México solo existen un par de estaciones de desarrollo para la construcción de LON (redes de operación local), contando la Unidad de Diseño Electrónico de la UNAM con una de estas estaciones de desarrollo. Por lo que, en México, no se cuenta con mucha información sobre esta nueva tecnología. Por tal motivo y para poder familiarizarnos con esta tecnología, es necesario dar a conocer los componentes que integran la tecnología LON, su funcionamiento, así como sus principales características. Además, de que posteriores trabajos que deseen utilizar esta tecnología, cuenten con información suficiente para poder realizarlos.

Para poder alcanzar los objetivos anteriormente expuestos el trabajo se divide en siete capítulos, de los cuales cuatro de ellos se enfocan exclusivamente a dar una explicación de la tecnología LON (componentes que lo integran, funcionamiento, principales características, etc.).

En el capítulo 1 "Tecnología LONWORK", Se describen las herramientas y componentes necesarios para la fabricación de redes LON, así como la comparación entre las redes LON y LAN.

El capítulo 2 "Circuitos Integrados Neuron", describe los Circuitos Integrados Neuron, que son la base de la tecnología LON. Como están organizados, como funcionan, cuales son sus componentes principales, y como interactúan entre ellos son algunas de los puntos que se exponen.

Capítulo 3 "Protocolo LONTALK", en este capítulo se explica el funcionamiento del protocolo que utiliza la tecnología LON, así como algunos aspectos referentes al direccionamiento y ruteo de los canales de comunicación. Se explican también algunos servicios que proporciona este protocolo.

Capítulo 4 "Transmisores/Receptores LONWORK", se enfoca principalmente en dar algunas alternativas para el uso de transmisores/receptores, así como alternativas para el uso de diferentes medios de comunicación. Y los modos en los que puede trabajar el puerto de comunicaciones del Circuito Integrado Neuron.

En los dos capítulos restantes se tiene por objetivo el diseño y la construcción de una red de comunicaciones distribuidas y control de energía eléctrica utilizando la tecnología LONWORK

En el capítulo 5 "Diseño de la red usando LONBUILDER" se tiene por objetivo dar a conocer el diseño del sistema, tomando como referencia lo descrito en los capítulos anteriores.

Capítulo 6 "Interface entre Circuitos Integrado Neuron y MCU's", tiene por objetivo explicar las diferentes interfaces que pueden realizarse entre un Circuito Integrado Neuron y los MCU's (microcontroladores). También, se explica la interface realizada entre los microprocesadores HC16 y el Circuito Integrado Neuron, necesaria para el desarrollo del proyecto.

Finalmente, se concluye con una serie de reflexiones, resultado del desarrollo de este trabajo, y se incluye un apéndice con las principales diferencias que existen entre el lenguaje de programación C y el lenguaje de programación NEURON C utilizado en este trabajo.

CAPÍTULO I

TECNOLOGÍA LONWORK

1.1 INTRODUCCIÓN

La tecnología LON (Local Operating Network - redes de operación local) ofrece medios poderosos para implementar una variedad de sistemas distribuidos, los cuales sensan, monitorean y controlan, además de otras aplicaciones. LONWORK es una colección de herramientas y componentes, los cuales pueden ser utilizados para aplicaciones de construcción específica de LONs. LONWORK puede ser usado dentro de una gran variedad de caminos para la automatización de fábricas, vehículos, casas y maquinaria a un bajo costo.

Las LONs constituye una clase de tecnología en la cual se incluyen dispositivos inteligentes, en donde la comunicación entre ellos es posible gracias a una variedad de medios de comunicación que utilizan un protocolo estándar. En este sentido, una LON es similar a una red de área local (LAN - Local Area Network) que permite la comunicación entre computadoras. La tecnología LON soporta un sistema distribuido, esto permite que no sea necesario un sistema de control central.

Lo que distingue a LONs de LANs es su propósito. LANs son diseñados para mover datos (tales como documentos, imágenes y base de datos) entre computadoras. Una ejecución de LAN es vista en términos de su rendimiento efectivo, por lo general medido en Megabits transmitidos por segundo. Una LON está diseñada para sensar y controlar mensajes los cuales son generalmente muy cortos, conteniendo comandos e información de estados que habilitan la acción. La ejecución de LON es vista en términos del tiempo de respuesta y las operaciones completadas por segundo.

La compañía Echelon, generadora de la tecnología LON, tiene una amplia experiencia en tecnología de comunicaciones, tecnología de control y tecnología en semiconductores, dando como resultado una gran colección de componentes y herramientas llamadas LONWORK, los cuales, son fáciles de instalar y flexibles. Además diferentes componentes (compatibles con LONWORK) podrían elegirse para el diseño de productos basados en LONWORK.

La combinación lógica de los dispositivos de las LONs, que cuentan con inteligencia artificial y dispositivos de comunicación, son llamados nodos. Un nodo puede ser un dispositivo alambrado semejante a un interruptor, un sensor analógico/digital o a un actuador. Los nodos LONWORK generalmente consisten de los siguiente:

- Uno o más sensores o actuadores y el hardware asociado a la aplicación.
- Un circuito integrado llamado NEURON.
- Un transmisor/receptor que proporciona el enlace para la comunicación.

Ver figura 1.1.

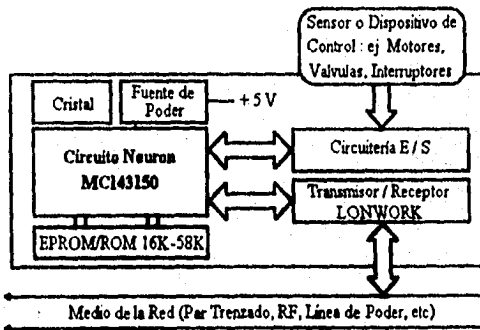


Figura 1.1 Diagrama de Bloques de un Nodo Típico

1.2 LOS ELEMENTOS DE LONWORK

La tecnología LONWORK es una plataforma completa para implementar sistemas de redes con un control distribuido. Estos sistemas consisten de dispositivos inteligentes o nodos que interactúan con su medio ambiente, y se comunican entre ellos a través de una gran variedad de medios de comunicación, usando mensajes basados en un protocolo de control. LONWORK incluye todos los elementos requeridos para diseñar, construir y mantener LONs: los Circuitos Integrados NEURON y el transmisor/receptor LONWORK que forman la parte física del nodo; el protocolo LONTALK que proporciona una comunicación formal y los fundamentos de las operaciones internas entre los productos; y un banco de desarrollo LONBUILDER que habilita un desarrollo rápido de productos y redes.

1.2.1 Los Circuitos Integrados NEURON

El corazón de cada nodo es un miembro de la familia de Circuitos Integrados NEURON, ya sea el NEURON 3120 o el NEURON 3150. Ambos proporcionan una solución para las comunicaciones, la entrada/salida y el control para muchas aplicaciones; El NEURON 3150 también soporta memoria externa para aplicaciones más complejas. Ambos dispositivos contienen tres CPUs idénticos de 8 bits; de los cuales dos son dedicados a la comunicación, y el tercero está dedicado al código de la aplicación.

El NEURON 3120 contiene memorias ROM, RAM y EEPROM. El NEURON 3150 contiene memorias RAM y EEPROM y provee una interface de memoria externa. Ambos contienen 11 terminales bidireccionales de entrada/salida para la interface de aplicaciones de hardware, una interface de 5 terminales para un transmisor/receptor LONWORK, un temporizador/contador, y muchas otras características, incluyendo un número serial de 48 bit permanentemente programado dentro de cada dispositivo.

Otras características únicas de los Circuitos NEURON es que están microprogramados (firmware). El NEURON 3120 contiene dentro de 10k de memoria ROM (o en los primero 10 K de memoria externa direccionada para el NEURON 3150) el microprograma NEURON. El microprograma implementa el llenado del protocolo de comunicaciones LONTALK y maneja todos los detalles de la lista de tareas, manejo de entrada y salida, manejo de la red, comunicación, y control interno.

1.2.2 Transmisores/Receptores LONWORK

Una característica de las redes LONWORK es su habilidad para comunicarse a través de diferentes medios en una red específica.

Los primeros productos desarrollados por Echelon, son un grupo de módulos que permiten la comunicación de los circuitos NEURON sobre líneas de potencia, radio frecuencia, e instalación de alambres eléctricos de par trenzado. Además, la flexibilidad de comunicación de los puertos de los circuitos NEURON habilitan al desarrollador para la implementación de un transmisor/receptor para otros medios de comunicación (ej. coaxial, fibra óptica, etc.) para necesidades especiales que se tengan.

1.2.3 El Protocolo LONTALK

Echelon utiliza LONTALK como el esqueleto para toda la comunicación que ocurra sobre LON. El protocolo LONTALK es versátil y soporta un rango amplio de aplicaciones. LONTALK es definido por el International Standard Organization (ISO) y el Open Systems Interconnection (OSI) que se utiliza como modelo de referencia.

El protocolo LONTALK está diseñado para proveer la entrega de mensajes cortos de control a través de una variedad de medios de comunicación. Con el apropiado transmisor/receptor, los nodos pueden comunicarse entre sí a través de algún medio, como par trenzado, línea de potencia y radio frecuencia.

Las LONs que siguen el protocolo LONTALK son llamados redes LONWORK, y los nodos que utilizan el protocolo LONTALK son llamados nodos LONWORK, o simplemente nodos.

1.2.4 Banco de Desarrollo LONBUILDER

El banco de desarrollo LONBUILDER es un medio que provea las herramientas necesarias para crear aplicaciones LONWORK. Una aplicación LONWORK consiste de dispositivos inteligentes, o nodos, que son conectados con uno o más medios de comunicación y que se comunican entre ellos usando el protocolo LONTALK. El sistema LONBUILDER es usado para programar nodos, para enviar

mensajes, para cambiar diferentes condiciones, y para tomar decisiones en respuesta a mensajes que ellos reciben. Como ya se había mencionado cada nodo contiene: un circuito NEURON, el cual es un dispositivo programable que implementa el protocolo LONTALK y ejecuta la función de aplicación deseada del nodo; un transmisor/receptor, para proveer la interface mecánica y electrónica entre los circuitos NEURON de los nodos y el medio de comunicación; y circuitos para conectar el circuito NEURON a dispositivos de entrada/salida como desplegados, etc.

1.3 EL SISTEMA LONBUILDER

El sistema LONBUILDER facilita el desarrollo independiente de cada uno de los nodos cuya función puede ser simple o compleja, y la integración de estos nodos dentro de aplicaciones LONWORK, los cuales pueden ejecutar tareas más complejas y sofisticadas.

El banco de desarrollo LONBUILDER (ver figura 1.2) consiste de una integración entre hardware y software, el cual opera en conjunto con una computadora personal. Durante el desarrollo, una aplicación LONWORK típica crece desde una red de dos nodos hasta una red de varios nodos. El sistema LONBUILDER ayuda a la expansión de nodos, y se compone de:

- La estación de desarrollo LONBUILDER
- Las tarjetas procesador LONBUILDER
- Las tarjetas de expansión LONBUILDER

Sobre una simple PC, el sistema LONBUILDER provee un medio ambiente de desarrollo que puede crecer desde un par de aplicaciones emuladas de nodos, residentes en la estación de desarrollo, hasta un sistema distribuido de 24 nodos emulados y 256 nodos remotos.

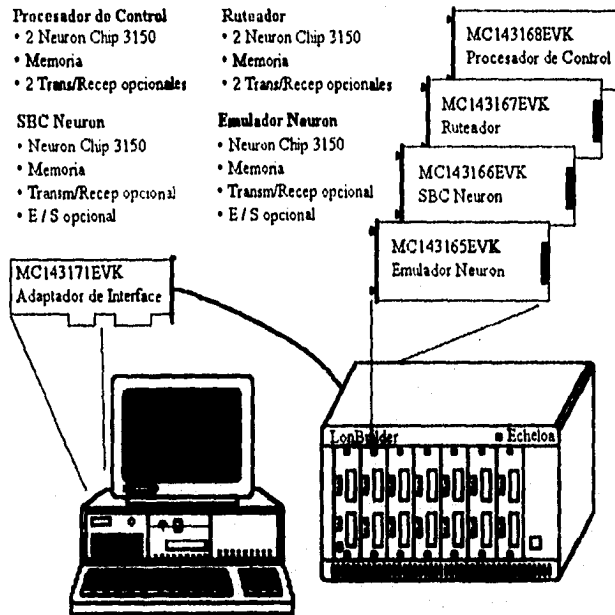


Figura 1.2 Banco de Desarrollo LONBUILDER

1.3.1 Estación de Desarrollo LONBUILDER

La estación de desarrollo LONBUILDER incluye:

- Un recinto con suministro de poder y conectores de hardware que soporta hasta seis tarjetas procesador opcionales.
- Un Procesador de Control LONBUILDER, el cual proporciona el control para una estación de desarrollo, manejando la comunicación de comandos y datos entre un adaptador de interface LONBUILDER (instalada en una PC) y las tarjetas procesador (instaladas en la estación de desarrollo). Un cable de interface enlaza el Procesador de Control LONBUILDER al Adaptador de Interface LONBUILDER. El Procesador de Control incluye los nodos manejador de red y analizador de protocolo, los cuales son nodos dedicados, usados para el software Manejador de Red LONBUILDER y software Analizador de Protocolo LONBUILDER.

En la figura 1.3 se muestra la Estación de Desarrollo LONBUILDER

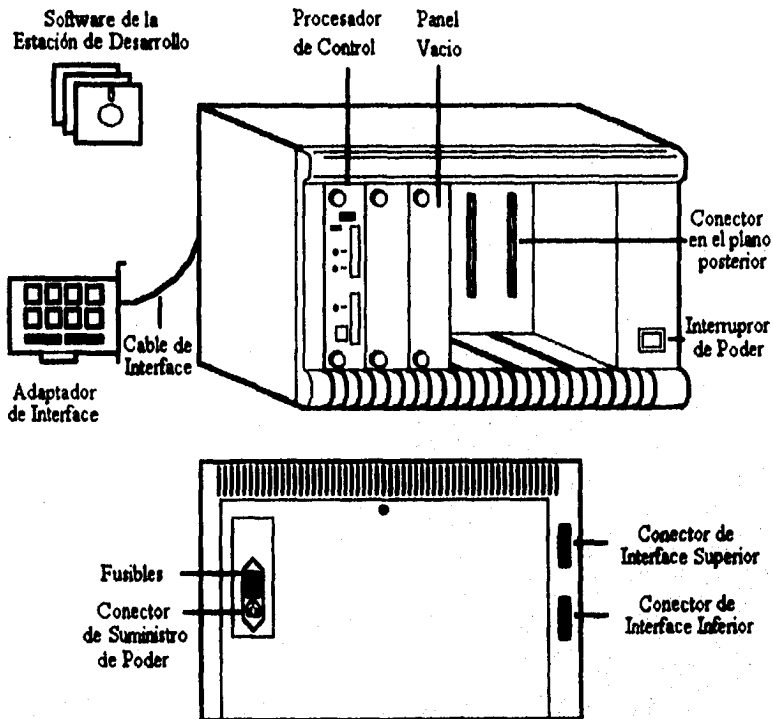


Figura 1.3 Estación de Desarrollo LONBUILDER

Un adaptador de interface LONBUILDER instalado en la PC provee una alta velocidad de comunicación entre el software de la PC y las tarjetas procesador en la estación de desarrollo. Hasta cuatro estaciones de desarrollo pueden ser conectadas en una PC, soportando un total de 24 tarjetas procesador, estas conexiones se muestran en la figura 1.4.

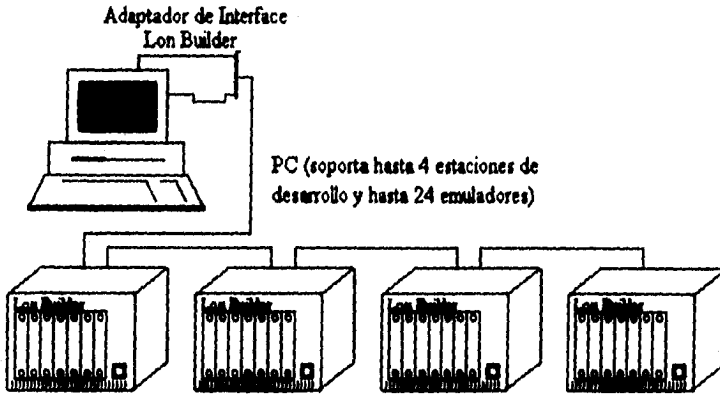


Figura 1.4 Conexión de Estaciones de Desarrollo

En el banco de desarrollo LONBUILDER existen seis ranuras que pueden alojar algunas mezclas de las tarjetas procesador LONBUILDER.

1.3.2 Tarjetas Procesador LONBUILDER

Existen tres tipos de tarjetas procesador LONBUILDER: Emulador NEURON LONBUILDER, Computadora en una tarjeta (SBC) LONBUILDER, y Ruteador LONBUILDER.

- Emulador NEURON para la depuración del software y hardware de los prototipos.
- LONBUILDER SBC (computadora en una tarjeta) para el uso en el desarrollo con una PC o remotamente con un transmisor/receptor y utilizando una fuente de alimentación externa.
- Ruteador LONBUILDER para proporcionar comunicación entre dos diferentes canales o medios de comunicación (ej. instalación de par trenzado y radio frecuencias).

Las tarjetas procesador pueden aceptar dos tipos opcionales de tarjetas de expansión. Las tarjetas de expansión pueden ser transmisores/receptores LONBUILDER, tarjetas de entrada/salida LONBUILDER, transmisores/receptores, o Interface entrada/salida.

En la figura 1.5 se muestran las tarjetas procesador y en la tabla 1-1 algunas características de éstas.

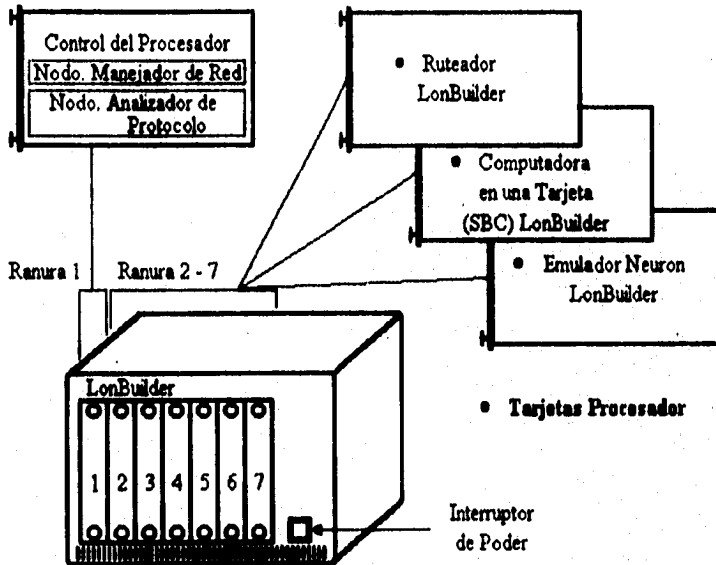


Figura 1.5 Tarjetas Procesador LONBUILDER

Tabla 1-1 Componentes de la tarjeta procesador LONBUILDER

Tarjeta Procesador	Procesadores	Cantidad de Memoria	Aceptación opcional de tarjetas de expansión transmisor/receptor	Aceptación opcional de tarjetas de expansión de E/S	Soporta la depuración NEURON C
Emulador	Un circuito NEURON 3150	64 KB de aplicación 64 KB de control	Si una tarjeta	Si una tarjeta	Si
SBC	Un circuito NEURON 3150	64 KB de aplicación	Si una tarjeta	Si una tarjeta	No
Ruteador	Dos circuitos NEURON 3150	32 KB de sistema para el circuito NEURON 3150	Si dos tarjetas	No	No

1.3.2.1 Emulador NEURON LONBUILDER

Los nodos son inicialmente desarrollados en el Emulador NEURON LONBUILDER, una tarjeta procesador que permite la depuración del software usando el depurador NEURON C. El emulador incorpora un circuito integrado NEURON 3150 con 64 KB de RAM. También incluye una memoria adicional de 64 K para usarla con el depurador en el control del programa de la aplicación.

El emulador siempre reside en la estación de desarrollo, donde proporciona el soporte de hardware para cargar la aplicación, hacer un punto de interrupción en el correr del programa de la aplicación, realizar pasos simples en el programa, hacer reinicio/comenzó/paro, y utilizar una memoria de protección de lectura/escritura. El emulador también proporciona un controlador de porcentaje de reloj de software el cual puede ser de 10 MHz, 5 MHz, 2.5 MHz, 1.25 MHz, y 625 KHz. El software LONBUILDER puede emular el mapa de memoria del circuito NEURON 3120 sobre los emuladores NEURON 3150.

1.3.2.2 Computadora en una tarjeta (SBC) LONBUILDER

El SBC LONBUILDER es una computadora en una tarjeta incorporando un circuito NEURON 3150 con 64 KB de memoria no volátil RAM. Después de depurar la aplicación de software sobre el emulador, éste puede ser bajado hacia un SBC. El SBC puede ser instalado en la estación de desarrollo o usado remotamente con una fuente de poder externa. El SBC acepta igual número de tarjetas de expansión que el emulador, permitiendo probar el prototipo.

1.3.2.3 Ruteador LONBUILDER

El Ruteador LONBUILDER es una tarjeta procesador que contiene dos circuitos NEURON 3150. Cada ruteador soporta dos tarjetas de expansión transmisor/receptor opcionales para proporcionar rutas entre dos comunicaciones o canales. Los ruteadores pueden ser instalados de la siguiente forma:

- Ruteador de aprendizaje, el cual monitorea el tráfico de redes para aprender la topología de la red. El ruteador utiliza la información de la topología de la red para seleccionar paquetes de rutas entre canales.
- Ruteador configurado, el cual utiliza tablas de rutas basadas en la configuración de la red para la ruta de los paquetes entre canales. El manejador de red LONBUILDER configura automáticamente las tablas de rutas.
- Puente, el cual transmite todo el paquete entre los dos canales conectados sobre el dominio en el cual el puente es instalado.
- Repetidor, el cual transmite todo el paquete entre los dos canales conectados sin estimar el dominio en el cual el repetidor es instalado.

Un ruteador LONBUILDER puede ser instalado en la estación de desarrollo o usado remotamente con una fuente de poder externa.

1.3.3 Tarjeta de Expansión LONBUILDER de Transmisión/Recepción

Las tarjetas procesador requieren un tarjeta de expansión transmisor/receptor para proporcionar la interface física de una comunicación entre canales. Los transmisores/receptores LONBUILDER son útiles para redes con par trenzado y radio frecuencia. Se puede desarrollar transmisores/receptores con un diseño propio.

Las tarjetas transmisor/receptor LONBUILDER provee la interface física entre un Emulador, SBC, Ruteador, o procesador de control y un canal de la red.

1.3.4 Tarjeta de Expansión LONBUILDER de Entrada/Salida

Las tarjetas procesador LONBUILDER acepta una segunda tarjeta de expansión además de una tarjeta de expansión transmisor/receptor. Hay diferentes opciones para desarrollar los prototipos de interface entrada/salida:

- Desarrollar una tarjeta de expansión entrada/salida.
- Usar la tarjeta de entrada/salida LONBUILDER para desarrollar el prototipo de interfaces entrada/salida. La tarjeta de entrada/salida LONBUILDER incluye memoria intermedia (buffers) para las señales entrada/salida del circuito NEURON y prototipos de hardware.
- Utilizar la tarjeta de interface de aplicación LONBUILDER para realizar la interface con dispositivos de entrada/salida y nodos externos para la estación de desarrollo.

1.4 SOFTWARE LONBUILDER

El Software LONBUILDER proporciona herramientas para la programación de la aplicación, también proporciona herramientas manejadoras de red que conocen las necesidades del desarrollo de la aplicación LONWORK.

Las herramientas para la programación de la aplicación LONBUILDER incluyen el manejador del proyecto, editor, compilador, y depurador del programa. Todas estas herramientas están enlazadas a través de un Medio Ambiente de Desarrollo Integrado (IDE), ilustrado en la figura 1.6.

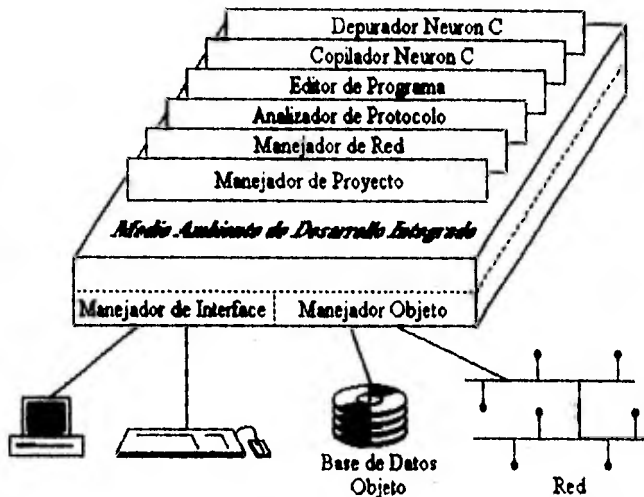


Figura 1.6 Herramientas para la Programación de la Aplicación

El banco de desarrollo LONBUILDER provee este poderoso juego de herramientas integradas de software:

- El Medio Ambiente de Desarrollo Integrado LONBUILDER incluye una base de datos, un manejador de proyecto, y un editor integrado.
- El equipo de desarrollo NEURON C contiene el compilador NEURON C y el depurador de programa NEURON C.
- El Manejador de Red LONBUILDER permite programar para cargar, configurar, y controlar redes LONWORK.
- El Analizador de Protocolo LONBUILDER permite la programación para el monitoreo y diagnóstico del tráfico de red.

1.4.1 Medio Ambiente de Desarrollo Integrado LONBUILDER

El medio ambiente de desarrollo integrado LONBUILDER (IDE) soporta programas de la aplicación para automatizar el ciclo de desarrollo, proporcionando

un almacén común para las herramientas de software. El IDE básico incluye lo siguiente.

- Base de datos enfocada a objetos, el cual almacena el valor de los parámetros de uso definible de la aplicación, ruteador, y nodos manejadores de redes. Estos valores son repartidos por todas las herramientas de software LONBUILDER.
- Navegador, el cual simplifica viendo y cambiando la base de datos. Un gran proyecto puede consistir de cientos de objetos. La definición de estos objetos son fácilmente manejados usando el navegador.
- Manejador de proyecto, el cual maneja la configuración de los nodos definidos dentro de la base de datos y habilita para la construcción de una aplicación basada en los valores de la base de datos. Los manejadores de objetos invocan opcionalmente al compilador NEURON C, para compilar los programas de la aplicación, y las herramientas de manejo de la red para construir la información de la configuración de la red, cargar la aplicación y emesarla.
- Editor Integrado para crear programas de aplicaciones y para llamar a el manejador de proyectos. El editor está integrado al compilador NEURON C, los errores encontrados con el compilador son fáciles de hallar y corregir.
- Herramientas adicionales, las cuales pueden ser sumadas por el IDE básico, incluyendo los componentes del equipo de desarrollo NEURON C.

1.4.2 Equipo de Desarrollo NEURON C

El equipo de desarrollo NEURON C incluye el compilador NEURON C y el depurador NEURON C.

El NEURON C (un lenguaje de alto nivel de programación basado en el lenguaje ANSI C) permite al desarrollador una producción de aplicaciones rápida que involucra el listado, la comunicación nodo a nodo, y la interface de hardware. La aplicación del NEURON C por otra parte, cuenta con el microprograma (firmware) para manejar e implementar los detalles en el listado, la comunicación, y la interface de hardware. De este modo, el microprograma proporciona un sistema operador de red para ejecutar el programa de la aplicación y liberar al desarrollador de tratos con bajo nivel, detalles de comunicación y el proceso de entrada/salida. Combinando el NEURON C y el microprograma NEURON se facilita la elaboración de programas compactos que son fáciles de desarrollar y de fácil mantenimiento.

Las redes LONWORK proveen la capacidad de conectar una variedad de nodos que realizan un control distribuido, sensan, realizan identificaciones, además de otras aplicaciones. El nodo usa uno o más medios de comunicación y se comunica con algún otro dispositivo usando un protocolo común, LONTALK. Los dispositivos sobre la red LONWORK son programados en NEURON C.

1.4.2.1 Compilador NEURON C

El compilador NEURON C, toma de la entrada NEURON C, el código fuente almacenado sobre la PC, y genera el código que se almacena en la base de datos para después ser cargada a uno o más nodos. El compilador NEURON C genera códigos para el circuito NEURON 3120 y el circuito NEURON 3150.

1.4.2.2 Depurador NEURON C

El depurador NEURON C corre sobre una PC mientras depura la corrida de aplicación NEURON C sobre uno o hasta 24 emuladores NEURON. El depurador NEURON C proporciona una vista en la pantalla de la PC de los programas de aplicación ejecutados sobre los emuladores NEURON. Utilizando esta vista en la pantalla, se puede colocar un punto de ruptura en el programa, empezar y parar la ejecución del programa y realizar simples pasos a través del programa. También se puede evaluar y modificar variables del programa usando nombres simbólicos NEURON C. El depurador NEURON C proporciona visibilidad dentro de la llamada a la pila del programa, proporcionando comandos para visualizar el llamado a la pila y cambiar el contexto corriente para alguna función dentro de la pila.

1.4.3 Herramientas para el Manejo de la Red

Las herramientas para el manejo de la red LONBUILDER incluyen el manejador de red LONBUILDER y el analizador de protocolo LONBUILDER. Estas herramientas son usadas para definir, configurar, cargar, monitorear, y controlar múltiples nodos en una red de desarrollo.

