

12  
zej



**UNIVERSIDAD NACIONAL AUTONOMA  
DE MEXICO**

**ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES  
PLANTEL "ARAGON"**

**DESARROLLO DEL SUBSISTEMA DE REPORTES DE  
CONTROL SUPERVISORIO DE SUBESTACIONES  
ELECTRICAS EN COMPUTADORA PERSONAL**

**T E S I S**

QUE PARA OBTENER EL TITULO DE:  
INGENIERO EN COMPUTACION  
P R E S E N T A :  
GUILLERMINA FUENTES CASTILLO



escuela nacional de estudios profesionales  
aragon

JUNIO DE 1993

**TESIS CON  
FALLA DE ORIGEN**



## **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 está 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.

# INDICE

INTRODUCCION.....	i
<b>1. DESCRIPCION DE SISTEMAS SCADA .....</b>	<b>1</b>
<b>1.1. ARQUITECTURA DE UN SISTEMA SCADA .....</b>	<b>2</b>
1.1.1. Estación Maestra de Control Supervisorio .....	3
1.1.2. Unidades Terminales Remotas (UTR).....	3
1.1.3. Sistema de comunicación .....	6
<b>1.2. FUNCIONES GENERALES DE UN SISTEMA SCADA .....</b>	<b>7</b>
1.2.1. Adquisición de datos.....	7
1.2.2. Despliegue de información.....	8
1.2.3. Control supervisorio.....	8
1.2.4. Procesamiento de alarmas.....	9
1.2.5. Almacenamiento de información y reportes.....	9
1.2.6. Adquisición de la secuencia de eventos .....	10
1.2.7. Cálculo de parámetros .....	10
<b>1.3. ESTACION MAESTRA MAC - 5000 PARA CONTROL         SUPERVISORIO.....</b>	<b>10</b>
1.3.1. Arquitectura general .....	11
1.3.2. Sistema de cómputo.....	12
1.3.3. Controlador de comunicaciones .....	13
1.3.4. Funciones SCADA .....	14

<b>2.</b>	<b>SISTEMA DE REPORTES.....</b>	<b>17</b>
2.1.	REPORTES DEL SISTEMA SUPERVISORIO .....	19
2.1.1.	Generador de Reportes de propósito general.....	19
2.1.2.	Generación de Reportes y mantenimiento.....	19
2.1.3.	Generación de la BDD y edición .....	20
2.2.	CARACTERISTICAS PRINCIPALES DE UN SISTEMA DE REPORTES.....	20
2.3.	REPORTES EN LA ESTACION MAESTRA MAC - 5000.....	21
2.4.	REPORTES ESPECIALES .....	22
2.4.1.	Reporte de Máximos y Mínimos de Voltaje en Barras .....	22
2.4.2.	Reportes Máximos y Mínimos de Corriente en circuitos.....	22
2.4.3.	Consumo Mensual .....	23
2.4.4.	Reportes de Operación de Interruptores, por falla y por maniobra.....	23
2.4.5.	Reporte de Operación de Circuitos.....	23
2.4.6.	Reporte de Histórico de Mediciones.....	24
<b>3.</b>	<b>ESPECIFICACION Y DISEÑO DEL SUBSISTEMA DE REPORTES DE LA EM MAC - 5000.....</b>	<b>27</b>
3.1.	JUSTIFICACION DE LA REALIZACION DE PROGRAMAS DEL SUBSISTEMA DE REPORTES EN UNA COMPUTADORA PERSONAL .....	28
3.1.1.	Ventajas al utilizar un sistema de reportes en una computadora personal.....	29
3.2.	ESPECIFICACION DEL SUBSISTEMA DE REPORTES .....	30
3.2.1.	Inicialización del sistema.....	33
3.2.2.	Subsistema de enlace PC - MAC.....	33
3.2.3.	Subsistema de generación de Reportes.....	35
3.2.4.	Valores Máximos y Mínimos de Voltaje en Barras .....	38
3.2.5.	Valores Máximos y Mínimos de Corriente en circuitos.....	41

3.2.6.	Operación de interruptores.....	44
3.2.7.	Operación de Circuitos .....	46
3.2.8.	Consumo Mensual de Transformadores.....	48
3.2.9.	Histórico de mediciones .....	50
3.2.10.	Subsistema de visualización.....	51
3.3.	DISEÑO DEL SUBSISTEMA DE REPORTES DE LA EM MAC - 5000 EN COMPUTADORA PERSONAL .....	52
3.3.1.	Definiciones de los procesos en el diagrama general (nivel 1) .....	55
3.3.2.	Definición de flujo de datos en el diagrama general (nivel 1).....	58
3.3.3.	Definición de estructuras de datos utilizadas en el diagrama general (nivel 1).....	63
3.3.4.	Definiciones de los diagramas del nivel 2 por proceso .....	69
3.3.5.	Definición de flujo de datos de los diagramas del nivel 2.....	74
3.3.6.	Definiciones de los diagramas del nivel 3.....	76
4.	ENLACE Y CONVERSION DE BDD MAC-5000 CON PC.....	105
4.1	ENLACE PARA COMUNICACION PC MAC - 5000 .....	106
4.1.1.	Norma RS - 485 .....	107
4.1.2.	Protocolo PAC.....	109
4.2	CONVERSION BDD MAC - PC.....	114
4.2.1.	Formato MAC (meta_r y meta_a).....	115
4.2.2.	Formato dBase III plus .....	118
4.2.3.	Implementación de conversión de formatos.....	120
5.	PRUEBAS E INTEGRACION DEL SISTEMA .....	123
5.1.	PRUEBAS DE LAS FUNCIONES PRINCIPALES.....	124
5.2.	CONVERSION DEL FORMATO MAC A FORMATO DBASE.....	126
5.3.	ACTIVACION DE LOS REPORTES.....	132

5.4. RELOJ.....	132
5.5. VERIFICACION DE OPERACIONES EN LOS REPORTES.....	134
5.6. ALMACENAMIENTO CADA 24 HORAS.....	139
5.7. SOLICITUD DE IMPRESION.....	140
5.8. CREACION MENSUAL DE ARCHIVOS DE REPORTES.....	141
5.9. PRUEBAS DE ESTABILIDAD DEL SISTEMA.....	141
5.10. GRAFICACION DEL REPORTE HISTORICO.....	142
5.11. PRESENTACION EN PANTALLA.....	143

CONCLUSIONES.....	148
-------------------	-----

REFERENCIAS.....	151
------------------	-----

## APENDICES

A. CARACTERISTICAS DE LAS HERRAMIENTAS DE DESARROLLO.....	154
A.1. LENGUAJE 'C' (TURBO C VERSION 2.0).....	155
A.2. UTILERIA CODE BASE 4.2.....	156
A.3. UTILERIA C-SCAPE 3.0.....	159
A.4. HERRAMIENTA CASE SYSTEM ARCHITECT.....	160
A.4.1. Modelado de procesos (Diagrama de Flujo de Datos).....	161
A.4.2. Modelo Entidad - Relación.....	161
A.4.3. Diagramas de estructura.....	162
A.2.4. Diagramas de descomposición.....	162
A.2.5. Sistemas de tiempo real.....	163

<b>B.</b>	<b>METODOLOGIA DE DESARROLLO .....</b>	<b>165</b>
	<b>B.1. SISTEMAS DE INFORMACION.....</b>	<b>166</b>
	<b>B.2. SISTEMAS DE TIEMPO REAL .....</b>	<b>166</b>
	<b>B.3. METODO DE ANALISIS ORIENTADO AL FLUJO DE DATOS.....</b>	<b>167</b>
	<b>B.3.1. Diagramas de flujo de datos (DFD).....</b>	<b>168</b>
	<b>B.3.2. Extensiones del modelo de flujo de datos .....</b>	<b>172</b>
	<b>B.4. METODOLOGIA UTILIZADA EN EL DESARROLLO DEL SUBSISTEMA DE REPORTES DE LA EM MAC - 5000 .....</b>	<b>174</b>
	<b>B.4.1. Consideraciones en el proceso de diseño .....</b>	<b>176</b>
<b>C.</b>	<b>ESTRUCTURA DE LOS MENSAJES DE REPORTES.....</b>	<b>178</b>
	<b>D.1. ESTRUCTURA MSJINT.....</b>	<b>179</b>
	<b>D.2. ESTRUCTURA MSJSOL.....</b>	<b>179</b>
	<b>D.3. MENSAJES DEL SISTEMA.....</b>	<b>180</b>
<b>D.</b>	<b>GLOSARIO DE TERMINOS.....</b>	<b>185</b>

## **INDICE DE FIGURAS**

FIGURA 1.1.	Arquitectura de un sistema SCADA.....	2
FIGURA 1.2.	Principales elementos de la UTR.....	4
FIGURA 1.3.	Vías de comunicación a la UTR.....	6
FIGURA 1.4.	Arquitectura de la Estación Maestra MAC -5000.....	11
FIGURA 1.5.	Sistema de Cómputo MAC.....	12
FIGURA 1.6.	Arquitectura controlador de comunicaciones CIC.....	13
FIGURA 3.1.	Arquitectura de Estación Maestra de distribución.....	31
FIGURA 3.2.	Configuración del subsistema de cómputo de la MAC - 5000.....	32
FIGURA 4.1.	Enlace PC - MAC usando las tarjeta MAC-851 y PC-851.....	107
FIGURA 4.2.	Interconexión entre transmisores, receptores.....	108
FIGURA 4.3.	(a) Diagrama de temporización, (b) Diagrama de conector.....	110
FIGURA 4.4.	Comandos de acceso a red.....	111
FIGURA 4.5.	Formato de mensaje.....	112
FIGURA 4.6.	Tarjeta MAC-851, PC-851.....	114
FIGURA 5.1.	Menú de graficación de Histórico de mediciones.....	143
FIGURA 5.2.	Menú principal opción VMMV.....	144
FIGURA 5.3.	Menú opción OPIN.....	144
FIGURA 5.4.	Menú opción OPERA.....	145
FIGURA 5.1.	Menú de graficación de Histórico de mediciones.....	145
FIGURA 5.5.	Menú opción Msje INT.....	146
FIGURA 5.6.	Menú opción OPERA.....	147

<b>FIGURA B.1. Símbolos básicos para construir diagramas de flujos de datos .....</b>	<b>169</b>
<b>FIGURA B.2. DFD de sistemas de información de una biblioteca .....</b>	<b>169</b>
<b>FIGURA B.3. Refinamiento parcial del DFD del sistema de información de una biblioteca.....</b>	<b>170</b>
<b>FIGURA B.4. Refinamientos del flujo de información.....</b>	<b>171</b>
<b>FIGURA B.5. Ejemplo de notación Ward &amp; Mellor.....</b>	<b>173</b>
<b>FIGURA B.6. Diagrama estructurado.....</b>	<b>174</b>
<b>FIGURA B.7. Diagrama de estructura decorado.....</b>	<b>175</b>

## **INDICE DE DIAGRAMAS**

DIAGRAMA 3.1.	Diagrama general del diseño de reportes MAC - PC.....	37
DIAGRAMA 3.2	Comunicación PC - MAC .....	70
DIAGRAMA 3.3.	Descripción del módulo de reportes.....	72
DIAGRAMA 3.4.	Proceso de creación del reporte OPIN .....	78
DIAGRAMA 3.5.	Decodifica mensaje OPIN .....	79
DIAGRAMA 3.6.	Actualiza reporte OPIN .....	80
DIAGRAMA 3.7.	Inicializa reporte OPIN.....	81
DIAGRAMA 3.8.	Obtiene relaciones para reporte.....	82
DIAGRAMA 3.9.	Crea archivo DBF OPIN.....	83
DIAGRAMA 3.10.	Realiza operaciones INTE_SE.....	84
DIAGRAMA 3.11.	Proceso de creación del reporte VMMV .....	86
DIAGRAMA 3.12.	Decodifica mensaje VMMV .....	88
DIAGRAMA 3.13.	Muestra y actualiza la relación BARRA .....	89
DIAGRAMA 3.14.	Actualiza reporte VMMV .....	90
DIAGRAMA 3.15.	Crea archivo DBF VMMV .....	91
DIAGRAMA 3.16.	Inicializa reporte VMMV.....	92
DIAGRAMA 3.17.	Proceso de creación del reporte OPERA.....	94
DIAGRAMA 3.18.	Crea archivo DBF OPERA .....	96
DIAGRAMA 3.19.	Decodifica mensaje OPERA .....	97
DIAGRAMA 3.20.	Inicializa reporte OPERA .....	98
DIAGRAMA 3.21.	Obtiene relación a cambiar .....	99
DIAGRAMA 3.22.	Verifica en archivo el estado .....	100

<b>DIAGRAMA 3.23. Actualiza reporte Opera.....</b>	<b>101</b>
<b>DIAGRAMA 3.24. Proceso de creación del reporte Histórico de Mediciones.....</b>	<b>103</b>
<b>DIAGRAMA 4.1. Convierte de MAC a dBase.....</b>	<b>122</b>

# INTRODUCCION

El aprovechamiento de los avances tecnológicos en diversas áreas de generación, transmisión y distribución de la energía eléctrica tiene como principal objetivo mejorar la calidad, continuidad y economía del servicio eléctrico, con este propósito, se incorporan continuamente sistemas que hacen más eficiente la supervisión de los diversas tareas que participan en el suministro del servicio eléctrico. Uno de esos sistemas es el *control supervisorio*.

El *control supervisorio* se define en la norma ANSI C37.1 [IEE 91] como: "Un arreglo para el control y supervisión por operador de aparatos localizados en zonas remotas, utilizando técnicas de multicanalización sobre un número relativamente pequeño de canales interconectados". Además se define un *sistema supervisorio* como: "Todo el control, indicación y equipo asociado de telemetría en la estación maestra, y todos los dispositivos complementarios en las estaciones remotas".

Un sistema supervisorio puede adquirir datos y controlar los dispositivos usados para generar, transmitir y distribuir la energía desde un punto geográfico distante. A esto se le conoce comúnmente como un sistema de control supervisorio y adquisición de datos "SCADA" por sus siglas en inglés.

En su forma más sencilla los sistemas SCADA consisten de una estación maestra con una Unidad Terminal Remota (UTR), conocidos generalmente como sistemas uno a uno, utilizados en aplicación.

Los sistemas SCADA más complejos pueden controlar la generación, transmisión y distribución de una área geográfica más extensa. Los sistemas SCADA pueden también adquirir datos y proveer información para sistemas avanzados de administración de la energía (EMSS) en el centro de operación.

Para realizar la supervisión y control de un sistema SCADA, se requiere básicamente de:

Una Estación Maestra (EM): compuesta generalmente de varias computadoras, periféricos, subsistema de entrada/salida (comunicaciones) que permiten a los operadores de un centro de control supervisar el estado de los sistemas de potencia y controlarlos [IEE 91].

Varias Unidades Terminales Remotas (UTR) distribuidas en diferentes puntos geográficos, y enlazadas directamente a elementos de campo de los cuales se obtienen dos tipos de datos; digitales o de estado y analógicos. Los datos digitales se obtienen por el monitoreo de contactos externos como son: interruptores, cuchillas, generación de pulsos, estado de protecciones, etc. Los datos analógicos son obtenidos normalmente de transductores con valores de las mediciones de corriente o voltaje de salida. Una tendencia reciente para adquirir esta información en forma digital directamente desde otros dispositivos basados en microprocesadores son: controladores, relevadores, medidores, etc. Estos datos son adquiridos periódicamente por la UTR.

Y el controlador de comunicaciones, el cual realiza los intercambios de información entre las estaciones maestras y las UTR. Los medios de comunicación pueden ser por diferentes vías, las más comunes son: teléfono, radio, onda portadora en líneas de transmisión o distribución, fibra óptica, microrondas, etc.

Por lo anteriormente descrito se observa que los sistemas SCADA manejan mucha información y requieren de una base de datos en tiempo real, que permita el análisis del comportamiento del sistema en línea y la generación de *reportes* para análisis fuera de línea.

El Departamento de Electrónica de la División de Sistemas de Control del Instituto de Investigaciones Eléctricas desarrolló la estación maestra MAC -5000 para supervisión y telecontrol de la red de distribución y subtransmisión de la zona urbana de Morelia y poblaciones vecinas.

La estación maestra MAC - 5000 se integra de un sistema de Cómputo MAC y del Controlador Inteligente de Comunicaciones (CIC) hacia las UTR, los cuales se enlazan por un canal serie tipo RS-485 a 38400 bauds. Incluye un disco duro, una unidad de disco flexible, dos estaciones de operador, integradas cada una por un monitor, un teclado y una impresora. Además la estación para el ingeniero de sistema se compone de una computadora personal (IBM-PC o compatible), unida al sistema de cómputo MAC. La estación del Ingeniero de sistema realiza funciones de generación/mantenimiento de la Base De Datos (BDD) fuera de línea y cargado de programas y datos a la MAC - 5000 [GUT 90].

El sistema de cómputo esta compuesto por múltiples procesadores en esquema multimaestro con acceso a una base de datos en memoria común. El sistema de cómputo se compone por cuatro módulos maestros MAC-1186, uno dedicado al procesamiento de información o módulo Central, un módulo para cada una de las estaciones del operador (IHM1 e IHM2), y el cuarto módulo en el cual se generan los *Reportes Especiales*, se realiza el manejo de discos (unidad de disco duro y flexible), y el enlace con la estación del ingeniero de sistema.

En apéndice C describe la estructura de los mensajes utilizados en el proceso "Genera reportes".

En el apéndice D es el glosario de términos y de las abreviaciones más frecuentemente utilizadas durante la elaboración del trabajo de tesis.

El capítulo uno da una breve descripción de un sistema supervisorio SCADA y la estación maestra MAC - 5000.

El capítulo dos describe el objetivo y la forma de generar los reportes de la estación maestra MAC - 5000 los cuales son:

- Máximos y Mínimos de Voltaje.
- Máximos y Mínimos de Corriente.
- Operación de interruptores.
- Operación de circuitos.
- Consumo Mensual.
- Histórico de mediciones

El capítulo tres contiene el diseño detallado del subsistema de reportes utilizando la herramienta de CASE (Computer Aided Software Engineering) denominada *System Architect*, describiendo la arquitectura general del diseño.

El capítulo cuatro, se describe el software de enlace a la Estación Maestra y la conversión del formato utilizado en la EM MAC-5000 al nuevo formato en la computadora personal.

En el capítulo cinco, se describe el plan de pruebas para la integración del sistema de reportes, estas pruebas se realizaron por función y de todo el sistema integrado, con el fin de tener una visión general de la confiabilidad del sistema.

Se presentan las conclusiones del trabajo desarrollado, las referencias utilizadas y finalmente los apéndices.

En el apéndice A se describen las características de las herramientas utilizadas para el desarrollo de los programas del subsistema de reportes para la estación maestra MAC - 5000.

En el apéndice B se describe la metodología utilizada para el diseño de sistemas en tiempo real, basada en el método de diagramas de flujos de datos de Ward & Mellor y utilizando el paquete CASE System Architect.

En apéndice C describe la estructura de los mensajes utilizados en el proceso "Genera reportes".

En el apéndice D es el glosario de términos y de las abreviaciones más frecuentemente utilizadas durante la elaboración del trabajo de tesis.

En apéndice C describe la estructura de los mensajes utilizados en el proceso "Genera reportes".

En el apéndice D es el glosario de términos y de las abreviaciones más frecuentemente utilizadas durante la elaboración del trabajo de tesis.

## **Capítulo 1**

# **DESCRIPCION DE SISTEMAS SCADA**

En este capítulo se explica en forma general las funciones y arquitectura de un sistema de control supervisorio SCADA. Además se describe la Estación Maestra MAC - 5000 la cual sirvió de base para integrar un sistema de Control Supervisorio de redes eléctricas de distribución.

Los sistemas SCADA están definidos en la norma ANSI C37.1 [IEE 91] como un arreglo para el control y supervisión por operador de dispositivos localizados en zonas remotas, utilizando técnicas de multicanalización sobre un número relativamente pequeño de canales interconectados, además estos sistemas pueden adquirir datos y controlar equipos que generan, transmiten y distribuyen la energía a diferentes puntos del país.

Una configuración frecuente de los sistemas SCADA consiste de una Estación Maestra y múltiples Unidades Terminales Remotas (UTR), unidas por un canal de comunicación. En los sistemas eléctricos el objetivo es proporcionar a los operadores del sistema la suficiente información para operarlo de una manera eficiente, segura y económica [IEE 91].

Los sistemas de control supervisorio SCADA tienen además diversas aplicaciones en la industria tales como: el tratamiento de aguas residuales, la transportación, distribución y producción de petróleo y gasolina, redes de comunicaciones, control industrial, etc. [GAU 87].

### 1.1. ARQUITECTURA DE UN SISTEMA SCADA

El sistema SCADA se integra de una Estación Maestra, establecida en un centro de control, y un número determinado de Unidades Terminales Remotas, distribuidas en puntos geográficos distantes, las cuales son enlazadas generalmente por radio u otras vías de comunicación alternas como son: teléfono, portadora en líneas de transmisión, fibra óptica, microondas entre las más comunes ver figura 1.1.