1.4.3.1 Manejador de red.

El manejador de red LONBUILDER proporciona las herramientas para definir, configurar, cargar, y controlar nodos LONWORK y redes. El manejador de red usa la interface de red incluida en el Procesador de Control de la estación de desarrollo para enviar comandos de manejo de red hacia otros nodos dentro de la misma red.

El manejador de red es usado para realizar lo siguiente:

- Definir la configuración física de los nodos para asociar el hardware con un programa de la aplicación. Este define la imagen de la aplicación.
- Definir las conexiones de la red, generando información acerca de la dirección de los nodos (llamado la imagen de la red), y cargando aplicación e imagen de la red sobre el hardware.
- Poner nodos sobre la línea de la red, sacar nodos de la línea, reiniciar nodos, leer memoria de nodos, escribir memoria de nodos, y probar nodos.
- Monitorear el estado de todos los nodos sobre la red.
- Examinar y cambiar variables de la red sobre algunos nodos en la red.
- Recuperar los datos de su propia identificación y su propia documentación describiendo las variables de la red en algún otro nodo.

1.4.3.2 Analizador de protocolo

El analizador de protocolo LONBUILDER monitorea, colecta, y despliega las estadísticas del tráfico de la red y la ejecución de la red selectivamente.

El analizador de protocolo es usado para realizar lo siguiente:

- Vigilar el tráfico de la red.
- Salvar todo el tráfico de la red en un archivo para desplegarlo después.
- Salvar un tráfico específico de la red en un archivo para la especificación de los filtros de paquetes basado en la fuente del nodo, la destinación de nodo, tipo de paquete, variable de la red y etiqueta de mensaje.
- Resumir la ejecución de la red, los contadores de paquetes por tipos de paquetes, y el contador de errores.

CAPÍTULO II

CIRCUITO INTEGRADO NEURON

2.1 INTRODUCCIÓN

Los Circuitos Integrados NEURON MC143150 y MC143120 de motorola, son dispositivos que hacen posible implementar a bajo costo aplicaciones de redes de operación local. Por medio de una combinación única de hardware y un microprograma (firmware), estos dispositivos proveen todas las funciones necesarias para procesar entradas de sensores y dispositivos de control, además de poder controlar la propagación de la información que viaja sobre una variedad de medios de comunicación. El MC143150 y MC143120 junto con el Banco de Desarrollo LONBUILDER ofrecen:

- Un fácil control de redes.
- Capacidad de volver a configurar después de la instalación de la red.
- Manejo de mensajes del protocolo LONTALK sobre la red.
- Un medio ambiente orientado a objetos para desarrollo de sistemas.

El MC143150 está diseñado para monitorear y controlar sistemas que necesitan de un programa de la aplicación grande. Una interfase hacia una memoria externa permite al sistema el uso de 42 K, de los 64 K de espacio total disponibles, para el almacenamiento del programa de la aplicación. El MC143150 no tiene memoria ROM interna. El equipo iniciador LONBUILDER (MC143160EVK) proporciona el código objeto del protocolo de comunicación, el sistema operativo y las 24 funciones de entrada/salida. El código del protocolo y el código de la aplicación puede ser localizados en memoria externa ROM, EPROM, NVRAM o RAM estática con baterías de respaldo.

La figura 2.1 muestra en diagrama de bloques el Circuito Integrado NEURON MC143150

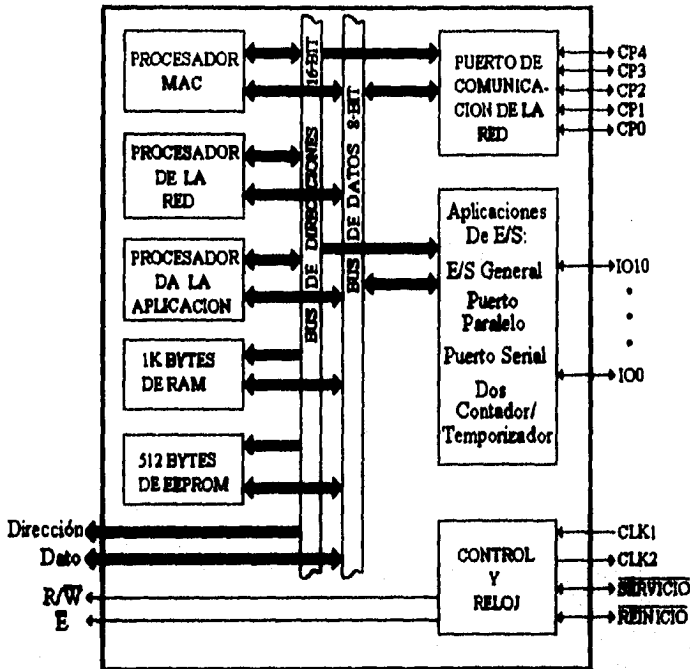


Figura 2.1 Diagrama de Bloques Simplificado del Circuito Integrado NEURON MC143150

El MC143120 no tiene interface para memoria externa y esta diseñado para aplicaciones que requieran programas de la aplicación pequeños. Este contiene 10 K de ROM de mascara que implementa el protocolo de comunicacion, el sistema operativo y las 24 funciones de entrada/salida que pueden ser accesadas por el programa de la aplicacion. El programa de la aplicacion reside dentro de la memoria EEPROM interna de 512 bytes y utiliza el microprograma (firmware) sobre la ROM de mascara para la aplicacion especifica.

En la figura 2.2 se muestra en diagrama de bloques el Circuito Integrado NEURON MC143120.

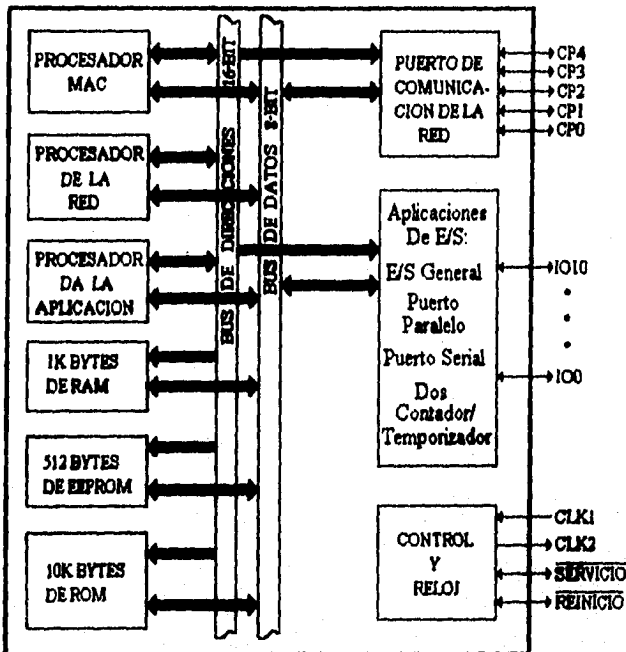


Figura 2.2 Diagrama de Bloques Simplificado del Circuito Integrado NEURON MC143120

Ambos dispositivos tienen 11 terminales (pines) de entrada/salida con hardware y software integrados para la conexión de motores, válvulas, manejadores de desplegados, convertidores A/D, sensores, termistores, interruptores, retardadores, triacs, tacómetros, otros microprocesadores, modems, etc. Cada uno de ellos tiene tres procesadores, de los cuales dos interactúan con un subsistema de comunicación, realizando la transferencia de información de un nodo hacia otro nodo en un proceso automático dentro de un sistema de control distribuido.

2.2 RECURSOS DE HARDWARE DE LOS CIRCUITOS INTEGRADOS NEURON

Los primeros miembros de la familia de procesadores Circuito Integrado NEURON son el MC143150 y el MC143120. El MC143150 (figura 2.3) soporta memoria externa para aplicaciones más complejas, mientras que el MC143120 (figura 2.4) es un sistema completo que soporta diversas aplicaciones. La mayoría de los bloques de hardware de ambos procesadores son iguales, excepto los que son especificados en la tabla 2.1.

Tabla 2.1 Comparación de los procesadores MC143150 y MC143120

Características	MC143150	MC143120
CPUs	3	3
Bytes de RAM	2,048	1,024
Bytes de ROM	---	10,240
Bytes de EEPROM	512	512
Temporizador/contador de 16-Bits	2	2
Interfaz de memoria externa	Si	No
Terminales	64/68	32

2.2.1 Unidades del Procesador

Los tres estados de máquina idénticos (un estado de máquina por procesador) de 8 bits forman un multiprocesador simétrico, cada uno ejecutando un microciclo diferente dentro de un ciclo de tres fases (una fase por cada microciclo) en cualquier tiempo. Cada uno de éstos tiene su propio juego de registros (tabla 2.2), pero los tres toman parte en los 8 bits de datos y direcciones de las ALUs y el circuito para acceso a la memoria. Esto reduce el tamaño del dispositivo sin afectar la ejecución.

Tabla 2.2 Juego de Registros

Mnemónicos	Bits	Contenido
BANDERAS	8	Número del CPU, selección rápida de E/S, y bit de acarreo
IP	16	Apuntador de la instrucción siguiente
BP	16	Dirección de los 256 bytes de la página base
DSP	8	Apuntador de datos de la pila dentro de la página base
RSP	8	Retorno al apuntador de pila dentro de la página base
TOS	8	Cima de los datos de la pila, entrada de la ALU
SAR	8	Registro del análisis de la asignatura

La arquitectura está orientada a la pila; una primera pila de un ancho de 8 bits es usada para la referencia de los datos, y la ALU opera sobre el registro TOS (cima de la pila) y la entrada siguiente en la pila de datos, la cual se encuentra en RAM. Una segunda pila almacena las direcciones de retorno para la instrucción CALL, y puede también ser usada para almacenar datos temporales (ver figura 2.5). Esta arquitectura maneja códigos muy compactos. Las Tabla 2.5, 2.6, y 2.7 delinear el juego de instrucciones.

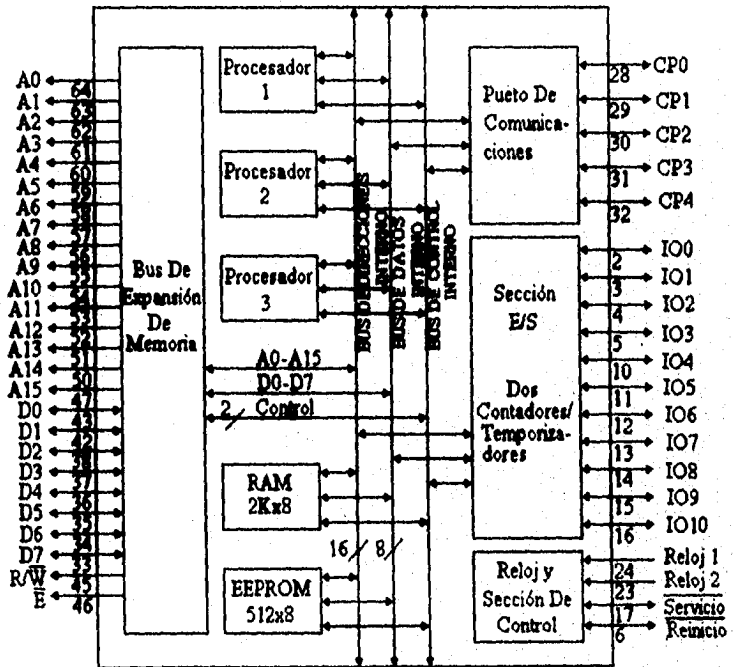


Figura 2.3 MC143150

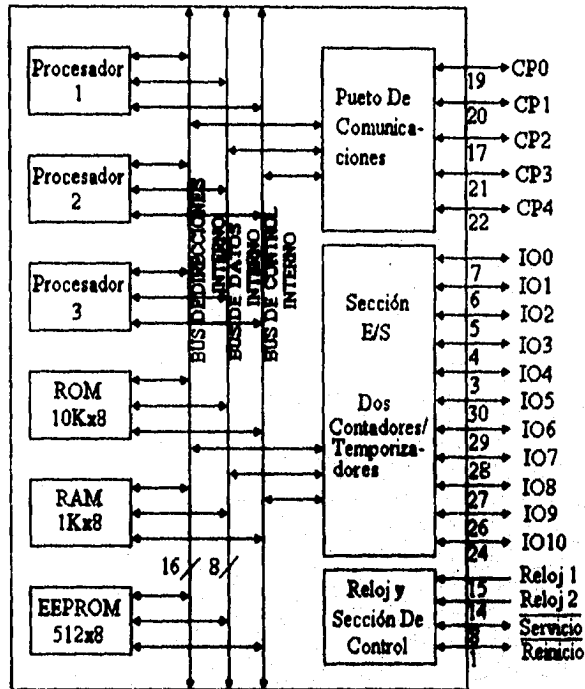


Figura 2.4 MC143120

La figura 2.5 muestra la organización de una página base, la cual puede ser de hasta 256 bytes de largo. Cada uno de los tres procesadores usa una página base diferente, cuya dirección está dada por el contenido del registro BP (página base) de este procesador. La cima de la pila de datos está dentro de los 8 bits del registro TOS, y el elemento siguiente en la pila de datos está localizado dentro de la página base en el desplazamiento dado por el contenido del registro DSP. La pila de datos crece desde la memoria baja hacia la memoria alta, y la pila para las direcciones de retorno crece desde la memoria alta hacia la baja.

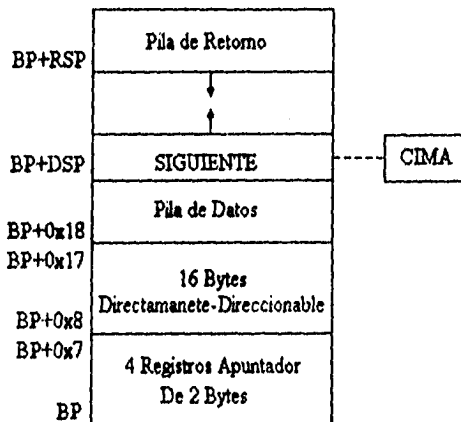


Figura 2.5 Organización de la página base de la memoria

2.2.2 Memorias

2.2.2.1 EEPROM

El MC143150 y el MC143120 tienen internamente 512 bytes de memoria EEPROM que se puede programar. Toda la memoria EEPROM puede ser escrita, con excepción de 8 bytes. Los 8 bytes faltantes son escritos durante la fabricación, y contiene un identificador único de 48 bits para cada parte, más 16 bits para el código de fabricación del dispositivo. El tiempo de borrado y el tiempo de escritura son cada uno de 10 ms por byte. La memoria EEPROM puede ser escrita hasta 10,000 veces. Para el MC143120 y el MC143150 la memoria EEPROM almacena la información de la instalación específica, tal como las direcciones de la red y parámetros de comunicación. Para el MC143120, la memoria EEPROM también almacena el programa de la aplicación generado por el Banco de Desarrollo LONBUILDER. El código de la aplicación para el MC143150 puede ser almacenado dentro del dispositivo o fuera de él, en una memoria externa.

2.2.2.2 RAM estática

El MC143150 tiene 2048 bytes de RAM estática y el MC143120 tiene 1024 bytes de RAM estática. El estado de la RAM es retenido por largo tiempo mientras se esté aplicando energía al dispositivo, igualmente en el modo "dormido".

2.2.2.3 ROM programada

El MC143120 contiene 10,240 bytes de ROM preprogramada. Esta memoria contiene el microprograma LONWORK, incluyendo el código del protocolo LONTALK, el tiempo real de la lista de tareas y las librerías de la función de la aplicación. Para todo esto el MC143150 accesa memoria externa. El código objeto es suplementado con el sistema de desarrollo LONBUILDER.

2.2.3 Interface de Memoria Externa (solo para el MC143150)

Esta interface soporta hasta 42 Kbytes de espacio de memoria externa para programa y datos adicionales. El espacio total de la dirección es de 64 Kbytes. Sin embargo, 6 Kbytes del espacio de dirección está reservado por el dispositivo, los restantes 58 Kbytes son el espacio para la dirección externa, de los cuales, 16 K son usados por el microprograma del Circuito Integrado NEURON, el depurador de desarrollo LONBUILDER, y espacio reservado. El espacio de la memoria externa puede ser poblada con RAM, PROM, EPROM ó EEPROM en 256 bytes adicionales. El mapa de memoria se muestra en la figura 2.12. El bus tiene 8 líneas de datos bidireccionales, y 16 líneas de dirección manejadas por el procesador. Dos líneas de interface (R/\bar{W} y \bar{E}) son usadas para el acceso a la memoria externa. A la velocidad máxima de reloj (10 MHz de reloj de entrada), se requiere de 70 ns o un tiempo de acceso menor para la memoria externa. Si la entrada del reloj es menor, una memoria más lenta puede ser usada. Los valores permitidos para la entrada del reloj son de 10 MHz, 5 MHz, 2.5 MHz, 1.25 MHz, y 625 KHz. La habilitación del reloj (\bar{E}) es 1/2 de la velocidad de entrada del reloj. La habilitación del reloj está en bajo siempre que el dato esté empezando ha ser transferido entre el Circuito Integrado NEURON y la memoria externa. Toda la memoria (interna y externa) puede ser accesada por algunos de los tres procesadores en la fase apropiada del ciclo de instrucción. Dado que el ciclo de instrucción de los tres procesadores se realiza en un tercio de ciclo con respecto uno del otro, el bus de la memoria es usada por un procesador a la vez.

La figura 2.6 muestra el diagrama de interfaces del MC143150 para diferentes tipos de memoria. Una configuración de hardware mínima usaría una memoria EEPROM externa para almacenar el código del protocolo y el código de la aplicación del usuario (figura 2.7). Esta configuración puede no permitir al operador del sistema cambiar el código de la aplicación después de la instalación. Sin embargo, la información de la paginación y dirección podría ser alterada, ya que esta información reside en los 512 bytes de la EEPROM interna. Cuando se están desarrollando nodos que deben ser escritos nuevamente sobre la red con el código de la aplicación, y éste sobrepasara los 512 bytes de la EEPROM interna, se requiere de una memoria externa EEPROM, NVRAM o SRAM con baterías de respaldo. También se requiere, cuando se diseña un nodo usando un reloj de cristal lento, donde posiblemente, reduciría significativamente la velocidad de la memoria. Las terminales usadas para la interfaz de memoria externa son listadas en la tabla 2.3. La señal de reloj \bar{E} no es frecuentemente usada para leer memoria externa, pero puede estar conmutando con líneas decodificadas de dirección y la señal \overline{RW} para generar la señal WRITE para memoria externa. Puede ser usada A15 (línea de dirección 15) o un decodificador de señal PAL conmutado con \overline{RW} para generar señales de lectura para la memoria externa. El MC143150 necesita ver el dato (tiempo de setup) 65 ns antes del flanco de subida del reloj \bar{E} .

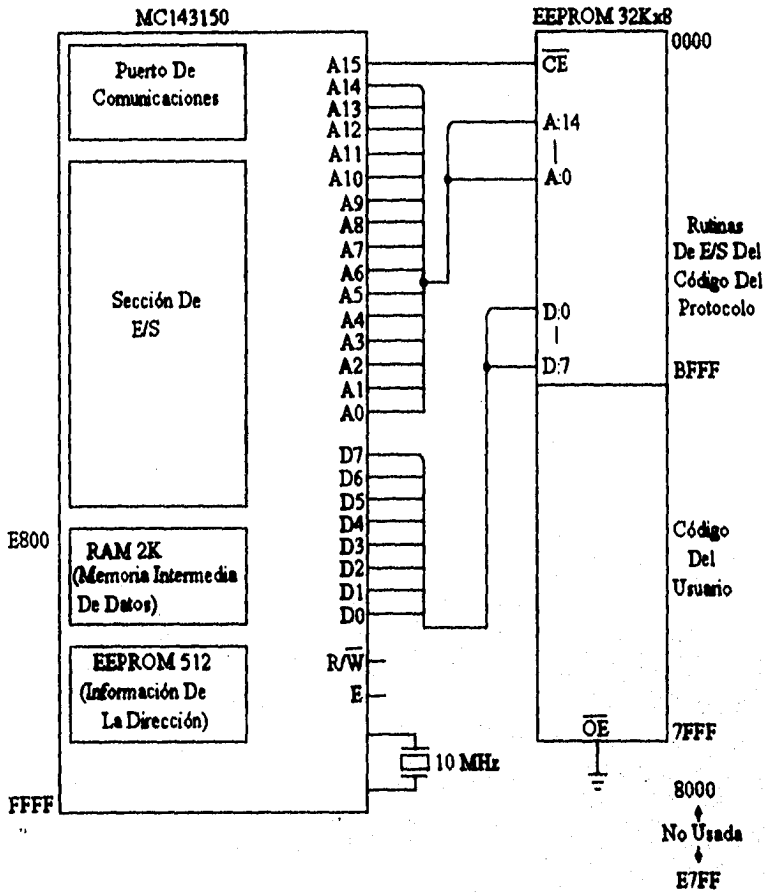


Figura 2.6 Interface de Memoria Externa

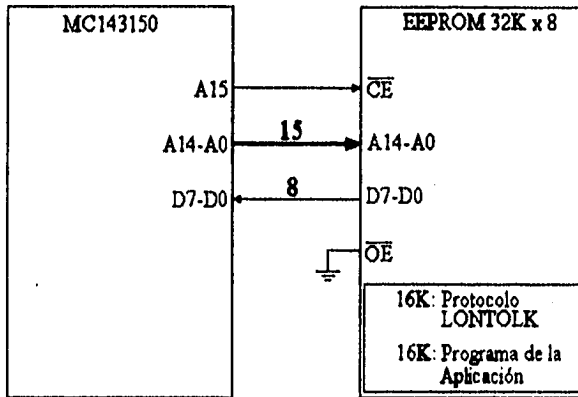


Figura 2.7 Mínima Interface de Memoria para el MC143150

Tabla 2.3 Terminales de la interface de memoria externa

Terminales	Dirección	Función
A0 - A15	Salida	Terminales de dirección
D0 - D7	Entrada/Salida	Terminales de datos
E	Salida	Habilitación de reloj
RAW	Salida	Selección Leer/Escribir

Ya que uno de los procesadores internos siempre está usando el bus de direcciones, la realización de una interface entre el Circuito Integrado NEURON y un MCU (microcontrolador) sólo es posible con el uso de manejadores externos tres estados sobre las 16 líneas del bus de direcciones, compartiendo así, el bus de direcciones y el bus de datos (y memoria) del Circuito Integrado NEURON con el bus de direcciones y el bus de datos del MCU. Este método no es recomendado o emulado por el Banco de Desarrollo LONBUILDER. El mejor método de interface del Circuito Integrado NEURON y el MCU es a través de los 11 terminales de entrada/salida. Existe modo serial y paralelo para este propósito el cual es implementado fácilmente usando el lenguaje de programación NEURON C.

2.2.4 Entrada/Salida

2.2.4.1 Once terminales bidireccionales E/S

Estas terminales son usadas en diferentes configuraciones para proveer una interface flexible hacia un hardware externo y para proveer el acceso a los temporizadores/contadores interno.

Las terminales IO4 - IO7 tienen transistores de pull-ups que son habilitados o deshabilitados con una directiva del compilador. Las terminales IO0 - IO3 tienen la capacidad de entregar una corriente alta (20 mA a 0.8 V). Las demás terminales tienen la capacidad estándar de entrega de corriente (1.4 mA a 0.4V). Todas las terminales (IO0 hasta IO10) tienen niveles de entrada TTL con histéresis. Las terminales IO0 hasta IO7 también tienen *latches* detector de nivel bajo.

2.2.4.2 Dos temporizadores/contadores de 16 bits

Los temporizadores/contadores están implementados con un registro que puede ser escrito por el procesador, un contador de 16 bits, y un *latch* que puede ser leído por el procesador. Los 16 bits del registro se accesan byte por byte. El MC143150 y el MC143120 tienen un temporizador/contador cuya entrada se puede seleccionar de entre las terminales IO4 hasta IO7, y cuya salida es la terminal IO0, y un segundo temporizador/contador con entrada en la terminal IO4 y salida por la terminal IO1 (figura 2.8). Note que las entradas/salidas no están dedicadas para la función del temporizador/contador. Si por ejemplo, el temporizador/contador 1 es usado para señal de entrada solamente, entonces IO0 es disponible para otra función de entrada o salida. El reloj para los temporizadores/contadores y las habilitaciones de las entradas pueden ser hechas desde terminales externas, o desde relojes escalados derivados del sistema de reloj; la velocidad de reloj de los dos temporizadores/contadores son independientes uno del otro. El funcionamiento del reloj externo ocurre opcionalmente sobre el flanco de subida, el flanco de bajada, o ambos flancos de bajada y subida de la entrada.

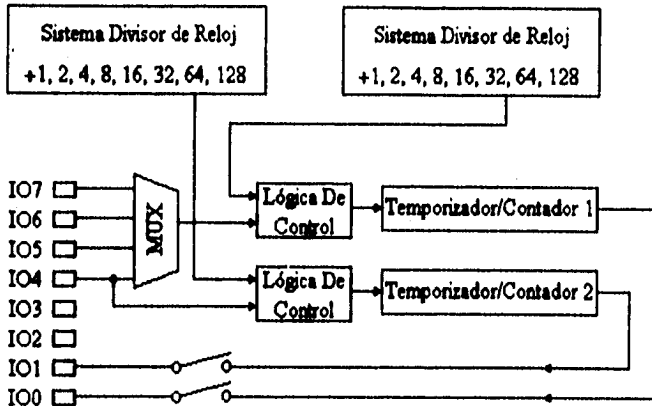


Figura 2.8 Circuitería del Temporizador/Contador

2.2.5 Comunicación de la Red

Las cinco terminales del puerto de comunicación son la conexión a la red, que hace interface con una amplia variedad de medios (se requiere de un transmisor/receptor). El puerto de comunicación puede ser configurado para operar en uno de dos modos: Modo directo (terminación simple o diferencial) o modo de propósito especial.

2.2.6 Sistema de Reloj

El Circuito Integrado NEURON incluye un oscilador que puede ser usado para generar una entrada de reloj usando un cristal externo. La transconductancia de este oscilador es menos de 2.1 milisiemens. El Circuito Integrado NEURON puede operar sobre un rango de velocidad de reloj de entrada desde 10 MHz hasta 625 KHz para aplicaciones de baja potencia. Las frecuencias de entradas válidas de reloj son: 10 MHz, 5 MHz, 2.5 MHz, 1.25 MHz, y 625 KHz. Alternativamente un generador de reloj externo puede manejar la terminal de entrada CMOS CLK1 de el Circuito Integrado NEURON, en tal caso CLK2 debe ser dejado desconectado. La precisión de la frecuencia de reloj debe ser de +/- 1.5% o de mayor precisión

para asegurar que los nodos puedan sincronizar correctamente sus bits de reloj cuando usan el transmisor/receptor en Modo Directo. Los valores para el cristal son listados en la tabla 2.4, hasta para una entrada de 625 KHz. Ver figura 2.9. Ya que la entrada de reloj es dividida entre dos, se permite un amplio rango de entrada del ciclo de trabajo.

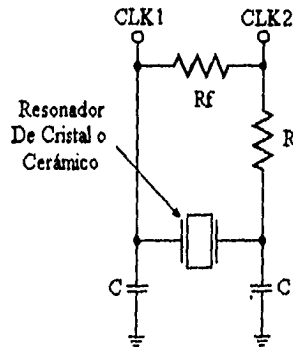


Figura 2.9 Circuito Generador de Reloj del Circuito NEURON

Tabla 2.4 Valores de los componentes del generador de reloj

Frecuencia del reloj de entrada	Cristal		Resonador cerámico	
	R	C	R	C
10.0 MHz	270 Ω	30 pF	270 Ω	30 pF
6.0 MHz	470 Ω	30 pF	270 Ω	30 pF
2.5 MHz	1 K Ω	36 pF	1 K Ω	36 pF
1.25 MHz	1.2 K Ω	47 pF	1.2 K Ω	47 pF
625 KHz	2.7 K Ω	47 pF	1.2 K Ω	100 pF

$R_f = 100 \text{ k}\Omega$ para todas las condiciones. La frecuencia de resonancia del reloj cerámico o del cristal es igual a la entrada de la frecuencia del reloj. El cristal podría estar en resonancia paralela o serie. La tolerancia de resistencias y capacitores es de +/- 5%.

2.2.6.1 Circuitería divisora de reloj

El Circuito Integrado NEURON divide la entrada del reloj por un factor de 2 para proveer un sistema de reloj simétrico sobre el dispositivo. La entrada del reloj puede ser generada por una fuente de oscilación externa TTL, o por un dispositivo oscilador usando un cristal externo o un resonador cerámico.

Un ciclo de instrucción del procesador es tres veces el ciclo de reloj del sistema, o seis ciclos del reloj de entrada. La mayoría de las instrucciones toman entre uno y siete ciclos de instrucción del procesador. A una velocidad de entrada de reloj máxima (10 MHz), el tiempo de instrucción son entre 0.6 μ s y 4.2 μ s. La fórmula para obtener el tiempo de instrucción es:

$$\langle \text{tiempo de instrucción} \rangle (\mu\text{s}) = \langle \# \text{ de ciclos} \rangle * 6 / \langle \text{entrada de reloj} \rangle (\text{MHz})$$

Las tablas 2.5, 2.6 y 2.7 listan las instrucciones del procesador y sus tiempos. Se provee éste para poder calcular el tiempos de respuesta para eventos realizados sobre las terminales de entrada/salida. Toda la programación del Circuito Integrado NEURON está hecha con NEURON C usando un Sistema de Desarrollo LONBUILDER.

Tabla 2.5 Tiempos de las instrucciones del programa de control

Mnemónicos	Ciclos	Descripción
NOP	1	No operación
SBR	1	Salto corto
GR/BR/BRNC	2	Salto, salto sobre (no) acarreo
SBRZ/SBRNZ	3	Salto corto sobre (no) cero
BRF	4	Salto largo
BRZ/BRNZ	4	Salto sobre (no) cero
RET	4	Regreso desde la subrutina
BRNEQ	4/6	Salto si no es igual (realizado/no realizado)
DBRNZ	5	Decrementa y brinca si no es cero
CALLR	5	Llamada a subrutina relativa
CALL	6	Llama a subrutina
CALLF	7	Llamada a subrutina lejana

Tabla 2.6 Tiempos de las instrucciones del programa de control

Mnemónicos	Ciclos	Modo de dirección
PUSH TOP	3	Registro TOS
PUSH/POP cpureg	4	Algún registro del CPU
PUSH/POP ID	4	Página base más desplazamiento (8-23)
PUSH/POP ITOS	4	Página base más contenido de TOS
PUSH (RSP)	4	Datos sobre la cima del regreso a la pila
PUSH/POP # literal	4	Valor literal de 3 ó 8 bits
PUSHPOP	5	Pop desde el regreso de la pila, push para regresar a los datos de la pila
POPPUSH	5	Pop desde los datos de la pila, push para regresar a la pila
LDBP dirección	5	Carga el apuntador de la página base
PUSH/POP (DSP)[D]	5	Contenido del mínimo desplazamiento del DSP (1-8)
PUSD # literal	6	Valor literal de 16 bits
POPD/PUSHD (PTR)	6	16 bits indirectos a través del apuntador de la página base
PUSH/POP (PTR)[TOS]	6	Indirecto a través del apuntador de la página base más TOS.
PUSH/POP (PTR)[D]	7	Indirecto a través del apuntador de la página base más desplazamiento
PUSH/POP absoluto	7	push en los datos de la memoria para los datos de la pila

Tabla 2.7 Tiempos de Instrucción de la ALU

Mnemónicos	Ciclos	Operación
INC/DEC/NOT	2	Incremento/decremento/negación
TOSROL/RORC	2	Rotación iz/der del TOS a través del acarreo
SHL/SHR	2	Corrimiento lógico iz/der del TOS
SHAL/SHRA	2	Corrimiento aritmético iz/der del TOS
DROP NEXT	2	Descenso al siguiente elemento desde los datos de la pila
DROP (RSP)	2	Descenso a la cima del retorno de la pila
ADD/AND/OR/XOR	3	Operando con el literal sobre TOS
DROP TOS	3	Descenso a la cima de los datos de la pila
ALLOC # literal	3	Mover el apuntador de datos de la pila
ADD/AND/OR/XOR	4	Operando con el siguiente elemento sobre TOS
ADC/SBC/SUB/XCH	4	Operando con el siguiente elemento sobre TOS
INC (PTR)	6	Incrementando el apuntador de la página base (0-3)
DEALLOC_R # literal	6	Moviendo el apuntador de datos de la pila y regresándolo
IN/OUT	7+4 #bytes	Bloque entrada/salida

2.2.7 Funciones Adicionales.

2.2.7.1 Modo dormido/desperto

El Circuito Integrado NEURON puede estar dentro de un modo de consumo de baja energía controlado por software. En este modo, el reloj del sistema y todos los temporizadores/contadores están apagados, y se encuentra guardado todo el estado de información (incluyendo el contenido de la RAM interna). La operación normal se reanuda cuando una transición de entrada ocurre en alguna de las siguiente terminales:

- Terminales de entrada/salida (de mascara):

Una de las terminales IO4 hasta IO7, es seleccionada por el multiplexor del temporizador/contador.

- Terminal de servicio (sin mascara)
- Puertos de comunicación (de mascara):

Modo Directo Diferencial	CP0 o CP1
Modo Directo de Terminación Simple	CP0
Modo de Propósito Especial	CP3

El software de la aplicación puede especificar que los transistores de pull-up (programables sobre la terminal de servicio y la terminal de entrada/salida IO4 - IO7) están deshabilitados para reducir el consumo de energía futura.

2.2.7.2 Temporizador perro de guardia

Los procesadores Circuito Integrado NEURON son protegidos contra un mal funcionamiento de software o memorias defectuosas por tres temporizadores perro de guardia, uno para cada procesador. Si la aplicación o el sistema de software falla, son reiniciados estos temporizadores y el Circuito Integrado NEURON es reiniciado automáticamente. El periodo del perro de guardia es aproximadamente de 0.84 segundos para la velocidad de entrada de reloj máxima (10 MHz) y es escalado inversamente con la velocidad de entrada del reloj. Cuando el Circuito Integrado NEURON está en el modo dormido, todos los temporizadores perro de guardia son deshabilitados.

2.2.8 Terminal de Servicio

Esta terminal actúa como entrada y como salida, alternando la entrada y la salida a una velocidad de 76 Hz con un ciclo de trabajo de 50%. Cuando esta terminal actúa como salida, éste puede suministrar 20 mA para manejar un LED. Cuando se usa exclusivamente como una entrada, éste tiene una opción de seleccionar un transistor de pull-up interno. Bajo el control del microprograma del Circuito Integrado NEURON, esta terminal es usada durante la configuración e instalación del nodo conteniendo el Circuito Integrado NEURON. El microprograma enciende el LED cuando el Circuito Integrado NEURON no ha sido configurado con la información de la dirección de la red. Poniendo la terminal de servicio a tierra causa que el Circuito Integrado NEURON transmita un mensaje manejador de red (conteniendo sus 48 bits únicos ID) sobre la red. Un circuito típico para la terminal de servicio de LED y push-button es mostrado en la figura 2.10. Durante el reinicio, el estado de la terminal de servicio esta indeterminada.

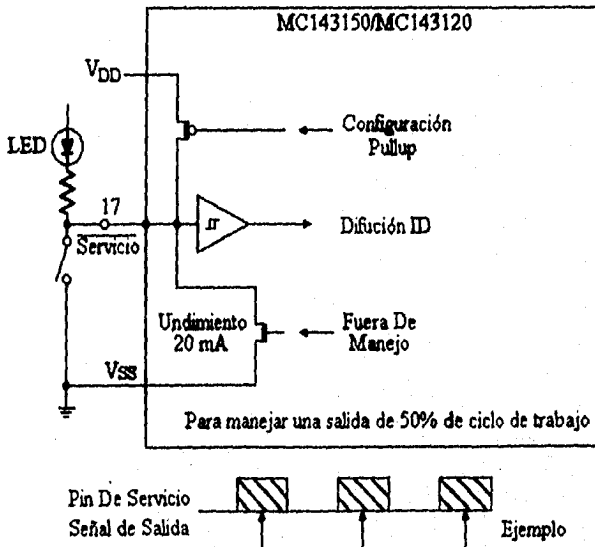


Figura 2.10 Circuitería de la Terminal de Servicio

2.3 ORGANIZACIÓN DEL PROCESADOR

2.3.1 Los Procesadores

Los tres procesadores están dedicados a las funciones siguientes para el software del sistema (ver figura 2.11).

Procesador 1, es el procesador del Control de Acceso al Medio (MAC) que manipula la primera de los siete niveles del protocolo de la red. Este incluye el manejo de hardware del subsistemas de comunicación, así como la ejecución del algoritmo que evita la colisión. El procesador 1 se comunica con el procesador 2 usando paquetes de memoria intermedia (buffers), localizados en alguna parte de la memoria.

Procesador 2, es el Procesador de la Red que implementa del segundo hasta el sexto nivel del protocolo de la red. Este manipula el procesamiento de las variables de la red, el direccionamiento, el procesamiento de las operaciones, la autenticidad, el diagnóstico, los tiempos de software, el manejo de redes y el encaminado de funciones. El procesador 2 usa paquetes de memoria intermedia (buffers), localizados en alguna parte de la memoria para comunicarse con el procesador 1, y el Servicio de Interface de Control de Bloques (SICBs) para comunicarse con el procesador 3.

Procesador 3, es el Procesador para la Aplicación. Este corre el código escrito por el usuario, junto con los servicios del sistema operativo llamado por el código del usuario. El NEURON C es el lenguaje principal usado para programar las aplicaciones, una derivación del lenguaje C, optimizado y realizado para aplicaciones de control distribuido LON. Lo sobresaliente de este lenguaje es:

- Una construcción interna de listas multitareas que permite al programador expresar lógicamente, manejos paralelos de eventos de tareas en un camino natural, y controlar la prioridad de la ejecución de esas tareas.
- Una sintaxis declarativa para objetos de entrada/salida dentro de las capacidades de entrada/salida de los procesadores.
- Una sintaxis declarativa para las variables de la red, las cuales son objetos del lenguaje NEURON C, cuyos valores son propagados automáticamente sobre la red siempre que los valores estén asignados a ellos.
- Una sintaxis declarativa para tiempos objetos de milisegundos y segundos, la cual activa las tareas del usuario al expirar estos tiempos.

- Una librería de funciones para ejecutar la verificación de un evento y la entrada/salida, para enviar y recibir mensajes hacia la red, y para el control de diversas funciones del NEURON.

El soporte para todas estas habilidades es parte del microprograma LONWORK. Esto permite aplicaciones de eventos complejos para ser escrito en el espacio de la memoria correspondiente a el programa de la aplicación del procesador MC143120.

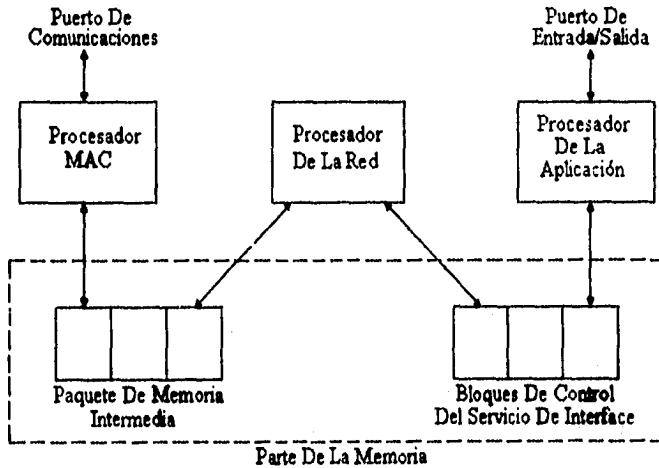


Figura 2.11 Distribución de la Organización de la Memoria del Procesador

2.3.2 Organización de las Memorias

2.3.2.1 Organización de la memoria del MC143150 (ver figura 2.12)

- 512 bytes de EEPROM programable, que contiene:
 - La configuración de la red y la información para el direccionamiento.
 - Un código de identificación LON de 48 bits (escrita por el fabricante).
 - El código de la aplicación escrito por el usuario y datos principales para lectura.

- 2,048 bytes de memoria RAM estática que contiene:
 - Segmento de la pila, aplicación y datos del sistema.
 - Memoria intermedia (buffers) para el protocolo de la red LON y memoria intermedia (buffer) para la aplicación.
- Hasta 65,536 bytes de direcciones de memoria libres (alguna mezcla de ROM, EPROM, EEPROM, o RAM). El procesador puede acceder 59,392 bytes de esta memoria por vía de la interface de memoria externa; 6,144 bytes del espacio de direcciones son manejadas internamente.
- 16,384 bytes de memoria externa son requeridas para almacenar:
 - El sistema operativo LONWORK, incluyendo el sistema de microprogramación ejecutado por el procesador MAC y el procesador de la red, y la ejecución que soporta el programa de la aplicación.
- El resto de la memoria externa es utilizada para:
 - El código de la aplicación escrito por el usuario.
 - Aplicación adicional de datos de lectura/escritura.
 - Memoria intermedia (buffers) adicional de la red y memoria intermedia (buffer) adicional para la aplicación.

2.3.2.2 Organización de la memoria del MC143120 (ver figura 2.12)

- 512 bytes de EEPROM programable que contiene:
 - La configuración de la red y la información para el direccionamiento.
 - Un código de identificación LON de 48 bits (escrito por el fabricante).
 - El código de la aplicación escrito por el usuario y datos principales para lectura.
- 1,024 bytes de RAM estática que contiene:
 - Segmento de la pila, aplicación y datos del sistema.
 - Memoria intermedia (buffers) para el protocolo de la red LON y memoria intermedia (buffer) para la aplicación.
- 10,240 bytes de ROM de máscara que contiene:
 - El microprograma LONWORK ejecutado por el procesador MAC y el procesador de la Red.
 - Sistema operativo soportando el programa de la aplicación.

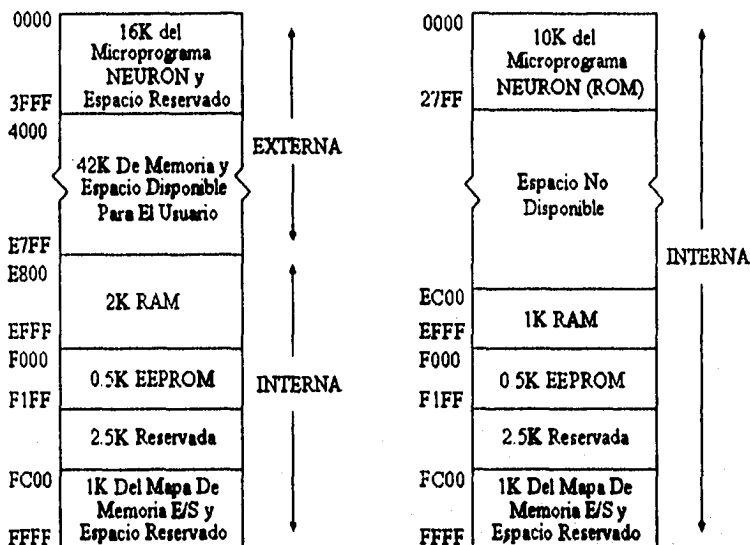


Figura 2.12 Mapa de Memoria de los Procesadores MC143150 y MC143120

2.4. INTERFACE DE ENTRADA/SALIDA

Las conexiones del Circuito Integrado NEURON para aplicaciones de hardware externo específicas son once terminales llamadas IO0 hasta IO10. Estas terminales pueden ser configuradas dentro de numerosos caminos para proveer flexibilidad en funciones de entrada y salida con un mínimo de circuitería externa. El modelo de programación (lenguaje NEURON C) permite al programador declarar una o más terminales como entrada/salida objeto. Un objeto es simplemente una definición de la forma de onda de entrada o salida. Estos están pensados como rutinas de la microprogramación prescritas dentro de la ROM, las cuales son accedidas por el programa de la aplicación del usuario. El programa del usuario, puede entonces, referirse a estos objetos en sistemas io_in e io_out (directivas del lenguaje NEURON C) llamados para ejecutar las funciones de entrada/salida durante la ejecución del programa. Existen 24 diferentes funciones útiles de entrada/salida (11 funciones para entrada, 11 funciones para salida, y dos funciones bidireccionales).

2.4.1 Consideraciones de Hardware

Diversas combinaciones de las terminales de E/S pueden ser configuradas como entradas ó salidas digitales como se lista en la tabla 2.8, 2.9, y 2.10. El programa de la aplicación puede especificar el valor inicial de las salidas digitales opcionalmente. Terminales configuradas como salida puede ser leídas también como entrada, regresando el último valor escrito. Las terminales IO4, IO5, IO6, e IO7 tienen suministro de corriente de pullup opcional (30 μ A a 300 μ A). Estos son habilitados con una directiva del compilador NEURON C (# pragma enable_io_pullup). Las terminales IO0, IO1, IO2, IO3 tiene gran capacidad de recepción de corriente (20 mA @ 0.8 V). Las otras terminales tienen la capacidad estándar de recepción de corriente (1.4 mA @ 0.4 V).

Tabla 2.8 Sumario de las funciones de entrada

Modo E/S	Terminales E/S	Valor de entrada
Bit	IO0 - IO10	0, 1 dato binario
Bit de Corrimiento	Algún par adyacente	Hasta 16 bits de datos de reloj
Byte	IO0 - IO7	0...255 datos binarios
Detector de Nivel	IO0 - IO7	Fianco negativo o fianco no detectado
Nibble	IO0...IO3 e IO4...IO7	0...15 datos binarios
Medición de tiempo	IO4 - IO7	Ancho de pulso de 0.2 μ s - 1.678 s
Periodo	IO4 - IO7	Una señal con periodo de 0.2 μ s - 1.678 s
Contador de Pulso	IO4 - IO7	0...65,535 flancos de entrada durante 0.839 s
Cuadratura	IO4 e IO5, IO6 e IO7	+/- 18,363 transiciones del código binario Gray
Serial	IO6	Caracteres de 8 bits a 600, 1200, 2400 ó 4800 bauds
Contador Total	IO4 - IO7	0... 65535 flancos de entrada

Tabla 2.9 Sumario de las funciones de salida

Modo E/S	Terminales E/S	Valor de salida
Bit	IO0 - IO10	0, 1 dato binario
Bit de corrimiento	Algún par adyacente	Hasta 16 bits de datos de reloj
Byte	IO0...IO7	0...255 datos binarios
Frecuencia	IO0, IO1	Onda cuadrada de 0.3 Hz a 2.5 MHz
Nibble	IO0...IO3, IO4...IO7	0...15 datos binarios
Un disparo	IO0, IO1	Pulso con duración de 0.2 μ s a 1.678 s
Conteo de pulsos	IO0, IO1	0...65,535 pulsos
Ancho de pulso	IO0, IO1	0...100 % de ciclo de trabajo del tren de pulsos
Serial	IO10	Caracteres de 8 bits a 600, 1200, 2400, ó 4800 bauds
Triac	IO0, IO1 e IO4 - IO7	Borrado del pulso de entrada escribe el fianco de entrada
Contador de disparos	IO0, IO1 e IO4 - IO7	Pulso de salida controlado por el conteo de los flancos de entrada

Tabla 2.10 Sumario de las funciones bidireccionales

Modo E/S	Terminales E/S	Valor de salida
NeuroAlambrado	IO8 (reloj) e IO9, IO10 (datos)	Hasta 255 bits de datos de reloj
E/S paralela	IO0 - IO10	Hasta 255 bytes de datos

2.4.2 Interfaces Digitales

2.4.2.1 Bit E/S

Las terminales desde IO0 hasta IO10, pueden ser configuradas individualmente como puertos de un simple bit de entrada o de salida. Las terminales configuradas como puerto de entrada pueden ser usadas para sensar señales lógicas compatible de niveles TTL desde una lógica externa. Las terminales configuradas como puerto de salida pueden ser usadas para manejar dispositivos externos CMOS, niveles compatibles lógicos TTL, y transistores de baja potencia, que retransmitan para mover dispositivos externos de alta corriente, tales como motores de paso y luces. La capacidad de suministrar corriente alta (20 mA) de las terminales IO0 hasta IO3 permite a estos manejar muchos dispositivos de entrada/salida directamente. Ver figura 2.13.

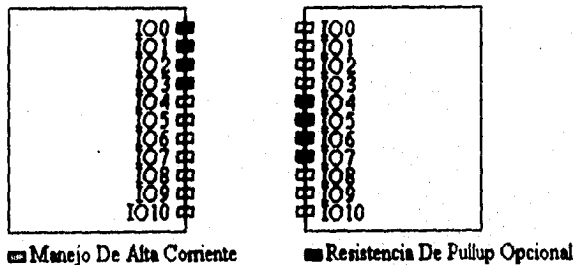


Figura 2.13 Bit E/S

2.4.2.2 Byte E/S

Las terminales IO0 hasta IO7 pueden ser configuradas como un puerto con un ancho de un byte de entrada o salida, el cual puede ser leído o escrito usando números enteros en el rango de 0 hasta 255 (en forma binaria). Esto es útil para manejar dispositivos que requieran datos en código ASCII, u otros datos de ocho bits. Por ejemplo un desplegador alfanumérico puede usar la función byte para los datos, y usar las terminales IO8 - IO10 en la función bit para el control y direccionamiento. Ver figura 2.14.

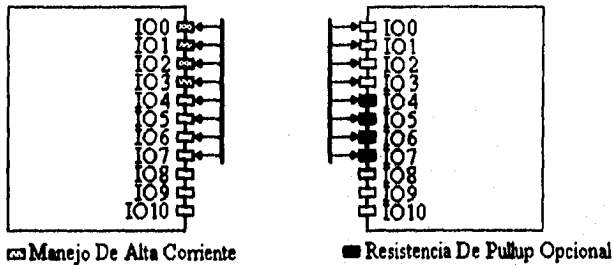
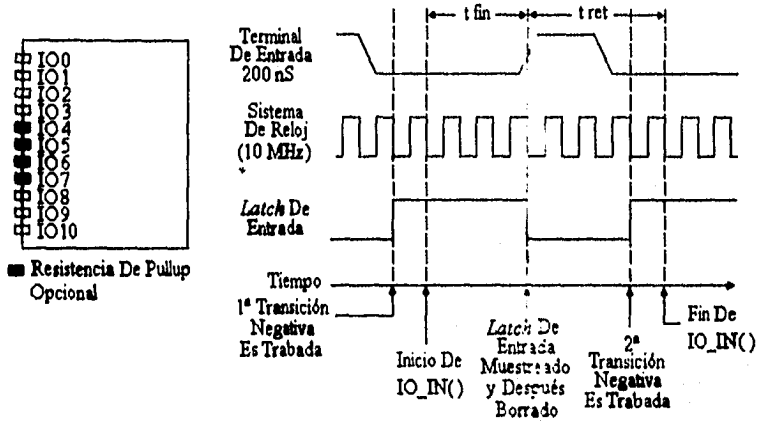


Figura 2.14 Byte E/S

2.4.2.3 Detección de nivel (nivel lógico bajo para entrada > 200 ns)

Las terminales IO0 hasta IO7 pueden ser configuradas individualmente como terminales de entrada de detección de nivel, el cual retiene la transición de un pulso que va de un nivel alto a un nivel bajo. Este utiliza un reloj con un ancho de pulso de 200 ns (con un reloj de 10 MHz) para detectar la transición del pulso entrante. La aplicación puede por lo tanto detectar pulsos cortos a la entrada. Esto es útil para leer dispositivos. Nótese que esto es sólo para funciones de entrada/salida, la cual es retenida antes que sea probada. Ver figura 2.15.



Símbolo	Descripción
t _{fin}	Tiempo entre el llamado de la función y la muestra
t _{ret}	Tiempo de retorno de la función

Figura 2.15 Detección de Nivel

2.4.2.4 Nibble E/S

Grupos de cuatro terminales consecutivas entre IO0 e IO7 pueden ser configuradas como puertos de entrada o salida de un ancho de 4 bits, las cuales puede ser leídas o escritas usando números enteros en el rango de 0 hasta 15 (en forma binaria). Esto es útil para manejar dispositivos que requieran datos en código BCD, u otros datos de cuatro bits. Ver figura 2.16

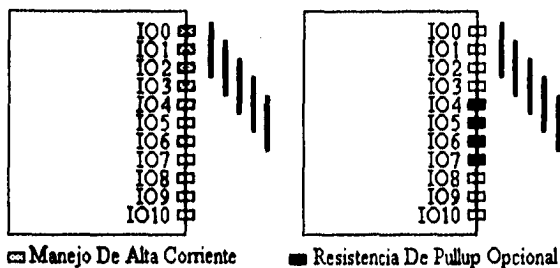


Figura 2.16 Nibble E/S

2.4.3 Funciones de interface en Paralelo E/S

Las terminales IO0 hasta IO10 pueden ser configuradas como un puerto bidireccional con 8 bits de datos y 3 bits de control para conectarlas a otro procesador. El otro procesador puede ser una computadora, un microcontrolador, u otro Circuito Integrado NEURON (para puente, ruteador, entrada, u otras aplicaciones). La interface en paralelo puede ser configurada en modo maestro/esclavo A, o maestro/esclavo B. Generalmente, la interface de dos Circuito Integrado NEURON se encuentra dentro del modo maestro/esclavo A y la interface entre un Circuito Integrado NEURON y un procesador diferente al Circuito Integrado NEURON, se encuentra en la configuración maestro/esclavo B, teniendo al procesador que no es Circuito Integrado NEURON como el maestro. El protocolo de saludo (handshaking) es usado en ambos modos para controlar la ejecución de la instrucción, y el proceso de la aplicación es suspendido en la duración de la transferencia de datos (hasta 255 bytes por transferencia).

2.4.4 Interface Serial

2.4.4.1 Bit de corrimiento de E/S

Pueden ser configurados pares de terminales adyacentes como líneas de entrada o salida serial, la terminal con numeración inferior será usada para el reloj (manejado por el Circuito Integrado NEURON) y la terminal con numeración mayor será usada para los 16 bits de dato serial. La velocidad del dato puede ser configurado como 1, 10, o 15 Kbps con la velocidad de reloj de entrada máxima (10 MHz).

2.4.4.2 Función NeuroAlambrada de E/S (interface SPI)

La función NeuroAlambrada implementa una transferencia de datos sincrónica full-duplex a algunos dispositivos periféricos. Este puede operar como el maestro (maneja un reloj de salida) o como el esclavo (acepta un reloj de entrada). En ambos modos maestro y esclavo, hasta 255 bits de datos pueden ser transferidos.

2.4.4.3 Serial de E/S

La terminal IO8 puede ser configurada como una línea de entrada serial asíncrona, y la terminal IO10 puede ser configurada como una línea de salida serial asíncrona. La velocidad del bit para la entrada y para la salida puede ser especificada independientemente, siendo de 600, 1200, 2400 o 4800 bps con la velocidad de reloj de entrada máxima (10 Mhz).

2.4.5 Función de Interface Temporizador/Contador

El MC143150 y el MC143120 tienen dos temporizadores/contadores de 16 bits. Para el primer temporizador/contador, IO0 es usado como la salida, y un multiplexor selecciona a una de las terminales IO4 a IO7 como la entrada. El segundo temporizador/contador usa la terminal IO4 como la entrada, y la terminal IO1 como la salida.

2.4.5.1 Funciones de entrada del temporizador/contador

La función de entrada del temporizador/contador tiene la ventaja (sobre la función sin temporizador/contador) que los eventos serán capturados aún cuando la aplicación del procesador esté ocupada haciendo alguna otra cosa cuando ocurra el evento.

• Medición del tiempo

Un temporizador/contador puede ser configurado para medir el tiempo por el cual su entrada está declarada. La declaración puede ser definida como lógica alta o lógica baja. Ver figura 2.17.

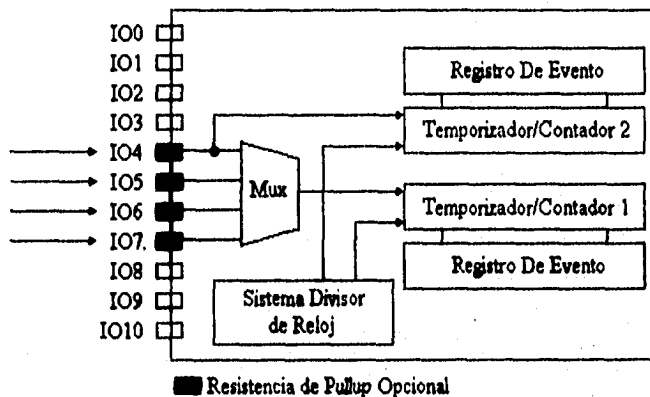


Figura 2.17 Medición del Tiempo

• Período

Un temporizador/contador puede ser configurado para medir el período de un flanco de subida o bajada al siguiente flanco sobre la entrada. Esta función es útil para frecuencias instantáneas o aplicaciones con tacómetros. Un convertidor analógico a digital puede ser implementado usando un convertidor de voltaje a frecuencia con esta función. Ver figura 2.18.

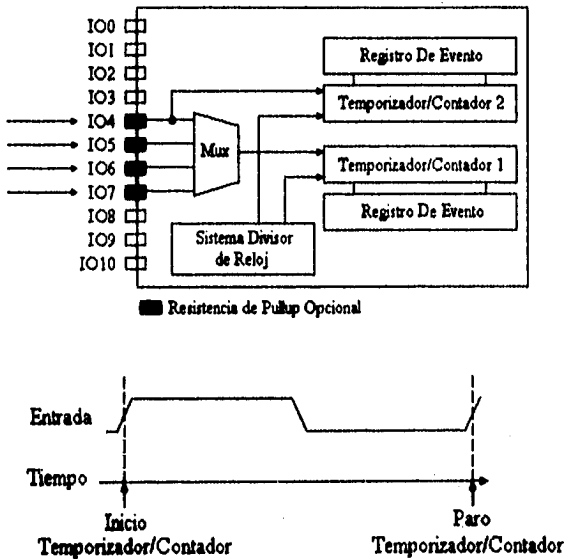


Figura 2.18 Período de entrada

• **Conteo de pulsos**

Un temporizador/contador puede ser configurado para contar el número de flancos de entradas (hasta 65,535) en un tiempo fijo (0.6388608 segundos) con la velocidad del reloj de entrada máxima. El flanco puede ser definido como de subida o de bajada. Esta función es útil para medir la frecuencia promedio, o aplicaciones de tacómetros. Ver figura 2.19.

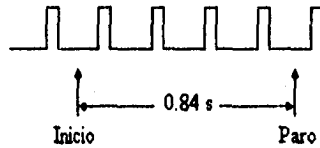
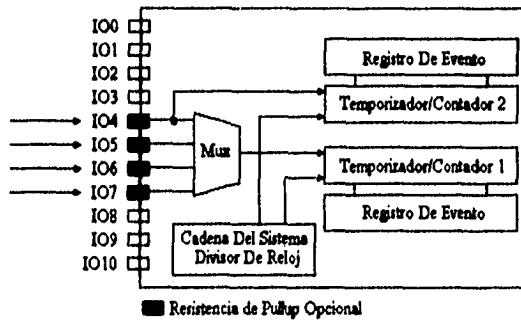


Figura 2.19 Conteo de Pulsos

• Cuadratura

Un temporizador/contador puede ser configurado para el conteo de la transición de un código de entrada binario Gray sobre dos terminales adyacentes de entrada. Este código Gray es generado por dispositivos tales como decodificadores y sensores de posición óptico, los cuales generan el patrón de bit (00, 01, 11, 10, 00...) para una dirección de movimiento y (00, 10, 11, 01, 00...) para la dirección opuesta. El valor leído de una cuadratura da la mitad del número de transiciones desde la última lectura (- 16,384 a 16,383). La frecuencia de entrada máxima es un cuarto de la velocidad del reloj de entrada, por ejemplo 1.25 MHz con la entrada de reloj máxima del Circuito Integrado NEURON. Los dispositivos de cuadratura pueden ser conectados por el temporizador/contador 1, via terminales IO6 e IO7, y temporizador/contador 2, via terminales IO4 e IO5. Ver figura 2.20.