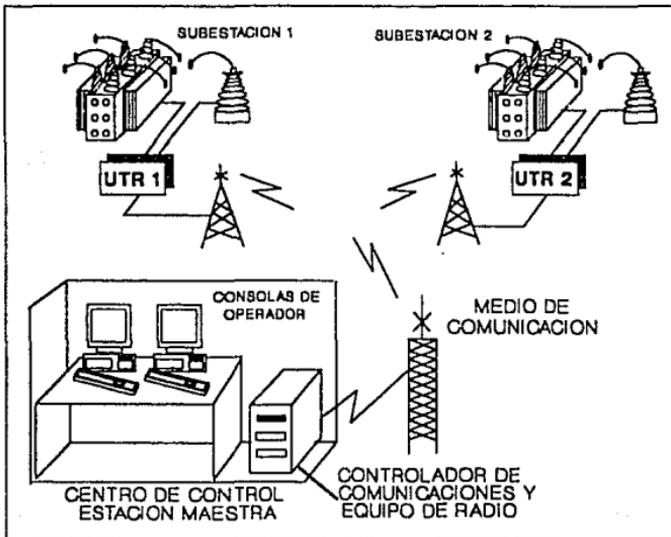


FIGURA 1.1 Arquitectura de un sistema SCADA

Entre los principales beneficios de los sistemas SCADA se encuentran:

- Reducir el costo de producción, el tiempo de mantenimiento y asegurar la satisfacción de los consumidores de energía, debido a una mejor calidad de servicio, a menor costo y con servicio ininterrumpido.
- La comunicación de los sistemas es a través de radio, lo cual permite incrementar la cobertura geográfica del operador sin la necesidad de usar y depender de líneas telefónicas costosas y a veces inaccesibles.

#### 1.1.1. Estación Maestra de Control Supervisorio

En los sistemas actuales la Estación Maestra se compone de una conexión de computadoras, periféricos y subsistemas apropiados de E/S que permiten a los operadores supervisar el estado de los sistemas de potencia y controlarlos. Este equipo proporciona diferentes medios para la supervisión, como son: en pantalla, por alarmas audibles e impresiones que reportan el estado del sistema. Estas funciones se realizan en el sistema de cómputo con la ayuda de la interfaz hombre-máquina.

#### 1.1.2. Unidades Terminales Remotas (UTR)

La UTR es un equipo electrónico instalado en sitios remotos, para monitorear o medir los sensores o dispositivos instalados en campo. La UTR convierte esta medición o señal adquirida en datos que pueden ser transmitidos por algún medio de comunicación, también recibe señales de comandos los cuales pueden controlar una función o un dispositivo en campo.

Aplicando la información anterior a los sistemas eléctricos, se puede considerar lo siguiente: las UTR se localizan en plantas o unidades de generación, subestaciones de transmisión y distribución las cuales reciben datos de la EM y ejecuta comandos de control y son el medio de interconexión entre el equipo de control de la subestación y la EM. Las funciones de la UTR se llevan a cabo colocando relevadores y circuitos de medición como transductores, medidores (corriente, voltaje, etc) directamente a los elementos de campo.

Los elementos que constituyen una UTR son los siguientes y se muestran en la figura 1.2:

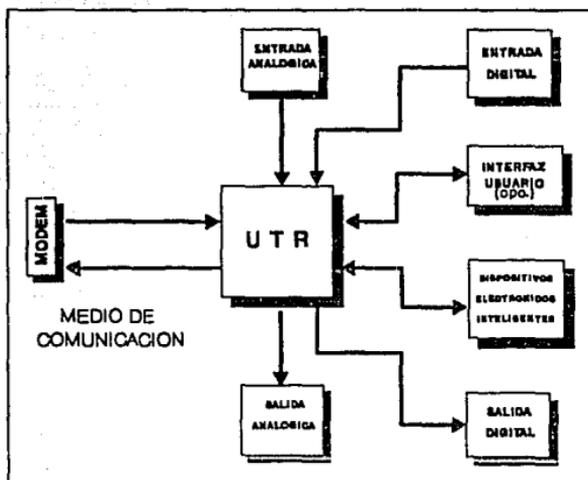


FIGURA 1.2. Principales elementos de la UTR .

**Entradas digitales.** La adquisición de datos digitales en la mayoría de los sistemas SCADA es obtenida por el monitoreo de contactos externos: interruptores, cuchillas, generador de pulsos, etc. Esta adquisición puede ser de cuatro tipos:

- El estado actual del interruptor (abierto/cerrado).
- Número de cambios del contacto desde el último reporte a la EM.
- Secuencia de eventos (SOE) registra en una tabla la hora de ocurrencia del cambio de estado del contacto.
- Valores acumulados (consumo de energía en KWH o KVARH, medición de acumuladores, etc.).

**Entradas analógicas.** Los datos analógicos se adquieren normalmente por transductores que obtienen el voltaje o corriente de salida del elemento medido. Existe una tendencia reciente en adquirir esta información en forma digital directamente de otros dispositivos basados en microprocesadores entre los que tenemos: controladores programables, relevadores, medidores inteligentes, etc. Estos datos son adquiridos periódicamente por la UTR y enviados a la EM.

**Dispositivos electrónicos inteligentes.** Las UTR pueden aceptar datos en forma serial de dispositivos electrónicos inteligentes instalados para ejecutar un procesamiento discreto o un monitoreo en la subestación. La UTR puede incorporar la información en la BDD de la EM. Entre estos dispositivos se incluyen relevadores digitales, registradores de fallas, registradores de secuencia de eventos, medidores inteligentes, etc. Esto beneficia en la reducción de transductores y puntos de estado/control alambrados.

**Salidas analógicas.** Las salidas analógicas son las menos utilizadas en las funciones de la UTR. El tipo más común produce una señal que es proporcional a la cantidad digital especificada en un comando de la EM. A veces se requieren salidas de volts o milivolts, pero los estándares de corriente son de 4 a 20 miliampers de 0 a 600 ohms.

**Salidas digitales.** La mayoría del control se ejecuta mediante salidas digitales comandadas desde la EM, y casi siempre involucran la operación de los contactos de los relevadores que se encuentran en la UTR o en los que se encuentran externamente.

**Interfaz Usuario.** La interfaz de usuario permite a los técnicos dar un mantenimiento más fácil a la UTR al interactuar localmente con los elementos que la componen. La UTR debe tener un determinado número de indicadores (usualmente LED's) que muestren las partes que estén funcionando correctamente y las que presenten fallas. Las UTR más avanzadas que utilizan microprocesadores pueden ofrecer interfaces de usuario más sofisticadas, de tipos permanentes o portables y pueden incluir paneles de control usando teclas especiales o pantallas en un CRT, así como dispositivos para generar impresiones y comunicación con algunos otros periféricos.

### 1.1.3. Sistema de comunicación

El sistema de comunicación es el encargado de procesar la información que se intercambia entre las UTR y la EM, reportando los datos en un formato predefinido. El procesamiento de los datos se hace en módulos inteligentes donde se ejecutan programas con protocolos de transmisión y recepción de datos hacia las UTR. El sistema de comunicación libera al sistema de cómputo de las tareas de comunicación y manejo del medio físico, independizando de esta manera las comunicaciones.

El sistema de comunicaciones se enlaza con el procesamiento principal (sistema de cómputo) por un puerto o red local y se recomienda utilizar una norma de comunicación que permita la transferencia de información en su velocidad más alta. Se utilizan protocolos orientados a adquisición de datos y control en tiempo real. Otra función del sistema de comunicaciones es llevar una estadística de mensajes enviados y recibidos por los puertos de comunicación, ver figura 1.3 [IEE 91].

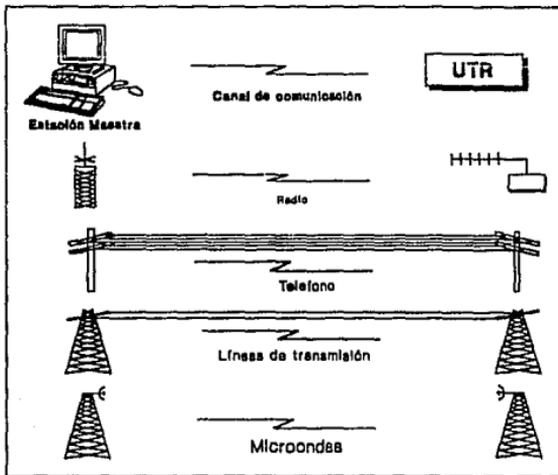


FIGURA 1.3. Vías de comunicación a la UTR

## 1.2. FUNCIONES GENERALES DE UN SISTEMA SCADA

La razón principal para instalar un sistema supervisorio es dar al operador la suficiente información y capacidad para operar y controlar un sistema, como puede ser un sistema eléctrico de potencia, de una manera eficiente, segura y económica. El operador puede determinar el estado del sistema de acuerdo con la información que se obtenga, incluyendo: estados (contactos abiertos/cerrados, etc.), valores analógicos (voltaje, corriente, etc.) y datos acumulativos (consumo de energía, etc.), asimismo, puede ejercer acciones de control hacia los dispositivos del sistema para que respondan eficientemente cuando sean requeridos, de tal manera que se alcancen los objetivos deseados de operación y calidad del servicio.

Ciertas funciones son básicas en los sistemas SCADA para subestaciones eléctricas, entre las más comunes se incluyen [GAU 87]:

- Adquisición de datos.
- Despliegue de información.
- Control Supervisorio.
- Procesamiento de Alarmas.
- Almacenamiento de información y reportes.
- Adquisición de la secuencia de eventos.
- Cálculo de parámetros.

### 1.2.1. Adquisición de datos

La adquisición periódica de datos provenientes de las UTR's es fundamental. La mayoría de los sistemas eléctricos adquieren datos muestreando, y los datos son transmitidos de la UTR a la estación maestra solo cuando se recibe un requerimiento de la maestra a la UTR. Existen dos opciones de respuesta de la UTR. Una opción es enviar los valores o estados de los puntos o grupos de puntos requeridos. La otra opción es enviar sólo aquellos que hayan cambiado de estado o su valor haya excedido un valor predeterminado. Esto último es conocido como "*reportes por excepción*" su principal ventaja es la reducción de mensajes hacia la estación maestra.

El proceso de adquisición de datos puede ser considerado como un conjunto de subprocesos especializados y altamente relacionados. Estos subprocesos incluyen:

- Exploración interna y actualización rápida de la base de datos de la UTR.
- Muestreo periódico a la UTR por la estación maestra.
- Transmisión de los datos requeridos de la UTR.
- Revisión de los datos para evitar errores introducidos en la transmisión.
- Conversión de los datos a unidades de ingeniería y etiquetación de tiempo.

### 1.2.2. Despliegue de información

Es el proceso de recuperar selectivamente datos fijos e información variable en tiempo real de la base de datos, combinandolos y presentandolos al operador usualmente en forma de gráficas a color en un monitor. Los datos fijos incluyen información del sistema la cual no varía con el tiempo. La información variable incluye los estados de los dispositivos y valores analógicos que varían en magnitud y posiblemente en signo. La designaciones de unidades e identificación de los puntos por su nombre o por algún número se consideran valores fijos que se añaden a las variables.

La selección de pantallas está organizada comúnmente en una estructura jerárquica donde un índice de páginas permiten al operador seleccionar una amplia variedad de ventanas usando técnicas de posicionamiento tales como: ratón, teclado, trackball, lápiz óptico o touchscreen. Un método alternativo es el uso de teclas de función dedicadas, este tipo de selección permite un acceso rápido a las pantallas más usadas, pero el número de tales teclas esta limitado por restricciones de espacio. Actualmente existe la tendencia a utilizar ventanas e iconos (botones, etc.) como medio de acceso a la información.

### 1.2.3. Control supervisorio

Control supervisorio es el proceso de ejecutar acciones de control a localidades remotas. Este proceso incluye selección de la UTR, del dispositivo a controlar, y la ejecución del comando deseado, como cambiar punto de ajuste, etc. La correcta selección y operación es crítica para la seguridad del personal y del sistema eléctrico. La operación

de un dispositivo no seleccionado, o la operación inadvertida de un dispositivo cuando no se ha enviado un comando debe ser evitada y es un parámetro crítico en el diseño de sistemas SCADA.

#### **1.2.4. Procesamiento de alarmas**

El procesamiento de alarmas da al operador la información necesaria para conocer eventos no programados, con la hora de ocurrencia, el lugar donde se encuentra, el identificador del dispositivo y la naturaleza del evento (fallas, licencias, maniobras, apertura/cierre, etc.). Existen varios tipos de presentación de las alarmas, las más comunes son: alarmas ordenadas cronológicamente en pantalla, impresiones y alarmas audibles.

#### **1.2.5. Almacenamiento de información y reportes**

El mantenimiento de registros siempre ha sido una tarea importante en la operación de sistemas eléctricos. Se necesitan registros para almacenar la información y cubrir con las necesidades del sistema, por ejemplo: los procesos de contabilidad, para soporte de operación de sistemas y como propósito de ingeniería de planeación. La práctica común para mantener estos registros es capturar un conjunto de datos preseleccionados a intervalos periódicos y salvarlos en un archivo (muestreo periódico). Ocasionalmente se usan esquemas de compresión de datos para los archivos con grandes cantidades de información.

El archivo histórico de un SCADA es una excelente fuente de información para la producción de varios reportes. Los reportes, su formato y contenido de información pueden ser personalizados para satisfacer las necesidades específicas de una amplia variedad de usuarios finales.

### 1.2.6. Adquisición de la secuencia de eventos

La secuencia de eventos es el proceso de capturar y registrar eventos no programados. Históricamente este registro sea realizado con equipo dedicado para este propósito. Los eventos se consideran generalmente como discretos (dos estados) tales como los que ocurren con los sistemas de protección por relevador. La secuencia de eventos de una UTR dada se pueden imprimir localmente en la UTR o pueden ser transmitidos a la estación maestra en donde son ordenados cronológicamente con eventos recibidos de otras localidades y entonces se pueden imprimir como una lista combinada.

### 1.2.7. Cálculo de parámetros

Existe la necesidad en los sistemas SCADA de ejecutar algunos cálculos utilizando los datos adquiridos. Algunos ejemplos de tales cálculos son la determinación de promedios, valores máximos y mínimos sobre intervalos dados y la integración con respecto al tiempo. Los cálculos con múltiples variables incluyen sumas, restas, multiplicaciones, divisiones, raíz cuadrada, exponenciación y otros.

Los cálculos booleanos se pueden aplicar a puntos de estado discreto y el resultado se considera como un nuevo punto de información. Este proceso puede determinar un estado particular de alguna parte del sistema eléctrico que no se puede definir sólo por una indicación (o no se tiene el equipo de medición).

## 1.3. ESTACION MAESTRA MAC - 5000 PARA CONTROL SUPERVISORIO

El Departamento de electrónica, del Instituto de Investigaciones Eléctricas ha tenido desde su inicio proyectos de control supervisorio, siendo los principales logros el desarrollo de la estación maestra MAC - 5000.

A partir de noviembre de 1989 se instaló en la División Centro Occidente de CFE, con base en Morelia, Michoacán una MAC -5000, la cual a la fecha controla y supervisa las 9 subestaciones que suministran la energía a la ciudad y otras 10 subestaciones en el estado consideradas puntos vitales a nivel de subtransmisión.

La estación Maestra MAC - 5000 es un equipo de adquisición y control supervisorio (SCADA) para la supervisión y telecontrol de redes de distribución eléctrica. La estación maestra adquiere información de un determinado número de UTR, que son enlazadas generalmente por radio, además de poder utilizar otras vías alternas como son: teléfono, portadora en líneas de transmisión, fibra óptica, entre las más comunes.

### 1.3.1. Arquitectura general

La arquitectura de la estación maestra MAC - 5000 se divide en dos módulos: sistema de Cómputo y el Controlador Inteligente de Comunicaciones CIC, la arquitectura del sistema se muestra en la figura 1.4. Estos módulos están unidos por un canal serie tipo RS-485 a 38400 bauds. Se tiene un disco duro y uno flexible. Dos estaciones de operador, cada una compuesta por un monitor, un teclado y una impresora. Se tiene una estación para el Ingeniero del Sistema, residente en una computadora personal, la cual se enlaza al sistema de cómputo por medio de un canal RS-232 [GUT 90].

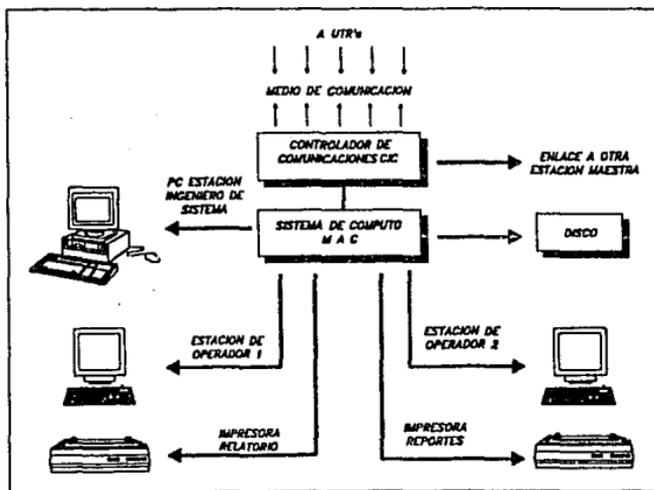


FIGURA 1.4. Arquitectura de la Estación Maestra MAC - 5000

### 1.3.2. Sistema de cómputo

El sistema de cómputo (figura 1.5) se compone de múltiples procesadores, basados en tarjetas MAC-1186 [MOL 89], en un esquema multimaster fuertemente acoplado mediante un área de memoria común. La comunicación es a través de un canal paralelo (IBUS-III), y los accesos al bus son controlados por un arbitraje en paralelo y de prioridades fijas. El sistema de cómputo cuenta con cuatro módulos maestros MAC-1186 (basados en el procesador Intel 8086): Uno dedicado al procesamiento de información o módulo central, un módulo para cada una de las estaciones de operador (IHM1 e IHM2), y el cuarto módulo dedicado a la generación de *reportes especiales*, manejo de discos, y enlace con la estación de Ingeniero de Sistema. Adicionalmente el sistema de cómputo cuenta con un módulo esclavo de comunicaciones (851 basado en el procesador 8031), para enlace con el CIC.

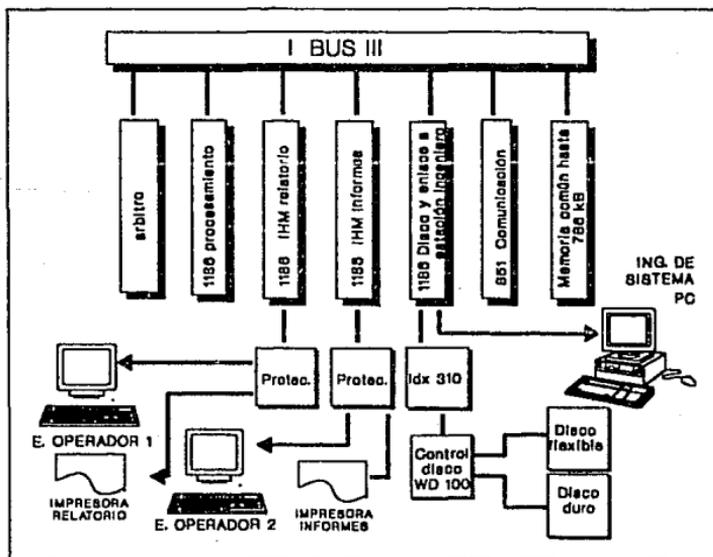


FIGURA 1.5. Sistema de Cómputo MAC

Cada estación de operador cuenta con un teclado con funciones especiales y alfanuméricas, un monitor a color de 19 pulgadas y una impresora. La impresora de la IHM1 esta dedicada al relatorio, donde se imprimen todas las alarmas, eventos o comandos del operador con la fecha y hora de ocurrencia. La impresora de la IHM2 esta dedicada a reportes o impresión de cualquier desplegado en pantalla de las dos consolas.

### 1.3.3. Controlador de comunicaciones

En la figura 1.6 se muestra la arquitectura del controlador inteligente de comunicaciones CIC, que se compone de un módulo maestro MAC-1186 y varios módulos esclavos inteligentes MAC-851, uno encargado de comunicarse al sistema de cómputo MAC y los otros como puertos de comunicación para las UTR o como enlace a una EM de nivel jerárquico superior.

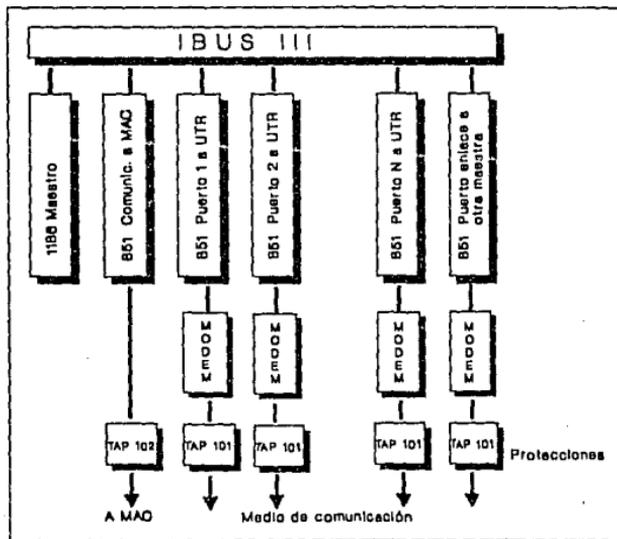


FIGURA 1.6. Arquitectura del controlador de comunicaciones CIC

El maestro del CIC está diseñado para liberar al sistema de cómputo de las labores de comunicación, es decir, del manejo de modems y radios o medio de comunicación, y hacerlo independiente del protocolo de comunicación. Cada puerto se personaliza con el protocolo de comunicación hacia las UTR, de manera que se tiene un sistema que puede tener varios protocolos diferentes trabajando en forma simultánea, independientes del sistema de cómputo.

El maestro del CIC efectúa las tareas de coordinación de los esclavos, y lleva las estadísticas de comunicaciones de todos los puertos.

Cada puerto contiene una imagen del estado de los puntos digitales en el campo y efectúa las detecciones de cambio de estado. El puerto efectúa los barridos cíclicos a las agrupaciones de puntos en las UTR, de manera autónoma. Para el caso de entradas digitales, sólo informa si el punto sufrió un cambio de estado o si se detectó un cambio momentáneo (cierre/apertura).