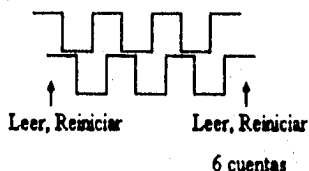
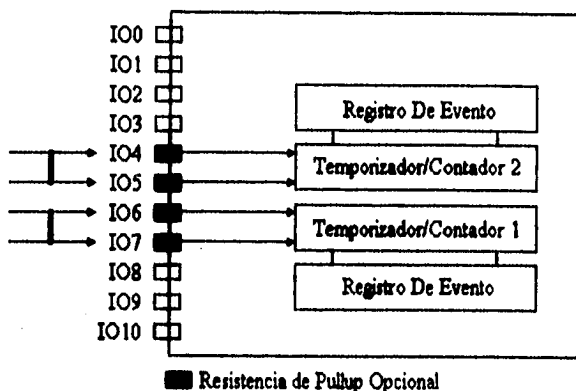


Figura 2.20 Cuadratura

• Contador total

Un temporizador/contador puede ser configurado para contar flancos de una señal de entrada (los flancos de subida ó los de bajada pero no ambos). El valor leído de un contador total da el número de transición (número de flancos de subida o bajada) desde que fue leído por última vez (de 0 a 65,535). La frecuencia de la entrada máxima es un cuarto de la velocidad de entrada de reloj. Esta función es útil para contar eventos externos. Ver figura 2.21

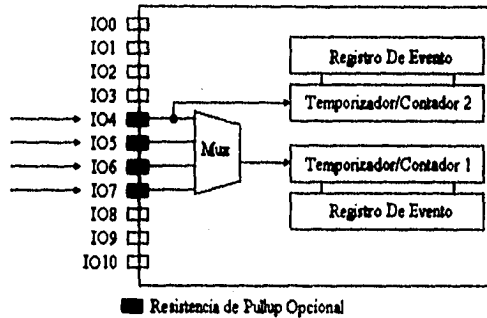
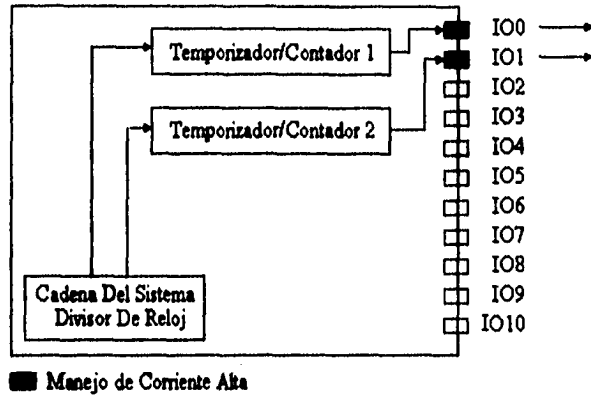


Figura 2.21 Contador Total

2.4.5.2 Funciones de salida del temporizador/contador

• Frecuencia

Un temporizador/contador puede ser configurado para generar una onda cuadrada con el 50% de ciclo de trabajo. Al escribir un valor de frecuencia nuevo en el dispositivo, éste toma efecto hasta el fin del ciclo corriente. Esta función es útil para síntesis de frecuencia, o al manejar una frecuencia en un convertidor de voltaje donde se genere una salida analógica, ver figura 2.22.



Reloj	Resolución	Rango
0	0.4	26.2
1	0.8	52.4
2	1.6	104.9
3	3.2	209.7
4	6.4	419.4
5	12.8	838.8
6	25.6	1677.6
7	51.2	3355.2

Figura 2.22 Frecuencia.

• Pulso

Un temporizador/contador puede ser configurado para generar un pulso en donde su duración puede ser programada. La declaración del estado puede ser "uno" lógico o "cero" lógico. Si se vuelve a disparar antes de que el pulso termine causará que este pulso continúe, con una nueva duración. Esta función es útil para generar un tiempo de retardo sin intervención de la aplicación del procesador. Ver figura 2.23.

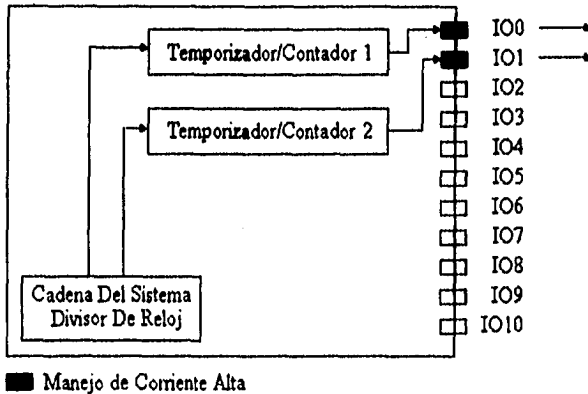


Figura 2.23 Pulso, Contador de Pulsos, Ancho de Pulso

• Generador de pulsos

Un temporizador/contador puede ser configurado para generar una serie de pulsos. El número de pulsos de salida está en el rango de 0 a 65,535, y la salida de la onda es una onda cuadrada con el 50% de ciclo de trabajo. Esta función suspende el proceso de aplicación hasta que el tren de pulso es completado. Este modo es útil para dispositivos de contador externo que pueden acumular tren de pulsos, tales como motores de pasos. Ver figura 2.23.

• Ancho de pulso

Un temporizador/contador puede ser configurado para generar una modulación por ancho de pulso repitiendo la forma de onda. En la función corta del ancho de pulso, el rango del ciclo de trabajo es desde 0% a 100% (0/256 a 255/256) de un ciclo, en pasos de alrededor de 0.4% (1/256). En la función larga del ancho de pulso, el rango de ciclo de trabajo es desde 0% hasta casi el 100% (0/65,536 a 65535/65536) de un ciclo, en pasos de 15.25 ppm (1/65536). El estado de declaración de la onda puede ser "uno" lógico o "cero" lógico. Al escribir un valor de ancho de pulso nuevo en el dispositivo, éste toma efecto hasta el fin del ciclo corriente. Una señal modulada en el ancho de pulso proporciona un medio de conversión digital a analógica simple. ver figura 2.23.

• Triac

En un Circuito Integrado NEURON un temporizador/contador puede ser configurado para controlar el disparo de una señal (un pulso) de salida con un ancho de $25 \mu\text{s}$ cuando a la entrada se tiene un flanco de subida o de bajada (este flanco funciona como disparo) de una señal. Para controlar circuitos de AC que utilizan un triac, la entrada sync es generalmente una señal de cruce por cero, y el pulso de salida es la señal que dispara un pulso hacia el triac. Ver figura 2.24.

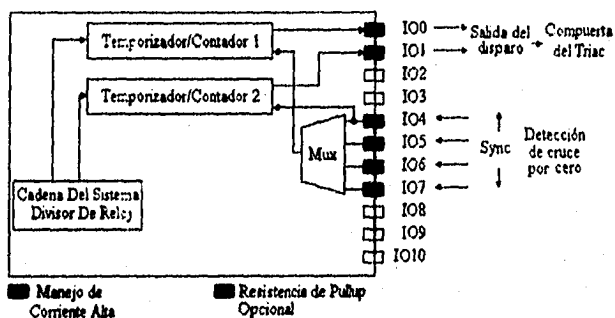


Figura 2.24 Triac

• Contador de disparos

Un temporizador/contador puede ser configurado para generar un pulso de salida que esté declarado dentro del programa de control, y no declarado cuando un número programable de flanco de entrada (de 1 hasta 65,535) haya sido contado en una terminal de entrada (IO4 - IO0). La declaración puede ser un nivel bajo o un nivel alto. Este modo es útil para controlar motores de paso o actuadores de posición el cual proporciona regeneración de posición en la forma de un tren de pulsos. El manejo para el dispositivo externo está habilitado hasta que se ha movido la distancia requerida, y luego el dispositivo es deshabilitado. Ver figura 2.25.

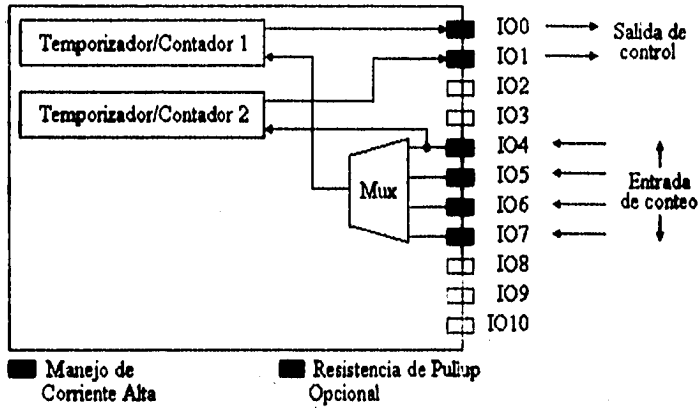


Figura 2.25 Contador de Disparos.

CAPÍTULO III PROTOCOLO LONTALK

3.1 PROTOCOLO LONTALK

El Circuito Integrado NEURON implementa un protocolo completo usando dos de los tres procesadores internos. Los tres procesadores dentro del Circuito Integrado NEURON son usados para ejecutar el software del protocolo junto con el programa de la aplicación. El protocolo LONTALK está diseñado para soportar las necesidades de las aplicaciones. El protocolo es presentado como una colección de servicios que pueden ser invocados. Los servicios pueden ser escogidos por el programador y fijados en el tiempo de compilación. Además, muchos de los servicios escogidos pueden ser cambiados por el instalador, cuando un nodo se está instalando o volviendo a configurar en una aplicación LONWORK particular.

El proceso del protocolo sobre el Circuito Integrado NEURON es independiente al medio de comunicación. Esto permite al Circuito Integrado NEURON soportar una amplia variedad de medios de comunicación, incluyendo: par trenzado, líneas de potencia, radio frecuencias, infrarrojo, cable coaxial, y fibra óptica.

El protocolo LONTALK sigue el modelo de referencia para sistemas abiertos de interconexión (OSI) desarrollado por la International Standard Organization (ISO). En la terminología del ISO el protocolo LONTALK proporciona servicios a los 7 niveles del modelo de referencia OSI como se muestra a continuación:

- Manejador de canales físicos (niveles 1 y 2).
- Nombrando, direccionando, y haciendo rutas (niveles 3 y 6).
- Comunicación confiable y uso eficiente del ancho de banda del canal (niveles 2 y 4).

- **Prioridad (nivel 2).**
- **Acción remota (nivel 5).**
- **Autenticidad (niveles 4 y 5).**
- **Manejo de la red (nivel 5).**
- **Transporte del sistema e interpretación de los datos (nivel 6).**
- **Compatibilidad de aplicación (nivel 7).**

La tabla 3.1 resume los niveles del modelo de referencia OSI y el servicio LONTALK proporcionado por cada nivel.

Tabla 3.1 Niveles del Protocolo LONTALK

Niveles OSI	Propósito	Servicios proporcionados
7 Aplicación	Compatibilidad de la aplicación	Tipos estándar de las variables de la red
6 Presentación	Interpretación de los datos	Transmisión de las variables de la red
5 Sesión	Acciones remotas	Solicitud-respuesta de la Autenticidad en el Manejo de la red
4 Transporte	Confiabilidad de inicio a fin	Unireparto y Multireparto Reconocido y No reconocido; Detección de duplicado
3 Red	Destinación de direcciones	Direccionando ruteadores
2 Enlace	Acceso al medio	Decodificando datos; CRC Verificando el error predictivo CSMA; Evasión de colisión; Opción de Prioridad y Detección de Colisión
1 Físico	Interconexión eléctrica	Interfases de medios específicos y Esquemas de modulación (par trenzado, líneas de potencia, radio frecuencia, cable coaxial, infrarrojo, fibra óptica)

El nivel más bajo del modelo de referencia OSI es el nivel físico. Las funciones incluidas dentro de este estrato se encargan de activar, mantener y desactivar un circuito físico entre un equipo terminal de datos y un equipo de comunicación de datos.

El nivel de enlace es el responsable de la transferencia de datos por el canal. Proporciona a los datos la sincronización necesaria para delimitar el flujo de bits del nivel físico. Asimismo, garantiza la identidad de los bits, encargándose de que

los datos lleguen sin error al equipo terminal receptor de datos. Se ocupa de controlar el flujo de datos para impedir que el equipo terminal de datos se desborde en algún momento. Una de sus funciones más importantes consiste en detectar errores en la transmisión y en recuperar, por distintos mecanismos, los datos perdidos, duplicados o erróneos.

El nivel de red define la interface entre el equipo terminal de datos y una red de comunicación de paquetes, además de la Interface de un equipo terminal de datos con otro a través de esta red. Especifica también las operaciones de encaminamiento por la red, y la comunicación entre distintas redes.

El nivel de transporte proporciona la interface entre la red de comunicación de datos y los tres niveles superiores. Es el nivel que permite al usuario elegir entre diversas opciones de calidad (y de precio) dentro de una misma red (es decir, dentro del nivel de red).

El nivel de sesión funciona como interface del usuario con el nivel de transporte ofrece un mecanismo organizado de intercambio de datos entre usuarios.

El nivel de sesión posee una serie de servicios específicos, primitivas y unidades del protocolo de datos, que están definidos en los documentos del ISO.

El nivel de presentación asigna una sintaxis a todos los datos, es decir, determina la forma de presentación de los datos según este modelo, sin preocuparse por su significado o semántica. Este nivel consta de muchas tablas sintácticas (correspondientes a códigos como el teletipo, ASCCI, Videotex, etc.).

El nivel de aplicación se encarga de atender el proceso de aplicación del usuario final. A diferencia del nivel de presentación, este nivel tiene en cuenta la semántica de los datos. Contiene varios elementos de servicios capaces de gestionar procesos de aplicación.

3.1.1 El Canal Físico

El protocolo LONTALK soporta redes segmentadas por el uso de diferentes medios de comunicación. Los medios soportados por el protocolo LONTALK incluye par trenzado, línea de potencia, radio frecuencia, infrarrojo, cable coaxial, y fibra óptica. Las especificaciones de cada transmisor/receptor LONWORK proporciona la distancia, la velocidad de los datos, y el soporte de la topología.

Todos los nodos LONWORK están físicamente conectados a un canal. Un canal es un medio de transporte físico para un conjunto de datos; una red LONWORK está compuesta de uno o más canales. La forma física de un canal depende del

medio. Por ejemplo, un canal de par trenzado es un alambre de par trenzado; un canal de RF (radio frecuencia) es un medio específico para radio frecuencia; un canal de línea de potencia es un sección continua de alambraje de potencia de corriente alterna.

Diversos canales son conectados por medio de puentes y ruteadores. Los puentes y ruteadores consiste de dos Circuitos Integrados NEURON conectados por vía de sus terminales de entrada/salida. Cada uno de los Circuitos NEURON están conectados a un canal diferente a través de un apropiado transmisor/receptor; un transmisor/receptor proporciona la interface entre un canal y un Circuito NEURON.

La velocidad de los datos en un canal, depende del medio de transmisión y el diseño del transmisor/receptor. Diversos transmisores/receptores, con diferentes velocidades de datos, podrian ser diseñados para un medio, permitiendo intercambios de: distancias, rendimiento efectivo, consumo de energia del nodo y costo.

La operación del rendimiento efectivo soportado por un canal dado, es limitado por diversos factores, además de la velocidad de los datos del canal. En una velocidad baja de los datos o con un conjunto de datos largo, el tiempo de transmisión del conjunto de datos y el retardo en un acceso al medio ordinario forman la región del rendimiento efectivo del paquete. En la velocidad más alta de la transmisión de los datos con paquetes cortos, el poder de procesamiento del paquete del Circuito NEURON limita la ejecución del canal.

La tabla 3.2 y 3.3 estiman el rendimiento efectivo aproximado de la red en función de la velocidad de los datos y el tamaño del paquete. Los números de mayor tráfico puede ser soportado para una ráfaga corta de datos.

Tabla 3.2 Rendimiento Efectivo del Canal del Protocolo LONTALK
(paquetes de 12 bytes)

Velocidad de los datos (K Bit / Seg.)	Número pico de paquetes / Seg.	Número de paquetes probados / Seg.
4.883	25	20
9.766	45	35
19.531	110	85
39.063	225	180
78.125	400	320
156.25	625	500
312.5	700	580
625.0	700	580
1.250.0	700	580

Tabla 3.3 Rendimiento Efectivo del Canal de Protocolo LONTALK
(paquetes de 16 bytes)

Velocidad de los datos (K Bits / Seg.)	Número pico de paquetes / Seg.	Número de paquetes probados / Seg.
4.883	7	5
9.766	13	10
19.531	25	20
39.063	50	40
78.125	100	80
156.25	200	170
312.5	340	270
625.0	500	470
1,250.0	700	560

3.1.2 Nombrando, Direccionando y haciendo Rutas

Un nombre únicamente identifica un objeto dentro de una clase de objetos. Un nombre es asignado cuando un objeto es creado y no habrá cambios en el tiempo que éste exista. Los 48 bits únicos del ID es un nombre para un Circuito NEURON, y no habrá cambio en el tiempo de vida del Circuito NEURON.

Una dirección únicamente identifica un objeto o grupo de objetos dentro de una clase de objetos. A diferencia de un nombre, una dirección puede ser asignada y cambiada en cualquier tiempo después de que un objeto es creado.

Las direcciones LONTALK identifican únicamente la fuente del nodo y la destinación del nodo (o nodos) de un paquete LONTALK. Estas direcciones son también usadas para ruteadores que pasan paquetes selectivamente entre dos canales.

El ID del Circuito NEURON puede ser usado como una dirección. Sin embargo, el ID del Circuito NEURON no es usado como una forma exclusiva de direccionar en el protocolo LONTALK, porque tal direccionamiento sólo soporta operaciones de una por una (no grupos), y las tablas de rutas basadas en las direcciones únicas del nodo tendrían que ser muy largas. Este modo de direccionar se usa principalmente durante la instalación y la configuración, después permite comunicaciones con nodos individuales cuando la topología de la red no se conoce todavía.

Para simplificar la realización de las rutas, el protocolo LONTALK define una forma jerárquica de direccionar, usando dominios, subred, y direcciones de nodo. Esta forma de direccionamiento puede ser usada para la dirección del dominio entero, una subred individual, o un nodo individual. Además facilita el

direccionamiento de diversos nodos dispersos, el protocolo LONTALK define otra clase de direcciones usando direcciones de dominio y grupo.

Esto también simplifica el reemplazamiento de los nodos en una red funcional. El nodo que se reemplaza se le asigna la misma dirección del nodo reemplazado. Así todas las referencias para este nodo desde otra parte de la red no necesitan ser modificadas, como sería el caso si la dirección ID fuera usada.

3.1.2.1 Dominio

Un dominio es una colección lógica de nodos sobre uno o más canales. Las comunicaciones pueden tener lugar únicamente entre nodos configurados en un dominio común; por lo tanto, un dominio forma una red virtual. Varios dominios pueden ocupar un mismo canal, los dominios pueden ser usados para impedir interferencia entre nodos en diferentes redes.

Por ejemplo, dos construcciones adyacentes de nodos usando transmisores/receptores de radio frecuencia en la misma frecuencia, estarían sobre el mismo canal. Para impedir interferencia en las aplicaciones llevadas a cabo por los nodos, los nodos en cada construcción estarán configurados para pertenecer a dominios diferentes.

El Circuito NEURON puede ser configurado para que un nodo LONWORK pueda pertenecer a uno o dos dominios. Un nodo que es miembro de dos dominios puede ser usado como una entrada entre los dos dominios. El protocolo LONTALK no soporta la comunicación entre dominios, pero un programa de aplicación puede ser implementado para hacer cruzar los paquetes entre los dos dominios.

Un dominio es identificado por un dominio ID. El dominio ID puede ser configurado con 0, 1, 3, o 6 bytes. Los 6 bytes para el dominio ID pueden ser usados para garantizar que el dominio ID sea único: por ejemplo, usando los 48 bits del ID de uno de los Circuitos NEURON dentro del dominio como dominio ID, garantiza que ninguna otra red pueda tener el mismo dominio ID, puesto que todos los Circuitos NEURON son únicos. Sin embargo, seis bytes de los dominios IDs suman seis bytes fijos para todos los paquetes. En un sistema donde no hay posibilidad de interferencia entre diversas redes, el dominio ID puede ser configurado con cero bytes. Por ejemplo, Las aplicaciones LONWORK utilizando canales de par trenzado pueden configurarse con dominios de cero bytes si sólo una aplicación será usada en los canales de par trenzado. Los dominios IDs pueden ser configurados con 1 o 3 bytes en sistemas donde un simple control de administración de asignamiento de dominios IDs puedan impedir duplicar los IDs.

3.1.2.2 Subred

Una subred es una colección lógica de hasta 127 nodos dentro de un dominio. Hasta 255 subredes pueden ser definidas dentro de un dominio. Todos los nodos en una subred deben estar sobre un mismo canal, o sobre canales conectados con puentes. Las subredes no pueden atravesar ruteadores.

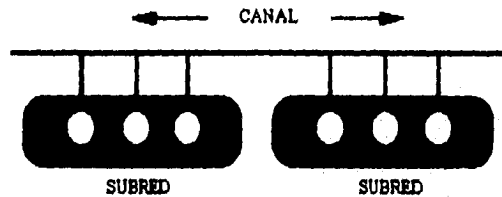


Figura 3.1 Varias SubRedes sobre un Canal Común

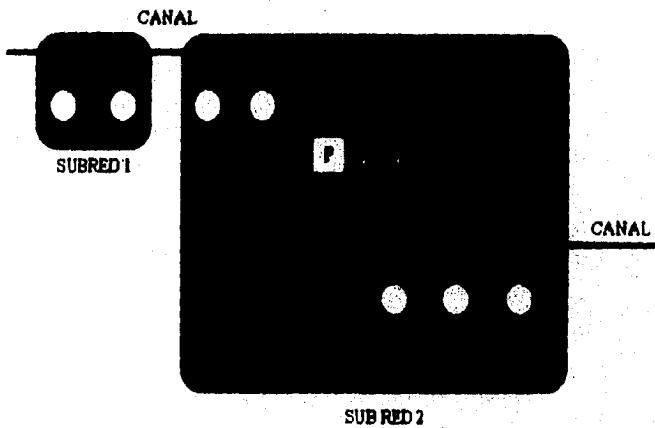


Figura 3.2 Las Subredes pueden atravesar Puentes

Si un nodo es configurado para pertenecer a dos dominios, éste debe ser asignado para una subred dentro de cada uno de los dominios.

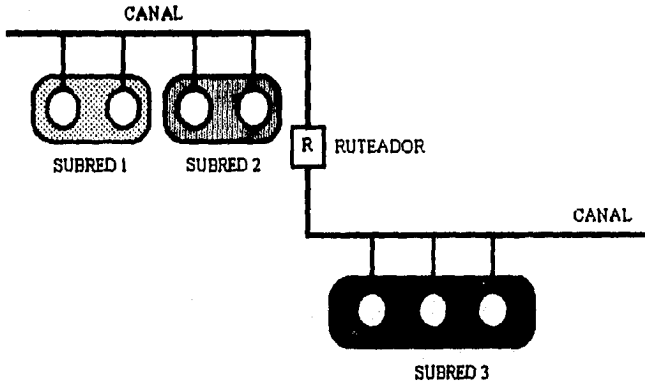


Figura 3.3 Las Subredes no pueden atravesar Ruteadores

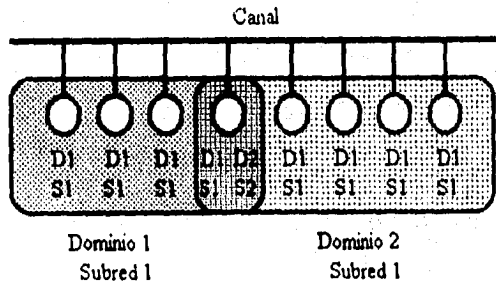


Figura 3.4 Un Nodo configurado en dos Dominios tiene asignado Subredes para cada Dominio

Todos los nodos dentro de un dominio son configurados generalmente en una misma subred, excepto en los siguientes casos:

- Que éstos estén localizados sobre diferentes canales con ruteadores intermedios.
- Configurando los nodos en la misma subred, excediendo el número máximo de nodos permitidos en una subred. Las subredes son limitadas por 127 nodos. Diversas subredes pueden ser configuradas sobre un conjunto de canales conectadas por puentes para incrementar la capacidad de los canales superiores a 127 nodos. Por ejemplo, un conjunto de canales conectados por puentes con dos subredes pueden tener hasta 254 nodos; tres subredes pueden tener hasta 381 nodos.

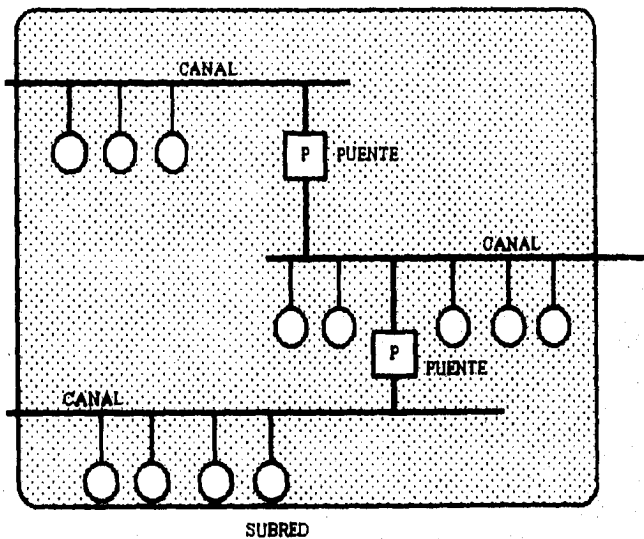


Figura 3.5 Todos los Nodos configurados en un Dominio como una Subred

3.1.2.3 Nodo

Todos los nodos dentro de una subred tienen asignados un número único de nodo dentro de la subred. El número del nodo es de 7 bits, así éstos pueden ser hasta 127 nodos por subred. Un máximo de 32,385 nodos (255 subred X 127 nodos por subred) pueden estar en un dominio.

3.1.2.4 Grupos

Un grupo es una colección de nodos dentro de un dominio. Diferentes a una subred, sin embargo, los nodos son agrupados juntos sin hacer caso de su localización física en el dominio. El Circuito NEURON le permite a un nodo ser configurado para ser un miembro de hasta 15 grupos

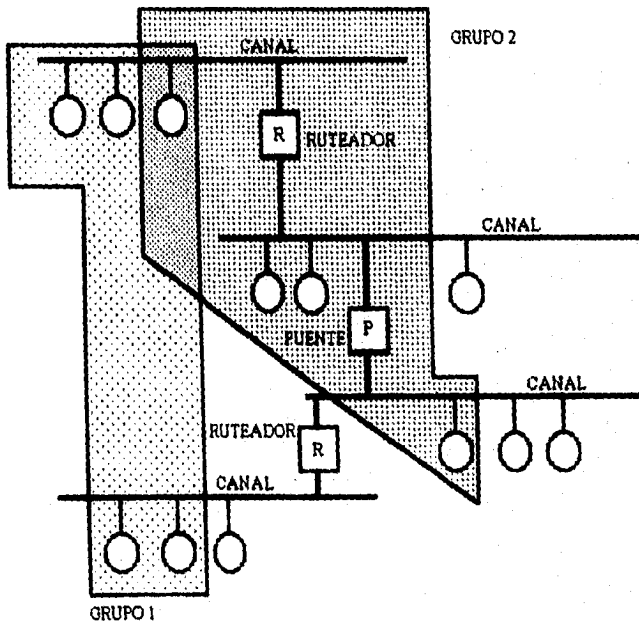


Figura 3.6 Los Grupos pueden atravesar cualquier número de Canales, Ruteadores, o Puentes

Los grupos son identificados por un byte del número de grupo, un simple dominio puede contener hasta 256 grupos.

3.1.2.5 El ID único

Para las dirección subred/nodo, un nodo puede ser siempre direccionado por este ID. El ID tiene 48 bits, y es asignado cuando cada Circuito NEURON es fabricado. Este ID es garantizado para ser único en el mundo

El formato de dirección Dominio/ID es usado por un nodo manejador de red en la configuración inicial de nodos durante el tiempo de instalación, para asignar cada nodo a uno o dos dominios, y para asignar los números de subred y nodo.

3.1.2.5 Los formatos de dirección

Los nodos son direccionados usando uno de los cinco formatos de direcciones. El formato de dirección particularmente usado determina el número de bytes requeridos para la dirección fuente y destino. La tabla 3.4 define los formatos y el número de los bytes requeridos para cada uno. El tamaño total de la dirección es calculada, sumando el número apropiado de bytes indicado en la tabla, al tamaño del dominio ID, el cual puede estar en el rango de 0 a 6 bytes dependiendo del tamaño de configuración del dominio ID.

Tabla 3.4 Formato de direcciones del protocolo LONTALK

Formato de dirección	Destinación	Tamaño de la dirección (bytes)
Dominio (Subred = 0)	Todos los nodos en el dominio	3
Dominio, Subred	Todos los nodos en la subred	3
Dominio, Subred, Nodo	Nodos específicos dentro de una subred	4
Dominio, Grupo	Todos los nodos en el grupo	3
Dominio, ID único	Nodos específicos	9

3.1.2.7 Manejo de redes y generación de direcciones

Dependiendo del nivel de una aplicación dada, una red LONWORK puede o no requerir el uso de un nodo Manejador de Red. Un nodo Manejador de Red es un nodo que tiene que ser específicamente diseñado para ejecutar funciones manejadoras de red, tales como:

- Encontrar los nodos no configurados y cargar la dirección de la red.
- Parar, empezar y reiniciar la aplicación de los nodos.
- Accesar las estadísticas de la comunicación del nodo.
- Configurar los ruteadores y puentes.
- Cargar nuevos programas de la aplicación.
- Extraer la topología de una red corriente.

En un medio ambiente de desarrollo, el papel del nodo manejador de red es ejecutado generalmente por el Manejador de Red LONBUILDER. El Manejador de Red LONBUILDER incluye las herramientas requeridas para definir, configurar, cargar y controlar redes LONWORK. El Analizador de Protocolo LONBUILDER proporciona la capacidad para: monitorear, coleccionar y desplegar el tráfico de la red y la ejecución estática.

El software LONBUILDER examina el contenido entre los nodos y asigna todos los grupos automáticamente, de una manera que optimiza el tráfico de la red

3.1.2.8 Ruteadores y puentes

Un ruteador (o puente) es un nodo especial que consiste de dos Circuitos NEURON conectados, cada uno conectado a un canal separado. Los ruteadores y los puentes pasan los paquetes de un canal a otro.

Existen cuatro tipos de ruteadores.

- Repetidor. Un repetidor es la forma más simple de un ruteador, simplemente hace seguir todos los paquetes, sin hacer caso del dominio, entre los dos canales conectados. Usando un repetidor, una subred puede existir en diversos canales.
- Puente. Un puente simplemente hace seguir todos los paquetes entre dos canales, sobre los dominios en los cuales el ruteador está definido. Usando un puente, una subred puede existir en diversos canales.
- Ruteador configurado. Al igual que un ruteador de aprendizaje, un ruteador configurado, hace rutas de paquetes selectivamente entre canales consultando tablas internas del ruteador. A diferencia de un ruteador de aprendizaje, el contenido de las tablas internas del ruteador son especificadas usando comandos manejadores de red.

- Ruteador de aprendizaje. Un ruteador de aprendizaje monitorea el tráfico de la red y aprende la topología para el nivel de dominio/subred. El ruteador de aprendizaje, entonces utiliza su conocimiento para selectivamente hacer rutas de los paquetes entre canales.

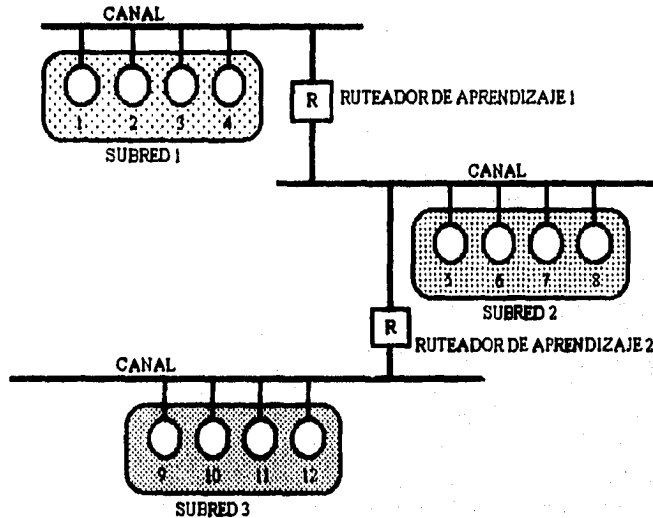


Figura 3.7 Ruteador de Aprendizaje

Inicialmente, cada ruteador coloca sus tablas de rutas internas para indicar que todos las subredes podrían estar situado sobre uno u otro lado del ruteador. Refiriendo a la figura 3.7, suponiendo que el nodo 6 genere un mensaje con destino al nodo 2. El ruteador de aprendizaje 1 inicialmente **capta el mensaje**. Examinando la fuente del campo de la subred del mensaje, el ruteador de aprendizaje apunta en su tabla de rutas internas que la subred 2 está situado abajo de él. El ruteador después comparara la fuente y el destino de las subred IDs, después de que ellos son diferentes, el mensaje es pasado.

Mientras tanto, el ruteador de aprendizaje 2 ha pasado también el mensaje, realizando una apropiada notación en su tabla de rutas internas respecto a la locación de la subred

Suponga ahora que el nodo 2 genera un reconocimiento. Este reconocimiento es captado por el ruteador de aprendizaje 1, que ahora apunta a la locación de la subred 1. El ruteador de aprendizaje 1 examina sus tablas de rutas internas, y, al descubrir que la subred 2 está situada abajo, pasa el mensaje. Cuando el mensaje aparece sobre la subred 2, éste es anotado por el nodo 6 (el nodo destino), y el ruteador de aprendizaje 2, quien no lo pasará, solamente anota que, esta subred 1, igual que la subred 2, está situada en alguna parte arriba. El ruteador de aprendizaje 2 no aprenderá de la existencia o localización de la subred 3 hasta que un mensaje sea originado desde ese lugar.

Cuando se hacen cambios entre ruteadores de aprendizaje y ruteadores configurados, deberá tomarse en cuenta lo siguiente:

- La inundación inicial del tráfico que ocurre mientras un ruteador de aprendizaje está aprendiendo la topología de la red, esto podría causar problemas de congestión.
- La topología de la red podría tener inadvertidos ciclos (loops), líneas de potencia y redes de radio frecuencia comunes, que pueden causar a un ruteador de aprendizaje desarrollar una inexacta imagen de la red.
- Un ruteador de aprendizaje está siempre aprendiendo, y pondrá al día sus tablas de rutas internas haciendo cambios en la topología de la red.
- Las tablas de rutas interna en un ruteador de aprendizaje no tendrán que ser programadas explícitamente.
- Los ruteadores de aprendizaje no aprenden acerca de grupos, pero ruteadores configurados pueden ser configurados para enviar selectivamente paquetes direccionados a grupos.

Las subredes no puede cruzar ruteadores. Mientras que los puentes y los repetidores permiten que la subred atraviere diversos canales, los dos lados de un ruteador deben pertenecer a subredes diferentes. El hecho que los ruteadores son selectivos acerca de los paquetes que ellos hacen pasar a cada canal, puede ser usado para incrementar la capacidad total de un sistema en términos de nodos y conexiones.

3.1.3 Servicios de Comunicación

3.1.3.1 Eficiencia, tiempo de repuesta, seguridad y confiabilidad

Existe una relación de los cambios realizados entre la eficiencia, el tiempo de respuesta, la seguridad y la confiabilidad de la red: El servicio más confiable es el servicio de reconocimiento, pero utiliza un ancho de banda más grande sobre la red que el servicio de sin reconocimiento o reconocimiento repetido para un número muy grande de grupos; asignando prioridad a los paquetes garantizará que estos paquetes sean enviados en una forma oportuna, en perjuicio de otros; sumando servicio de autenticidad para las operaciones diseñadas se requiere de dos veces el número de paquetes para completar una operación con relación a aquellas operaciones que no tengan servicio de autenticidad.

3.1.3.2 Seleccionando servicios de mensaje para la confiabilidad y eficiencia

El protocolo LONTALK ofrece cuatro tipos de servicios básicos de mensaje:

El servicio más confiable es el de reconocimiento, o servicio de reconocimiento inicio a fin, donde un mensaje es enviado a un nodo o grupo de nodos, y los reconocimientos individuales son esperados de cada uno de los receptores. Si un reconocimiento no es recibido de todas las fuentes de destino, el emisor realizará un tiempo fuera y volverá a procesar la operación. Se puede seleccionar el número de reprocesos y el de tiempo fuera. Los reconocimientos son generados por el CPU de la red sin intervención de la aplicación. Las operaciones IDs son usadas para guardar la trayectoria de mensajes y reconocimientos así que la aplicación no recibirá mensajes duplicados.

Un servicio igualmente confiable es el de solicitud/respuesta, donde un mensaje es enviado a un nodo o grupo de nodos y respuestas individuales son esperadas de cada uno de los receptores. El mensaje entrante es procesado por la aplicación sobre el lado recibido antes de que una respuesta sea generada. Las mismas operaciones de reproceso y de tiempo fuera son disponibles como en el servicio de reconocimiento. Las respuestas podrían incluir los datos, así que ese servicio es particularmente conveniente para convocar procedimientos remotos, o aplicaciones cliente/servidor.

El siguiente servicio de mayor confiabilidad es el de repetidor sin reconocimiento, donde un mensaje es enviado, a un nodo o grupo de nodos varias veces, y no se espera la respuesta. Este servicio es usado generalmente cuando se está emitiendo a un gran número de grupos de nodos, en el cual el tráfico generado por todas las respuestas sobrecargarían la red.

El servicio menos confiable es el servicio no reconocido, donde un mensaje es enviado una vez a un nodo o grupo de nodos y no se espera respuesta. Este servicio es usado generalmente cuando se requiere la máxima ejecución, el ancho de banda de la red es limitada, y la aplicación no es sensible a la pérdida de un mensaje.

3.1.3.3 Detección de colisión

El protocolo LONTALK usa un algoritmo único para evitar la colisión.

Cuando se utiliza un medio de comunicación que soporta hardware de detección de colisión (por ejemplo par trenzado), el protocolo LONTALK puede cancelar la transmisión de un paquete tan pronto como una colisión es detectada por el transmisor/receptor. Esto permite que el nodo inmediatamente retransmita algún paquete que ha sido dañado por una colisión. Sin la detección de colisión, el nodo tendría que esperar la duración del tiempo de reproceso para darse cuenta que el reconocimiento no fue recibido, tiempo en el cual éste retransmitiría el paquete, asumiendo el servicio de reconocimiento o solicitud/respuesta. Para el servicio sin reconocimiento, una colisión no detectada significa que el paquete no es recibido y el reproceso no es intentado.

3.1.3.4 Prioridad

El protocolo LONTALK ofrece opcionalmente un mecanismo de prioridad para mejorar el tiempo de respuesta de paquetes críticos. El protocolo permite al usuario asignar el tiempo de las ranuras de prioridad sobre un canal, dedicado a la prioridad de los nodos. La red manejadora de nodos, que asigna individualmente las ranuras de prioridad a los nodos, puede asegurar que uno y sólo un nodo esté asignado para una ranura de prioridad particular sobre el canal. Cada tiempo de ranura de prioridad sobre un canal suma un mínimo de dos bits de tiempo para la transmisión de todo el mensaje. La cantidad del encabezado variará basándose en la exactitud de la velocidad del oscilador del dato y el transmisor/receptor necesario. Por ejemplo usando un transmisor/receptor LONWORK de 1.25 Mbps de par trenzado con todos los nodos sobre el canal y teniendo un oscilador con exactitud de 0.2% o uno mejor, cada ranura de prioridad es de 30 bit de ancho de tiempo. Ya que no existe contención por el medio durante la porción de prioridad de un ciclo de paquete, los nodos configurados con prioridad tienen mejor tiempo de respuesta que los nodos que no tiene prioridad. La combinación de prioridad y detección de colisión permite limitar el tiempo de respuesta.

La ranura de prioridad asignada a un nodo, es aplicada a todos los paquetes de prioridad enviados desde este nodo. Uno, todos, o algunos de los paquetes

enviados desde un nodo podría ser marcada con el uso del servicio de prioridad. La designación de la prioridad dentro de un nodo es realizada sobre cada una de las variables de la red o sobre cada etiqueta de mensaje base, y puede ser colocada en el tiempo de compilación, en el caso de las variables de la red, la designación de prioridad puede ser cambiada opcionalmente durante o después de la instalación.

Los números de prioridad más bajos indican los más altos niveles de prioridad: un paquete de prioridad desde un nodo con ranura de prioridad de número 2 será transmitido antes que un paquete de prioridad desde un nodo con ranura de prioridad con número 4. Colocando una prioridad de nodo a 0 indica que ninguno de estos paquetes será transmitido en un ranura de prioridad, sin atender la asignación del servicio de mensaje hecha en la compilación del nodo o tiempo de instalación. La ranura número 1 es reservada generalmente para un nodo manejador de red, para garantizar que la aplicación no pueda ser ejecutada por un canal incapaz de ser interrumpido por un manejador de red. La ranura 2 hasta el 127 (dependiendo del medio, y el número de ranuras alojados sobre el canal) están disponibles para realizar prioridades de paquetes desde nodos diseñados.

Cuando un paquete de prioridad es generado dentro de un nodo, éste viaja fuera del nodo sobre la cola de prioridad antes de que algún paquete pendiente sin prioridad puesto en la memoria intermedia para ser transmitido.

3.1.3.5 Autenticidad

El protocolo LONTALK soporta mensajes de autenticidad; los receptores de un mensaje determinan si el envió está autorizado para enviar ese mensaje. Este puede impedir el acceso sin autorización a nodos y sus aplicaciones. El uso de la autenticidad es configurada individualmente para cada variable de la red. La operación del manejador de red puede también hacer uso de la autenticidad.

La autenticidad es implementada para distribuir 48 bits llaves a los nodos en el tiempo de instalación, con el envió y receptor de un mensaje de autenticidad ambos poseyendo la misma llave. Cuando un mensaje de autenticidad es enviado, el receptor desafía a el envió a proporcionar autenticidad, cada vez usando un desafío diferente al azar. El envió después utiliza la llave de autenticidad y el dato desde el paquete original, para ejecutar una transformación en el desafío, y responderlo. El receptor compara la respuesta a el desafío con su propia transformación sobre el desafío. Si las transformaciones son iguales, la operación tendrá éxito. La transformación usada es diseñada, así que es extremadamente difícil deducir cual llave es, igual que si el desafío y la respuesta son conocidas.

3.1.4 Temporizadores del Protocolo LONTALK

Existen diversos temporizadores los cuales necesitan ser colocados en orden en el protocolo LONTALK para funcionar eficientemente. Estos incluyen:

- Temporizador de operación.
- Temporizador de repetición.
- Temporizador de recepción para grupo.
- Temporizador de recepción para no grupo.
- Temporizador de espera de liberación de memoria intermedia.

La siguiente sección detalla, para cada uno de los cuatro servicios de mensaje, como los paquetes fluyen directo a el Circuito NEURON y donde los temporizadores son utilizados.

3.1.5 Paquetes Fluyendo al Circuito NEURON

Existen cuatro diferentes servicios de mensaje en el protocolo LONTALK: sin reconocimiento, con reconocimiento, repetidor sin reconocimiento, y solicitud/respuesta. Usualmente, estos servicios pueden ser usados con algún modo de direccionamiento. Existen dos excepciones: primero, cuando se ejecuta una emisión solicitud/respuesta, la aplicación recibirá solo la primera respuesta; todo lo demás serán descartado por el procesador de la red. Segundo, la emisión de operaciones con reconocimiento se completa una vez que un reconocimiento es recibido.

SIN RECONOCIMIENTO: Cuando este servicio es usado, el único temporizador que es requerido es el de espera de liberación de memoria intermedia. Este temporizador determina la máxima extensión de tiempo que el Circuito NEURON esperará para una liberación de la memoria intermedia cuando se envíe un mensaje. Este temporizador puede ser desactivado (el Circuito NEURON esperará por siempre), colocando el valor del temporizador en cero. Si éste es colocado en otro número n , el microprograma esperará entre $2n$ y $2n+1$ segundos. Por ejemplo, si el número de configuración es colocado a 2, el microprograma esperará para una liberación de memoria intermedia entre 4 y 5 segundos. Si una memoria intermedia no es obtenida antes de que el conteo expire, el Circuito NEURON asume un error fatal y reiniciará.

RECONOCIMIENTO: El servicio de reconocimiento también utiliza el temporizador de espera de liberación de memoria intermedia. Pero son también requeridos algunos temporizadores adicionales:

1. El temporizador de operación determina cuanto tiempo el nodo espera a un reconocimiento antes de volver a procesar. Si el nodo no recibe un reconocimiento antes de que el temporizador de operación expire, éste procesará de nuevo, enviando el mismo paquete. Este reprocesamiento continuará hasta que la cuenta de reproceso haya sido agotada o hasta que todos los reconocimientos hayan sido recibidos. La cuenta de reproceso se puede configurar desde 0 hasta 15.

Note que un paquete podría ir a ruteadores y puentes para extender su destino final. El tiempo de operación deberá ser bastante largo, para que un paquete pueda alcanzar el "más lejano" destino y el reconocimiento desde este destino puede ser recibido antes de que el temporizador de operación expire. Si el temporizador de operación es muy corto, el acceso a reprocesar será generado; si es muy largo, el tiempo para una operación a completar incrementará en promedio.

El sistema LONBUILDER y el LONMANAGER API automáticamente calculan el tiempo de operación basándose en la topología, la velocidad del dato, y la entrada de frecuencia de reloj del Circuito NEURON.

2. Cuando un paquete llega a su destino final, los nodos receptores mirarán hacia la dirección fuente del paquete y la operación ID. Si no están activadas las "operaciones recibidas" con esta pareja de fuentes dirección/operación ID, una nueva anotación de operación recibida es creada. Si la anotación de operación recibida no existe porque el nodo ha usado todas con operaciones activadas, la entrada del mensaje es perdida. Si se asume que el nodo receptor puede alojar una anotación de operación recibida, éste iniciará a un temporizador de recepción. Este elige cual temporizador de recepción usa, basándose en el modo de dirección que el transmisor utilizó. Si el transmisor utilizó grupo direccionado, existe una entrada de tabla de dirección de reproceso para este grupo, y el valor del temporizador de recepción para grupo es tomada desde que se reprocesa en la tabla de dirección. Si algún otro modo de direccionamiento es usado, el nodo usa sus valores de temporizador de recepción para no grupo.

Cuando el temporizador de recepción expira, la anotación de operación es borrada, y si alguna nueva operación tiene la misma operación ID desde la misma fuente de dirección será tratada como una nueva operación. Por lo tanto, ese temporizador tendrá que ser mayor que el más grande producto del conteo de reproceso y el temporizador de operación que puede ser recibida desde el transmisor.

Si el temporizador de operación de recepción es muy largo, entonces es probable que el nodo se termine la memoria por recibir las operaciones de la memoria intermedia (buffers). Si es muy corto, entonces el nodo podría tener reprocesos equivocados para nuevas operaciones, causando mensajes duplicados por error.

SIN RECONOCIMIENTO/REPETIDO: Este servicio sigue esencialmente el mismo flujo de mensaje del servicio de reconocimiento, con algunas excepciones. En la tabla de dirección del transmisor existe un temporizador separado conocido como el temporizador de repetición. Este temporizador especifica que tan frecuentemente el mensaje es repetido cuando se utiliza el servicio de sin reconocimiento/repetido. Este tiempo puede ser más corto que el temporizador de operación, porque no espera un reconocimiento cuando esos mensajes son enviados. Las operaciones IDs y la detección de duplicado deben estar vigentes por esas operaciones. El transmisor envía el mensaje n veces a intervalos de m milisegundos, donde n es el temporizador de reproceso y m es el valor del temporizador de repetición.

SOLICITUD/RESPUESTA: El flujo del mensaje para este servicio es idéntico al del servicio de reconocimiento, excepto que la aplicación envía una respuesta en lugar de un reconocimiento. El hecho de que el programa de aplicación suma tiempo de procesos extra a la generación del reconocimiento deberá ser tomado en cuenta cuando se coloque el temporizador de operación y el temporizador de recepción.

3.1.6 Interpretación de los Datos

El protocolo LONTALK emplea un protocolo de aplicación orientado a datos. En este acercamiento, la aplicación de los datos tales como temperaturas, estados de presiones, y otros datos son cambiados entre nodos en ingeniería normal y otras unidades predefinidas. Las funciones de comandos son después encapsuladas dentro de los programas de la aplicación de los nodos receptor antes de que sean enviados sobre la red. En este camino, el mismo valor ingenieril puede ser enviado a varios nodos, los cuales tendrán diferente aplicación para esos datos.

3.1.6.1 Variables de la red

Los datos en el protocolo de la aplicación LONTALK son llamados variables de la red. Las variables de la red pueden ser algún dato o estructura de datos. Los escritores de la aplicación necesitan también declarar esas variables usando la palabra llave *network*, y la variable estará disponible para algún otro nodo dentro

de la red. Cuando la salida de las variables de la red cambien por vía de las operaciones de asignamiento dentro del programa de la aplicación, la ejecución construida dentro del microprograma NEURON, propagará automáticamente el nuevo valor sobre la red usando el servicio de protocolo LONTALK. Este mensaje implícito libera al escritor de aplicación del manejo de memoria intermedia, inicialización de mensaje, analizar mensajes, y errores de manejo.

3.1.7 Compatibilidad de la Aplicación

La compatibilidad de la aplicación se facilita a través del uso de Tipos de Variable de Red Estándar (Standard Network Variable Types), o SNVTs. La lista inicial de SNVTs incluye casi 100 tipos y cubren un amplio rango de aplicaciones. La definición de un SNVT incluye las unidades, un rango, y una resolución. Usando el apropiado comando manejador de red, un nodo LONWORK puede extraer la información SNVT de algún otro nodo.

3.1.8 Servicios del Protocolo y Parámetros

Los servicios del protocolo LONTALK pueden ser seleccionados por un programa de la aplicación de nodo o por un nodo manejador de red. En general, un nodo manejador de red podría pasar por alto los servicios seleccionados por el programa de la aplicación.

Durante el desarrollo, el manejador de red LONBUILDER ejecuta el papel del nodo manejador de red que será usado para instalar nodos en la aplicación final. Los servicios seleccionados y configurados por el manejador de red LONBUILDER en el tiempo de desarrollo del nodo, podría ser diferente del servicio seleccionado y configurado por el nodo manejador de red en la aplicación final.

Tabla 3.6 Parámetros de imagen de la Red

Parámetro	Cuando/donde instalarse	Bases para la configuración	¿ Se puede cambiar cuando el nodo está instalado ?	¿ La opción del tiempo de compilación puede prevenir el sobremanejo de la puesta inicial ?
Velocidad del canal de los datos	Compilación o Instalación	Por nodo	Si	No
Dominio ID	Instalación	Por dominio	Si	No
Dirección subred/nodo	Instalación	Por dominio	Si	No
Dirección(es) de grupo	Instalación	Por nodo	Si	No
Nodo único ID	Fabricación	Por nodo	No	No
Servicio de reconocimiento -Mensaje explícito	Compilación	Por variable de la red o por mensaje explícito	No	No
Servicio de reconocimiento -Variable de la red	Compilación o Instalación	Por variable de la red o por mensaje explícito	Si	Si
Conteo de reproceso	Instalación	Por variable de la red o por mensaje explícito	Si	No
Servicio de autenticidad -Mensaje explícito	Compilación	Por variable de la red o por mensaje explícito	No	No
Servicio de autenticidad -Variable de la red	Compilación o Instalación	Por variable de la red o por mensaje explícito	Si	Si
Llave de autenticidad	Compilación o Instalación	Por dominio	Si	No
Número de ranuras de prioridad	Instalación	Por nodo	Si	No
Servicio de prioridad -Mensaje explícito	Compilación	Por variable de la red o mensaje explícito	No	No
Servicio de prioridad -Variable de la red	Compilación o Instalación	Por variable de la red o mensaje explícito	Si	Si
Tipos de las variables de la red	Compilación	Por variable de la red o mensaje explícito	No	No

3.1.9 Límites y Regiones

La extensión del dominio IDs LONTALK podría ser 0, 1, 3, o 6 bytes. Todos los nodos en un dominio común deben tener indicado el dominio IDs de la misma extensión. Dentro de cada dominio LONTALK, los siguientes límites se aplican.

Límites LONTALK

- Un máximo de 225 subredes.
- Un máximo de 127 nodos por subred.
- Un máximo de 256 grupos.
- Un máximo de 64 nodos por grupo (sólo servicio de reconocimiento, no existe límite de nodos sobre grupos usando servicio sin reconocimiento).
- Un máximo de 32,385 nodos.
- Un nodo tiene una subred y una dirección de nodo por Dominio al cual pertenece el nodo.
- Grupo asociados deben estar en un Dominio al cual pertenece el nodo.
- Un nodo puede tener hasta 255 variables de la red definidas.

Límites del nodo LONWORK

- Un nodo basado en un Circuito NEURON 3120 o NEURON 3150 puede pertenecer a un máximo de 2 Dominios.
- Un nodo puede tener sólo una salida de operación en el proceso a la vez.
- Un nodo puede tener sólo una autenticidad de operación en el proceso a la vez.
- Una aplicación basada en un Circuito NEURON 3120 o un NEURON 3150 puede tener hasta 62 variables de la red definidas.
- Un nodo basado en un Circuito NEURON 3120 o un NEURON 3150 puede ser un miembro de hasta 15 grupos.
- Todos los nodos deben ser capaz de recibir 60-byte.
- Todos los nodos deben ser capaz de enviar 32-byte.

3.1.10 La Tabla de Dirección

El tamaño de la tabla de dirección de un nodo puede ser configurado por el programador de la aplicación. El *default* especifica 15 direcciones separadas de destino en la tabla de dirección. La tabla de dirección se usa para obtener la dirección a donde se va a enviar un mensaje, y, en el caso de direcciones de grupo, se usa para determinar si el mensaje recibido es de un grupo, donde el nodo es un miembro. Por cada entrada a la tabla de dirección, el primer byte contiene uno de los siguientes tres tipos de entrada de tablas de dirección: grupo, subred/nodo, y de reparto.

La estructura de grupo tiene entradas para el tamaño del grupo, cual dominio utiliza, cual es el número del nodo en el grupo, un temporizador de operaciones, un temporizador de repetición, un conteo de reproceso, un temporizador de recepción, y el grupo ID.

La estructura subred/nodo tiene entradas para el dominio, el número de nodo de destino, número de la subred de destino, un temporizador de repetición, un conteo de reproceso, y un temporizador de recepción.

La estructura de reparto tiene entradas para el dominio y el destino de la subred (0 implica todas las subredes), un temporizador de repetición, un conteo de reproceso y un temporizador de recepción.

3.2 MEDICIÓN DEL TIEMPO DE RESPUESTA DEL LONTALK

La optimización del tiempo de respuesta describe diversas técnicas proporcionadas en el protocolo LONTALK para perfeccionar el tiempo de respuesta de la red. Este incluye:

- Diseñar la red permitiendo suficiente ancho de banda para conocer el tiempo crítico de respuesta de la aplicación.
- Establecer la prioridad para paquetes críticos
- Implementar el hardware de detención de colisión para nodos críticos.
- Hacer colas en la comunicación de memoria intermedia colocándolos apropiadamente.

- **Escoger los mejores servicios de protocolo teniendo en cuenta la aplicación y el ancho de banda disponible de la red.**
- **Colocar el conteo del reproceso, temporizador de operación, temporizador de repetición, y el tamaño de la puesta de operación de recepción, de tal modo que no fracasen por causas: de la expiración de la anotación de la operación recibida, por incorrecto conteo de reproceso, o por la mala colocación de los temporizadores (tiempos demasiados cortos o demasiados largos).**

CAPÍTULO IV TRANSMISORES/RECEPTORES LONWORK

4.1 INTRODUCCIÓN

Una característica de las redes LONWORK es su habilidad para comunicarse con diferentes medios de transmisión en una red.

La habilidad de los nodos para comunicarse con una variedad de medios es soportada por una familia de productos transmisores/receptores para las redes LONWORK. Los primeros productos, desarrollado por Echelon, son un juego de módulos que permiten a los Circuitos Integrados NEURON comunicarse sobre alambrados de líneas de potencia, radio frecuencia y par trenzado. Además, el puerto de comunicación de los Circuitos Integrados NEURON habilitan el desarrollo de la implementación de transmisores/receptores para otros medios (ejemplo: cable coaxial, fibra óptica, etc.) para necesidades especiales.

La circuitería del puerto de comunicación del Circuito Integrado NEURON habilitan a los nodos para ser conectados directamente via las terminales de E/S del puerto de comunicaciones. No obstante, los nodos LONWORK que se comunican a grandes distancias o sobre medios ruidosos requieren un transmisor/receptor para proveer la interface física entre el nodo y el medio sobre el cual se comunican. El transmisor/receptor codifica y decodifica los bits de datos transmitidos y recibidos en un formato apropiado para el medio. Para algunos medios como el par trenzado, el Circuito NEURON contiene la mayoría de la circuitería del transmisor/receptor, y se requiere de muy poca circuitería de interface externa para conectarla con el medio. Para otros medios, como líneas de poder y radio frecuencia, se requiere de un transmisor/receptor externo para soportar un esquema más complejo de decodificación.

Para cada medio, los transmisores/receptores LONWORK manipulan el esquema de codificación particular del medio y provee la interface completa

mecánica y eléctrica entre el medio y los puertos de comunicación de los Circuitos NEURON. Las especificaciones para algunos transmisores/receptores LONWORK son definidos en la tabla 4.1.

Tabla 4.1 Especificaciones para algunos Transmisores/Receptores

Medio	Velocidad de datos	Rango	Comentario
Línea de potencia	9600 bps	55 db de atenuación en presencia de fuentes de ruido comunes de línea de potencia	Diseñado para la implementación usando un IC para el Circuito NEURON
Radio frecuencia	5 kbps	9.1 m de interior. 45.7 m de exterior	Baja potencia, el modo de operación soporta baterías
Par trenzado	78 kbps 1.25 Mbps	1,372 m, con 64 nodos 457 m, con 64 nodos	Bus de terminación doble. Usa 1 par para potencia local opcional y 2 pares cuando la potencia es suministrada sobre la red.