#### **1.3.4. Funciones SCADA**

El sistema de Cómputo efectúa todas las funciones propias de un SCADA, facilitando la operación por medio de desplegados adecuados. Además el operador tiene a su disposición teclas dedicadas y funcionales que lo guían para lograr una operación segura.

##### *Unifilares y tabulares*

Entre las principales funciones están los desplegados del estado de puntos en las subestaciones mediante diagramas unifilares y tabulares. Los unifilares presentan el diagrama eléctrico de la subestación con el estado de interruptores, cuchillas y mediciones importantes de la subestación. Los tabulares muestra toda la información monitoreada de la subestación, incluso la información presentada en los unifilares más el estado de las protecciones, de los servicios de la subestación o de diversos dispositivos eléctricos. En los tabulares, la información se encuentra dividida en cuatro grupos: puntos digitales, analógicos, acumuladores de potencia y de contadores de operaciones de interruptores.

En ambas presentaciones, unificar o tabular, pueden efectuarse todas las operaciones permitidas sobre los puntos; operaciones digitales: abrir, cerrar, otorgar licencias, retirar licencias, inhibir, desinhibir y reconocer alarma de operaciones; para analógicos tales como: cambio de límites, inhibir, desinhibir y reconocer alarma.

### *Procesamiento de alarmas*

Para prevenir al operador, las alarmas se clasifican en críticas y no críticas; al solicitar el resumen de alarmas, primero se despliegan las críticas y en páginas subsecuentes, las no críticas.

Se tienen cuatro procedimientos de alarma; cambio de normal a anormal, de anormal a normal, en ambos sentidos o no alarma. En el caso de mediciones, verifica el cruce por límites superior e inferior con verificación de banda muerta.

El operador puede inhibir o desinhibir el procesamiento de alarma. La detección de una alarma causa que el mnemónico de la subestación donde ocurrió, empiece a parpadear; que el punto que origina la alarma cambie a color rojo en el desplegado del unificar o tabular donde se encuentre; que los datos del punto se inserten en el archivo de resumen de alarmas; que aparezca en la línea las últimas alarmas en pantalla; que se envíen los datos a la impresora de relatorio, y que se accione una alarma audible. El archivo de resumen de alarmas, con capacidad de 150 alarmas, puede consultarlo y administrarlo el operador.

### *Manejo de telecontrol*

Se tienen diversas verificaciones para aceptar una operación de control a diferentes niveles en el sistema supervisorio que van desde el módulo de interfaz hombre-máquina y base de datos, hasta el punto de salida en la UTK. La verificación final del cambio de estado del dispositivo en campo, se hace con la detección del cambio de estado de la entrada digital asociada.

### ***Estado de las subestaciones***

El estado general de las subestaciones siempre está disponible para el operador a través de mnemónicos permanentes que cambian de color de acuerdo a las siguientes condiciones:

- a) Si la subestación está dentro de exploración con todos sus grupos actualizados, según los tiempos de barridos cíclicos programados.
- b) Si la subestación está en exploración, aunque tiene al menos un grupo no actualizado.
- c) Si la subestación está fuera de exploración.
- d) Si la subestación está siendo explorada, no contesta y después de un número de intentos se sometió a un "barrido lento".
- e) Si el puerto donde está la subestación está fuera de exploración.

Además, al recibirse una alarma de la subestación, estos mnemónicos se encienden y apagan intermitentemente.

### ***Desplegado del estado de subestaciones y barridos cíclicos***

Este desplegado indica el estado de las subestaciones: dentro/fuera de exploración, y el estado actual de los grupos de puntos en la UTR con sus respectivas frecuencias de barrido en horas-minutos-segundos. El operador puede modificar los parámetros (dentro/fuera) y la frecuencia de operación de estos grupos.

### ***Desplegado del estado del puerto***

Indica el estado de los puertos del controlador inteligente de comunicaciones CIC: dentro/fuera de exploración y los parámetros de comunicación que pueden modificarse en línea (tiempos de modem CTS [clear to send], para manejo de radio PTT [push to talk], de espera por respuesta remota TOUT, de inversión de canal de transmisión a recepción y viceversa, utilizables en la comunicación por radio, sobre todo en el caso de repetidores). Asimismo, para cada puerto se indican las UTR que tienen asignadas.

## Capítulo 2

# SISTEMA DE REPORTES

Un sistema tan complejo como lo es un SCADA requiere el uso de herramientas específicas de software para operar eficientemente. El sistema requiere informar al usuario el tipo de datos que se está adquiriendo y procesando, para lo cual se utiliza un almacenamiento en BDD en tiempo real.

La adquisición de datos por un sistema SCADA requiere generar *reportes* de acuerdo a la aplicación y el tipo de datos adquiridos, permitiendo realizar reportes específicos del sistema. Estas características le dan gran flexibilidad a los sistemas SCADA utilizándolos no sólo en el área eléctrica, sino que también en diversas áreas de la industria como las mencionadas en el capítulo uno.

Para el sector eléctrico, los reportes informan el funcionamiento del sistema, reportando los valores de la energía suministrada al día, semana, mes o año (voltajes o corrientes), costo promedio, el cambio de estado de interruptores: con la hora, fecha de ocurrencia y causa, el número de operaciones de interruptores clasificadas por tipo (falla, maniobra o licencia) y la corriente suministrada antes de la apertura, y el reporte histórico de las mediciones analógicas del sistema.

La generación de reportes semi-automáticos o automáticos son ahora estándares en el ambiente de los sistemas SCADA. Los reportes son generados al muestrear periódicamente la BDD en tiempo real y generar archivos históricos.

A continuación se da una lista de los datos que son adquiridos por un sistema SCADA (IEE 91):

- DATOS DE DISTRIBUCION:

- Parámetros de datos analógicos
- Balance de alimentadores
- Operación de seccionalización y control
- Operación de switcheo
- Cierre de alimentadores
- Operación de recierre

- DATOS DE SUBESTACION:

- Voltajes de BARRA
- Línea de parámetros analógicos
- Parámetros analógicos del transformador
- Posición del TAP del transformador
- Estado del transformador (auto, manual)
- Cierre y estado de la cuchilla
- Estado del relevador (tarjetas)
- Alarmas de subestación
- Alarmas de transformador (temperatura, presión, diferencial)
- Lectura del medidor en Mw y Mvar-hora
- Estado de protección
- Información SOE

## 2.1. REPORTES DEL SISTEMA SUPERVISORIO

Los sistemas SCADA requieren de herramientas de software específicas para que el operador maneje la información que es adquirida de una manera eficiente y almacenada en BDD, por ello se requiere de diferentes funciones para llevar a cabo el correcto manejo de esta información. A continuación se describen algunas funciones que generan las estructuras de la BDD de reportes y le dan mantenimiento a los mismos.

### 2.1.1. Generador de Reportes de propósito general

Este tipo de herramienta permite la construcción de la estructura de la BDD de reportes, en la cual se definen los registros donde se almacenará la información muestreada periódicamente por este sistema. El nivel de sofisticación de la estructura varía desde un formato fijo, en donde solo se insertan datos dinámicos, hasta un sistema verdaderamente flexible en el cual se soportan modificaciones en la estructura de los reportes, mostrando también los datos dinámicos solicitados. Sin embargo, la generación de algunos reportes depende de la capacidad del sistema general.

### 2.1.2. Generación de Reportes y mantenimiento

La generación de reportes puede ser un evento programado produciéndose *Reportes* específicos a tiempos predefinidos (mes, día, minuto) y/o a intervalos regulares. Otros reportes pueden ser generados al ocurrir un evento determinado (por ejemplo, el cambio de estado de un interruptor particular de abierto a cerrado o viceversa generará en el reporte de Operación de Circuitos que se actualice ese cambio). El método más simple de iniciar un reporte es cuando un operador lo solicita *manualmente*, sin importar la técnica de generación. Los generadores de reportes deben de ser flexibles para la incorporación de datos dinámicos históricos en formatos variables de reportes.

### 2.1.3. Generación de la BDD y edición

La generación de la BDD SCADA se puede dividir en dos procesos principales: la generación y la edición de la BDD.

**Generación de la BDD.** Como norma en la industria se requiere utilizar técnicas semiautomáticas en la generación de la BDD. La generación de un archivo se puede hacer por medio de la conversión de los datos muestreados periódicamente de los sistemas de la red eléctrica enviados a la estación maestra. Esto se procesa fuera de línea por funciones que validan los datos y construyen una BDD la cual se puede cargar al sistema. Los reportes de error producidos automáticamente durante el proceso de generación permiten la depuración de los archivos con la definición de la BDD. Este procedimiento se utiliza generalmente para creación inicial.

**Edición de la BDD.** Una vez que una BDD validada, está instalada y ejecutándose en un sistema, la mayoría de los sistemas SCADA ofrecen características de edición en línea. Estos editores manejan la protección de los segmentos que se están editando, para prevenir interferencia con el sistema ejecutándose. esto permite una operación continua mientras se llevan a cabo las funciones de mantenimiento.

## 2.2. CARACTERISTICAS PRINCIPALES DE UN SISTEMA DE REPORTES

Los reportes deben ser claros y concisos. El sistema de reportes debe seleccionar de manera específica las variables a monitorear y las frecuencias de monitoreo. La información desplegada puede ser en forma gráfica o en forma tabular.

El procesamiento de los reportes no debe representar una sobre carga de procesamiento al sistema en tiempo real, es decir, no debe degradar las funciones prioritarias de supervisión y telecontrol del sistema eléctrico.

La información proporcionada por los sistemas SCADA, puede ser útil a diferentes departamentos de la compañía eléctrica, (distribución, planeación, mantenimiento, producción, etc.). Es conveniente identificar todos los posibles usuarios de manera que toda la información necesaria del sistema en tiempo real se pueda adquirir y procesar.

### 2.3. REPORTES EN LA ESTACION MAESTRA MAC - 5000

En la Estación Maestra MAC - 5000 se realizan un conjunto de *reportes especiales* que, aunque no son vitales para la operación del sistema de control, constituyen una característica adicional muy deseable para análisis del comportamiento de la red y para la parte administrativa de CFE.

La función de los reportes es generar en forma periódica (por hora, día o mes) o a solicitud del operador una serie de reportes del sistema monitoreado por la Estación Maestra.

La estación maestra MAC - 5000 cuenta con un sistema de reportes para información gerencial o comportamiento de la red eléctrica. Los programas residen en una tarjeta MAC-1186 del sistema, la cual también realiza el enlace a disco y a la PC de la estación del ingeniero de sistema, con lo que se logra una independencia del CPU de procesamiento de datos Central de la MAC y de los sistemas encargados de la interfaz hombre-máquina, sin interferir en sus tareas prioritarias.

Tratando de mejorar la generación de estos reportes se traslado su generación a una PC donde se incluye el software necesario para realizarlos accediendo mínimamente la tarjeta de la MAC donde se almacenan estos datos. Esto incluye procesos de conversión de formato de la BDD MAC a una BDD en un formato comercial (dBase), también cuenta con un módulo de reloj para la verificación periódica de la BDD y realizar con esto los reportes periódicamente.

## 2.4. REPORTES ESPECIALES

Los reportes especiales que efectúan actualmente son los siguientes:

Valores Máximos y Mínimos de Voltaje	(VMMV)
Valores Máximos y Mínimos de Corriente	(VMMI)
Consumo Mensual	(COME)
Operación de Interruptores	(OPIN)
Operación de circuitos	(OPERA)
Histórico de mediciones	(HISTORICO)

### 2.4.1. Reporte de Máximos y Mínimos de Voltaje en Barras

Los reportes del valor máximo y mínimo de voltajes en barras de las subestaciones se realizan cada 24 horas automáticamente, los datos son obtenidos por muestras que se toman y analizan cada 15 minutos o de acuerdo al periodo que el operador necesite. El reporte se genera mensualmente por subestación, e indica los valores máximos y mínimos diarios de cada barra, así como la hora que se registro.

Estos reportes son almacenados en disco mensualmente por un periodo de un año, después se va reescribiendo el reporte del mes del año anterior. El reporte se imprime cada mes en forma automática si está programado por el operador para la impresora de reportes. Además, el operador puede solicitar la impresión del reporte actual o de otro mes anterior. Los reportes se ordenan e imprimen por subestación.

### 2.4.2. Reportes Máximos y Mínimos de Corriente en circuitos

Los reportes del valor máximo y mínimo de corriente en circuitos alimentadores de las subestaciones se realizan cada 24 horas. Las muestras se toman y analizan cada 15 minutos o pueden ser cambiados los periodos de acuerdo con las necesidades del operador. Para el cálculo del mínimo no se consideran valores menores al 10% del valor nominal.

Los reportes se almacenan en disco mensualmente por un período de un año, y después se va reescribiendo en el reporte del mes del año anterior. El reporte se imprime en forma automática mensualmente en la impresora de reportes, si está programada (el operador puede solicitar la impresión del reporte actual o de uno anterior). El reporte se ordena e imprime por subestación, informando al operador de los valores máximos y mínimos de cada circuito y la hora en que se registro.

#### **2.4.3. Consumo Mensual**

Este reporte registra diariamente la medición del consumo de potencia de todos los transformadores en KWH y KVARH, así como el día del registro, de todas las subestaciones. Los reportes son diarios, a partir de la inicialización del sistema, y son almacenados durante un año, si la opción de impresión automática esta activa este se imprimirá mensualmente o en forma manual por solicitud del operador.

#### **2.4.4. Reportes de Operación de Interruptores, por falla y por maniobra**

Este reporte proporciona en forma diaria y ordenada por subestación, todos los cambios de estado de los interruptores y la causa. Se indica el número de operaciones por cada interruptor, separándose por el tipo de operación (por falla o por maniobra).

El sistema genera automáticamente un reporte cada 24 horas, el cual contiene el número de operaciones diarias por subestación y la causa de la operación.

#### **2.4.5. Reporte de Operación de Circuitos**

El reporte de operación de circuitos es un reporte por evento que se genera al realizarse el cambio de cada operación de un interruptor, el reporte se manda a imprimir cada 24 horas. Para cada operación, se indica la corriente registrada antes de la apertura, el hora de ocurrencia de la apertura, cierre y la causa. La causa de la apertura se registra por maniobra, falla o licencia de línea desenergizada.

El reporte permite al operador evaluar la cantidad de energía dejada de vender al momento de una apertura.

#### 2.4.6. Reporte de Histórico de Mediciones

Proporciona un historial, durante 24 horas, de todas las mediciones de las subestaciones, monitoreando cada 15 minutos los valores analógicos de la BDD, el total de muestras por día puede ser de 96, si comenzo a las cero horas y termina a las 23:59 horas, pero si el reporte es activo a otra hora del día sólo lleva a cabo el número de muestras que apartir de ese momento y hasta las 23:59 sean realizadas. Este reporte se actualiza diariamente en un archivo general para el número de subestaciones que monitorea la EM, en caso que el reporte este desactivado no se crea el archivo HISTORICO.DBF del día en curso, en caso de estar activado se crea el archivo HISTORICO conteniendo el número de muestras por día para cada subestación.

En la versión anterior el reporte se realizaba para una sola subestación, reescribiéndose en el archivo generado los datos adquiridos diariamente y sólo conservandolos en formatos de impresión. Por lo tanto, uno de los objetivos de diseñar estos reportes en la computadora personal es mejorar el reporte de Histórico de Mediciones haciendo posible la generación de los reportes para todas las subestaciones monitoriadas por el sistema, utiliza un archivo general donde se almacenan las n-ésimas muestras de información para cada una de las subestaciones, las cuales son accedadas posteriormente por otro procedimiento que genera el reporte de una subestación específica de forma gráfica (esto se realiza en el paquete de lotus, ya que el archivo histórico esta en un formato compatible como lo es dBase).

La funciones que se pueden realizar en los seis reportes anteriormente descritos son las siguientes:

- Inicialización
- Actualización de relaciones
- Actualización de reportes

La **inicialización** puede ser de dos tipos: inicialización de las relaciones de la BDD (SUBEST, ANALOGICO, ACUMULADO, DIGITAL, REPORTES, INTE\_SE, OPERA\_CT, BARRA, CORRI\_SUB, HISTORICO) involucradas en la generación del reporte e inicialización del archivo del reporte (VMMV, VMMI, COME, OPIN, OPERA, HISTORICO), el cual es inicializada con cero.

La **actualización** de las *relaciones* se realiza por el muestreo periódico de los datos adquiridos en tiempo real almacenados en una BDD o por evento dependiendo del tipo de reporte. Involucra, en algunos casos, la realización de cálculos sobre el valor de ciertos campos de las relaciones y la actualización de otros. El intervalo entre cada actualización depende del período definido para cada uno de ellos.

La **actualización** del reporte consiste en agregar al archivo de reporte los datos obtenidos de las relaciones muestreadas de la BDD, al cumplirse el tiempo del período de actualización definido.

Para cada reporte se le asigna una variable con el valor del período de muestreo que en la función de reloj se compara con el valor definido en el archivo de REPORTES.DBF, con lo cual se realiza el muestreo periódico, la actualización e impresión automática. Por lo anterior, al efectuar el operador un cambio en el período de muestreo, se activa una función que reasigna a la variable que contiene el valor del período para que el siguiente muestreo sea con el nuevo valor asignado por el operador.

Debido a que la adquisición de datos se relaciona con el tiempo, se tiene un proceso "reloj", encargado de actualizar la hora y fecha de la tarjeta de reportes y que de manera automática realiza la adquisición de los datos de las relaciones de la BDD. Este proceso se activa cada minuto, verificando de forma automática la función que debe realizar cada reporte, pero siempre y cuando este activa, en caso contrario no realiza ninguna operación hasta no ser activado por el operador. Este proceso también verifica si hubo cambios en los parámetros del reporte indicados más adelante, tales como periodos de activación o desactivación.

Uno de los pasos importantes al iniciar el proceso de generación de los reportes es contar con la BDD para realizar el acceso a los datos, de igual manera el operador puede modificar algunos parámetros del reporte en línea y estos son [GUT 90]:

**Tiempo de muestreo.** Es el tiempo requerido para la activación del algoritmo. Está dado en minutos y puede ser de 1 a 1440 (un día = 1440 minutos).

**Tiempo de reporte.** Es el tiempo para actualizar el reporte de forma automática o mandarlo a imprimir. El tiempo puede estar dado en días, o no estar activo indicándose con un cero el que se encuentra desactivado para mandarlo a la impresora.

**Disco y directorio del reporte.** Indica el disco lógico y directorio o usuario donde se guarda la información.

Al disco duro del sistema MAC se le asignan las condiciones iniciales para guardar la información organizada mes por mes con una antigüedad de un año. En el caso del histórico de mediciones, la forma de archivo es diario, con antigüedad de una semana.

## Capítulo 3

# ESPECIFICACION Y DISEÑO DEL SUBSISTEMA DE REPORTES DE LA EM MAC - 5000

Este capítulo contiene la justificación, especificación y el diseño de los programas para realizar los reportes de la estación maestra MAC-5000 en la estación del ingeniero. Se decidió dividir en tres secciones el capítulo para su mejor entendimiento:

Justificación  
Especificación  
Diseño

La *justificación* explica brevemente el uso potencial de estos reportes y el medio donde se realizan indicando las ventajas logradas al desarrollar este sistema.

La *especificación* da a conocer todos los requerimientos necesarios para el desarrollo de los programas del subsistema de reportes.

Por ultimo, la sección de *diseño* describe en forma detallada la estructura de cada módulo del subsistema realizado, utilizando la herramienta gráfica de CASE *System Architect* y la metodología Ward & Mellor [SYS 90], la cual esta orientada al flujo de datos con extensiones para tiempo real.

### 3.1. JUSTIFICACION DE LA REALIZACION DE PROGRAMAS DEL SUBSISTEMA DE REPORTES EN UNA COMPUTADORA PERSONAL

La Estación Maestra MAC-5000 diseñada en el departamento de Electrónica contiene dos tipos básicos de programación, desde el punto de vista de la aplicación:

- Programación para el control y adquisición de datos de las subestaciones eléctricas.
- Programación de Reportes Especiales los cuales dan información global sobre el comportamiento de la red eléctrica supervisada.

Los reportes especiales ejecutados actualmente son los siguientes:

- Valores Máximos y Mínimos de Voltaje (VMMV).
- Valores Máximos y Mínimos de Corriente (VMMI).
- Consumo Mensual (COME).
- Operación de Interruptores (OPIN).
- Operación de Circuitos (OPERA).
- Histórico de Mediciones (HISTORICO).

Los reportes son generados muestreando periódicamente la BDD. El procesamiento se realiza en una de las tarjetas MAC-1186 de la canasta de cómputo de la EM, por medio de comandos el operador puede solicitar su despliegue en las consolas de operador o la impresión de los mismos en la impresora de reportes.

El Histórico de mediciones, se realiza de una manera diferente, para este reporte la EM almacena todos los datos requeridos y cuando el operador o el administrador necesitan visualizarlo, deben extraer los datos y llevarlos a una PC, en donde deben ejecutar un programa especial para el despliegue de la información de una sola subestación en forma gráfica o tabular. La tendencia actual es generar gráficamente este reporte para hacer más eficaz el análisis de la energía suministrada.

Existen varios puntos que justifican la ejecución de los reportes en una Computadora Personal (PC), entre los cuales destacan los siguientes:

La tarjeta MAC-1186, en la cual se lleva actualmente todo el procesamiento de los reportes, está llegando a su límite de capacidad de memoria, tanto para datos como para código, lo cual representa una limitante para ampliar el rango de operación de los reportes ya existentes y para realizar otros reportes que requiera el usuario.

Las capacidades gráficas de las consolas de operador actuales son limitadas, por lo cual reportes como el Histórico de mediciones no se pueden desplegar adecuadamente y es necesario extraer los datos de la MAC-5000 para graficar en una PC.