Para un rápido desarrollo, los módulos de transmisores/receptores están disponibles para cumplir con las especificaciones de los transmisores/receptores de línea de potencia, radio frecuencia, y par trenzado como una parte del Banco de Desarrollo LONBUILDER.

El alambrado de par trenzado ofrece ventajas para aplicaciones típicas de control. El cable está disponible en versiones de blindaje y no blindaje. En relación con la fibra óptica y cable coaxial, el cable de par trenzado es barato, simple de trabajar con él, y de fácil mantenimiento.

Echelon ofrece un módulo de par trenzado que se incorpora a un Circuito NEURON. El pequeño tamaño de estos módulos les permite ser montados sobre o dentro de un producto de diseño propio.

Usando un módulo de control de par trenzado puede salvar literalmente cientos de horas de tiempo de desarrollo comparado con el diseño de un módulo.

Puede haber situaciones en las cuales un módulo de control de par trenzado no sea la mejor solución debido al tamaño de restricciones. Para implementar un diseño de transmisor/receptor de par trenzado, existen cuatro tipos de diseños de transmisores/receptores. Estos incluyen:

- Conexión directa diferencial.
- EIA RS-485.

- Transformador de aislamiento de 78 Kbps.
- Transformador de aislamiento 1.28 Mbps.

Cada transmisor/receptor tiene limitaciones específicas asociadas con su diseño. Estas limitaciones incluyen el número total de nodos que pueden ser interconectados, la distancia ente los nodos, el largo total del cable de la red, características del alambre, tiempo de respuesta, aislamiento eléctrico, y el costo del transmisor/receptor.

El transmisor/receptor de par trenzado LONBUILDER permite a los usuarios desarrollar y depurar redes LONWORK usando par trenzado en el medio.

4.2 COMUNICACIÓN DE LA RED

Las cinco terminales del puerto de comunicación son la conexión a la red, que hace interface con una amplia variedad de medios de interfaces (transmisor/receptor). El puerto de comunicación puede ser configurado para operar en uno de dos modos: Modo directo (terminación simple o diferencial) o modo de propósito especial. Ver tabla 4.2 y figura 4.1.

Tabla 4.2 Características de las Terminales del Puerto de Comunicaciones

Terminal	Manejo de corriente (mA)	Modo directo diferencial	Modo directo de Terminación simple	Modo de propósito especial
CP0	1.4	RX+(entrada)	RX (entrada)	RX (entrada)
CP1	1.4	RX-(entrada)	TX (salida)	TX (salida)
CP2	35	TX+(salida)	TX habilitada (salida)	Bit de Reloj (salida)
CP3	35	TX-(salida)	Dormido (salida)	Dormido (salida) o Despierto (salida)
CP4	1.4	DC(entrada)	DC (entrada)	Sistema de reloj (salida)

RX-Receptor TX-Transmisor DC-Detector de colisión

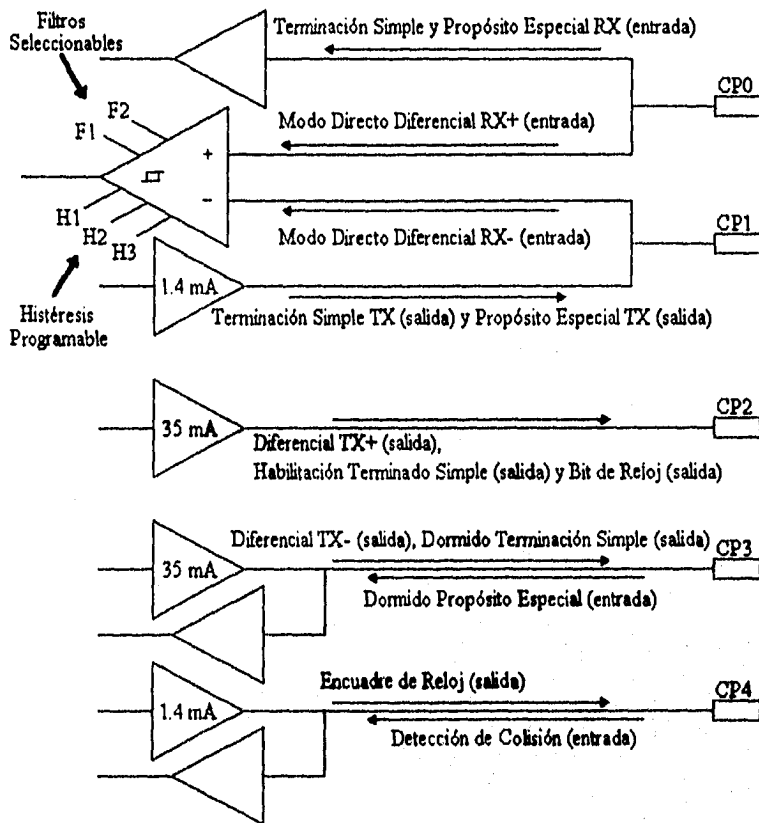


Figura 4.1 Diagrama de Bloques del Transmisor/Receptor interno

4.2.1 Modo Directo

En modo directo, las comunicaciones del procesador con el transmisor/receptor usan una codificación de datos Manchester Diferencial (ver figura 4.2). Esto garantiza una transmisión en cada periodo de bit con el propósito de sincronizar el reloj receptor. El dato cero/uno es indicado con la presencia o ausencia y una segunda transición a la mitad de una de las transiciones de reloj.

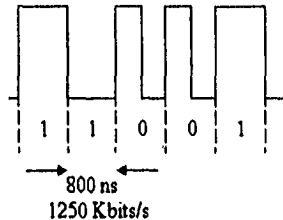


Figura 4.2 Datos Codificados del Modo Directo Manchester Diferencial

Tabla 4.3 Velocidad de los Datos de la Red de Terminación Simple y Diferencial

Velocidad de los bits de la red (kbps)	Reloj de entrada mínimo (MHz)	Reloj de entrada máximo (MHz)
1.250	10.0	10.0
625	5.0	10.0
312.5	2.5	10.0
156.3	1.25	10.0
78.1	0.625	10.0
39.1	0.625	10.0
19.5	0.625	10.0
9.8	0.625	10.0
4.9	0.625	5.0

4.2.1.1 Modo directo de terminación simple

El modo de terminación simple es el más comúnmente usado con transmisores/receptores con componentes externos activos, realizando interface de medios de comunicación como con radio frecuencia, infrarrojo, fibra óptica, y cable coaxial. Le figura 4.3 muestra la configuración de la comunicación de puertos para la operación del modo de terminación simple. En el modo de terminación simple la comunicación de los datos ocurre sobre las terminales CP0 y CP1. Los modos de terminación simple y diferencial usan el código Diferencial Manchester, el cual es un formato ampliamente usado y confiable para la transmisión de datos sobre varios medios de comunicaciones. La velocidad disponible de los bits de la red para el modo diferencial y de terminación simple son dadas en la tabla 4.3 como una función de la entrada de velocidad de reloj del Circuito NEURON.

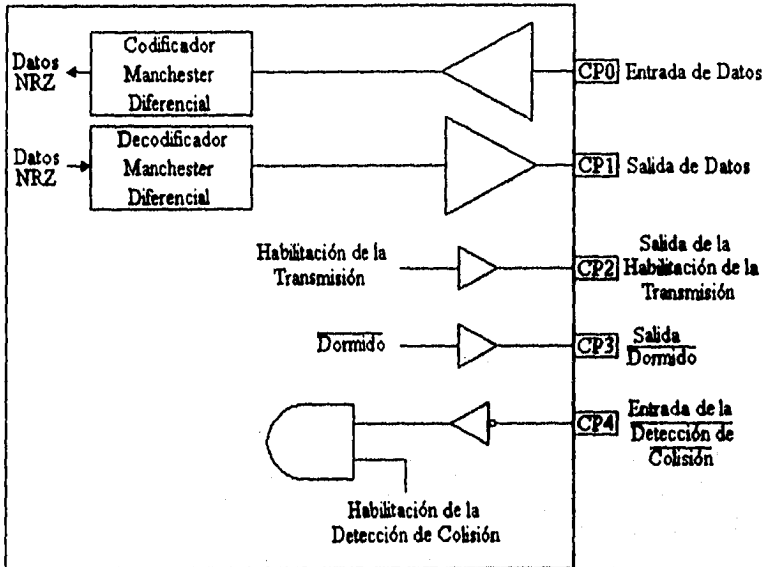


Figura 4.3 Configuración del Modo de Terminación Simple

En el modo de terminación simple, el puerto de comunicación codifica los datos transmitidos y decodifica los datos recibidos usando el código Manchester Diferencial (también conocido como código bifásico especial). Este esquema garantiza una transición al inicio del periodo de todos los bits para sincronizar el reloj receptor. El dato de cero/uno es indicado con la presencia o ausencia de una segunda transición a la mitad de una de las transiciones del reloj. Una caída a la mitad de la transición indica un cero. El carecer de una caída a la mitad de la transición indica un "uno". Así, el cambio de polaridad en la comunicación ligada no afectará los datos recibidos. La figura 4.4 muestra un conjunto de datos típico donde T es el periodo del bit, igual a $1/(\text{velocidad del bit})$.

El transmisor envía un preámbulo al empezar un conjunto de datos para permitir a los otros nodos la sincronización de sus relojes receptores. El preámbulo consiste de una serie de 1's (usando el código Diferencial Manchester). Esta duración es menor de seis bits y puede ser seleccionado por el usuario.

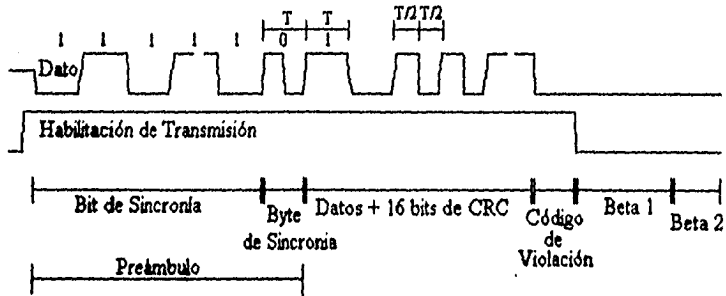


Figura 4.4 Formato de Datos del Modo de Terminación Simple

El preámbulo finaliza con una señal de byte de sincronía, la cual marca el inicio de los bits de datos. El byte de sincronía es un cero Manchester Diferencial.

El Circuito NEURON termina el conjunto de datos forzando un código de violación Manchester Diferencial. Después envía los últimos bits CRC (comprobación por redundancia cíclica), éste mantiene la salida del dato sin transición por 2 1/2 bits de tiempo. La habilitación de la terminal de transmisión sobre CP2 es después puesta en bajo, indicando el final de la transmisión. Para un Circuito NEURON, el tiempo para realizar un formato y empezar a enviar un mensaje de 12 bytes es desde 2.8 hasta 44.8 ms dependiendo de la velocidad de entrada del reloj (10 MHz hasta 625 KHz).

Como una opción, el Circuito NEURON acepta una entrada de activación baja para el detector de colisión desde el transmisor/receptor. Si la detección de colisión es habilitada y CP4 está en bajo por al menos un periodo de reloj (200 ns con un reloj de 10 MHz) durante la transmisión, al Circuito NEURON se le señala que tiene una colisión o que está ocurriendo y que el mensaje tiene que ser tomado como erróneo. El nodo después intenta volver a acceder el canal.

Si el nodo no está usando el detector de colisión, entonces el único camino para poder determinar que un mensaje no ha sido recibido es pedir un reconocimiento. Cuando el servicio de reconocimiento es usado, se inicia un proceso de conteo para permitir el tiempo suficiente a que un mensaje sea enviado y reconocido (48 a 96 ms generalmente para 1.25 Mbps cuando no hay ruteador en la trayectoria de transmisión). Si se procesa de nuevo el conteo del tiempo fuera, el nodo intenta reaccessar el canal. El beneficio de usar el detector de colisión es que el nodo no tiene que esperar el nuevo procesamiento del contador de tiempo fuera

antes de intentar volver a enviar el mensaje, porque el nodo detecta la colisión cuando ésta envía el conjunto de datos.

De la figura 4.5 el tiempo beta 1 es el periodo desocupado después de que un conjunto de datos ha sido enviado

Las ranuras de prioridad (P) y no prioridad son definidas por el tiempo beta 2. Los nodos obedecen a la prioridad de la red para transmitir un conjunto de datos. Esto impide a los nodos transmitir conjuntos de datos uno en cima de otro, excepto cuando los conjuntos de datos son inicializados casi al mismo tiempo. Además, los nodos tienen un tiempo aleatorio antes de empezar la transmisión sobre la red. Cuando la red está desocupada, todos los nodos tienen un retardo de tiempo aleatorio sobre 16 ranuras. Cuando la carga de la red estimada se incrementa, los nodos aumentan las ranuras a fin de bajar la probabilidad de una colisión. El número de ranuras para los tiempos aleatorios (R) varía desde 16 hasta 1008, basado en "n", la estimación de la acumulación de canal (el número de ranuras es $n \times 16$ donde "n" tiene un rango de 1 a 63). Ver figura 4.5

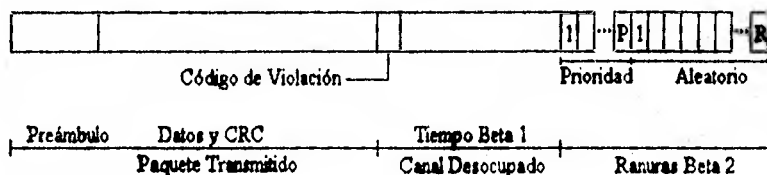


Figura 4.5 Tiempos de los Paquetes

Cuando se pone en modo directo de terminación simple el Circuito NEURON puede hacer una interfaz para un transmisor/receptor IC externo EIA-485. Ver figura 4.6. Mientras no tenga la ejecución de un circuito transformador acoplado, éste puede ofrecer una solución más económica. El dispositivo EIA-485 puede ser encontrado en versiones bipolar y CMOS. El puerto de comunicación tiene que ser puesto en modo de terminación simple para implementar una red EIA-485. El dispositivo EIA-485 especifica 32 manejadores y 32 recibidores sobre un alambrado simple de par trenzado. Muchos tipos de alambrados pueden ser usados con este acercamiento pero la detección de colisión puede ser difícil de implementar.

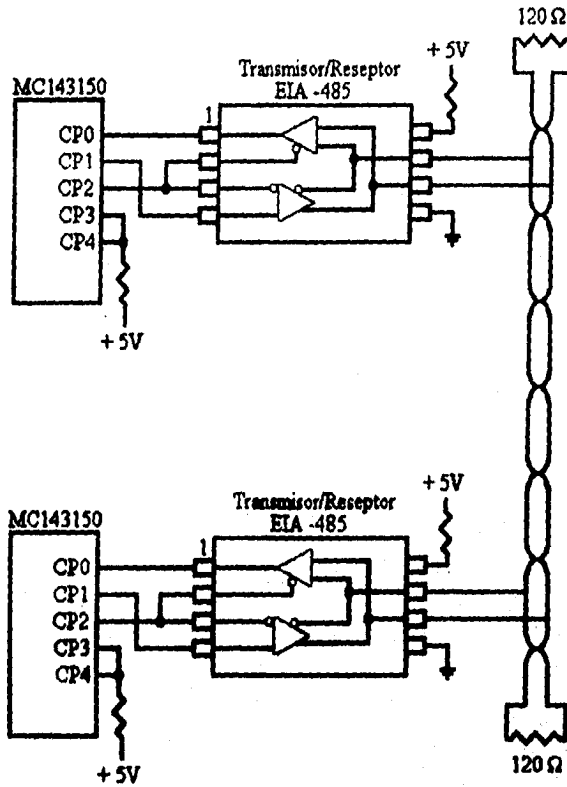


Figura 4.6 Interfaz de Par Trenzado
(Usado con el Modo de Terminación Simple)

4.2.1.2 Modo directo diferencial (conexión directa)

En el modo diferencial, La construcción interna del transmisor/receptor del Circuito NEURON maneja y sense diferencialmente una línea de transmisión de par trenzado con componentes externos pasivos. El modo diferencial es similar en la mayor parte al modo de terminación simple; la diferencia es que el circuito manejador/recibidor es configurado para líneas de transmisión diferencial. Los

datos de salida de las terminales CP2 y CP3 son manejados para estados opuestos durante la transmisión y puestos en un estado de alta impedancia (sin manejar) cuando no está transmitiendo. El circuito de recepción diferencial tiene selección de histéresis sobre las terminales CP0 y CP1, con ocho niveles de selección de voltaje seguido por un filtro paso bajas que puede seleccionarse con cuatro valores de supresión de pulsos transitorios (ruido). Ver figura 4.7. La selección de la histéresis y el filtro permiten optimizar la ejecución de recepción. Ver tabla 4.4 y 4.5 para valores específicos.

La figura 4.8 muestra la forma de un conjunto de datos típico en modo diferencial. Note que el formato del conjunto de datos es idéntico al de la interface de terminación simple descrita anteriormente: la codificación y la tolerancia al ruido, también se aplican idénticamente.

Los manejadores soportan una corriente de 35 mA a 4.2 V y pueden entregar 35 mA de corriente a 0.8 V ($V_{dd} = 5V$). El valor programado de histéresis mostrado en la figura 4.1 corresponde al umbral diferencial del receptor. Es recomendable que para el nivel de la señal, el receptor guarde de 2 a 3 veces más arriba del umbral programado. Ver tabla 4.5.

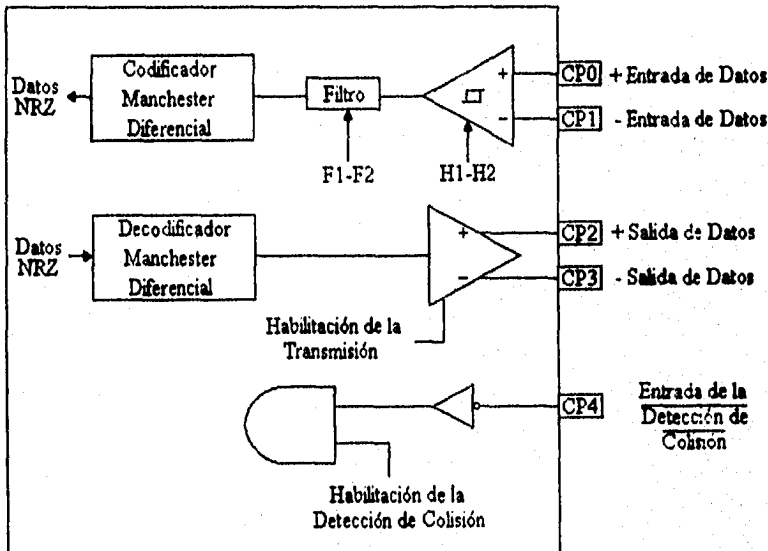


Figura 4.7 Modo Diferencial

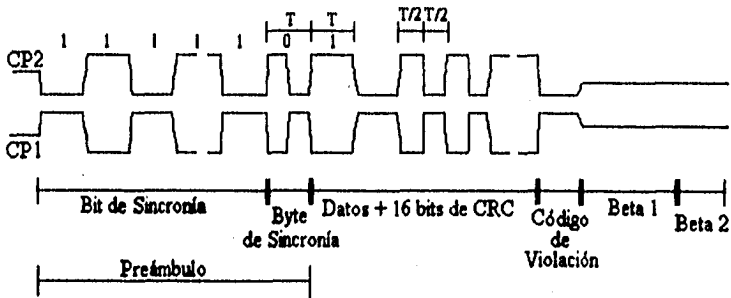


Figura 4.8 Formato de Datos del Modo Diferencial

**Tabla 4.4 Valores de Histéresis del Puerto de Comunicaciones
(expresado como voltaje diferencial pico a pico en términos de VDD)**

Histéresis	V _{th} Min	V _{th} Típico	V _{th} Max
0	0.019 V _{DD}	0.027 V _{DD}	0.035 V _{DD}
1	0.040 V _{DD}	0.054 V _{DD}	0.068 V _{DD}
2	0.061 V _{DD}	0.081 V _{DD}	0.101 V _{DD}
3	0.081 V _{DD}	0.108 V _{DD}	0.135 V _{DD}
4	0.101 V _{DD}	0.135 V _{DD}	0.169 V _{DD}
5	0.121 V _{DD}	0.162 V _{DD}	0.203 V _{DD}
6	0.142 V _{DD}	0.189 V _{DD}	0.236 V _{DD}
7	0.162 V _{DD}	0.216 V _{DD}	0.270 V _{DD}

Tabla 4.5 Valores de Filtros del Puerto de Comunicaciones

Filtro (F)	Min	Típico	Max	Unidad
0	2	6	6	ns
1	90	270	540	ns
2	200	535	690	ns
3	410	1070	1920	ns

Para que el receptor detecte el flanco de transición, dos detectores son puestos para cada periodo de bit, T . El primer detector se encuentra en $T/2$ y determina si un cero es el comienzo de la recepción. El segundo detector se encuentra en T y define un 1. Esta transición después preparará los siguientes dos detectores ($T/2$ y T). Si la transición no ocurre, una violación de código Manchester es detectada y el conjunto de datos se asume que ha terminado. La tabla 4.6 muestra el ancho del detector como una función de la entrada de reloj del Circuito NEURON (MHz) y la velocidad del bit (Mbps) de la red seleccionada. Si una transición cae fuera de lugar de cualquiera de los dos detectores, ésta no será detectada. El tiempo instantáneo de la transición, conocido como ruido, puede ser causado por cambios en el medio de comunicación, o instantáneamente en la transmisión o recepción de la entrada de reloj de los nodos. La tolerancia al ruido de los detectores está expresada como fracciones del periodo de bit, T , en la tabla 4.6.

Tabla 4.6 Tolerancia de Fluctuaciones del Receptor

Ratio del reloj de entrada del Circuito NEURON para la velocidad de bits de la red	Siguiete borde de dato			Siguiete flanco de reloj		
	Min	Nom	Max	Min	Nom	Max
8:1	0.375T	0.500T	0.622T	0.875T	1.000T	1.122T
16:1	0.313T	0.600T	0.665T	0.813T	1.000T	1.165T
32:1	0.345T	0.500T	0.717T	0.845T	1.000T	1.155T
64:1	0.330T	0.500T	0.702T	0.830T	1.000T	1.170T
128:1	0.323T	0.500T	0.695T	0.823T	1.000T	1.177T
256:1	0.313T	0.600T	0.690T	0.818T	1.000T	1.162T
512:1	0.316T	0.600T	0.687T	0.816T	1.000T	1.165T
1024:1	0.315T	0.600T	0.687T	0.815T	1.000T	1.165T

La velocidad permisible de los bits de la red para modo directo son dados en la tabla 4.3. Todos los componentes activos necesarios para implementar redes de par trenzado utilizando el modo directo están contenidos dentro del Circuito NEURON. Para una red local pequeña, una conexión simplificada de interface directa de la red, como la mostrada en la figura 4.9, puede ser empleada. Se utiliza este circuito sólo cuando todos los Circuitos NEURON sobre la red parten de la misma fuente de poder. Hasta 64 nodos pueden ser puestos en la red en modo de conexión directa a 1.25 Mbps. La resistencia de 51Ω proporciona corriente limitada cuando dos Circuitos NEURON podrían estar intentando manejar la línea para estados alternos (una colisión) y también polarizando la señal de transmisión para estar dentro del rango de voltaje en modo común del receptor. El diodo proporciona alguna protección transitoria.

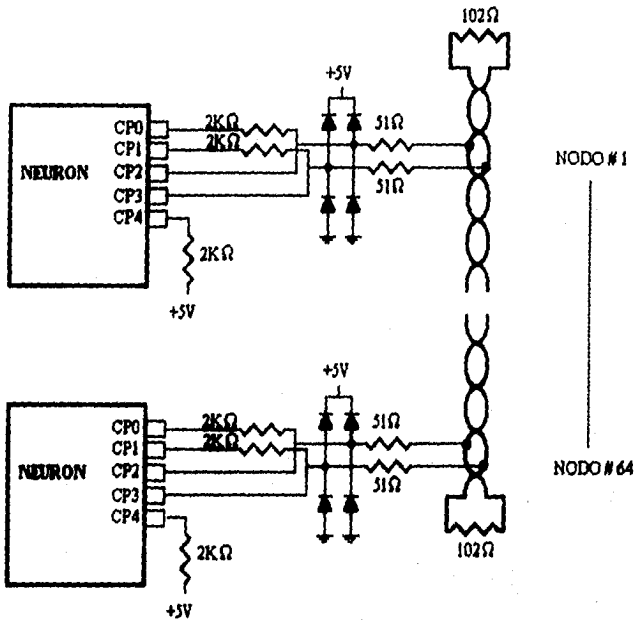


Figura 4.9 Interface Simple de Conexión Directa a la Red

El transmisor/receptor de conexión directa usa la corriente alta de salida del circuito integral de transmisor/receptor para el Circuito Integrado NEURON manejando la red directamente. El modo diferencial proporciona alguna inmunidad al ruido en modo común. Este diseño de transmisor/receptor es utilizado para aplicaciones en las cuales los nodos LONWORK están en una cercana proximidad, generalmente menor a 30 metros.

Cada terminación del cable de par trenzado debe ser concluida con una resistencia de 102Ω para igualar la impedancia característica de la línea. El no hacer esto causa reflejos, los cuales interactúan con la señal principal, resultando cancelaciones o señal falsas, teniendo una mala recepción en la velocidad más alta de los datos. Los cuatro diodos IN4148 son conectados para proteger al Circuito NEURON de pequeñas señales transitorias que podrían ser recolectadas desde el cable de par trenzado.

4.2.1.3 Modo directo diferencial (transformador acoplado)

Los transformadores acoplado del transmisor/receptor de par trenzado son diseñados para la comunicación entre varios nodos remotos, e incorpora circuitos de rechazo de modo común, detección de colisión, línea de aislamiento, protección para la energía baja, etc. Un ejemplo es mostrado en la figura 4.10 y 4.11.

Cuando se usa el transformador descrito en la tabla 4.7, la detección de colisión puede ser soportada sobre alambrados específicos y alto modo de aislamiento así como inmunidad al ruido. Otra velocidad de datos y tipos de alambrado podrían requerir diferentes transformadores. El diseñador podría desarrollar sus propios circuitos de transformador acoplado. Apropiadamente diseñado el circuito de transformador acoplado puede soportar hasta 64 nodos sobre un alambrado de par trenzado.

Tabla 4.7 Sugerecias de Fabricación del Transformador para Par Trenzado de 78 kbp y 1.25 Mbps

	78 kbps	1.25 Mbps
Parte del número	0505-0542	PE-65948
Turno del radio	2:1	1:1
Tamaño de inductancia	= 35 mH	= 3.5 mH
Compañía	Prección Components Inc.	Pulse engineering, Inc

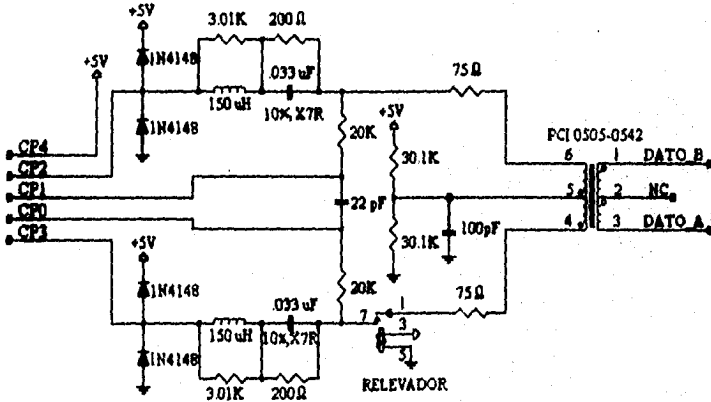


Figura 4.10 Transmisor/Receptor de Par Trenzado a 78 Kbps

4.2.1.3 Modo directo diferencial (transformador acoplado)

Los transformadores acoplado del transmisor/receptor de par trenzado son diseñados para la comunicación entre varios nodos remotos, e incorpora circuitos de rechazo de modo común, detección de colisión, línea de aislamiento, protección para la energía baja, etc. Un ejemplo es mostrado en la figura 4.10 y 4.11.

Quando se usa el transformador descrito en la tabla 4.7, la detección de colisión puede ser soportada sobre alambrados específicos y alto modo de aislamiento así como inmunidad al ruido. Otra velocidad de datos y tipos de alambrado podrían requerir diferentes transformadores. El diseñador podría desarrollar sus propios circuitos de transformador acoplado. Apropiadamente diseñado el circuito de transformador acoplado puede soportar hasta 64 nodos sobre un alambrado de par trenzado.

Tabla 4.7 Sugerencias de Fabricación del Transformador para Par Trenzado de 78 kbp y 1.25 Mbps

	78 kbps	1.25 Mbps
Parte del número	0505-0542	PE-65848
Turno del radio	2:1	1:1
Tamaño de inductancia	= 35 mH	= 3.5 mH
Compañía	Precisión Components Inc.	Pulse engineering, Inc

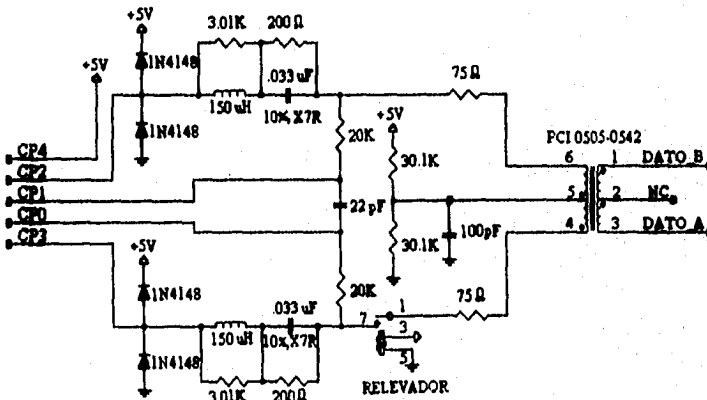


Figura 4.10 Transmisor/Receptor de Par Trenzado a 78 Kbps

4.2.1.4 Transmisor/receptor RS-485

El transmisor/receptor RS-485 soporta las especificaciones electrónicas del EIA RS-485. El transmisor/receptor presentado en la figura 4.13 debe ser puesto para operar a 39 Kbps. El RS-485 estándar especifica que el nodo es conectado directamente al cable de par trenzado dentro de un fragmento del alambrado.