Sin embargo, para extraer los datos del Histórico de mediciones, es necesario convertir la información del formato CP/M a formato DOS por medio de un gran número de pasos, lo cual desmotiva y desalienta al operador en el uso de este reporte.

De la misma manera se desarrolló un programa mínimo en PC para que el administrador de la MAC-5000 pueda desplegar y/o imprimir cualquiera de los reportes fuera de línea a fin de analizarlos detalladamente, sin estar interfiriendo con las funciones de la EM. Sin embargo, para transportar la información hasta la computadora, también se necesitan ejecutar todos los pasos referentes a la conversión de formato de CP/M a formato DOS, por lo cual esta opción prácticamente no es utilizada por los administradores de la MAC-5000.

Cabe enfatizar, que para que el operador o administrador de la MAC-5000 haga uso del programa mínimo de la PC que existe actualmente, necesitan realizar un gran número de pasos debido a que la operación es totalmente MANUAL y FUERA DE LINEA, ya que no existe una conexión física con la PC y por lo tanto, no existen comandos que permitan transmitir fácil y directamente los datos de los reportes.

### **3.1.1. Ventajas al utilizar un sistema de reportes en una computadora personal**

Las ventajas que se esperan obtener con el desarrollo de esta tesis son las siguientes:

- 1) Facilitar el manejo de los reportes, ya que la conexión directa entre PC y MAC-5000 permite eliminar el uso del sistema operativo CP/M y como se trabajará directamente en formato DOS, se eliminan todos los pasos intermedios que desmotivan el uso de los mismos.
- 2) Al generar los reportes en un formato estándar, permite un mecanismo más eficiente para pasar la información a otros usuarios o computadoras, a fin de revisar y/o analizar más detalladamente cada uno de los reportes.
- 3) Con el traslado de los reportes a la PC, se da el primer paso para eliminar el sistema operativo CP/M de la MAC - 5000, al cual prácticamente le quedaría como única función la del arranque de la EM. Este arranque posteriormente podría implementarse desde la misma PC.
- 4) Similarmente, con el traslado de los reportes a la PC, también se da el primer paso para reducir el uso del el disco duro en la MAC-5000. Al pasar los reportes a la PC, al disco duro solamente le quedarían las funciones de almacenar la base de datos de respaldo, los datos de los digitalizadores y el código de los programas, lo cuales se cargan en memoria RAM al realizar el arranque de la MAC-5000.
- 5) El traslado permite liberar a la MAC-5000 de la carga fuerte de programación y de tiempo de procesamiento que se necesita para la generación de los reportes, concentrando toda su capacidad a una adquisición de datos en tiempo real, rápida y eficiente.

### 3.2. ESPECIFICACION DEL SUBSISTEMA DE REPORTES

Para realizar los reportes especiales que actualmente realiza la estación maestra MAC-5000 en una computadora personal, no es necesario modificar la arquitectura actual de la estación maestra MAC-5000 (Ver figura 3.1) [HID 92].

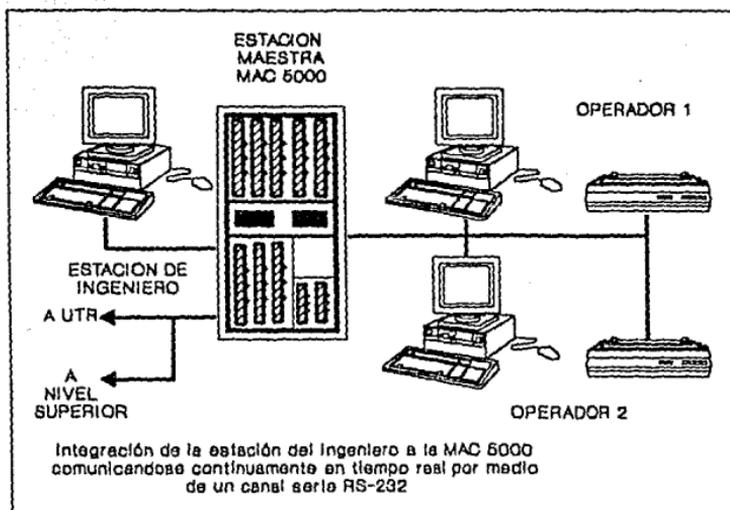


FIGURA 3.1 Arquitectura de Estación Maestra de distribución

Lo que se requiere es modificar la programación que corre en la tarjeta de disco del subsistema de cómputo de la EM MAC -5000, establecer un enlace permanente por medio del puerto serie de la misma tarjeta con la estación del ingeniero y realizar los programas de los reportes especiales en la PC. En la figura 3.2 se muestra la configuración del subsistema de cómputo de la MAC -5000 con el enlace a la estación del ingeniero.

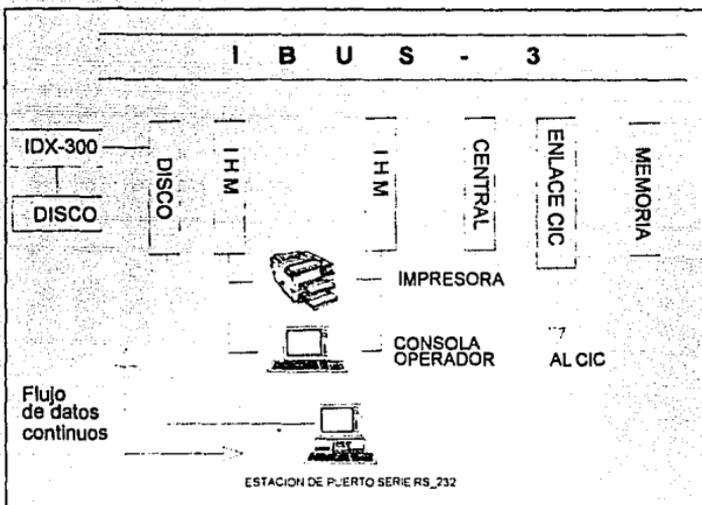


Figura 3.2 Configuración del subsistema de cómputo de la MAC - 5000

Las funciones que se realizan en la computadora personal se pueden agrupar en cuatro bloques principales:

- Inicialización
- Subsistema de enlace PC - MAC
- Subsistema de generación de reportes
- Subsistema de visualización de reportes

El módulo de *inicialización* se encarga de llevar a la computadora personal a un estado apropiado para ejecutar sus funciones.

El módulo de *enlace EM* se encarga de establecer la comunicación entre la PC y la estación maestra MAC-5000. A través de este enlace la MAC-5000 envía periódicamente a la PC las relaciones de la base de datos requeridas por el subsistema de generación de reportes en un formato propietario de la MAC - 5000.

El módulo de *generación* de reportes se encarga de muestrear continuamente la base de datos para generar los reportes de: Valores Máximos y Mínimos de Voltaje en barras, Valores Máximos y Mínimos de Corriente en circuitos, Operación de Circuitos, Operación de Interruptores, Consumo Mensual e Histórico de Mediciones.

El módulo de *visualización* de reportes, por medio de la interfaz del operador, presenta los menús de comandos y los procesa. También se encarga de presentar la información de los reportes en un formato adecuado para el operador, tanto en la pantalla como en la impresora.

A continuación se explica más a detalle cada uno de estos bloques.

### 3.2.1. Inicialización del sistema

El módulo de inicialización se encarga de inicializar tanto los elementos de hardware, como los de software, requeridos para la correcta operación de la computadora personal en la EM.

- Realiza la programación de los dispositivos de hardware requeridos para establecer la comunicación con la estación maestra MAC-5000.
- Ejecuta el diagnóstico inicial del enlace PC - MAC para verificar su funcionamiento.
- Realiza la sincronización inicial de fecha y hora de la PC con la EM.
- Realiza la transferencia inicial de la base de datos en tiempo real de la estación maestra MAC-5000 a la PC.
- Este módulo verifica también la existencia de los subdirectorios y archivos de los reportes, si no existen, los crea.

### 3.2.2. Subsistema de enlace PC - MAC

El subsistema de enlace PC - MAC se encarga de manejar la comunicación entre la PC y la estación maestra MAC-5000.

La información que maneja puede ser de dos tipos: relaciones de la base de datos en tiempo real o mensajes. La información de las bases de datos en tiempo real está en formato MAC, para poder trabajar con esta información en la PC, es necesario que se convierta a formato dBase (este proceso de conversión es explicado detalladamente en el capítulo 4). Los mensajes se envían a otros subsistemas dentro de la PC para su procesamiento.

Las funciones anteriores se realizan mediante los procesos en comunicación PC - MAC y Convierte de MAC a dBase. Ver diagrama 3.1 "Diagrama general de diseño de reportes MAC-PC".

#### *Proceso de Comunicación PC - MAC*

El proceso de comunicación PC - MAC lleva a cabo su función de la siguiente manera:

Implementa el protocolo (PAC) para sincronizar el intercambio de información entre la PC y la estación maestra. El flujo de información es en ambos sentidos.

Identifica el tipo de información que recibe de la MAC - 5000 y la distribuye a los subsistemas correspondientes:

- Las relaciones de base de datos las envía al proceso de conversión de relaciones, dentro del mismo subsistema.
- El mensaje de cambio de estado de interruptor es enviado desde las UTR, las cuales al detectar el cambio de estado (apertura/cierre) de algún interruptor envían el mensaje de cambio, el cual es recibido por el subsistema de enlace y este lo envía al subsistema de generación de reportes donde modifica directamente el archivo del reporte correspondiente a Operación de Circuitos y al reporte de Operación de Interruptores.
- El mensaje de sincronización del reloj calendario es enviado desde el subsistema de enlace y verifica mediante una función el reloj de la EM MAC - 5000 y la PC para que de alguna manera se encuentren en sincronía.

- El mensaje de respuesta de diagnóstico del enlace PC - MAC es enviado desde la MAC a la PC para diagnosticar mediante un código un posible error en el canal de comunicaciones, sino envía un mensaje por medio del subsistema de visualización para que el operador verifique cual es el problema del canal o envíe nuevamente el código de prueba del canal de comunicación.

#### ***Proceso Convierte de MAC a DBase***

Este proceso se encarga de convertir las relaciones de la base de datos en formato MAC a formato dBase y las almacena en el directorio: C:\IMAGEN\RELACIONES.

Las relaciones que envía la MAC a la PC son las siguientes:

- |             |             |            |
|-------------|-------------|------------|
| * ANALOGICO | * DIGITALES | * REPORTES |
| * ACUMULADO | * HISTORICO | * SUBEST   |
| * BARRA     | * INTE_SE   | * TRAN_SE  |
| * CORRI_SUB | * OPERA_CTO | * UTR      |

La conversión a formato dBase, de la mayoría de estas relaciones se realiza sólo durante la inicialización del sistema. Las relaciones de la base de datos en tiempo real que se actualizan constantemente son SUBEST, UTR, ANALOGICO, DIGITALES y ACUMULADO.

### **3.2.3. Subsistema de generación de Reportes**

La generación de reportes es un proceso que se realiza en varias etapas :

- Inicialización del reporte y bases de datos
- Actualización de datos en las relaciones
- Actualización del reporte

La *inicialización* puede ser de dos tipos: inicialización de las relaciones de la BDD involucradas en la generación del reporte e inicialización del archivo de reporte, en el cual se limpian los valores de la información que contiene.

La **actualización de las relaciones** con los datos contenidos en la BDD de tiempo real o por evento en los reportes de operación de circuitos e interruptores. Involucra, en algunos casos, la realización de cálculos sobre el valor de ciertos campos de las relaciones y la actualización de otros.

La **actualización del reporte** consiste en agregar al archivo de reporte los datos obtenidos de las relaciones muestreadas periódicamente de la BDD. Por otra parte, al cumplirse el periodo de actualización definido para cada reporte, se almacenan estos datos en la BDD de los reportes.

Las funciones anteriores se realizan por medio de dos procesos: base de tiempos y generación de reportes. Ver diagrama 3.1 "Diagrama general del diseño de reportes MAC-PC".

#### **Base de Tiempos.**

Este módulo maneja las bases de tiempo para las diferentes etapas de la generación de los reportes.

El módulo obtiene de la relación "*reportes*" los tiempos definidos para el muestreo de las relaciones de la base de datos en tiempo real involucradas con la generación del reportes, siempre y cuando los reportes esten activos de no ser así se inhiere la actualización del reporte automática, el muestreo de las relaciones de la BDD y el comando de impresión automática. El período de muestreo puede ser de 1 a 1440 minutos (24 horas). Al cumplirse el tiempo especificado en la BDD envia un mensaje al proceso de genera reportes para que efectúe la actualización o muestreo de dicho reporte.



De la relación "*reportes*" se toma también el período definido para la impresión automática. Este período es de 0 a 31 días. Al cumplirse el tiempo especificado en la BDD para la activación de impresión se envía un mensaje al subsistema de visualización de reportes para que realice la impresión del reporte. Si es cero significa que está deshabilitada la opción de impresión del reporte automático, de cualquier manera, el reporte puede ser enviado a impresión a petición del operador.

### *Generación de reportes*

Este módulo se encarga de generar los reportes especiales a través de todas sus etapas. Los reportes que genera son:

- Valores máximos y mínimos de voltaje en barras
- Valores máximos y mínimos de corriente en circuitos
- Operación de interruptores
- Operación de circuitos
- Consumo mensual de transformadores
- Histórico de mediciones

A continuación se presenta la descripción detallada de cada uno de ellos.

#### **3.2.4. Valores Máximos y Mínimos de Voltaje en Barras**

El Valor Máximo y Mínimo de Voltaje (VMMV) en barras de las subestaciones se realizan cada 24 horas automáticamente, los datos son obtenidos por muestras que se toman y analizan cada 15 minutos o de acuerdo al período que el operador necesite. Para que los valores sean considerados como máximos o mínimos deben ser mayores al 50% del voltaje nominal. Si el valor de la medición actual es mayor que el valor de V\_MAX, se sustituye V\_MAX en la relación BARRA. En los campos de HR\_MAX, MIN\_MAX y DIA\_MAX se registra la hora y fecha de ocurrencia del máximo, pero si el valor de la medición actual es mayor o igual que el 50% del valor del voltaje nominal de la barra y menor que el valor de V\_MIN, se sustituye V\_MIN. En los campos de HR\_MIN, MIN\_MIN y DIA\_MIN se registra la hora y fecha de la ocurrencia del mínimo.

Por lo anterior, si el valor de la medición actual es menor que el 50% del valor de voltaje nominal de la barra, el valor de V\_MIN se sustituye por cero. Los campos de HR\_MIN, MIN\_MIN y DIA\_MIN también se sustituyen por cero. Para la obtención de la información se utilizan las relaciones SUBBEST, ANALOGICO y BARRA de las cuales se extrae los datos necesarios para construir el reporte VMMV.

El reporte se genera mensualmente por subestación, e indica los valores máximos y mínimos diarios de cada barra, así como la hora que se registro. Estos reportes son almacenados en disco mensualmente por un período de un año, después se va reescribiendo el reporte del mes del año anterior. El reporte se imprime cada mes en forma automática, en la impresora de reportes, si está programado por el operador. Además, el operador puede solicitar la impresión del reporte actual o de otro mes anterior. Los reportes se ordenan e imprimen por subestación.

Los reportes se generan en el directorio C:\VMMV, que tiene doce subdirectorios correspondientes a cada mes del año. En cada subdirectorio de mes se crea un archivo por cada subestación con el nombre de la subestación y la extensión DBF, el cual contiene toda la información necesaria para dicho reporte.

Además, se genera un archivo llamado ACTVMMV.DBF que contiene la fecha y hora de inicio y terminación de reporte, válida para todas las subestaciones.

Para la generación de estos reportes se requiere desarrollar las siguientes funciones:

### *Inicialización*

Para el funcionamiento correcto del reporte VMMV, al arrancar el sistema o iniciar el mes, se verifica si los archivos correspondientes a cada subestación existen en el subdirectorio adecuado al mes actual, de no existir los crea.

Se inicializa la relación "barra" cada 24 horas con los siguientes valores:

<i>Campo</i>	<i>Valor</i>
V_MAX, V_MIN	Medición actual del punto analógico asociado a la barra, al momento de la inicialización.
HR_MAX, HR_MIN	hora en que se realiza la inicialización.
MIN_MAX, MIN_MIN	minuto en que se realiza la inicialización
DIA_MAX, DIA_MIN	fecha en que se realiza la inicialización.

#### *Actualización de Relaciones*

Para realizar esta función se requiere obtener de la base de tiempos el intervalo de tiempo de muestreo el valor de la medición actual del punto analógico asociado, comparandose con los valores de los campos V\_MAX y V\_MIN de la barra.

Si el valor de la medición actual es mayor que el valor de V\_MAX, se sustituye V\_MAX en la relación BARRA. En los campos de HR\_MAX, MIN\_MAX y DIA\_MAX se registra la hora y fecha de ocurrencia del máximo.

Si el valor de la medición actual es mayor o igual que el 50% del valor del voltaje nominal de la barra y menor que el valor de V\_MIN, se sustituye V\_MIN. En los campos de HR\_MIN, MIN\_MIN y DIA\_MIN se registra la hora y fecha de la ocurrencia del mínimo.

Si el valor de la medición actual es menor que el 50% del valor de voltaje nominal de la barra, el valor de V\_MIN se sustituye por cero. Los campos de HR\_MIN, MIN\_MIN y DIA\_MIN también se sustituyen por cero.

#### *Actualización de Reporte*

La función de actualiza reporte se realiza cada 24 horas, exportando la información de los valores máximo y mínimo de voltaje del día, la hora y fecha de ocurrencia contenida en la relación BARRA a el reporte correspondiente.

Los archivos de reporte de máximos y mínimos de voltaje en barras está en formato DBase. La información que contiene está organizada secuencialmente por día, con el nombre de la barra, el valor máximo y mínimo de voltaje registrados ese día y la hora de ocurrencia. Los campos asociados a estos datos son de tipo texto para que el subsistema de visualización de reportes los despliegue sin realizar ningún tipo de procesamiento. El formato para los datos en el archivo de reporte es el siguiente:

<i>Campo</i>	<i>Tipo de campo</i>
NOMBRE DE LA BARRA	NNNNNNNN
VOLTAJE MAXIMO	VALOR (en KV)
HORA DEL VALOR MAXIMO	HH:MM
DIA	DD
VOLTAJE MINIMO	VALOR (en KV)
HORA DEL VALOR MINIMO	HH:MM

### 3.2.5. Valores Máximos y Mínimos de Corriente en circuitos

En el reporte de Valores Máximos y Mínimos de Corriente (VMMI) los valores medidos de los circuitos se almacena en la subestación correspondiente donde son agrupados todos los circuitos correspondientes, almacenando también la hora y fecha de ocurrencia de la lectura. Con estos registros diarios se genera un reporte mensual que se conserva durante un año.

Los reportes se generan en el directorio C:\VMMI, que tiene doce subdirectorios correspondientes a cada mes del año. En cada subdirectorio de mes se crea un archivo por cada subestación con el nombre de la subestación con extensión DBF, el cual contiene la información de cada subestación. La generación del reporte VMMI requiere de las relaciones SUBEST, CORRI\_SUB y ANALOGICO de las cuales se obtiene la información necesaria para la realización del reporte.

Además, se tiene un archivo llamado ACTVMMI.DBF que contiene la fecha y hora de inicio y terminación de reporte, válida para todas las subestaciones durante el periodo indicado.

Para la generación del reporte VMMI se lleva a cabo una serie de etapas que a continuación son descritas:

### *Inicialización*

Esta función verifica al arrancar el sistema o iniciar el mes, la existencia de los archivos con los nombres de cada una de las subestaciones con extensión DBF en el subdirectorio adecuado al mes actual, de no existir los crea.

Se inicializa la relación "corri\_sub" cada 24 horas con los siguientes valores:

<i>Campo</i>	<i>Valor</i>
I_MAX, I_MIN	Medición actual del punto analógico asociado al circuito, al momento de inicialización
HR_MAX, HR_MIN	Hora en que se realiza la inicialización
MIN_MAX, MIN_MIN	Minuto en que se realiza la inicialización
DIA_MAX, DIA_MIN	Fecha en que se realiza la inicialización

### *Actualización de Relaciones*

Esta función se realiza al cumplirse el tiempo definido en la base de tiempos del archivo de reportes.dbf el cual ejecuta la actualización, verificando el valor de la medición de campo del punto analógico asociado al circuito. Si es mayor o igual que 66H se toma el valor de la medición actual, en unidades de ingeniería, del punto analógico asociado. Si es menor que 66H se asume que es de 66H y se calcula el valor de la medición actual en unidades de ingeniería:

$$\text{medición actual} = \text{medición de campo} \times \text{escala}/2048 + \text{offset}$$

El valor de la medición actual se compara con los valores de los campos I\_MAX e I\_MIN del circuito:

Si el valor de la medición actual es mayor que el valor de I\_MAX, se sustituye I\_MAX. En los campos de HR\_MAX, MIN\_MAX y DIA\_MAX se registra la hora y fecha de ocurrencia del máximo.

Si el valor de la medición actual es menor que el valor de I\_MIN, se sustituye I\_MIN. En los campos de HR\_MIN, MIN\_MIN y DIA\_MIN se registra la hora y fecha de la ocurrencia del mínimo.

#### *Actualización de Reporte*

El reporte VMMI actualiza los datos obtenidos a lo largo del día cada 24 horas, utilizando la información almacenada en la relación "corri\_sub", los valores registrados como máximo y mínimo del día, la hora y fecha de ocurrencia se añade al archivo del reporte VMMI. Una vez terminada la actualización se procede a inicializar la relación "corri\_sub" y se continúa el ciclo.

Los archivos del reporte VMMI está en un formato DBase. La información que contiene está organizada secuencialmente por día, con el nombre del circuito, el valor máximo y el valor mínimo de corriente registrados ese día y la hora de ocurrencia.

Los campos asociados a estos datos son de tipo texto para que el subsistema de visualización de reportes los despliegue sin realizar ningún tipo de procesamiento.

El formato para los datos en el archivo de reporte es el siguiente:

<i>Campo</i>	<i>Tipo de campo</i>
NOMBRE DEL CIRCUITO	NNNNNNNN
CORRIENTE MAXIMA	VALOR (en AMP)
HORA DEL VALOR MAXIMO	HH:MM
DIA	DD
CORRIENTE MINIMA	VALOR (en AMP)
HORA DEL VALOR MINIMO	HH:MM

### 3.2.6. Operación de interruptores

Este reporte lleva un registro diario de todas las operaciones realizadas sobre los interruptores en campo. Se contabiliza por separado el número de operaciones ocurridas por falla del interruptor en la subestación o por maniobra del operador, registrándose en el reporte el número de fallas y maniobras de los interruptores específicos de cada subestación. Esta información se agrupa por subestación. Con los registros diarios se genera un reporte mensual el cual se conservará durante un año.

Los reportes se generan en el directorio C:\OPIN, que tiene doce subdirectorios correspondientes a cada mes del año. En cada subdirectorio de mes se crea un archivo por cada subestación con el nombre de la subestación con extensión DBF que contiene el reporte propiamente dicho. La información se obtiene de las siguientes relaciones SUBEST, INTE\_SE, DIGITAL Y ACUMULADO.

Además, se cuenta con un archivo llamado ACTOPLDBF que registra la fecha y hora de inicio y terminación de reporte, válida para todas las subestaciones. Para la realización de los reportes es necesario efectuar las siguientes etapas:

#### *Inicialización*

Esta función se ejecuta al poner en funcionamiento el sistema por primera vez o iniciar el mes, revisando si los archivos con el nombre de cada una de las subestaciones estan creadas en el subdirectorio correspondiente del mes actual, en caso contrario lo crea, para ir registrando las operaciones realizadas en los interruptores correspondientes.

Se inicializa la relación "inte\_se" cada 24 horas con los siguientes valores:

<i>Campo</i>	<i>Valor</i>
DIA	con la fecha en que se realiza la inicialización.
FALLAS	0
MANIOBRAS	0

### *Actualización de Relaciones*

Esta función se realiza cada vez que ocurre un cambio de estado en algún interruptor, por lo cual se genera un mensaje que es enviado de acuerdo a la causa del cambio de estado, es decir si es maniobra se incrementa el número registrado en el campo *maniobra* de la relación INTE\_SE en caso de no ser una maniobra, incrementa solo el contador del registro de la relación ACUMULADO siendo en este caso una falla, con lo que posteriormente se actualiza el reporte de Operación de Interruptores (OPIN). Una operación se considera como maniobra del operador cuando se confirma un telecontrol.

Se tiene un contador total de operaciones del interruptor que es actualizado cada veinticuatro horas por un proceso interno de la MAC - 5000 llamado "adquisición del grupo de acumuladores y contadores", este contador resulta útil dado que las operaciones por falla se obtienen de la diferencia del contador total de operaciones y las operaciones por maniobra.

### *Actualización del reporte*

La actualización del reporte se hace cada 24 horas. Se obtiene de la relación "inte\_se" el nombre del interruptor, la cantidad de operaciones por falla, la cantidad de operaciones por maniobra y el día del registro y se añade esta información al reporte. Se procede a inicializar de nuevo la relación "inte\_se" y continua la operación.

El archivo del reporte de operación de interruptores está en formato DBase. La información que contiene se registra secuencialmente por día, con el nombre del interruptor, la cantidad de operaciones por falla, la cantidad de operaciones por maniobra del operador y el día del registro.

Los campos asociados a estos datos son de tipo texto para que el subsistema de visualización de reportes los despliegue sin realizar ningún tipo de procesamiento.

El formato para los datos del archivo de reporte es el siguiente:

<i>Campo</i>	<i>Tipo del campo</i>
NOMBRE DEL INTERRUPTOR	NNNNN
FALLAS	FF
MANIOBRAS	MM
DIA	DD

### 3.2.7. Operación de Circuitos

Este reporte se realiza por evento, es decir, al recibir un mensaje que indica que un circuito tuvo una operación de apertura o cierre, se registra directamente en el archivo correspondiente con el nombre de la subestación a la que pertenece ese circuito, en el archivo se registra la hora de cierre o apertura, la corriente antes de la apertura y la causa de la apertura, que puede ser por falla o por maniobra. Con estos registros diarios se genera el reporte mensual Operación de Circuitos (OPERA) que se conserva durante un año.

Los reportes se generan en el directorio C:\OPCI, que tiene doce subdirectorios correspondientes a cada mes del año. En cada subdirectorio de mes se crea un archivo por cada subestación con el nombre de la subestación con extensión DBF el cual contiene el reporte. Para la generación de este reporte es necesario obtener algunos datos de las siguientes relaciones: SUBEST, ANALOGICO, OPERA\_CT y DIGITAL. Además, se tiene un archivo llamado ACTCTO.DBF que contiene la fecha y hora de inicio y terminación de reporte, válida para todas las subestaciones.

Para la generación del reporte de Operación de Circuitos es necesario seguir ciertos pasos los cuales se describen a continuación:

#### *Inicialización*

Al recibir por primera vez el mensaje de cambio de estado de algún interruptor, se crea un archivo en el subdirectorio correspondiente al mes actual y se almacena la información directamente en el reporte de Operación de Circuitos.

Se inicializa la relación "opera\_cto" cada 24 horas con los siguientes valores:

<i>Campo</i>	<i>Valor</i>
EST TUPLE	vacio (1)

### *Actualización del reporte*

La actualización del reporte se hace por evento, cada vez que se recibe un mensaje de cambio de estado de interruptores se verifica si fue cierre o apertura de circuito. En caso de cierre se registra directamente en el reporte el nombre del circuito y la fecha y hora de cierre; en la relación "opera\_cto" se marca el campo est\_tuple como vacío. En caso de apertura se registra en el reporte el nombre del circuito, la fecha y hora de apertura, la causa y la corriente se obtiene de la relación de DIGITAL antes de la apertura; en la relación "opera\_cto" se marca el campo est\_tuple como no vacío.

El archivo del reporte de operación de circuitos está en formato DBase. La información que contiene se registra secuencialmente por día, con el nombre del circuito, la hora y fecha de cierre, la hora y fecha de apertura, la corriente medida antes de la apertura y la causa de la apertura. Los campos asociados a estos datos son de tipo texto para que el subsistema de visualización de reportes los despliegue sin realizar ningún tipo de procesamiento.

El formato para los datos es el siguiente:

<i>Campo</i>	<i>Tipo de campo</i>
NOMBRE DEL CIRCUITO	NNNNN
HORA DIA A	HH:MM DD
HORA DIA C	HH:MM DD
CORRIENTE	VALOR AMP
CAUSA	FAL, MAN

### 3.2.8. Consumo Mensual de Transformadores

En este reporte se registra diariamente la medición del consumo de potencia de todos los transformadores en KWH y KVARH, así como el día del registro, agrupados por subestación. Con estos registros diarios se genera un reporte mensual que se conserva durante un año, para conocer el consumo diario de estos transformadores. En la generación de este reporte se utilizan las relaciones SUBEST, TRAN\_SE y ACUMULADO para obtener la información necesaria al elaborar este reporte.

Los reportes se generan en el directorio CACOME, que tiene doce subdirectorios correspondientes a cada mes del año. En cada subdirectorio de mes se crea un archivo por cada subestación con el nombre de la subestación y la extensión DBF que contiene el reporte propiamente dicho.

Además, se tiene un archivo llamado ACTCM.DBF que contiene la fecha y hora de inicio y terminación de reporte, válida para todas las subestaciones. Para la realización del reporte de COME se deben llevar a cabo las siguientes etapas:

#### *Inicialización*

Esta función se realiza al arrancar el sistema por primera vez o iniciar el mes, para la verificación de la existencia de los archivos con el nombre de cada una de las subestaciones en el subdirectorio correspondiente al mes actual, de no ser así los archivos son creados.

Además, se inicializa la relación "tran\_se" cada veinticuatro horas con los siguientes valores:

<i>Campo</i>	<i>Valor</i>
DIA	con la fecha de inicialización
VALOR DIA	0

#### *Actualización de Relaciones*

El intervalo de actualización de las relaciones involucradas en este reporte es de una hora. Este valor es fijo y no puede ser modificado por el usuario.

Los valores del consumo de potencia del transformador en KWH y en KVARH son actualizados cada hora en la relación "acumulado" por un proceso interno de la MAC-5000 el cual obtiene el valor desde campo y lo almacena en la relación ACUMULADO.

A su vez, tanto los valores de potencia en KWH, como los de KVARH se muestrean cada hora y se suman con el valor del anterior muestreo en la relación TRAN\_SE y a las 24 horas son actualizados en el reporte el cual se conserva durante un mes.

#### *Actualización de Reporte*

El reporte COME es actualizado cada 24 horas, obtiene de la relación "tran\_se" los valores de consumo de los transformadores en KWH y KVARH y el día del registro y se añade esta información al reporte. Se procede de nuevo a inicializar la relación "tran\_se" y continúa el proceso.

Los archivos del reporte de consumo mensual de transformadores está en formato DBase. La información que contiene está organizada secuencialmente por día, con dos registros por transformador.

El primer registro con el nombre del transformador, el valor de la medición de consumo de potencia en el transformador en KWH y el día de la medición.

El segundo, con el nombre del transformador, el valor de la medición de consumo de potencia en el transformador en KVARH y el día de la medición. Los campos asociados a estos datos son de tipo texto para que el subsistema de visualización de reportes los despliegue sin realizar ningún tipo de procesamiento.

El formato para los datos del archivo de reporte es el siguiente:

<i>Campo</i>	<i>Tipo de campo</i>
NOMBRE DEL TRANSFORMADOR	NNNNN
MEDICION	MMMM
UNIDAD	KWH, KVARH
DIA	DD

### 3.2.9. Histórico de mediciones

En este reporte se registran los valores muestreados a intervalos programables durante el día en todos los puntos analógicos, agrupados por subestación. Se pueden almacenar un máximo de 96 mediciones diarias por punto analógico. Estos registros diarios se conservan por un número de días ya que esta limitado a la capacidad de memoria del disco que se defina para este reporte. Cuando se termina el espacio en el disco, se reescribe sobre la información más antigua.

Los reportes se almacenan en el directorio CAHISTORICO, en el cual se crea un subdirectorío por cada día de la semana que se registra. En cada subdirectorío se crea un archivo que contiene todos los puntos analógicos con sus 96 muestras, este archivo almacena las 96 muestras para todos los puntos analógicos de todas las subestaciones. El nombre del reporte en el subdirectorío es HISTORICO.DBF. En la realización de este reporte se utilizan las relaciones SUBEST, HISTORICO y ANALOGICO.

Además, en cada subdirectorío de día se tiene un archivo llamado ACTHST.DBF que contiene la fecha y hora de inicio y terminación de reporte y el intervalo de actualización del reporte.

#### *Inicialización*

Se crea un archivo que contiene el identificador de cada una de las subestaciones, las unidades de ingeniería, los decimales, el punto analógico, el nombre del punto analógico y los 96 campos correspondientes a las 96 muestras. Este archivo tiene extensión DBF y se encuentra en el subdirectorío correspondiente al día actual.

El archivo del reporte histórico de mediciones está en formato DBase. La información que contiene registra el nombre del punto analógico y los valores obtenidos en él durante el día. Los campos asociados a estos datos son de tipo texto para que puedan ser procesados por otros paquetes de software, como es el Lotus que mediante su función de graficación por tabulares, puede obtener un esquema más detallado de las mediciones de los puntos analógicos realizadas para el histórico de mediciones.

El formato para los datos del archivo de reporte es el siguiente:

<i>Campo</i>	<i>Tipo de campo</i>
IDENTIFICADOR DEL PUNTO	NNNNN
UNIDADES DE INGENIERIA	NNNNN
DECIMALES	NNNN
NOMBRE DEL PUNTO ANALOGICO	NNNNN
VALOR NN	VALOR UI

### 3.2.10. Subsistema de visualización

Este subsistema de visualización es el cargado de presentar en pantalla, menues que desplieguen las operaciones que puede realizar el sistema para que el manejo sea lo más sencillo posible, por lo que cuenta con una interfaz gráfica la cual contiene cinco botones que mediante el rellenarse de color rojo y una señal audible (bip) le avisan al operador que se esta procesando la información de muestreo.

Se cuenta con un sistema que simula la obtención de los datos de la BDD la cual se utiliza para generar los seis reportes que se requieren. Esta demostración servirá para facilitar al usuario la comprensión de como esta estructurado el sistema y para pruebas de funcionalidad de los reportes.

### 3.3. DISEÑO DEL SUBSISTEMA DE REPORTES DE LA EM MAC - 5000 EN COMPUTADORA PERSONAL

El diseño del subsistema de reportes en la computadora personal se describe como una serie de pasos en los cuales la representación del flujo de datos, los diagramas de procesos y procedimientos son obtenidos de las especificaciones del sistema, dando la información necesaria para su interpretación.

Es importante hacer hincapie de que se utilizó una herramienta gráfica de CASE

(Computer Aided Software Engineering) llamada *SYSTEM ARCHITECT* en la elaboración de los diagramas, ajustándose a la metodología "*Data Flow Ward & Mellor*" [SYS 90] para el diseño de sistemas de tiempo real lo cual define las diferentes etapas que transforman el flujo de información en estructura de módulos que pueden ser refinados mediante un pseudocódigo y posteriormente en programas.

En el diagrama 3.1 se aprecian los cinco procesos principales numerados como P1 a P5 los cuales definen el diagrama principal del Diseño del subsistema de reportes, así como también las interacciones de flujo de datos entre ellos, mostrando los mensajes transferidos a los procesos correspondientes y la forma en que estos accesan las dos bases de datos (BDD PC en tiempo real y REPORTES). La herramienta *SYSTEM ARCHITECT* [SYS 90] tiene la flexibilidad de permitir la definición de cada elemento del diagrama según el tipo al que pertenezca y generar un reporte con esta información, es decir cuenta con un sistema de reportes que proporcionará las descripciones de cada uno de los procesos del diagrama principal, como de los submódulos pertenecientes a cada proceso y que también describen cada uno de sus elementos.

Los procesos se representan en el diagrama por medio de un círculo en el cual se especifica el número de proceso y el nombre del mismo, tres círculos más pequeños en la parte superior con el círculo del centro relleno indican que el proceso se expande a otro diagrama, cada proceso cuenta con una descripción que define su función.

Las líneas continuas indican flujo de datos, las cuales tiene una pequeña descripción de la información que les es transferida y son simbolizadas en el diccionario del SYSARCH como comentarios, los cuales son representados por medio de tres círculos pequeños con el primero de ellos relleno. Estos flujos se describen en el reporte de "Definición Flujo datos" que se encuentra definido en el nivel 1 de nuestro diseño (diagrama 3.1).

Las estructuras de datos forman lo que llamamos la Base de Datos del sistema, estas estructuras estan definidas formalmente en un almacen de datos como estructuras de datos que contienen elementos, cada elemento a su vez se describe por medio de una definición, el tipo de dato y su longitud. Las dos Bases de datos son definidas, una externamente en los documentos de la EM MAC - 5000 y la otra mediante las utilerías del paquete SYSARCH para la definición de estructuras de datos; estas descripciones son

presentadas en el reporte "Definición de Estructura de datos" que contiene el nombre de la estructura y los atributos que la forman.

Los reportes generados por *SYSTEM ARCHITECT* pueden ser modificados por el usuario para facilidad de ajustarse a las necesidades del diseño a realizar, están contenidos en el archivo "REPORTES.RPT" que se incluye en el paquete en el diccionario de datos.

En los diagramas de estructura el rectángulo con un rombo en la parte central inferior se conoce como centro de transacción e indica la función a llevarse a cabo mediante toda la estructura jerárquica. Se utilizo la técnica TOP-DOWN en el desarrollo de los diagramas, las flechas indican llamadas a otros módulos y la estructura total debe leerse de izquierda a derecha, existen varios diagramas que además contienen un flujo de datos interno al mismo programa que se simboliza mediante una flecha saliendo de un círculo, la dirección de la flecha indica el sentido del flujo de los datos, ver apéndice B, para una mayor descripción de la simbología dentro de esta metodología. En el diagrama 3.1 se muestra el nivel 1 correspondiente al diseño del Subsistema de Reportes.



La definición de cada uno de los elementos del diagrama se encuentra en el reporte llamado "Definiciones del diagrama general del diseño de Reportes MAC - PC" elaborado con el sistema de reportes de la herramienta *System Architect* y del cual se muestra su contenido.

### 3.3.1. Definiciones de los procesos en el diagrama general (nivel 1).

Para este punto se presentan todas las definiciones de los procesos utilizados en el diagrama principal del diseño del subsistema de reportes, representado en el diagrama 3.1.

#### *Cargado de BDD a la MAC*

Es el encargado de obtener la base de datos generada fuera de línea por el MBAD\_FL y que es almacenarla en la MAC-5000 en formato MAC.

#### *Comunicación PC - MAC*

Es el encargado de atender y diagnosticar el canal de enlace entre la PC y la EM MAC - 5000. Al recibir un mensaje de la EM, determina a que proceso debe de ser enviado. Las relaciones dinámicas se envían al proceso de conversión y el mensaje de cambio de estado de un interruptor se envía directamente al proceso de generación de reportes. En caso de detectar falla en el canal de comunicación, éste manda un mensaje de aviso al proceso de interfaz de usuario, para que se despliegue el mensaje correspondiente en pantalla.

#### *Convierte de MAC a dBase*

Este proceso cambia el formato MAC que está definido por medio de estructuras, llamadas relaciones, que sirven como referencia para crear los archivos correspondientes. Este proceso adquiere las relaciones uno a uno de las estructuras Meta\_r y el Meta\_a y con estos datos va creando los archivos en un formato comercial (dBase III plus).

### *Genera Reportes*

Este proceso se encarga de crear, actualizar, inicializar y muestrear los REPORTES a partir de la información historica almacenada en la base de datos .

#### *Inicia*

Proceso que se encarga de la inicialización del sistema, la cual incluye las siguientes tareas:

- Inicializa puerto de enlace
- Recibe señal de puerto listo
- Inicializa semaforos de comunicación
- Inicializa Tablas BDD de MAC

Después de inicializar el sistema este proceso ya no se utiliza.

#### *Interfaz de usuario*

La interfaz de usuario es la encargada de manejar todos los comandos disponibles al operador , para accesar los reportes. Alguno de los comandos son los siguientes:

- Desplegado en pantalla (gráficos, texto, etc) de los reportes.
- Impresión de reportes.
- Formato de reportes.
- Cambio de frecuencia de muestreo.
- Activa/Desactiva generación de reportes periódicos.

#### *MAC*

El sistema de control supervisorio MAC - 5000 mantiene actualizada en su base de datos (RAM) todos los estados de los puntos adquiridos de campo. Esta información es la que se envia periódicamente a la PC para realizar los reportes.

### ***MBAD\_FL***

El sistema de base de datos fuera de línea (MBAD\_FL) ofrece al usuario las herramientas para la definición de estructuras, captura de datos, emisión de reportes y conversión de formato de acuerdo a lo requerido por el sistema en tiempo real en la EM MAC - 5000.

### ***Reloj BDD DBF***

El proceso de reloj es el encargado de controlar la generación de los reportes ya sea manualmente o de manera automática, verifica si el reporte esta activado o no, con lo cual puede enviar mensajes para obtener algún reporte, esto es, enviar un mensaje al proceso Genera Reportes para que active los reportes, también puede enviar un mensaje para la impresión automática en función de la información que se encuentra en la BDD imagen, obtener datos en un tiempo periodico predefinido (muestreo). El tiempo de activación de cada proceso puede ser modificado en línea, y el proceso reloj se ajusta a estos nuevos tiempos.

### ***Visualiza***

Este proceso se encarga de presentar en pantalla o impresora los REPORTES solicitados por el operador, además manda el mensaje de solicitud de impresión automática.

### ***Operador***

Es la persona encargada de utilizar el Manejador de la Base de Datos fuera de línea (MBAD\_FL) para almacenar los datos que componen el sistema a supervisar.

### 3.3.2 Definición de flujo de datos en el diagrama general (nivel 1)

En el diagrama 3.1 se muestra el diseño general del sistema, las líneas continuas indican flujo de datos, interpretados en nuestro diseño como mensajes de un módulo a otro, estos símbolos están definidos como un comentario dentro de la herramienta utilizada y explican su función:

#### *Adquiere relaciones*

Este mensaje obtiene de la BDD la información necesaria de las relaciones que utiliza el proceso de Genera Reportes para realizar el muestreo y la actualización específica de los reportes a generar.

#### *Comandos PC*

El operador del sistema puede realizar ciertos comandos que son atendidos por el proceso interfaz de usuario. Los comandos que puede solicitar son los siguientes:

1. Inicia programa
2. Manda mensaje de inicialización de la BDD.
3. Muestra menues en pantalla para la elección de reportes.
4. Regresa a la pantalla los datos de selección y queda en espera de que el operador elija nuevamente.
5. Puede cambiar la frecuencia de muestreo de la BDD y fecha.
6. Permite al operador generar la impresión de un reporte específico de forma directa.

#### *Error en el canal de comunicación*

Mensaje que envia el proceso de comunicación PC - MAC para informar la pérdida de comunicación entre la PC y la MAC, generando un mensaje en pantalla por medio de la interfaz de usuario para que el operador intente la conexión del canal nuevamente.

### ***Información de muestreo***

Este mensaje obtiene la frecuencia de muestreo de cada uno de los reportes, y la utiliza para actualizar los valores de las relaciones de la BDD para la generación de reportes específicos. Es cargada inicialmente de la BDD que contiene estos tiempos.