Tres aisladores opcionales pueden ser usados cuando el aislamiento es requerido entre los cables de par trenzado y el transmisor/receptor: uno para la transmisión de señal, uno para la recepción de señal, y un tercero para la señal de habilitación de transmisión. La tierra local del transmisor/receptor RS-485 debe ser referida a la tierra local de los demás transmisores/receptores RS-485 operando en la misma red.

El transmisor/receptor debe ser concluido en las terminales con las terminaciones complejas mostradas en la figura 4.12, o con una resistencia de 121Ω al 1% a fin de minimizar los reflejos.

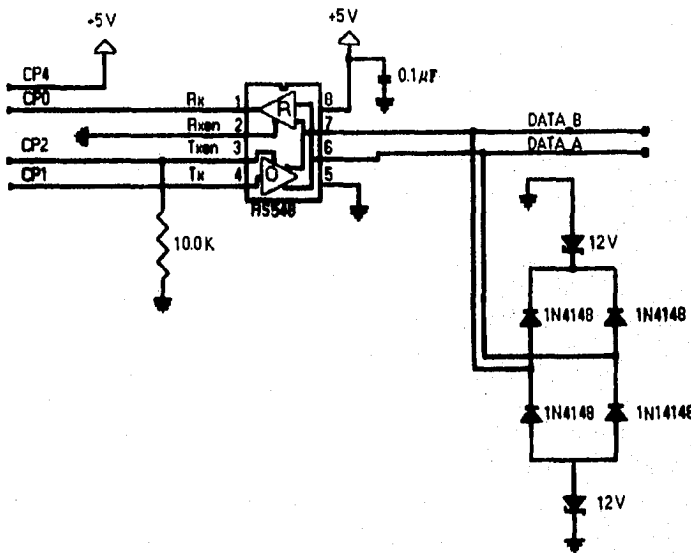


Figura 4.13 Esquema del Transmisor/Receptor de Par Trenzado RS-485

4.2.2 Modo de Propósito Especial

En situaciones especiales, el modo de propósito especial es deseable para el Circuito NEURON ya que proporciona el paquete de datos en un formato sin codificar y sin un preámbulo. En ese caso, una transmisor inteligente acepta el dato sin codificar y hace el propio formato e inserción de preámbulo. El receptor inteligente después de detectar y de despojar al dato del preámbulo y del formato, retorna el dato decodificado al Circuito NEURON, tal transmisor/receptor inteligente contiene la entrada y salida de los datos propios de la memoria intermedia (buffers) y funciones de control inteligente, y proporciona las señales del protocolo de saludo para el paso apropiado del dato anterior y posterior entre el Circuito NEURON y el transmisor/receptor, además, hay muchas características que pueden ser definidas e incorporadas dentro de una situación especial del transmisor/receptor.

- **Habilidad para configurar varios parámetros del transmisor/receptor desde el Circuito NEURON.**
- **Habilidad para reportar sobre varios parámetros del transmisor/receptor desde el Circuito NEURON.**
- **Diversas operación de canales.**
- **Diversas operación de velocidad del bit.**
- **Uso del corrector de error.**
- **Medios técnicos de modulación específica, requiriendo mensaje especiales de encabezados y construcción.**
- **Detector de colisión.**

Cuando el modo de propósito especial es usado, un protocolo es utilizado entre el Circuito NEURON y el transmisor/receptor. Este protocolo está contenido en el Circuito NEURON y el transmisor/receptor, cada uno intercambiando 16 bits, 8 bits de estado y 8 bits de datos (ver figura 4.14) simultánea y continuamente a velocidad de 1.25 Mbps (cuando la entrada de reloj del Circuito NEURON es de 10 MHz). La velocidad máxima de bits es de 156 Kbps debido al encimamiento asociado con el protocolo de saludo.

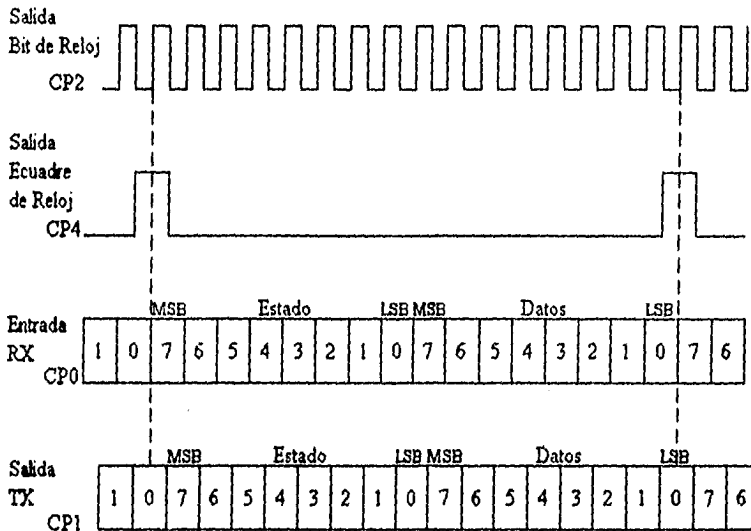


Figura 4.14 Formato de Datos del Modo de Propósito Especial

Hay tres tipos de datos que pueden ser enviados y recibidos durante cada marco.

- 1.- Paquete de dato de la red: Datos actuales (8 bits a la vez) para ser transmitido y recibido.
- 2.- Datos de configuración: Información del Circuito NEURON, el cual llama al transmisor/receptor para saber como está organizado o configurado.
- 3.- Estado del dato: Reporte de los parámetros de información desde el transmisor/receptor al Circuito NEURON cuando es solicitado por el Circuito NEURON.

Todos los Circuitos NEURON tienen alguna cantidad de memoria EEPROM interna. ésta es inicializada con la configuración de información necesaria para iniciar el asociado transmisor/receptor. Sobre la recepción de una red manejadora de mensaje, el microprograma del circuito NEURON puede leer el estado del registro dentro del transmisor/receptor e informar su contenido a una solicitud del dispositivo para la depuración de la red y el análisis.

4.2.2.1 E/S física del modo de propósito especial

Cuando el Circuito NEURON es configurado para operar en el modo de propósito especial, El Circuito NEURON y el transmisor/receptor intercambian estados y datos sobre un bus serial síncrono en las terminales CP0, CP1, CP2, CP4.

CP0 (entrada RX) es el enlace serial desde el transmisor/receptor hacia el Circuito NEURON. Durante cada periodo de encuadre (frame), ocho bits de estado y ocho bits de datos son enviados desde el transmisor/receptor hacia el Circuito NEURON.

CP1 (salida TX) es el enlace serial desde el Circuito NEURON hacia el transmisor/receptor. durante cada periodo de encuadre, ocho bits de estado y ocho bits de datos son enviados desde el Circuito NEURON hacia el transmisor/receptor.

CP2 (salida de reloj) es un bit de reloj generado por el Circuito NEURON, éste cronometra el dato y el estado de entrada y salida del Circuito NEURON y del transmisor/receptor. El flanco de bajada de la salida de reloj es usado por el Circuito NEURON para entrada de reloj RX y para salida de reloj TX. La velocidad de los bits del reloj es la misma que la del reloj del puerto de comunicación del Circuito NEURON. La velocidad del reloj del puerto de comunicación está relacionado con la velocidad de entrada del Circuito NEURON; para el modo de propósito especial, el reloj del puerto de comunicación debe ser configurado entre 8 o entre 16 veces la entrada de reloj del Circuito NEURON. Es recomendable que el reloj del puerto de comunicación sea menor a 8 veces la velocidad del dato sobre el medio de comunicación.

CP4 (salida de encuadre del reloj) es un encuadre del reloj generado por el Circuito NEURON, éste cambia a alto cada 16 bits sobre el enlace serial. El encuadre del reloj establece un periodo de 16 bits durante el cual ocho bits de estado y ocho bits de datos son intercambiados.

La quinta terminal de E/S, CP3, del puerto de comunicaciones es definida en modo "dormido". El soporte de esta función es opcional sobre el transmisor/receptor. Cuando la terminal es conectada a un transmisor/receptor con capacidad de modo "dormido", el Circuito NEURON puede ser configurado para.

1) El Circuito NEURON puede ser configurado de modo que algún mensaje entrante lo "despierte". En este caso, CP3 es una entrada para el Circuito NEURON.

2) El programa de la aplicación corriente sobre el Circuito NEURON puede decidir el modo "dormido" periódicamente declarando la terminal en bajo. En este

caso, la terminal es una salida. Cuando la terminal del modo "dormido" está en alto, el transmisor/receptor asume al Circuito NEURON "despierto" y cuando la terminal se encuentra en bajo, el transmisor/receptor asume que está "dormido".

El transmisor/receptor también tiene una entrada de reinicio que es enlazada a la entrada de reinicio del Circuito NEURON. El transmisor/receptor reinicia sus estados y registros internos cuando esta entrada es activada. El transmisor/receptor no debe causar que el Circuito NEURON se reinicie por declaración de esta línea.

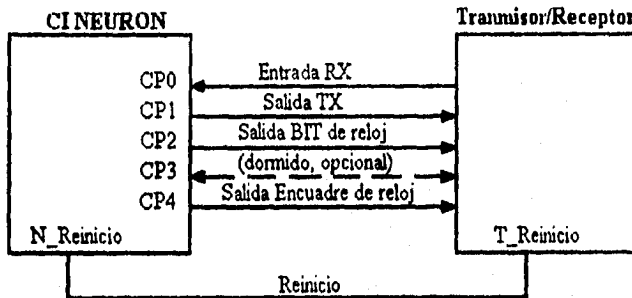


Figura 4.15 Puerto de Comunicaciones de E/S

4.2.2.2 Estado de los bits

El contenido de la configuración de datos y el estado del dato son definidos por el transmisor/receptor. El estado de las banderas son listadas en la tabla 4.8. Si se determina que el modo de propósito especial es necesario y práctico para el diseño del transmisor/receptor, entonces el Banco de Desarrollo LONBUILDER puede ser usado para probar este modo de operación con el transmisor/receptor de diseño propio.

Tabla 4.8 Estado de Transmisión y Recepción de los Bits en el Modo de Propósito especial

Salida TX, Bits de estado		
Bit 7	TX_FLAG	Circuito NEURON en el proceso de transmisión de paquetes
Bit 6	TX_REQ_FLAG	Circuito NEURON solicitando transmitir sobre la red
Bit 5	TX_DATA_VALID	Circuito NEURON está aprobando los datos de la red para el transmisor/receptor en este marco
Bit 4	No importa	Inutilizado
Bit 3	TX_ADDR_R/W	Si es negado, Circuito NEURON está escribiendo el registro interno del transmisor/receptor
Bit 2	TX_ADDR_2	Dirección del bit 2 del registro interno del transmisor/receptor (1..7)
Bit 1	TX_ADDR_1	Dirección del bit 1 del registro interno del transmisor/receptor (1..7)
Bit 0	TX_ADDR_0	Dirección del bit 0 del registro interno del transmisor/receptor (1..7)
Entrada RX, Bits de estado		
Bit 7	SET_TX_FLAG	Transmisor/receptor aceptando la solicitud para transmitir paquetes
Bit 6	CLR_TX_REQ_FLAG	Transmisor/receptor reconociendo la solicitud para transmitir paquetes
Bit 5	RX_DATA_VALID	Transmisor/receptor está aprobando los datos de la red para el Circuito NEURON en este marco
Bit 4	TX_DATA_CST	Transmisor/receptor indica que el Circuito NEURON es borrado para enviar byte de los datos de la red
Bit 3	SET-COLL_DET	Transmisor/receptor ha detectado una colisión mientras transmite el preámbulo
Bit 2	RX_FLAG	Transmisor/receptor ha detectado un paquete sobre la red
Bit 1	RD/WR_ACK	Transmisor/receptor reconociendo lectura/escritura para el registro interno
Bit 0	TX_ON	Transmisor/receptor está transmitiendo sobre la red

Mientras que el modo de propósito especial ofrece características propias, este modo supone que la mayoría de los transmisores/receptores usaran el modo de terminación simple para mayor número de medios de comunicación, desde cables coaxiales, radio frecuencia hasta fibra óptica, puesto que éste ofrece el código Manchester Diferencial, el cual toma el cargo de recuperar el reloj.

4.3 TRANSMISORES/RECEPTORES LONBUILDER

Las tarjetas procesador requieren una tarjeta de expansión transmisor/receptor para proporcionar la interface física para una canal de comunicación. Los transmisores/receptores LONBUILDER son útiles para redes de par trenzado y radio frecuencias.

Todas las tarjetas procesador incluyen un conector para el transmisor/receptor que puede ser usado para instalar la tarjeta transmisor/receptor LONBUILDER sobre el conector, dentro de una estación de desarrollo. La configuración de la velocidad de bits es 4.8 Kbps, 9.8 Kbps, 19.5 Kbps, 39.1 Kbps, 156.3 Kbps, 312.5 Kbps, 625 Kbps, y 1.25 Mbps. La detección de colisión requiere un transmisor/receptor LONBUILDER de par trenzado.

4.3.1 Transmisor/Receptor LONBUILDER de Par Trenzado

El transmisor/receptor LONBUILDER de par trenzado proporciona la interface física para una red de par trenzado LONWORK TP/XF-78 o TP/XF-1250, habilitando el nivel del sistema de ejecución sobre un medio de par trenzado. Este transmisor/receptor incluye dos diferentes circuitos transmisores/receptores que pueden ser configurados para operar en dos velocidades de bit: 78 Kbps y 1.25 Mbps. Ambos transmisores/receptores incluyen circuito detector de colisión que puede ser usado para evaluar el impacto de la detección de colisión sobre la red an funcionamiento.

Note que el número de nodos soportados en el medio ambiente de desarrollo LONBUILDER con el transmisor/receptor de par trenzado es un poco menor que las especificaciones de la capacidad máxima de interface del LONWORK de par trenzado

El transmisor/receptor LONBUILDER de par trenzado puede ser usado para comunicarse con nodos de diseño propio que usen los módulos de control LONWORK TP/XF-78 o TP/XF-1250. Este también puede ser usado para comunicación con el transmisor/receptor de transformador acoplado. El transmisor/receptor LONBUILDER de par trenzado no puede ser usado para comunicación con nodos que usen un módulo de control LONWORK TP-RS585 o una conexión directa o transmisor/receptor RS-485.

4.3.2 Transmisor/Receptor LONBUILDER TP-RS485

El transmisor/receptor LONBUILDER TP-RS485 es un multifuncional transmisor/receptor diseñado para usarse con alguna tarjeta procesador LONBUILDER. El transmisor/receptor LONBUILDER TP-RS485 proporciona la

Interface física para una red de par trenzado LONWORK TP-RS485 basado en el EIA RS-485 estándar. El transmisor/receptor TP-RS485 soporta velocidades de bits estándar LONWORK: 4.8 Kbps, 9.8 Kbps, 19.5 Kbps, 39.1 Kbps, 78.1 Kbps, 156.3 Kbps, 312.5 Kbps, 625 Kbps, y 1.25 Mbps. La velocidad de bits recomendada es de 39.1 Kbps. En 39.1 Kbps, el largo del bus es de 1,200 m.

El transmisor/receptor LONBUILDER TP-RS485 puede ser usado para comunicación con nodos de diseño propio que usen el módulo de control LONWORK TP-RS485. Este puede también ser usado para comunicación con un nodo usando un transmisor/receptor EIA RS-485. El transmisor/receptor LONBUILDER TP-RS485 no puede ser usado para comunicación con nodos que usen un transmisor/receptor TP/XF-78 o TP/XF-1250 de par trenzado.

4.3.3 Transmisor/receptor LONBUILDER RF

El transmisor/receptor LONBUILDER RF proporciona la interface física para una red de radio frecuencia. Generalmente el rango es de 15 a 50 metros en lugares cerrados y hasta 170 metros en lugares abiertos. Este transmisor/receptor proporciona una velocidad de bits de 4,822 Kbps usando una portadora de 49 MHz. El transmisor/receptor LONBUILDER puede ser usado para comunicación con nodos de diseño propio usando el transmisor/receptor LONWORK RF-10.

CAPÍTULO V

DISEÑO DE LA RED USANDO LONBUILDER

5.1 DESARROLLO DE LA RED LONWORK

5.1.1 Definición del Proyecto

El objetivo del proyecto es el monitoreo de los parámetros de funcionamiento de motores eléctricos (factor de potencia, voltaje y corriente). Este objetivo se lleva a cabo con la ayuda de microprocesadores HC16 y el diseño de una red LONWORK.

Cada motor eléctrico cuenta con un microprocesador, el cual sensa los parámetros de su funcionamiento. Estos datos obtenidos por el microprocesador son almacenados en su memoria.

El proyecto sólo se enfoca a la transmisión de estos datos (obtenidos y almacenados ya por el microprocesador HC16) sobre una red LONWORK, es decir, que se parte del hecho de que el microprocesador ya ha obtenidos los datos y están éstos almacenados en su memoria. Con lo cual solamente nos enfocamos al desarrollo de la red LONWORK, y a la interface entre los microprocesadores HC16 y los nodos de la red LONWORK.

La figura siguiente muestra como se encuentra conectada la red LONWORK y la conexión con los microprocesadores HC16.

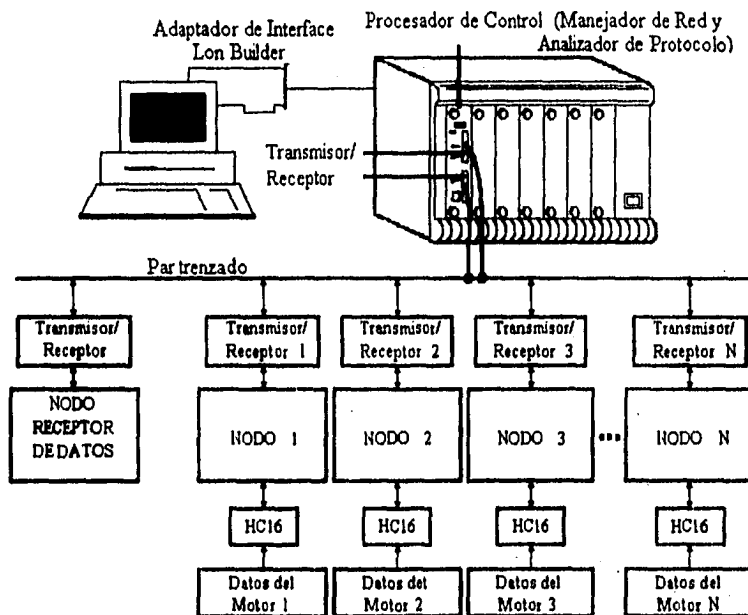


Figura 5.1 Conexión de la Red LONWORK y los Microprocesadores HC16

5.1.1.2 Transmisión de los datos del HC16 al nodo LONWORK

Los datos que se encuentran disponibles en la memoria del microprocesador HC16 son enviados al nodo de la red LONWORK a través de una interface entre el microprocesador HC16 y el nodo.

Para la realización de esta interface se cuenta con el desarrollo de dos programas; un programa para el microprocesador HC16 y otro para el nodo LONWORK (la interface será explicada con detalle en el capítulo siguiente).

5.1.1.3 Red LONWORK

Cada HC16 estará conectado a un nodo LONWORK, y los datos de los parámetros del motor, que se encuentran en la memoria del microprocesador HC16, serán transmitidos de cada uno de los microprocesadores a cada nodo LONWORK conectado a él.

Los nodos al obtener los datos los envía (a través de la red LONWORK) hacia un nodo receptor de datos, como se muestra en la figura 5.2. El nodo receptor de datos se encarga de almacenar los datos de todos los nodos conectados a la red.

5.1.1.4 Visualización de los datos

La visualización de los datos se hace a través de una PC, por medio de la interface de la PC con el nodo manejador de red (este nodo está incluido en el banco de desarrollo LONBUILDER) como se muestra en la figura 5.1.

El nodo manajador de red se encarga de manejar, controlar y monitorear la red LONWORK.

Es importante distinguir entre un manejo da red, un control de red y un monitoreo de red. El manejo de la red simplemente es la capacidad de instalar y configurar los nodos de la red. Una herramienta manejadora de red no participa en el intercambio de mensajes de aplicación y mensajes de las variables de la red, y no es necesario que esté presente para que la red pueda operar, el control da la red LONWORK se encarga de coordina el sentir y al control del proceso del sistema, y el monitoreo de la red LONWORK recibe mensajes de aplicación o variables de red actualizadas de los demás nodos sobre la red.

5.1.2 Desarrollo del Proyecto

5.1.2.1 Identificación de los nodos y asignación de su función

Como ya se ha indicado anteriormente se tienen dentro de la red dos tipos diferentes de nodos. Unos son los que se encargan de llevar la información desde el microprocesador hacia un nodo receptor de datos; y el otro, el nodo receptor de datos, que constituye un tipo de nodo diferente a la de los anteriores, ya que se encarga de almacenar los datos transmitidos por los demás nodos, a través de la red LONWORK.

Al tener dos tipos diferentes de nodos, lógicamente se tienen dos programas da la aplicación diferentes (una para cada tipo de nodo).

El programa del nodo receptor de datos se encargará, primero, de recibir los datos que transmiten todos los demás nodo, a través de la red LONWORK; y después, almacenarlos.

El otro programa contendrá la interface del microprocesador HC16 con el nodo (con el cual el nodo recibirá los datos del microprocesador), y la transmisión de estos datos hacia el nodo receptor de datos, a través de la red LONWORK.

Se utiliza también un nodo manejador de red, que se encarga de monitorear los datos transferidos y de visualizarlos en la PC, por medio de una interface. Este nodo, está contenido dentro de la estación de desarrollo LONBUILDER.

5.1.2.2 Definición de la interface interna de cada nodo

Las variables de la red de un nodo son sólo parte del nodo que son visibles para otros nodos. Definiendo los nodos en términos de su interface externa les permite ser desarrollados independientemente, y minimizar el impacto de la red o cambios de aplicación. La figura siguiente muestra las variables de la red para los nodos transmisores y las variables para el nodo receptor de datos.

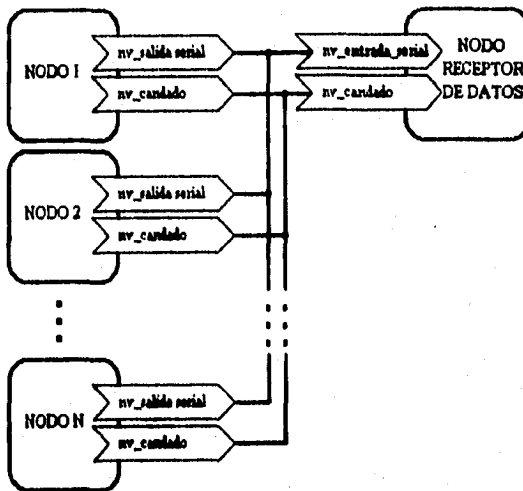


Figura 5.2 Interface Interna de los Nodos

En la figura 5.2 también se muestra el flujo de información, que debe seguir cada nodo.

5.2.3 Los programas de la aplicación de cada nodo

Los programas de las aplicaciones están escritos con lenguaje NEURON C.

En las figuras 5.3 y 5.4 se muestran los diagramas de flujo de los programas de la aplicación y a continuación los programas de la aplicación.

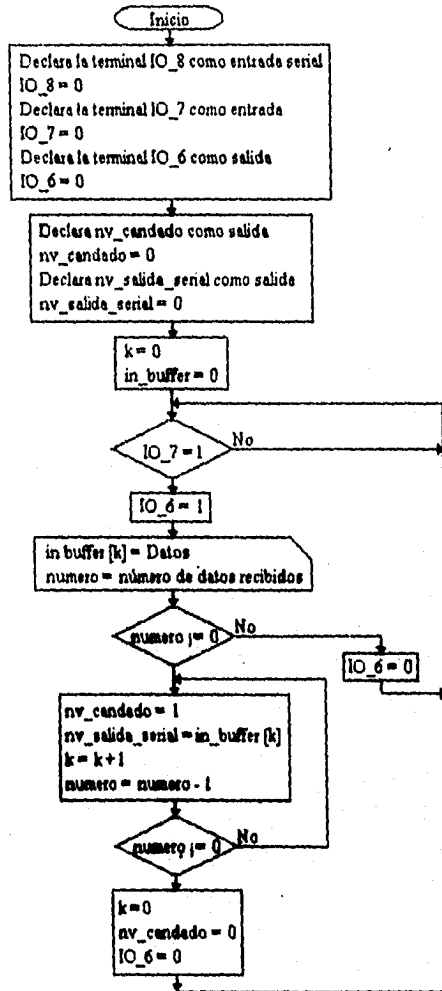


Figura 5.3 Diagrama de Flujo del Programa de los Nodos Transmisores

```
// Programa: Nodo transmisor
// Propósito: Programa para una interface entre el
//           Circuito NEURON y el HC16.
//           Interface en serie asincrona.

#include <io_types.h>
#include <snvt_lev.h>

// Declaración de las variables de la red
network output SNVT_char_ascii nv_salida_serial;
network output SNVT_lev_disc nv_candado;
network output SNVT_char_ascii nv_num_char;

// Declaración de objetos de Entrada/Salida
IO_8 input serial baud (2400) io_inter_serial;
IO_7 input bit io_inicio;
IO_6 output bit io_proto_saludo = 0;
IO_0 output bit io_led = 0;

//Declaración del tiempo objeto
timer countdown;

char in_buffer[5];
unsigned int num_chars;
unsigned int numero;
int k=0;
int i;
int l;

when (reset)
(
    io_change_init(io_inicio);
)

when (io_changes(io_inicio) to 1)
(
    io_out (io_proto_saludo , 1);
    num_chars = io_in(io_inter_serial, &in_buffer, 5);
    io_out(io_led , 1);
    countdown = 5;
    numero = num_chars;
)
```

```
when (numero != 0)
{
    nv_candado = ST_ON;
    nv_salida_serial = in_buffer[k];
    for (l=0;l<99;l++)
    {
        for(l=0;l<9;l++)
        {
        }
    }
    k=k+1;
    numero = numero - 1;
    if (numero == 0)
    {
        k=0;
    }
}

when (timer_expires(cooldown))
{
    io_out (io_proto_saludo,0);
    io_out (io_led,0);
}
```

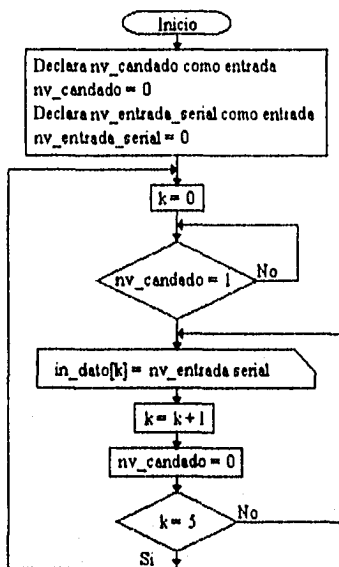


Figura 5.4 Diagrama de Flujo del Programa del Nodo Receptor

//Programa: Nodo Receptor de datos
//Propósito: Recoge los datos de los nodos transmisor

```
#include <io_types.h>  
#include <snvt_lev.h>
```

```
//Declaración de las variables de la red  
network input SNVT_char_ascii nv_entra_serial;  
network input SNVT_lev_disc nv_candado = ST_OFF;  
network input SNVT_char_ascii nv_num_char;
```

```
unsigned int in_dato[5];  
int k=0;  
unsigned int num_char;  
unsigned numero;
```

```
when (nv_candado == ST_ON)  
{  
    in_dato[k] = nv_entra_serial;  
    k=k+1;  
    nv_candado = ST_OFF;  
    if (k==5)  
    {  
        k=0;  
    }  
}
```

5.1.2.4 Depuración y prueba individual de cada nodo

Durante el paso de depuración se usó el banco de desarrollo LONBUILDER para ejecutar estas tareas en cada nodo de aplicación:

1. Se instaló y configuró un Emulador NEURON LONBUILDER en una estación de desarrollo.
2. Se compiló y ligó el código de la aplicación para el nodo, y se cargó el programa de la aplicación sobre el emulador instalado y configurado.
3. Se depuró el programa de aplicación de los nodos, corriéndolo sobre el emulador usando el Depurador NEURON C.

5.1.2.5 Instalación de los nodos sobre la red

Durante este paso los nodos se instalaron en la red como se muestra en la figura 5.5 quedando la configuración definitiva de la red.

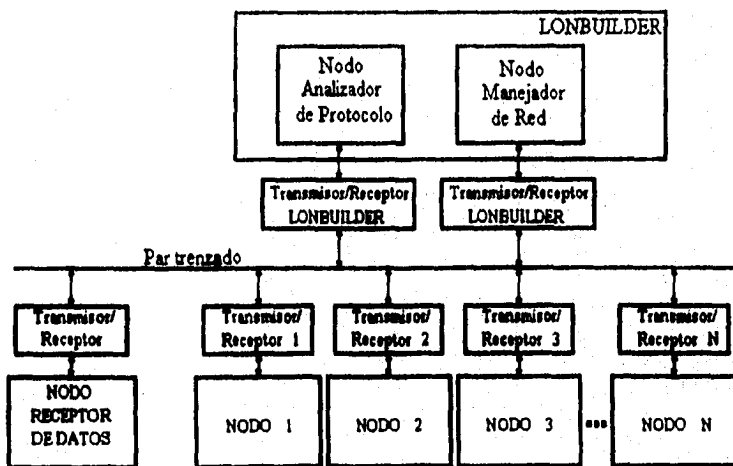


Figura 5.5 Red LONWORK

5.1.3 Transmisor/Receptor utilizado

Para la comunicación de los datos a través de la red LONWORK se utilizó como medio de comunicación par trenzado, usando en el Circuito Neuron una configuración directa para un transmisor/receptor de diseño propio.