### ***Información de tiempo real***

La información de tiempo real que envía la MAC-5000 es la siguiente:

- 1.- Relaciones dinámicas  
ANALOGICO  
DIGITAL  
ACUMULADO  
UTR  
SUBEST  
REPORTES
- 2.- Mensaje de estado de cambio de interruptor
- 3.- Mensaje de sincronización del reloj calendario
- 4.- Respuesta de diagnóstico del enlace PC-MAC

### ***Inicializa***

Comando que indica al módulo "Genera reportes" que la base de datos de la PC debe inicializarse al finalizar el día, después de haber almacenado la información actualizada en la base de datos de REPORTES.

### ***Mensaje de cambio***

Mediante este comando, generado por el operador, se indica desde la IHM alguna acción para modificar los parámetros de generación de reportes. Los mensajes de cambio son los siguientes:

**1.- Cambio de la frecuencia de muestreo de la base de datos**

Este mensaje de cambio afecta directamente la relación de la BDD Reportes.DBF y modifica los campos de activación o de frecuencia.

**2.- Cambio de hora y fecha**

Este mensaje modifica la hora y fecha del reloj de la computadora directamente.

***Mensaje interruptores***

Este mensaje es generado cada vez que existe un cambio de estado en algún interruptor y es enviado directamente al subsistema de generación de reportes para actualizar de forma inmediata el cambio sufrido por el circuito, en el reporte OPERA. El mensaje de interruptor esta definido de la siguiente manera:

<b><i>ESTRUCTURA</i></b>	<b><i>TIPO</i></b>
Longitud	del word
mensaje	
Función	word
Error	word
n_msje	byte
proc	byte
n_pto	word
edo_fun	word
edo_vis	word

***Petición de reporte por usuario***

El operador puede solicitar en cualquier momento la visualización del contenido de un reporte, por lo tanto el proceso de interfaz de usuario envía la petición de reporte al subsistema de visualización para que se muestre inmediatamente.

## **REPORTE**

Este mensaje obtiene la información específica de los archivos de la BDD REPORTES, y son los siguientes:

- Reporte VMMV
- Reporte VMMI
- Reporte OPIN
- Reporte COME
- Reporte OPERA
- Reporte HISTORICO

### **Relaciones REPORTES**

Envía la información adecuada para la actualización o creación de algún reporte específico, como son:

VMMV	Valores Máximos y Mínimos de Voltaje
VMMI	Valores Máximos y Mínimos de Corriente
OPIN	Operación de Interruptores
OPERA	Operación de Circuitos
COME	Consumo Mensual
HISTORICO	Reporte historico

### **Relaciones dinámicas**

Las relaciones que son accedadas frecuentemente por el subsistema de reportes, y que necesitan refrescar su información en periodos fijos para la correcta operación son los siguientes:

ACUMULADO	Cuentas de la operación de interruptores y consumo de energía.
ANALOGICO	Valores analógicos obtenidos por la UTR que indican el valor de ese tipo de variables en la subestación.

REPORTES	Contiene las variables del periodo de muestreo y activación de los reportes.
DIGITAL	Valores digitales obtenidos por las UTR que indican el estado de dispositivos monitoreados por la subestación.
SUBEST	Contiene información de las subestaciones.
UTR	Estado y configuración de las UTR en el sistema.

### *Relaciones formato MAC*

Por este mensaje circulan los archivos generados por MBAD-FL con el formato MAC, los cuales son cargados del disco duro de la MAC - 5000, y trasladados al proceso que se encarga de realizar la conversión. La descripción de las relaciones y sus atributos se definen en las estructuras META\_A y META\_R, las cuales se encuentran al inicio del mensaje.

### *Relaciones modificadas*

Este mensaje permite obtener la información que es enviada al proceso de "Genera Reportes" el cual lee la información actualizada en las relaciones DBF de la BDD, y permite la elaboración de los reportes periódicos; durante todo momento esta controlado por el proceso "Reloj" y por las peticiones requeridas en la interfaz de usuario.

### *Solicitud REPORTE*

Este mensaje es enviado periódicamente por el proceso de "Reloj" para avisar al proceso de "Genera Reporte" que es necesario actualizar y/o muestrear las relaciones específicas que se utilizan para generar los reportes posteriormete. Esta solicitud se realiza cuando el proceso "reloj" determina que se cumplió el tiempo de espera para la generación de reportes.

**Visualiza reporte periódico**

Comando que se genera cada fin de mes de manera automática por medio de un mensaje que envía el proceso de "Reloj" para la impresión de los reportes, el tiempo de solicitud de impresión puede ser modificado de manera manual por petición del operador.

**3.3.3 Definición de estructuras de datos utilizadas en el diagrama general (nivel 1)**

Las estructuras de datos forman a la base de datos del sistema y están definidas formalmente como un almacén de información, los elementos que la constituyen son descritos en este punto, y son definidos por el tipo de dato y su longitud; la estructura se definió externamente y las descripciones de las estructuras son definidas en el reporte obtenido de la herramienta CASE SYSARCH. El reporte se denominó "Definición de estructuras de datos" y contiene la siguiente información de la BDD y la BDD REPORTES:

**Estructura de la relación TRAN\_SE**

Se definen los elementos de la relación TRAN\_SE para almacenar el valor del consumo de energía de la red eléctrica, por lo cual se define la siguiente tabla con el nombre de los elementos que son utilizados, el tipo de dato (carácter, entero) y la definición del elemento.

<b>NOMBRE</b>	<b>TIPO</b>	<b>DESCRIPCION</b>
DEC	Entero (8)	Número de decimales del valor
DIA	Entero	Día del mes
N PTO AC	Entero (16)	Operaciones asociado al interruptor
N TRAFO	Entero	Identificador del transformador
NOM AC	Texto (10)	Nombre del punto
UI	Entero (8)	Unidades de ingeniería
VALOR DIA	Entero (32)	Valor consumo diario

**Estructura de la relación BARRA**

Define los elementos necesarios para almacenar los valores máximos y mínimos de voltajes en cada elemento, en la siguiente tabla se muestra los elementos de la relación BARRA utilizados, presentando el nombre del elemento, el tipo (caracter, entero) y una breve definición.

<i>NOMBRE</i>	<i>TIPO</i>	<i>DESCRIPCION</i>
DEC	Entero (8)	Número de decimales del valor
DIA MAX	Entero (8)	Día del mes ocurrencia del máximo
DIA MIN	Entero (8)	Día del mes ocurrencia min
V MAX	Entero (16)	Valor voltaje máximo registrado
V MIN	Entero (16)	Valor voltaje mínimo registrado
HR MAX	Entero (8)	Hora en que se registro el max
HR MIN	Entero (8)	Hora en que se registro min
MIN MAX	Entero (8)	Minuto en que registro el max
MIN MIN	Entero (8)	Minuto en que se registro el min
N PTO AN	Entero (16)	Identificador del punto analógico asociado
N SE	Entero (8)	Identificador de la subestación
N BARRA	Entero (8)	Identificador de la barra
NOM BARRA	Entero (8)	Nombre de la barra

**Estructura de la relación CORRI\_SUB**

Define los elementos necesarios para almacenar los valores máximos y mínimos de corriente en cada elemento, describiendo en la siguiente tabla el nombre del elemento utilizado, el tipo de su dato (entero, caracter) y una breve descripción de estos elementos.

<i>NOMBRE</i>	<i>TIPO</i>	<i>DESCRIPCION</i>
DEC	Entero (8)	Número de decimales del valor
DIA MAX	Entero (8)	Día del mes ocurrencia del máximo
DIA MIN	Entero (8)	Día del mes ocurrencia del mínimo
I MAX	Entero (16)	Valor corriente máxima registrada

<i>NOMBRE</i>	<i>TIPO</i>	<i>DESCRIPCION</i>
I MIN	Entero (16)	Valor corriente mínima registrada
IHR MAX	Entero (8)	Hora en que se registro el i_max
IHR MIN	Entero (8)	Hora en que se registro i_min
IMIN MAX	Entero (8)	Minuto en que registro el i_max
IMIN MIN	Entero (8)	Minuto en que se registro el i_min
N PTO AN	Entero (16)	Identificador del punto analogico asociado
N SE	Entero (8)	Identificador de la subestación
NOM	Entero (8)	Nombre del circuito

#### *Estructura de la relación SUBEST*

Esta estructura define los elementos de la relación SUBEST, proporcionando el nombre del elemento, el tipo de dato que utiliza (entero, caracter) y una descripción de cada elemento.

<i>NOMBRE</i>	<i>TIPO</i>	<i>DESCRIPCION</i>
N SE	Entero (8)	Identificador de la subestación
NOM SE	Caracter (10)	Nombre de la subestación
N UNIF	Entero (8)	Identificador del unifilar de la subestación
NOM CRTO	Caracter (10)	Nombre corto (mnemónico)
T ESP CTL	Entero (16)	Tiempo de espera para completar un control

#### *Estructura de la relación OPERA\_CTO*

Esta estructura define los elementos de la relación OPERA\_CTO, indica el nombre del elemento, tipo de dato utilizado (entero, caracter) y una descripción de para cada elemento.

<i>NOMBRE</i>	<i>TIPO</i>	<i>DESCRIPCION</i>
CAUSA	Entero (8)	Causa de la apertura del interruptor
CORRIENTE	Entero (16)	Valor de la corriente registrada antes de la apertura.
DIA APER	Entero (8)	Día del mes de la apertura
<i>NOMBRE</i>	<i>TIPO</i>	<i>DESCRIPCION</i>

DIA_CIERRE	Entero (8)	Día del mes del cierre
EST_TUPLE	Entero (8)	Indicador del tuple completo/vacio
HR_APER	Entero (8)	Hora en que ocurrió apertura
HR_CIERRE	Entero (8)	Hora en que ocurrió el cierre
MIN_APER	Entero (8)	Minuto en que ocurrió apertura
MIN_CIER	Entero (8)	Minuto en que ocurrió el cierre
N_CTO	Entero (8)	Identificador del número del circuito
N_PTO_AN	Entero (16)	Identificador del punto analogico asociado
N_PTO_DI	Entero (16)	Identificador del punto digital asociado al circuito
N_SE	Entero (8)	Identificador de la subestación
NOM_CTO	Entero (8)	Identificador del número del circuito

#### *Estructura de la relación INTE\_SE*

Define los elementos de la relación INTE\_SE para la elaboración del reporte de Operación de Interruptores, dando una breve definición de cada uno de sus elementos, el tipo de dato que maneja (entero, caracter) y el nombre del elemento.

<i>NOMBRE</i>	<i>TIPO</i>	<i>DESCRIPCION</i>
DIA	Entero (8)	Día del mes
FALLAS	Entero (32)	Cuenta el número de fallas
MANIOBRAS	Entero (32)	Cuenta el número de maniobra
N_PTO_DI	Entero (16)	Identificador del punto digital asociado al circuito
N_SE	Entero (8)	Identificador de la subestación
NOM_INTE	Caracter	Nombre del interruptor

#### *Estructura del reporte Valores Máximos y Mínimos de Corriente (VMMI)*

Define los elementos donde es almacenada la información que genera el reporte VMMI, la siguiente tabla contiene, el nombre corto del elemento a utilizar, el tipo dato requerido (entero, caracter) y la definición de cada elemento.

<i>NOMBRE</i>	<i>TIPO</i>	<i>DESCRIPCION</i>
NOM SE	Entero (8)	Nombre de la subestación
NOM	Entero (8)	Nombre del circuito
I MAX	Entero (16)	Valor corriente máxima registrada
HORA IMAX	Entero (8)	Hora en que se registro el i max
DIA	Entero (8)	Día del mes ocurrencia del máximo
I MIN	Entero (16)	Valor corriente mínima registrada
HORA MIN	Entero (8)	Hora en que se registro i min

*Estructura del reporte de Valores Máximo y Mínimo de Voltaje (VMMV)*

Define los elementos utilizados para la generación del reporte VMMV; el nombre corto del elemento, el tipo de dato usado (entero, caracter) y la definición de cada elemento.

<i>NOMBRE</i>	<i>TIPO</i>	<i>DESCRIPCION</i>
NOM BARRA	Entero (8)	Nombre de la barra
NOM SE	Entero (8)	Nombre de la subestación
V MAX	Entero (8)	Valor voltaje máximo registrado
V MIN	Entero (16)	Valor voltaje mínimo registrado
HORA MAX	Entero (8)	Hora en que se registro v max
DIA	Entero (8)	Día del mes ocurrencia del máximo
HORA MIN	Entero (8)	Hora en que se registro el v min

*Estructura del reporte de Consumo Mensual (COME)*

Define los elementos utilizados para la generación del reporte COME; el nombre, tipo y una breve definición.

<i>NOMBRE</i>	<i>TIPO</i>	<i>DESCRIPCION</i>
NOM	Caracter	Nombre del transformador
NOM SE	Caracter	Nombre de la subestación
MEDICION	Entero (16)	Valor consumo diario
UNIDAD	Caracter	Unidades de ingeniería (KWH o KVARH)
DIA	Caracter	Día del registro en el mes

#### *Estructura del reporte de Operación de Interruptores (OPIN)*

Define los elementos utilizados para la generación del reporte OPIN; el nombre corto del elemento, el tipo de dato utilizado y el nombre completo.

<i>NOMBRE</i>	<i>TIPO</i>	<i>DESCRIPCION</i>
NOM CTO	Entero (16)	Número del circuito
NOM SE	Entero (8)	Nombre de la subestación
FALLAS	Entero (8)	Número de fallas registradas en el día
MANIOBRAS	Entero (16)	Número de maniobras registradas
DIA	Entero (8)	Día del mes ocurrencia del máximo

#### *Estructura del reporte de Operación de Circuitos (OPERA)*

Define los elementos utilizados para la generación del reporte OPERA; define el nombre corto, tipo de dato utilizado y una breve definición de cada elemento.

<i>NOMBRE</i>	<i>TIPO</i>	<i>DESCRIPCION</i>
NOM CTO	Entero (8)	Nombre del circuito
NOM SE	Entero (8)	Nombre de la subestación
DIA APER	Caracter	Día del registro de apertura
DIA CIERRE	Caracter	Día del registro de cierre
CORRIENTE	Entero (16)	Valor de corriente registrada en el día

**Estructura del reporte Histórico de mediciones (HISTORICO)**

Define los elementos utilizados en la generación del reporte HISTORICO; el nombre corto, el tipo de dato utilizado (entero, caracter) y una breve definición.

<i>NOMBRE</i>	<i>TIPO</i>	<i>DESCRIPCION</i>
N PTO AN	Entero (16)	Identificador del punto analógico
NOM PTO	Entero (8)	Nombre del punto analógico
NOM SE	Entero (8)	Nombre de la subestación
UI	Entero (8)	Unidades de ingeniería
VALOR 01	Entero (16)	Valor de la muestra 1
VALOR 02	Entero (16)	Valor de la muestra 2
...	...	...
VALOR 96	Entero (16)	Valor de la muestra 96

**3.3.4 Definiciones de los diagramas del nivel 2 por proceso**

A continuación se presenta la descripción correspondiente al nivel 2 de algunos de los procesos mostrados en el diagrama general 3.1. Para cada uno de los diagramas se muestra la información obtenida mediante el sistema de reportes del SYSARCH.

En el diagrama de la figura 3.2 se muestra el nivel 2 correspondiente al proceso de comunicación PC - MAC, donde se observan los dos procesos encargados del enlace del canal de comunicación entre la computadora personal y la estación maestra, además se da una breve descripción de los mensajes que son generados y enviados a otros procesos.

***Comunicación PC - MAC***

Este proceso permite el coordinar la información de enlace entre la computadora y la MAC, mediante una función que recibe, manda y verifica que tipo de mensaje debe enviarse y a cual proceso va dirigido.

### Decodifica mensajes

Proceso que se encarga validar un mensaje recibido. Una vez validado el mensaje, este proceso se encarga de enviarlo a otros módulos responsables del tratamiento de la información. Los mensajes pueden ser los siguientes:

- 1.- Mensaje de comunicación
- 2.- Mensaje interruptores
- 3.- Relaciones dinámicas

#### Destino

- Diagnóstico de canal  
Módulo genera reportes  
Módulo de conversión de MAC a dBase

### Diagnóstico de canal

El proceso que verifica continuamente que el canal de comunicación se encuentre funcionando en caso contrario genera un mensaje de error a pantalla.

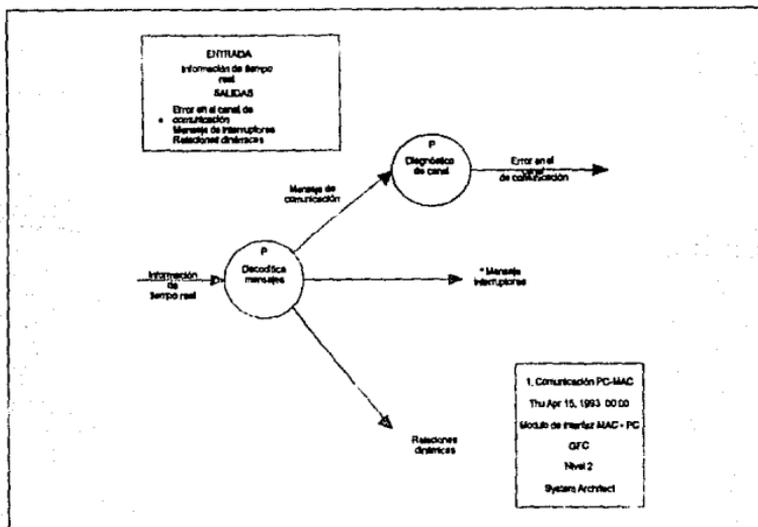


DIAGRAMA 3.2 Comunicación PC - MAC

Otro de los procesos descritos en el nivel 2 es "Genera Reportes" que se muestra en el diagrama 3.3, donde se observan los 7 módulos encargados de la generación de reportes individualmente, además se indica para cada uno las relaciones que utiliza de la BDD que son necesarias para la generación de los reportes.

### *Genera Reportes*

Proceso que contiene los parámetros adecuados para generar los reportes de acuerdo a la información que accese, a continuación se describe cada uno de los módulos:

#### **Selección de Reportes**

Este proceso recibe la solicitud de reporte del proceso de reloj, el cual indica que se cumplió el tiempo de espera para la inicialización de reportes, así como el comando de inicializar los reportes desde la interfaz de usuario. En base al mensaje recibido, este módulo se encarga de crear un mensaje y enviarlo al módulo correspondiente para que ahí se ejecute la acción solicitada.

Cada uno de los 6 módulos encargados de la generación de reportes (OPIN, VMMV, VMMI, OPERA\_CTO, COME e HISTORICO) tiene funciones similares, como son: crear, generar y actualizar cada uno ellos. Tales acciones se especifican en los mensajes provenientes de la solicitud, los mensajes contienen la información necesaria para crear, almacenar, muestrear y controlar (activa/desactiva) la generación del reporte.

A continuación se explica cada uno de los procesos del nivel 2 para el diagrama 3.3 correspondientes al módulo "Genera Reporte" del nivel superior, en cada uno de ellos se realizan la operaciones de crear, generar y actualizar el reporte correspondiente:

#### **OPIN**

El *reporte* de OPeración de INterruptores (OPIN), utiliza las relaciones ACUMULADO, INTE\_SE, DIGITAL Y SUBEST para su elaboración. El reporte se genera por subestación conteniendo los siguientes parámetros: nombre del circuito, el nombre de la subestación, el número de fallas, el número de maniobras y el día de ocurrencia.



### VMMV

El *reporte* de Valores Máximo y Mínimo de Voltaje (VMMV), utilizando las relaciones de la BDD son las siguientes: ANALOGICO, SUBEST y BARRA para la elaboración del reporte. El reporte se genera con el nombre de cada uno de las subestaciones proporcionando la siguiente información: el nombre de la subestación, el nombre de las barras de cada subestación, la hora de ocurrencia del valor máximo, la hora de ocurrencia del valor mínimo y el día en que se registraron los valores.

### VMMI

El *reporte* de Valores Máximo y Mínimo de Corriente (VMMI), utilizando las siguiente relaciones de la BDD son las siguientes: CORRI\_SUB, ANALOGICO y SUBEST, de las cuales obtiene la información necesaria para la generación del reporte. El reporte se genera por el nombre de cada subestación y proporciona la siguiente información: el nombre de la subestación, el nombre del circuito, el valor de la corriente suministrada en el día tanto máxima como mínima, el día y hora de ocurrencia de los valores máximo y mínimo.

### OPERA\_CTO

El *reporte* de Operación de Circuitos, recibe un mensaje que le indica que un interruptor cambio de estado y que debe almacenar los datos que cambiaron al reporte de OPERA en forma inmediata o por solicitud del operador (mensaje opera). El reporte contiene los siguiente parámetros: nombre de la subestación, nombre del circuito, hora y día de la apertura, hora y día de la cerradura, la lectura de la corriente antes de la apertura y la causa que provocó el cambio de estado del interruptor (falla o maniobra). Además utiliza las siguientes relaciones de la BDD: ANALOGICO, DIGITAL y OPERA\_CTO la cual indica si el registro utilizado del reporte esta completo (0) o incompleto (1), es decir, si el circuito registrado paso de apertura a cierre o de cierre a apertura.

## COME

El *reporte* de Consumo Mensual. Utilizando las relaciones de la BDD siguientes: SUBEST, TRAN\_SE y ACUMULADO de las cuales se obtiene la información necesaria para elaborar el reporte. Los parámetros que contiene el reporte son los siguientes: nombre de la subestación, nombre del transformador, el valor de la medición del transformador, la unidad en que se midió (KWH o KVARH) y el día de la medición.

## HISTORICO

El *reporte* Histórico utiliza las relaciones: SUBEST y ANALOGICO de las cuales obtiene los valores de las mediciones para la generación del mismo. El reporte despliega la siguiente información: El nombre de la subestación y una gráfica del valor de la medición Vs tiempo, donde se muestra el comportamiento de la medición durante el día.