La circuitería de la red utilizando el transmisor/receptor se muestra en la figura 5.6.

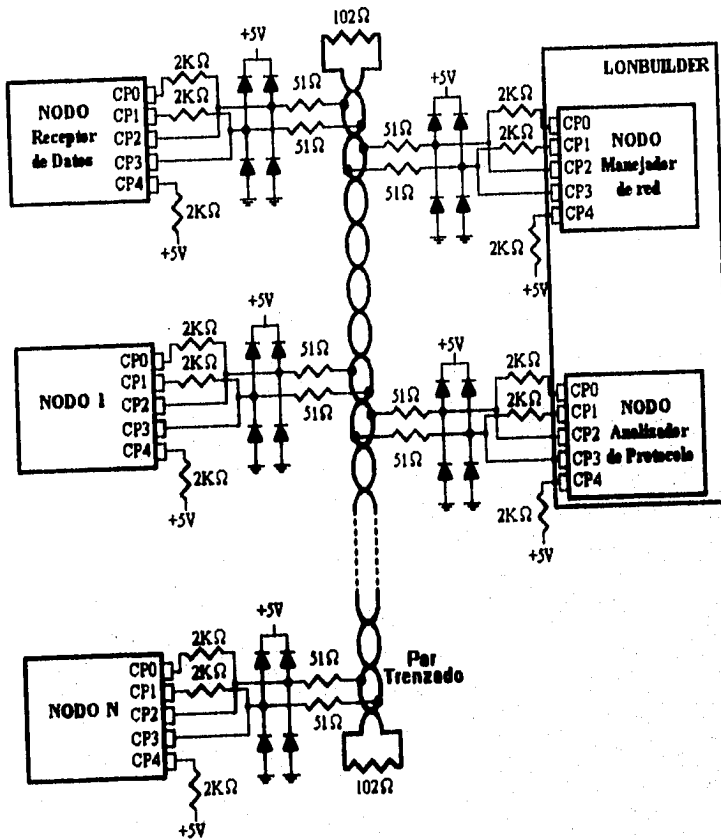


Figura 5.6 Red LONWORK del Proyecto

Se utilizó esta configuración de par trenzado debido a:

- A que es fácil de implementar.
- La distancia corta que existe entre los motores.
- El número de nodos es inferior a 64 (número máximo que soporta este tipo de configuración).
- El bajo costo de la implementación del transmisor/receptor.
- Fácil mantenimiento.

CAPÍTULO VI

INTERFACE ENTRE Circuito Integrado NEURON Y MCU'S

6.1 INTRODUCCIÓN

En el capítulo anterior se definió el sistema definitivo de la red LONWORK, únicamente nos resta definir la interface entre el Circuito Integrado NEURON y el microprocesador HC16.

Para la realización de la interface entre un MCU (Microcontrolador) y un Circuito Integrado NEURON se cuentan con diversas configuraciones. Estas configuraciones son expuestas a continuación.

6.1.1 Interface en Paralelo

6.1.1.1 Modo maestro/esclavo A

Este modo es recomendado cuando se hace interface entre dos Circuitos Integrados NEURON. En el modo maestro/esclavo A, el maestro maneja la terminal IO8 como selección del dispositivo y la terminal IO9 para especificar el ciclo de lectura o escritura, y el esclavo maneja la terminal IO10 como un protocolo de saludo de reconocimiento (ver figura 6.1). La velocidad de la transferencia máxima de los datos es un byte por cuatro ciclos de instrucción del procesador, o 2.4 μ s por byte con la velocidad de entrada de reloj máxima (10 MHz). La velocidad de la transferencia de los datos tiene una escala proporcional a la velocidad del reloj de entrada. Después de que todos los bytes son leídos o escritos, la línea del protocolo de saludo es monitoreada por el maestro para verificar si el esclavo ha completado el proceso (cuando HS = 0) y el esclavo está listo para la siguiente transferencia de bytes. Esto se realiza automáticamente

dentro de la transferencia de datos de un Circuito Integrado NEURON al otro Circuito Integrado NEURON (maestro/esclavo A).

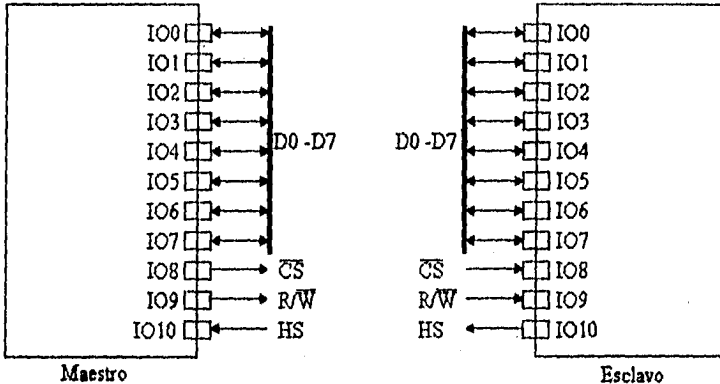


Figura 6.1 Entrada/Salida en Paralelo - Maestro / Esclavo A

6.1.1.2 Modo maestro/esclavo B

El modo maestro/esclavo B es recomendado para hacer interface entre un Circuito Integrado NEURON (como esclavo) y un procesador diferente al Circuito Integrado NEURON (como maestro). Cuando se configura en modo esclavo B, el Circuito Integrado NEURON acepta a la terminal IO8 como un seleccionador del dispositivo, y la terminal IO9 para especificar si el maestro leerá o escribirá, acepta también a la terminal IO10 como un selector de entrada. Cuando CS (selector del dispositivo) es habilitado y la terminal IO10 está en bajo o alto, y R/W está en bajo, las terminales IO0 hasta IO7 forman el bus de datos bidireccional. Cuando la terminal IO10 está en alto, R/W está en alto, y CS es habilitado, la terminal IO0 es manejada como la señal del protocolo de saludo de reconocimiento para el maestro. El Circuito Integrado NEURON puede aparecer como dos registros, en el espacio de direcciones del maestro, uno de los registros será el registro de datos de lectura/escritura, y el otro será el registro de control de sólo lectura. Por lo tanto, la lectura que realiza el maestro de una dirección es accesada por el registro de control del reconocimiento del protocolo de saludo y las restantes lectura o escritura son accesadas por el registro de datos para la transferencia de entrada/salida. El bit menos significativo del registro de control, el cual es leído a través de la terminal IO0, es el bit del protocolo de salud (HS). El maestro lee el bit del protocolo de salud después de toda la lectura o escritura

del maestro. En la interface del Circuito Integrado NEURON hacia el microprocesador que no es Circuito NEURON, el Circuito NEURON en el modo esclavo B manipula automáticamente el protocolo de saludo y el paso de testigo (token passing). Sin embargo, un maestro que no sea un Circuito NEURON tiene que leer el bit de protocolo de saludo después de cada operación y tiene también que rastrear internamente el paso de testigo. Este modo está diseñado para usarse con un procesador maestro que use un mapa de memoria de entrada/salida, por esto el bit menos significativo del bus de direcciones del maestro es generalmente conectado a la terminal IO10 del Circuito NEURON. Esto es ilustrado en la figura 6.2.

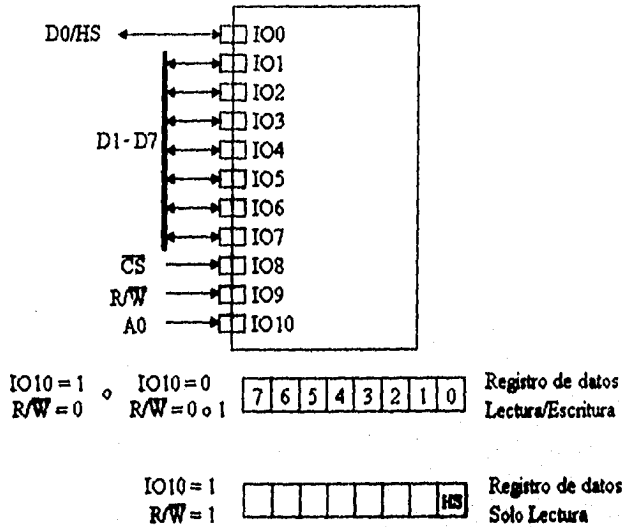


Figura 6.2 Entrada/Salida en Paralelo - Maestro / Esclavo B

6.1.1.3 Paso de testigo (token passing)

El paso de testigo está implementado para eliminar la posibilidad de contención del bus de datos. El testigo es poseído en el maestro después de la sincronización y es pasado entre los nodos maestro y esclavo. El paso de testigo es ejecutado automáticamente en una interface de Circuito Integrado NEURON a Circuito Integrado NEURON.

6.1.1.4 Protocolo de saludo

El protocolo de saludo le permite al maestro monitorear al esclavo en toda la transferencia del byte, garantizando que ambos procesadores están listos para transferir el siguiente byte. Si el maestro reconoce la prueba, el maestro esperará a el protocolo de saludo del esclavo antes de escribir el dato en el bus. Si el esclavo reconoce la prueba, el maestro monitorea el flanco de bajada del protocolo de saludo antes de escribir en el bus.

6.1.2 Bit de Corrimiento de Entrada/Salida

Se pueden configurar como líneas de entrada o salida pares de terminales adyacentes, la terminal con numeración inferior será usada para el reloj (manejado por el Circuito Integrado NEURON) y la terminal con numeración mayor será usada para los 16 bits de dato serial. La velocidad del dato puede ser configurado como 1, 10, o 15 K bps con la velocidad del reloj máxima de entrada (10 Mhz). La velocidad de entrada es proporcional a la velocidad de entrada del reloj. La activación del flanco del reloj puede ser especificada como flanco de bajada o de subida. Esta función es usada para transferencia de datos de lógica externa empleando registros de corrimiento. Esta función suspende el proceso de la aplicación hasta que la operación es completada. Ver figura 6.3.

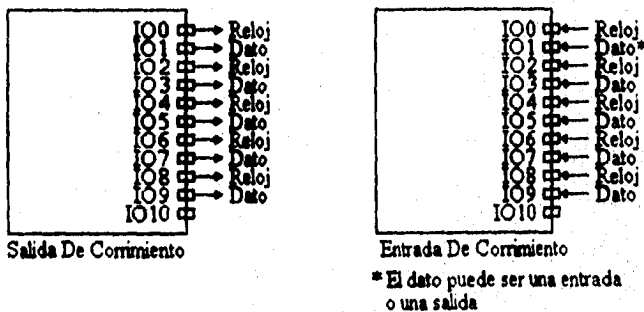


Figura 6.3 Bit de Corrimiento de Entrada/Salida

6.1.3 Función NeuroAlambrada de Entrada/Salida (interface SPI)

6.1.3.1 Modo maestro NeuroAlambrado

En el modo maestro NeuroAlambrado, la terminal IO8 es el reloj (manejador para el Circuito Integrado NEURON), la terminal IO9 es el dato serial de salida, y la terminal IO10 es el dato serial de entrada. El dato serial con ayuda del reloj es sacado por la terminal IO9 al mismo tiempo que el dato con la ayuda del reloj es introducido por la terminal IO10. El dato es enviado con el flanco de subida de la señal de reloj. Además, uno o más de las terminales IO0 hasta IO7 pueden ser usados como un selector del dispositivo, permitiendo a varios dispositivos NeuroAlambrados ser conectados sobre un bus de tres alambres. La velocidad del reloj puede ser especificada como 1, 10 o 20 Kbps a una velocidad de reloj de entrada de 10 MHz; Estas escalas están en proporción con el reloj de entrada.

6.1.3.2 Modo esclavo NeuroAlambrado

En un modo esclavo NeuroAlambrado, la terminal IO8 es el reloj (manejador para el maestro externo), la terminal IO9 es el dato serial de salida, y la terminal IO10 es el dato serial de entrada. El dato serial es con la ayuda del reloj sacado por la terminal IO9 al mismo tiempo que el dato con la ayuda del reloj es introducido por la terminal IO10. El dato es enviado con el flanco de subida de la señal de reloj, el cual puede ser hasta de 18 Kbps. Una de las terminales IO0 hasta IO7 puede ser designada como una terminal de tiempo fuera. Un "uno" de nivel lógico sobre la terminal de tiempo fuera causa que la operación de entrada/salida del NeuroAlambrado esclavo termine antes de que el número especificado de bits haya sido transferido.

6.1.4 Interface Serial

La terminal IO8 puede ser configurada como una línea de entrada serial asíncrona, y la terminal IO10 puede ser configurada como una línea de salida serial asíncrona. La velocidad del bit para la entrada y para la salida puede ser especificada independientemente, siendo de 600, 1200, 2400 o 4800 bps con la velocidad de reloj de entrada máxima (10 MHz). Las escalas de la velocidad de datos son proporcional a la velocidad del reloj de entrada. El formato es fijado en un bit de inicio, ocho bits de datos y un bit de paro, y hasta 255 bytes pueden ser transferidos en un tiempo. La operación de entrada serial o de salida serial (pero no ambos) pueden estar sin efecto en algún tiempo. La interface es semiduplex. Esta función suspende el proceso de aplicación hasta que la operación es completada. Esta función es útil para transferencia de datos de dispositivos

seriales tal como terminales, modem e interface con computadoras en forma serial.

6.2 ELECCIÓN DE LA INTERFACE DE LA RED LONWORK

Después de haber explicado las diferentes configuraciones de la interface entre el Circuito Integrado NEURON y un MCU, podemos elegir la mejor opción de la interface para nuestra red en particular.

Para la interface entre los microprocesadores HC16 y el Circuito Integrado NEURON (nodos de la red LONWORK), se utilizó una interface serial asíncrona de 2400 bps, y el formato se fijó en un bit de inicio, ocho bits de datos y un bit de paro.

Para lograr realizar la transmisión de datos desde el microcontrolador hasta el Circuito NEURON se envían los datos como se muestra en la figura 6.4. El programa de aplicación del Circuito NEURON requiere la directiva IO_in() para esperara la transmisión de datos serial. El tiempo que el circuito NEURON espera para la entrada de datos después de haber declarado la directiva, y el tiempo de regreso de esta función son mostrados en la tabla 6.1.

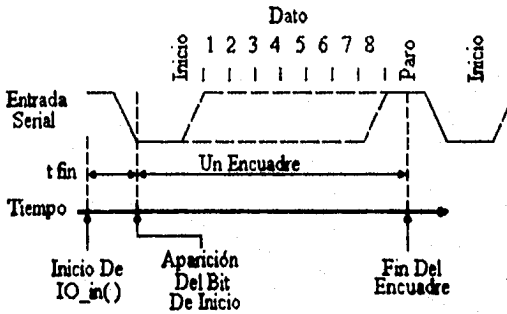


Figura 6.4 Transmisión Serial

Tabla 6.1 Tiempos de la función serial

Símbolo	Descripción	typ 10 MHz
t _{fin}	llamada a función para una muestra de salida Min (primera muestra) Max (tiempo fuera)	67 µs 20 byte de encuadre
t _{ret}	llamada a función	10 µs

Se escogió este tipo de interface debido a que en la transmisión asincrónica no se requiere de un reloj de sincronía, y por tanto el Circuito Integrado NEURON podría recibir la transmisión de datos en cualquier momento, sin que se tenga que sincronizar algún reloj. Además, que solamente se requiere de una línea de transmisión para los datos. También, dentro del programa de la aplicación del Circuito Integrado NEURON sólo se necesita una directiva para habilitar la transmisión serial asincrónica.

Para esta interface se realizó un programa de la aplicación para el Circuito Integrado NEURON y un programa para el microprocesador HC16. El programa del microprocesador HC16 se encarga de enviar los datos que se encuentran almacenados en su memoria a través de la interface asincrónica, mientras que el programa de la aplicación del Circuito Integrado NEURON se encarga de recibir los datos que llegan a través de la interface para después transmitirlos sobre la red LONWORK. La figura 6.5 muestra la interface entre el microprocesador HC16 y el Circuito Integrado NEURON.

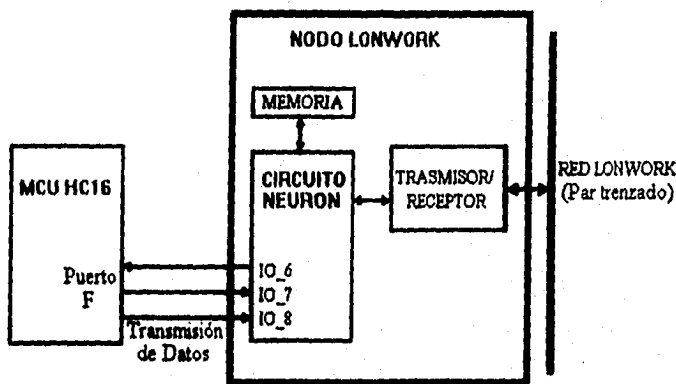


Figura 6.5 Interface entre el Microprocesador HC16 y el Circuito NEURON

El programa de la aplicación del Circuito Integrado NEURON se mostró en el capítulo anterior, en él se encuentra la programación referente a la interface con el microprocesador HC16 y la transmisión hacia la red LONWORK. El diagrama de flujo del programa del microprocesador HC16 realizado para la interface se muestra en la figura 6.6, y a continuación se muestra el programa.

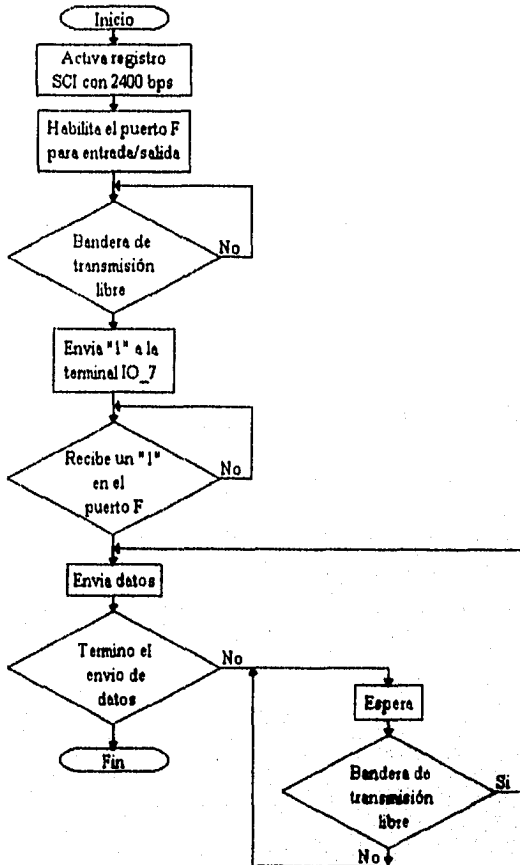


Figura 6.6 Diagrama de Flujo del Programa del HC16

```
$INCLUDE 'EQUATES.ASM'
      ORG $200
$INCLUDE 'INITSYS.ASM'
$INCLUDE 'INITRAM.ASM'

      LDD #$00DA
      STD SCCR0 ;sel the SCI baud rate to 2400 baud

      LDD #$000C
      STD SCCR1 ;enable the SCI receiver and transmitter

      LDAB #$0F
      TBXK

      LDX #$FA1F
      LDAA #$00
      STAA $00,X

      LDX #$FA1D
      LDAA #$8F
      STAA $00,X

      LDX #$FA1B
      LDAA #$00
      STAA $00,X

      LDY #$0000
RET    AIY #$01
      CPY #$FFFF
      BNE RET

      LDAB #$01
      TBYK
      LDY #$0000

ET4   LDX #$FC0C
      LDD $00,X
      BITA #$01
      BEQ ET4

      LDX #$FA1B
      LDAA #$8F
      STAA $00,X
```

```
ET5 LDX #FA1B
    LDAA $00,X
    BITA #40
    BEQ ET5
    JMP ET6

ET2 LDY #0000
ET1 LDX #FC0C
    LDD $00,X
    BITA #01
    BEQ ET1

ET6 LDAB #FF
ET3 DECB
    CMPB #00
    BNE ET3

    LDX #FC0F
    LDAA $00,Y
    STAA $00,X

    AIY #01
    CPY #0009
    BNE ET1

    LDX #FA1B
    LDAA #00
    STAA $00,X

    JMP ET2
FINAL NOP
```

CONCLUSIONES

En la realización del proyecto se llegaron a las siguientes conclusiones:

- El control distribuido de las redes LONWORK permite que cualquier nodo pueda ser desconectado y vuelto a conectado a la red sin que esta operación modifique al programa de aplicación de los demás nodos conectados a la misma red (los nodos son definidos individualmente debido a que las variables de la red que contiene cada nodo son independientes de nodo a nodo, permitiendo así, desconectar y volver a conectar a los nodos sobre la red). Además en el control distribuido la red LONWORK no necesita de un nodo de control para manejar la red del sistema.
- La programación del Circuito Integrado NEURON es relativamente sencilla y rápida comparado con la programación de otro tipo de microcontroladores, gracias al lenguaje de programación NEURON C. Con este lenguaje nos ahorramos tratos con la programación de bajo nivel, que se utiliza generalmente para la programación de microcontroladores.
- Al trabajar con el sistema LONBUILDER nos dimos cuenta que facilita el desarrollo independiente de los nodos conectados a la red, gracias a la emulación de los nodos dentro del sistema, además con la ayuda del Ambiente de Desarrollo Integrado que contiene se puede depurar el programa y cargarlo al nodo correspondiente sobre la red.

- Las redes LONWORK permiten utilizar medios de comunicación como par trenzado, radio frecuencia, líneas de poder y otros medios de comunicación dependiendo de las necesidades del sistema. Sin embargo, se utilizó el par trenzado como medio de comunicación para los Circuitos Integrados NEURON debido a su fácil implementación, a la sencilla configuración que tiene, a su fácil mantenimiento, y a su bajo costo.
- Con las once terminales con que cuenta el Circuito Integrado NEURON se realizó la interface con los microprocesadores HC16. Pero, con este puerto de once terminales se puede hacer la interface con cualquier otro tipo de microcontrolador dependiendo de las necesidades que requiera el sistema.

Se observó que es difícil el trabajar con tecnología nueva, debido a que generalmente los equipos son caros y es difícil conseguir recursos económicos para la adquisición de éstos, además la información acerca de éstos no es sencilla de conseguir.

Para el desarrollo del proyecto, se necesitaron algunos componentes que no fue posible comprarlos, a consecuencia de las limitaciones presupuestales en la UNAM que afectaron a la Unidad de Diseño Electrónico, debido a la crisis económica que sufrió el país.

Sin embargo, podemos llegar a la conclusión de que es importante el trabajar con nuevas tecnologías y el estar actualizado en todas las áreas de la ingeniería para que así el país pueda salir adelante en estos campos, y por ende pueda ser más competitivo a nivel mundial.

La experiencia que nos dejó este trabajo serán de gran ayuda para el futuro, ya que nos sirvió de complemento con los estudios realizados en la Facultad de Ingeniería, formándonos un criterio más amplio de lo que pueden hacer los ingenieros y el alcance que pueden tener.

Podemos concluir también que los conocimientos adquiridos en la Facultad de Ingeniería son de gran ayuda y nos sirven de base para poder entender el manejo y funcionamiento de nuevas tecnologías, así como para el desarrollo de trabajos de investigación.

Sin embargo, es necesario que el ingeniero esté continuamente estudiando, ya que la tecnología ha ido avanzando rápidamente y es importante estar al día en estos avances.

APÉNDICE

LENGUAJE NEURON C

Diferencias entre el lenguaje NEURON C y el lenguaje ANSI C

El lenguaje NEURON C es un lenguaje semejante al lenguaje estándar ANSI C; sin embargo, El lenguaje NEURON C no es una parte del lenguaje C estándar.

La siguiente lista contiene algunas diferencias entre el NEURON C y el ANSI C.

- El NEURON C no soporta cálculos con punto flotante.
- El ANSI C define un *short int* como de 16 bits o más, y un *long int* como de 32 bits o más. El NEURON C define un *short int* como de 8 bits y un *long int* como de 16 bits. En el NEURON C, *int* es por omisión *short int*.
- El NEURON C no soporta las clases *register* o *volatile*.
- El NEURON C no soporta inicialización en declaraciones de variables automáticas.
- El NEURON C no soporta operaciones de dirección para variables automáticas o parámetros de función formal.
- El NEURON C no soporta estructuras o uniones como parámetros de procedimiento, como variables automáticas, o como valores de regreso a función.

- El NEURON C no soporta arreglos como variables automáticas.
- Las estructuras de las variables de la red no pueden incluir apuntadores. El NEURON C no soporta apuntadores para las variables de la red, cronómetros, variables eeprom, tareas de mensajes, y objetos de E/S.
- Los nombres de las variables de la red y tareas de mensajes son limitadas a 16 caracteres.
- Sólo son incluidas las funciones de librería *memcpy()* y *memset()* del ANSI C en el NEURON C.
- El NEURON C contiene palabras reservadas adicionales y sintaxis no incluido en el ANSI C.
- El NEURON C soporta constantes binarias además del octal y hexadecimal.
- El NEURON C soporta el // para comentarios, a diferencia del /* */ del C++.
- La construcción *main()* no es usada. En su lugar, una ejecución de objetos del programa en NEURON C consiste de declaraciones *when* en lugar de las funciones.
- El NEURON C no soporta varios archivos fuente en unidades de compilación separadas (sin embargo, soporta la directiva *#include*).
- Las directivas que soporta el NEURON C, del ANSI C, son: *#define*, *#include*, y *#pragma*.

Tipos de variables del lenguaje NEURON C

El lenguaje NEURON C soporta los siguientes tipos de variables:

<code>(signed) long int</code>	cantidad	16 bits
<code>unsigned long int</code>	cantidad	16 bits
<code>signed char</code>	cantidad	8 bits
<code>(unsigned) char</code>	cantidad	8 bits
<code>(signed) [short] int</code>	cantidad	8 bits
<code>unsigned [short] int</code>	cantidad	8 bits
<code>enum (int type)</code>	cantidad	8 bits

Clases de almacenamiento

Si la clase de almacenamiento no es especificada y la declaración está en la extensión del archivo, el dato o función es global. (La extensión del archivo forma parte del programa NEURON C que no está contenido dentro de una función o tarea). Los datos globales son presentados a lo largo de la ejecución del programa. Al energizar el Circuito Integrado NEURON o reiniciarlo, el dato global es inicializado con la expresión del valor inicial, si se presenta, o si no con un cero (la declaración de la variable con las clases *eprom* o *config* son también inicializados cuando la imagen de la aplicación es cargada por primera vez).

El NEURON C soporta las siguientes clases de almacenamiento del ANSI C.

auto	es una variable de extensión local. Generalmente, ésta deberá estar dentro de una función.
const	es una variable la cual no puede ser modificada por el programa de la aplicación.
extern	son artículos o funciones de datos que están definidos en otro módulo, en una librería, o en la imagen del sistema microprogramado.
static	es un artículo o función de datos el cual no es útil en otros módulos. Además, si el artículo de datos es local para una función o una tarea <i>when</i> , el valor de los datos estarán preservados antes de la invocación.

Además de las clases de almacenamiento para el ANSI C, el NEURON C también provee las siguientes clases:

config	esta clase de palabra llave puede ser combinada sólo con una declaración de una variable de entrada de la red, está alojada en la EEPROM, y puede ser cambiada sólo por otro nodo.
network	esta clase de palabra llave introduce una variable de la red.
system	esta clase de palabra llave es usada en NEURON C solamente para acceder la librería de la función microprogramada del Circuito Integrado NEURON.

Tipos de constantes enteras

Las constantes enteras tienen los siguientes tipos:

0 .. 127	signed short
128 .. 32767	signed long
32768 .. 65535	unsigned long

Las constantes hexadecimal, octal y binario tienen los siguientes tipos.

0x0 .. 0x7f	signed short
0x80 .. 0xff	unsigned short
0x100 .. 0x7ff	signed long
0x8000 .. 0xffff	unsigned long

Declaraciones del lenguaje NEURON C

El ANSI C y el NEURON C soportan declaraciones de:

Declaración	Ejemplo
• Datos simples	int a,b,c;
• Tipos de datos	typedef unsigned long ULONG;
• Enumeraciones	enum hue {RED, GREEN, BLUE};
• Apuntadores	char *p;
• Funciones	int f(int a, int b);
• Arreglos	int a [4];
• Estructuras y uniones	struct s { int field1; unsigned field2 : 3; unsigned field3 : 4; };

Además, El NEURON C soporta declaraciones de:

Declaración	Ejemplo
• Objetos de E/S	IO_output oneshot relay_trigger;
• Cronómetros	mtimer led_on_timer;
• Variables de la red	network input int temperature;

BIBLIOGRAFÍA

1. Echelon
LONBUILDER User's Guide
Echelon Corporation
1992
 2. Motorola Semiconductor
NEURON CHIP Distributed Communications and Control Processors
Motorola Inc
1992
 3. Echelon
NEURON C Programmer's Guide
Echelon Corporation
1992
 4. Echelon
LONBUILDER Startup and Hardware Guide
Echelon Corporation
1992
 5. Echelon
LONBUILDER Notes and News
Echelon Corporation
1992
 6. Ulyses Bleck
Redes de Computadors
Re-Ma Editorial
1989
-