### 3.3.5 Definición de flujo de datos de los diagramas del nivel 2

El diagrama 3.3 presenta mensajes (líneas con flechas) que son designados como flujo de datos dentro de la metodología utilizada, y están definidos con un comentario dentro del SYSTEM ARCHITECT [SYS 90] simbolizados con tres círculos pequeños en la parte superior de la línea con el primer círculo relleno generando una narrativa que se puede obtener desde el sistema de reportes del paquete. A continuación se describen los 6 mensajes utilizados en el diagrama 3.3.

#### *Mensaje OPIN*

Mensaje que contiene la información necesaria para crear, actualizar, inicializar y controlar (activa/desactiva) la generación del archivo de *reporte* de Operación de interruptores. Se define además, la estructura del mensaje así como el mensaje de cambio de estado de un interruptor.

Dependiente de la función que se deba realizar (crear, actualizar, muestrear, inicializar o mensaje de interruptores) se definen dos estructuras para el mensaje, las cuales son: *MsjInt* y *MsjSol* (descritos en el apéndice D).

A continuación se definen como ejemplo los 4 mensajes para este proceso, los mensajes para los otros cinco procesos siguen la misma estructura por lo cual estos son descritos en el apéndice D.

#### Crea archivo de reporte OPIN

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_OPIN	"Reporte OPIN"
MsjSol.operac	=	Crea	"Crea archivo OPIN"

#### Actualiza archivo de reporte OPIN

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_OPIN	"Reporte OPIN"
MsjSol.operac	=	Actualiza	"Actualiza archivo OPIN"

#### Inicializa archivo de reporte OPIN

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_OPIN	"Reporte OPIN"
MsjSol.operac	=	Inicializa	"Inicializa reporte OPIN"

#### ^ Mensaje de interruptores

MsjInt.funcion	=	Msje_Int	"Mensaje por interrupción"
MsjInt.npto	=	npto	"número de interruptor"
MsjInt.edo_fun	=	1 o 0	"Control completo/no control"
MsjInt.edo_vis	=	1 o 0	"apertura/cierre"

#### Mensaje VMMV

Mensaje que contiene la información necesaria para realizar las funciones de: crear, actualizar, muestrear, inicializar y controlar (activa/desactiva) la generación de los reportes de Valores Máximos y Mínimos de Voltaje. Sus mensajes son definidos en el apéndice D.

### ***Mensaje VMMI***

Mensaje que contiene la información necesaria para crear, guardar, muestrear y controlar (activa/desactiva) la generación de los *reportes* de Valores Máximos y Mínimos de Corriente. La estructura de los mensajes de las funciones anteriores son definidos en el apéndice D.

### ***Mensaje OPERA***

Mensaje que contiene la información necesaria para crear y controlar (activa/desactiva) la generación de los *reportes* de Operación de Circuitos. Los mensajes son definidos en el apéndice D.

### ***Mensaje COME***

Mensaje que contiene la información necesaria para crear, guardar, muestrear y controlar (activa/desactiva) la generación del *reporte* de Consumo Mensual. Los mensajes son definidos en el apéndice D.

### ***Mensaje HISTORICO***

Mensaje que contiene la información necesaria para crear, guardar, muestrear y controlar (activa/desactiva) la generación del *reporte* Histórico de Mediciones. Las estructuras del mensaje se definen en el apéndice D.

### **3.3.6 Definiciones de los diagramas del nivel 3**

En el desarrollo del tercer nivel, nuestro diseño se comienza a extender a niveles muy detallados, por lo que es necesario hacer hincapié en que se describirán sólo los diagramas principales del diseño, que son representativos en el proceso de "Genera Reporte".

### ***Proceso OPIN***

Para llevar a cabo el proceso de Operación de Interruptores descrito en el nivel 2, se elaboraron 6 módulos; cada uno de los cuales se encarga de ejecutar una función correspondiente al mensaje recibido, como lo son: inicializa OPIN, actualiza OPIN, crea OPIN y el mensaje de interruptores; el diagrama 3.4 del nivel 3 hace referencia a estos módulos e incluye una pequeña descripción para cada uno de ellos en este nivel.

#### **Decodifica mensaje OPIN**

Módulo que enruta el mensaje recibido al módulo encargado de procesarlo, de acuerdo al tipo de proceso a realizar. Las opciones del mensaje son las siguientes:

- 1.- Actualizar archivo de reporte OPIN (mensaje actualiza OPIN).
- 2.- Inicializar el archivo del reporte OPIN (mensaje inicializa OPIN).
- 3.- Crear archivo del reporte OPIN (mensaje de crea OPIN).
- 4.- Mensaje de cambio de estado (\*mensaje de interruptores).

#### **Actualiza reporte OPIN**

Para realizar las acciones correspondientes a este mensaje, se deben llevar a cabo diferentes operaciones para que el proceso obtenga de la relación INTE\_SE el número de cuantas operaciones por maniobras se efectuó en ese día, y al mismo tiempo obtener de la relación ACUMULADO el total de operaciones realizadas, para obtener mediante la diferencia de estos dos rubros el número de fallas, el cual se almacena en el archivo del reporte. Este módulo se encarga de realizar las modificaciones y/o actualizaciones en el archivo OPIN que ya esta en formato Dbase III plus.



### Crea Archivo DBF OPIN

Proceso encargado de crear el archivo "Operación de Interruptores" en el directorio adecuado, el cual corresponde a la fecha obtenida en la PC. Este archivo contiene los siguientes campos:

NOMBRE	TIPO	DESCRIPCION
NOM	Caracter 10 byte	Nombre del interruptor
FALLAS	Entero 4 byte	Número de Fallas
MANIOBRAS	Entero 4 byte	Número de Maniobras
DIA	Entero 1 byte	Día del mes de ocurrencia
NOM_SE	Entero 1 byte	Nombre de la subestación
FECHA	Entero 1 byte	Fecha de periodo muestreado

### Inicializa REPORTE OPIN

Proceso que inicializa el archivo OPIN (borra información anterior) solamente si esta creado, de no ser así llama a la función "crea archivo DBF OPIN".

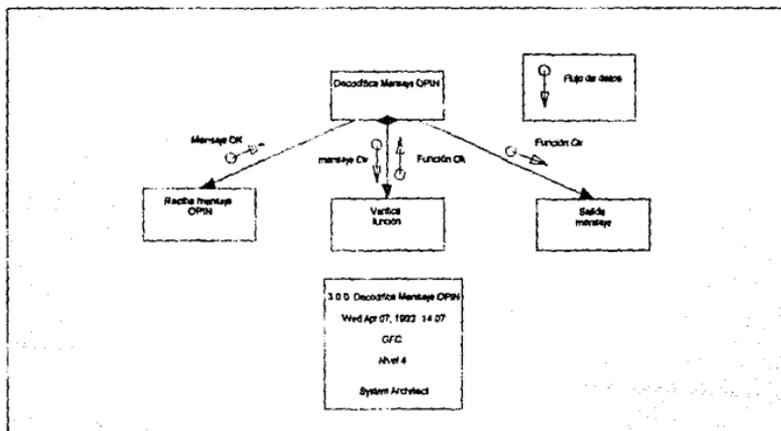


DIAGRAMA 3.5 Decodifica mensaje OPIN

### Realiza Operaciones INTE\_SE

Al efectuarse el cambio de estado de algún interruptor, este proceso revisa la relación INTE\_SE para conocer si es cierre o apertura y determinar si el cambio de estado se realizó por falla o maniobra. Además tomando en cuenta el total de operaciones que se encuentra en la relación ACUMULADO determina el total de maniobras y fallas.

En los diagramas del 3.5 al diagrama 3.10 se muestran los correspondientes niveles de detalle de los módulos integrantes del proceso OPIN, los cuales no presentan narrativa por ser muy cercanos al código fuente.

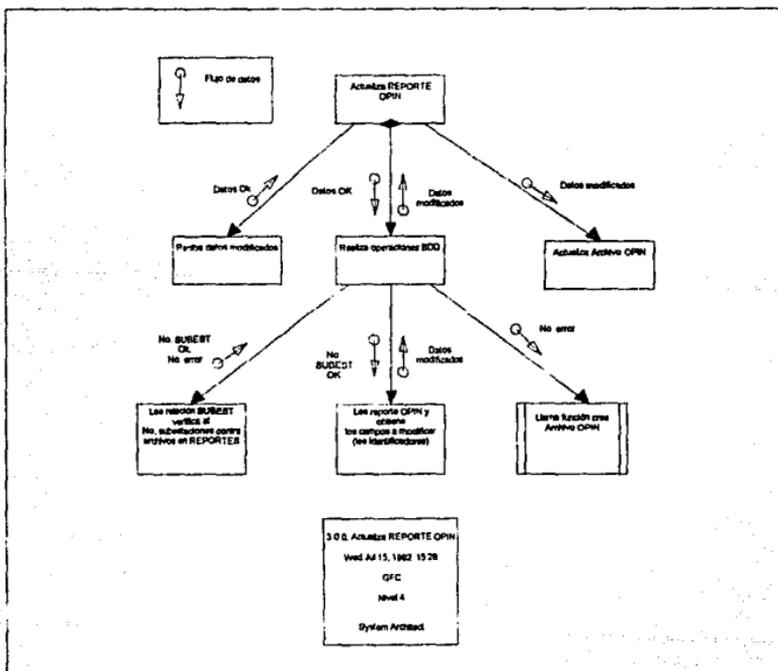


DIAGRAMA 3.6 Actualiza reporte OPIN

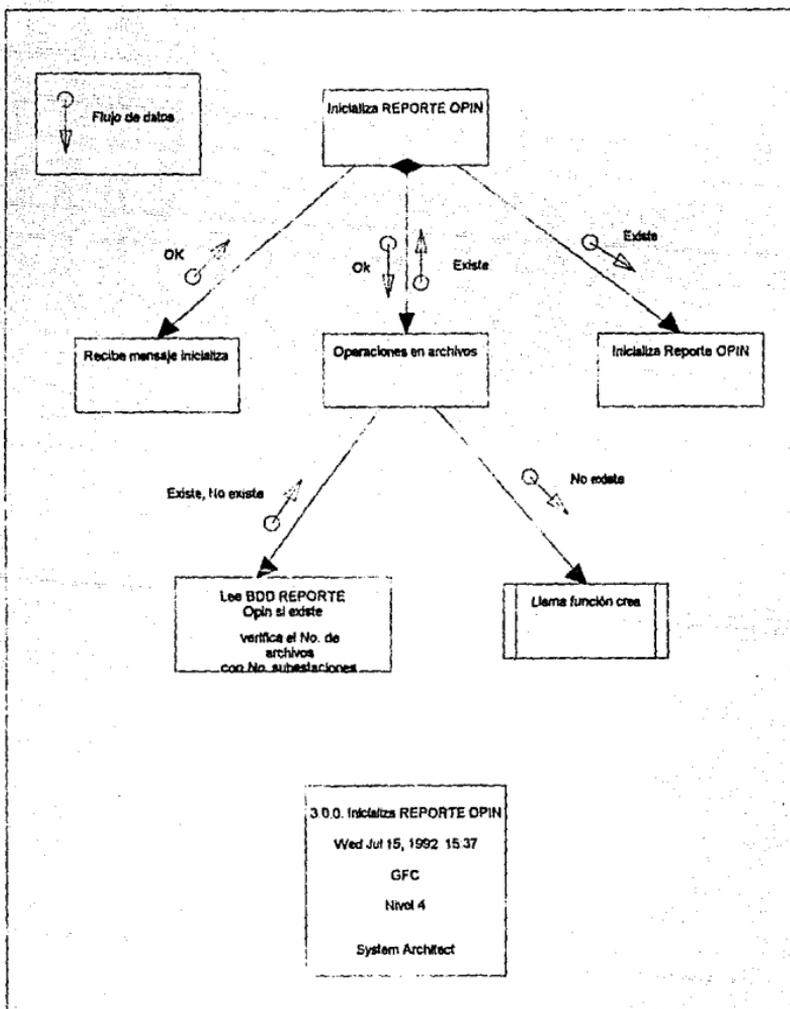


DIAGRAMA 3.7 Inicializa reporte OPIN

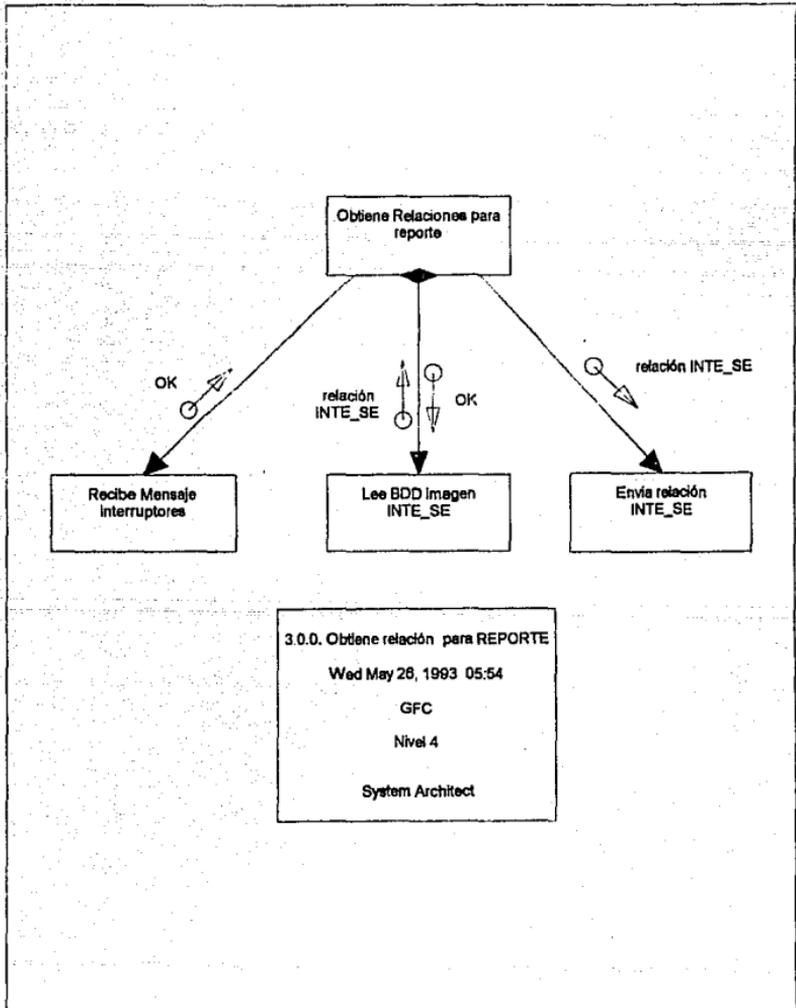


DIAGRAMA 3.8 Obtiene relaciones para reporte

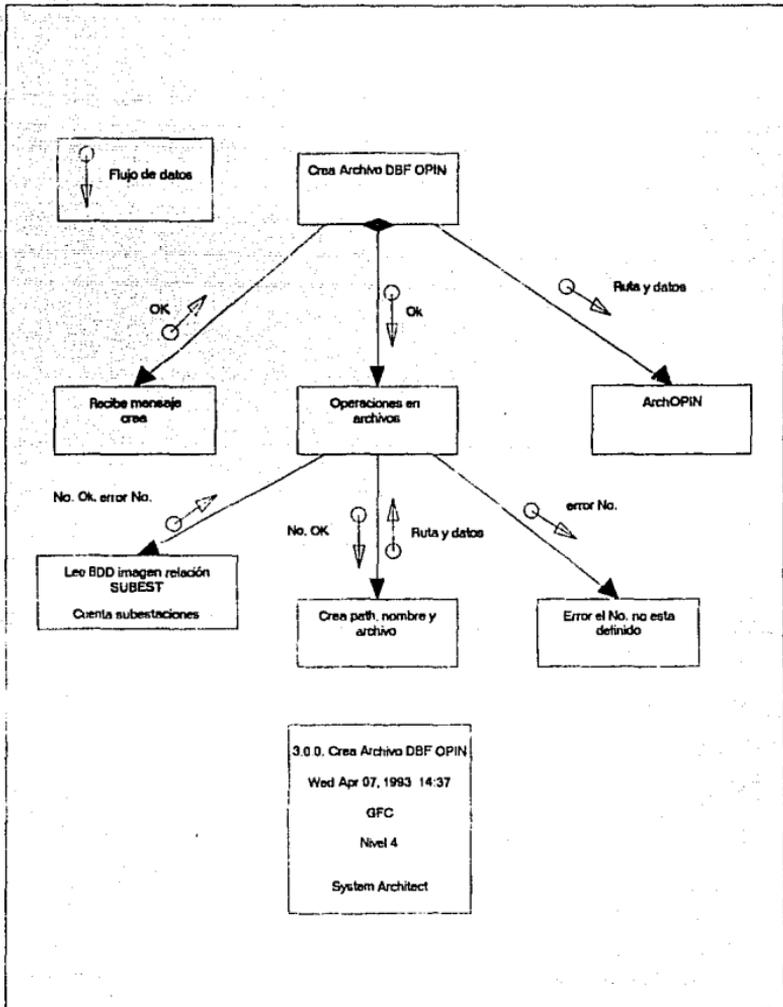


DIAGRAMA 3.9 Crea archivo DBF OPIN

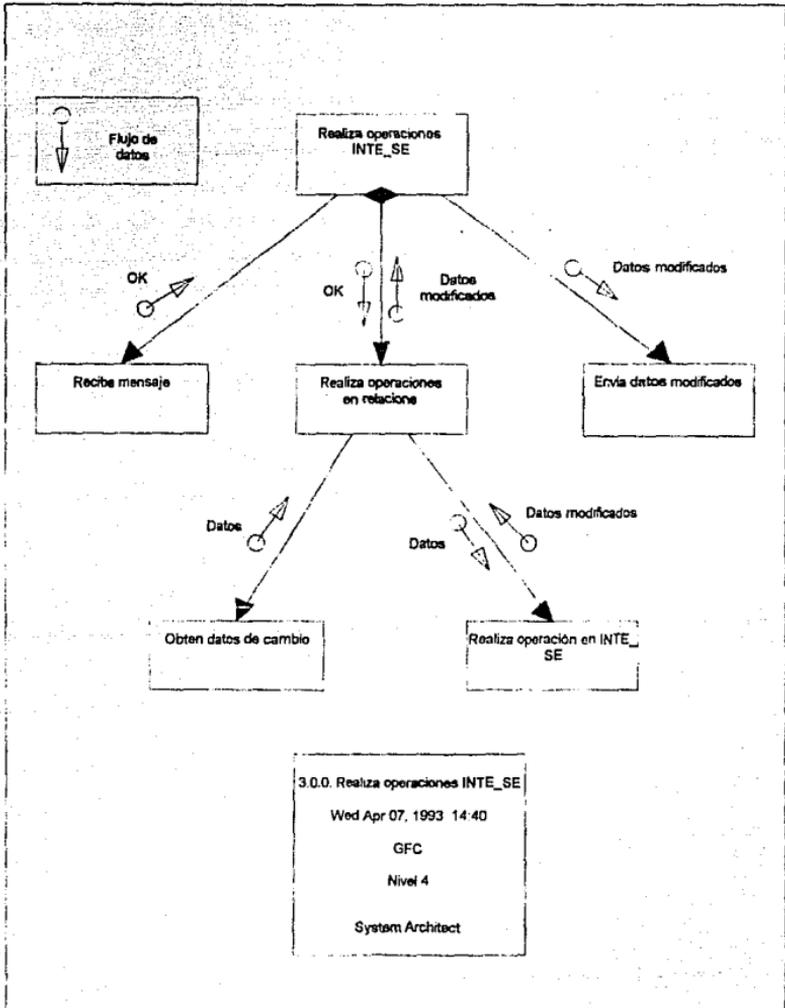


DIAGRAMA 3.10 Realiza operaciones INTE\_SE

**Proceso VMMV**

Este proceso esta descrito en el diagrama 3.11 a continuación se describe cada uno de sus módulos. En los procesos VMMI y COME las funciones que efectuan son similares a las descritas para este proceso, por lo cual no se incluye su descripción ni los diagramas.

**Actualiza REPORTE VMMV**

Proceso que actualiza el archivo VMMV al final del día con los datos almacenados en la relación BARRA. También se encarga de la inicialización de la relación BARRA para el día siguiente.

La actualización se ejecuta en el archivo del reporte VMMV modificando la información de los siguientes campos:

<i>NOMBRE</i>	<i>TIPO</i>	<i>DESCRIPCION</i>
DEC	Entero (8)	Número de decimales del valor
DIA MAX	Entero (8)	Día del mes ocurrencia del máximo
DIA MIN	Entero (8)	Día del mes ocurrencia min
V MAX	Entero (16)	Valor voltaje máximo registrado
V MIN	Entero (16)	Valor voltaje mínimo registrado
HR MAX	Entero (8)	Hora en que se registro el max
HR MIN	Entero (8)	Hora en que se registro min
MIN MAX	Entero (8)	Minuto en que se registro el max
MIN MIN	Entero (8)	Minuto en que se registro el min
N PTO AN	Entero (16)	Identificador del punto analógico asociado
N SE	Entero (8)	Identificador de la subestación
N BARRA	Entero (8)	Identificador de la barra
NOM BARRA	Entero (8)	Nombre de la barra

**Crea Archivo DBF VMMV**

Proceso que crea el archivo del reporte VMMV, al crear el archivo se define un subdirectorio, y otro archivo con el nombre y la fecha de inicio.

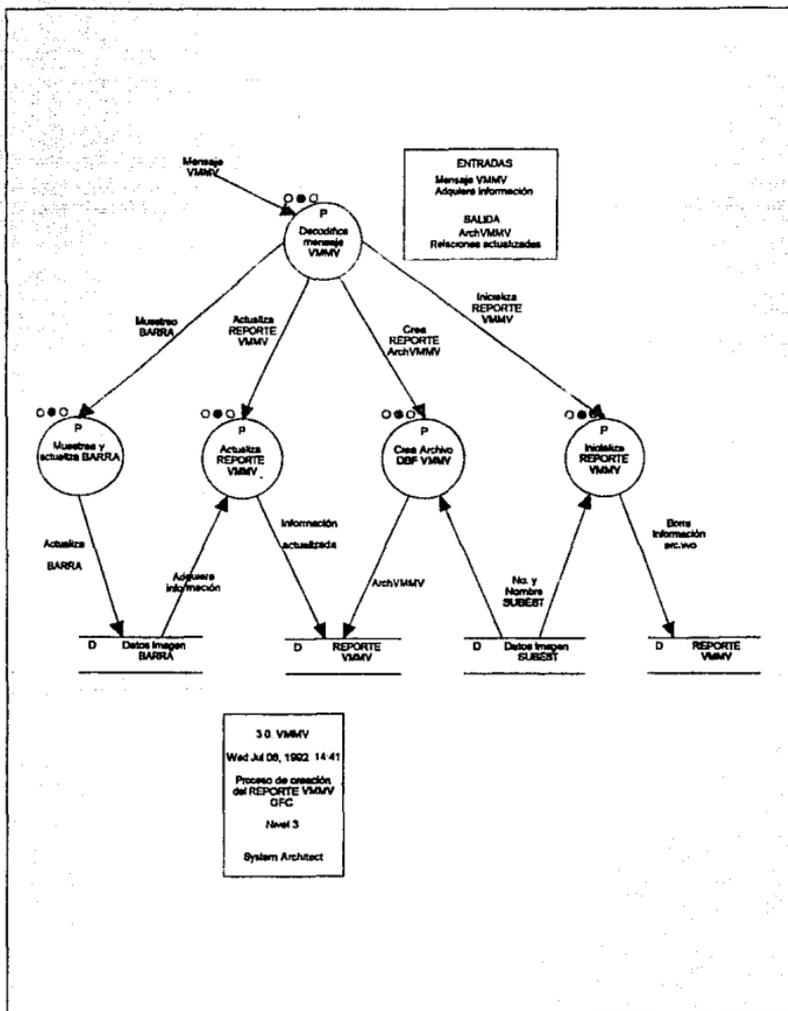


DIAGRAMA 3.11 Proceso de creación del reporte VMMV

### **Decodifica mensaje VMMV**

Proceso que enruta el mensaje recibido al módulo correspondiente encargado de procesarlo. Las posibles acciones a ejecutar indicadas en el mensaje son las siguientes:

- 1.- Actualizar reporte VMMV
- 2.- Crear base de datos reporte VMMV
- 3.- Inicializar base de datos reporte VMMV
- 4.- Muestreo

### **Inicializa reporte VMMV**

Proceso que inicializará el archivo VMMV colocando en cero sus registros, solo si esta creado, de no ser así llama a la función "Crea Archivo DBF VMMV".

### **Muestrea y actualiza BARRA**

Este proceso se efectúa periódicamente, y obtiene de la BDD de la relación ANALOGICO los valores de los puntos asociados a las barras y almacena esta información en la relación BARRA donde se guardan los valores máximos o mínimos de dicha barra.

Se muestran también los diagramas del 3.12 al 3.16 correspondientes a los niveles subsecuentes del diseño. No poseen narrativa por ser cercanos al código fuente. Se utiliza otra metodología la cual es descrita en el apéndice B.

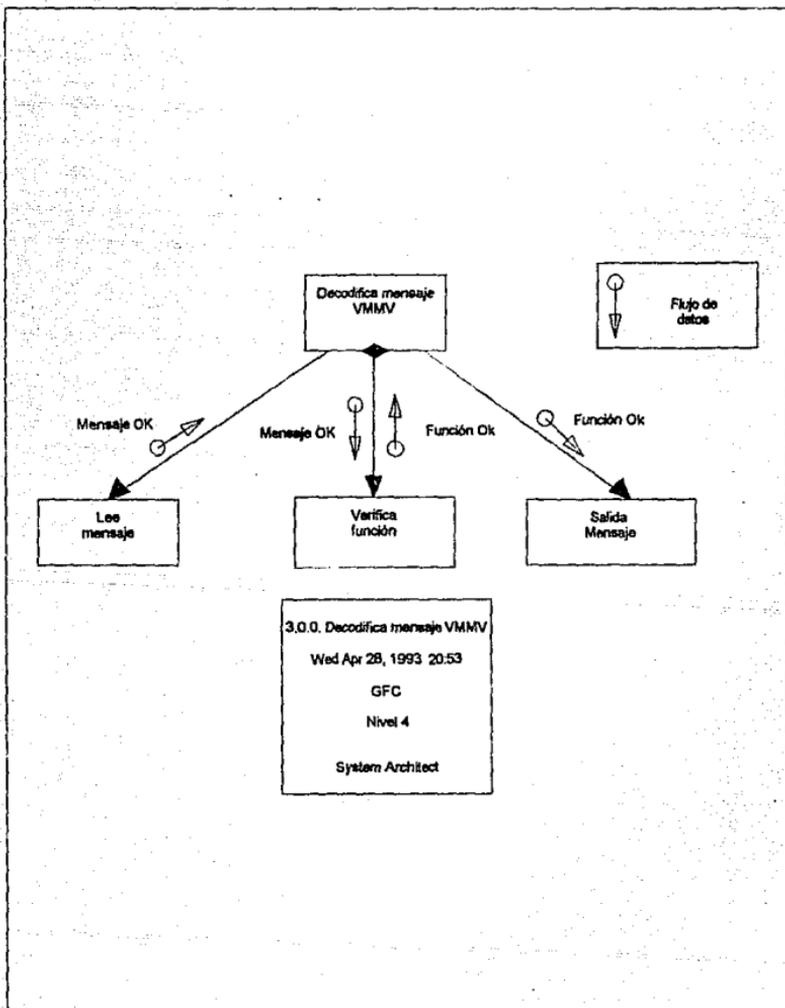


DIAGRAMA 3.12 Decodifica mensaje VMMV

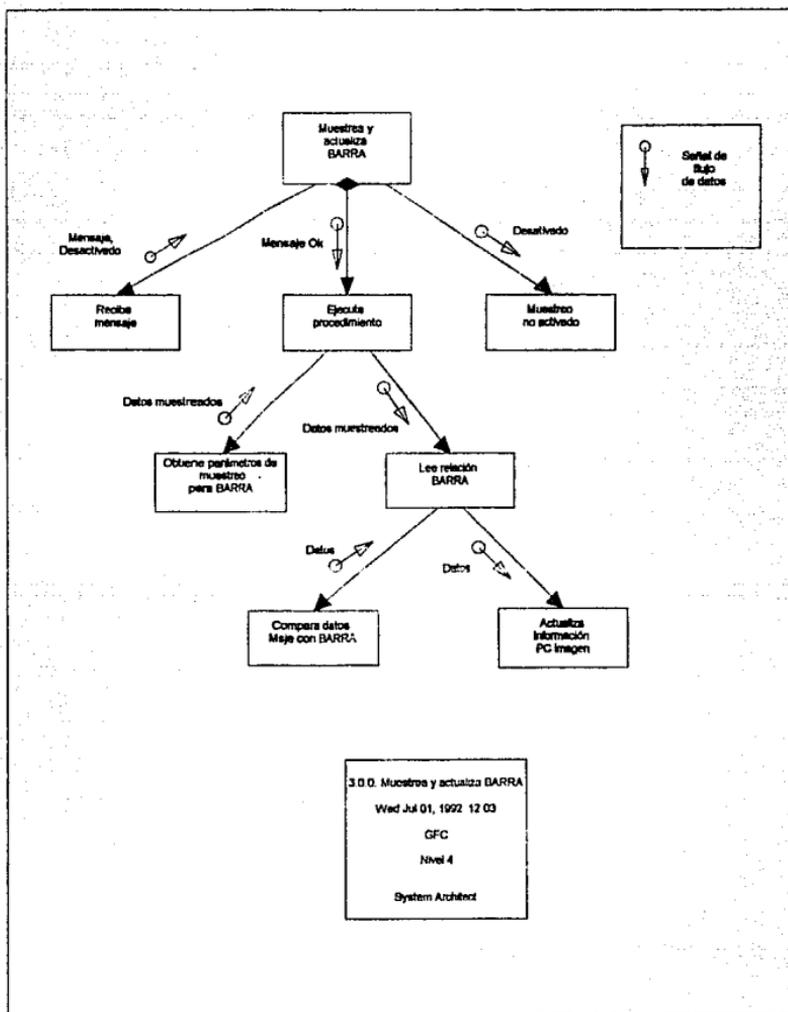


DIAGRAMA 3.13 Muestra y actualiza la relación BARRA

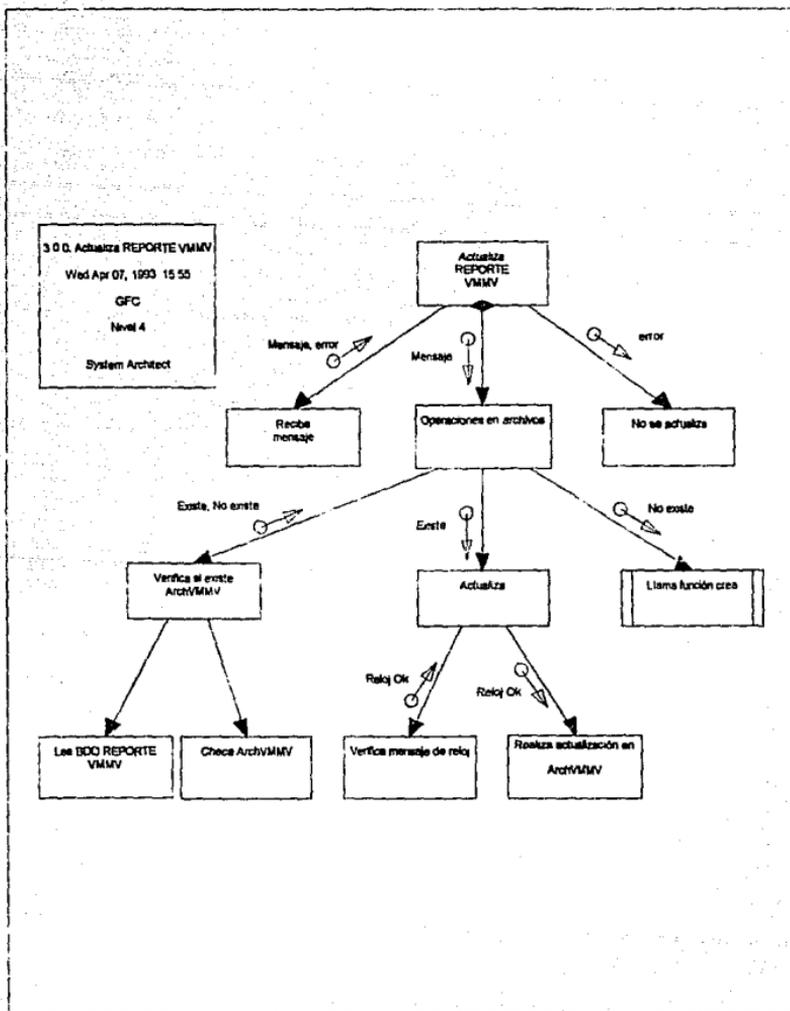


DIAGRAMA 3.14 Actualiza reporte VMV

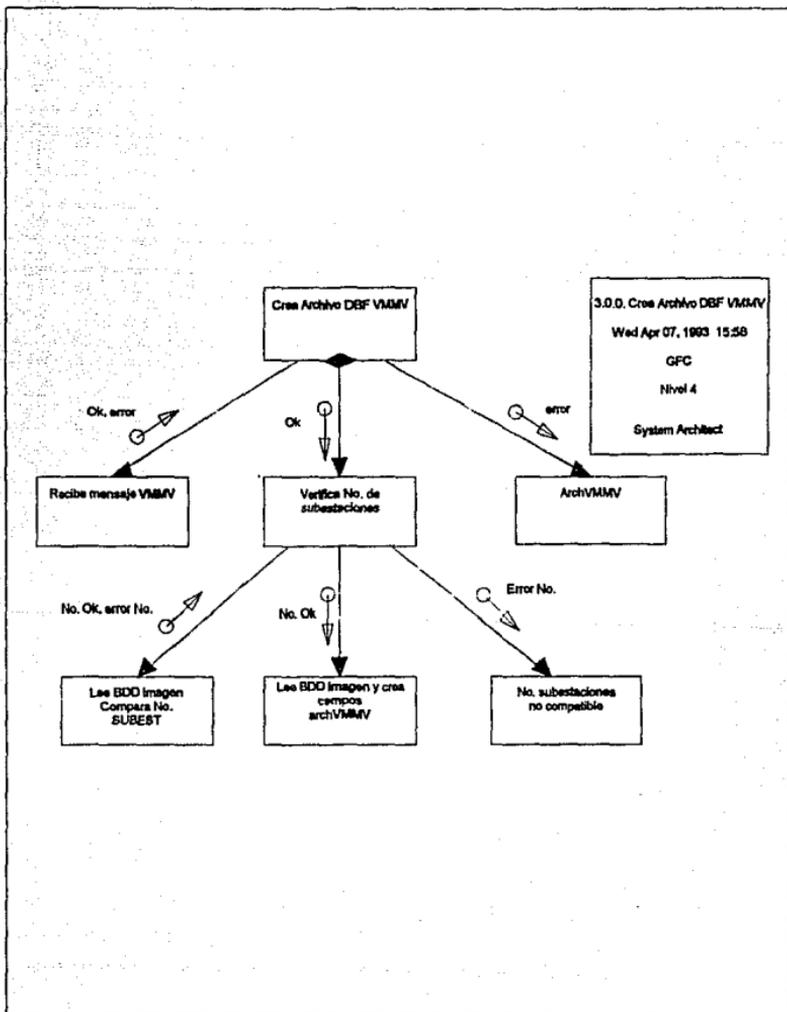


DIAGRAMA 3.15 Crea archivo DBF VMMV

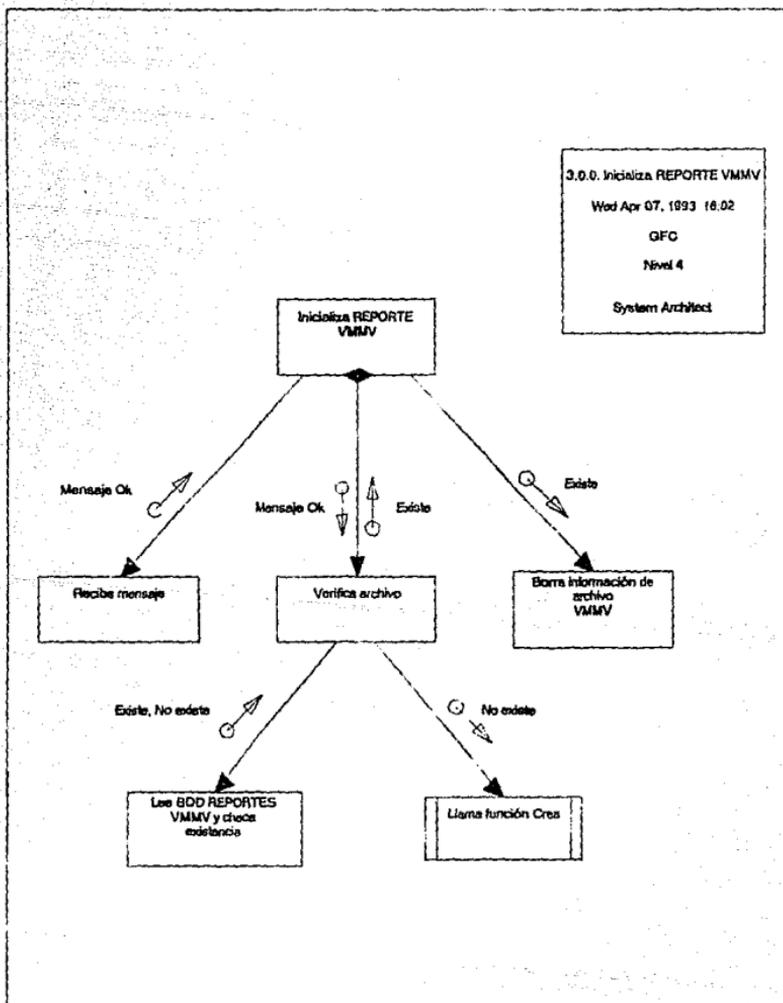


DIAGRAMA 3.18 Inicializa reporte VMMV

**Proceso OPERA**

Este proceso esta definido en la figura 3.17 que muestra las partes que lo componen y la forma en que obtienen y envian sus datos.

**Actualiza reporte OPERA**

Proceso que actualiza en el reporte OPERA los cambios efectuados en alguno de los circuitos indicados en la base de datos. En el siguiente punto se muestra la estructura del archivo OPERA.

**Crea archivo DBF OPERA**

Proceso que crea el archivo de reporte de Operación de Circuitos en el subdirectorio correspondiente. A continuación se define la estructura a crear:

<b>NOMBRE</b>	<b>TIPO</b>	<b>DESCRIPCION</b>
SUBESTACION	Caracter	Nombre de la subestación
FECHA/HORA	Entero (8)	Fecha de impresión
CIRCUITO	Entero (8)	Identificador del número de circuito
CORRIENTE	Entero (16)	Corriente registrada antes de apertura
CAUSA	Entero (8)	Causa apertura
DIA APER	Entero (8)	Día del mes de apertura
HORA APER	Entero (8)	Hora en que ocurrió apertura
DIA CIERRE	Entero (8)	Día del mes de cierre
HORA CIERRE	Entero (8)	Hora en que ocurrió cierre

Al crear el archivo se define su ruta de búsqueda, el nombre y la fecha de inicio del mismo.

**Decodifica Mensaje OPERA**

Proceso que enruta el mensaje al módulo encargado de procesarlo. Las operaciones que se efectuan de acuerdo al mensaje son las siguientes:

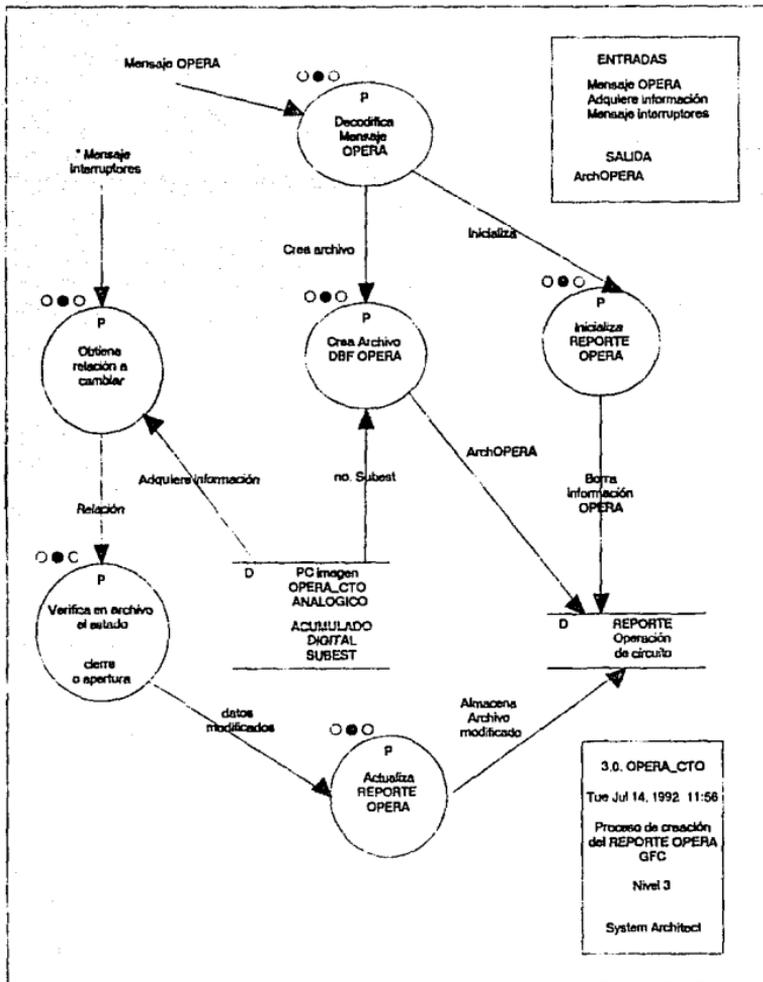


DIAGRAMA 3.17 Proceso de creación del reporte OPERA

- 1.- Inicializar base de datos reporte OPERA\_CTO
- 2.- Crear base de datos reporte OPERA\_CTO

#### **Inicializa reporte OPERA**

Este proceso inicializa, es decir, pondrá en ceros al archivo OPERA solo si esta creado de no ser así llama la función "Crea Archivo DBF OPERA".

#### **Obtiene relación a cambiar**

Proceso que obtiene de la relación OPERA\_CTO los puntos que seran modificados al llegar el cambio de estado del interruptor.

#### **Verifica en archivo el estado cierre o apertura**

Este módulo verifica en el archivo OPERA, en el campo que contiene el punto digital, el estado actual y el anterior para realizar la modificación del punto de acuerdo al estado en que se encuentre el interruptor; en caso que este completo el registro de lectura, es decir que ya hubo una apertura/cierre se incrementa la cuenta de maniobras y suma otro registro, en caso que sea cierre/apertura determinará la causa de apertura por falla o maniobra.

A continuación se muestran los diagramas del 3.18 al diagrama 3.23 que integran el proceso OPERA, los cuales no presentan narrativa por estar muy cercanos al código fuente. En esta parte del diseño se utiliza otra metodología la cual es descrita en el apéndice B.

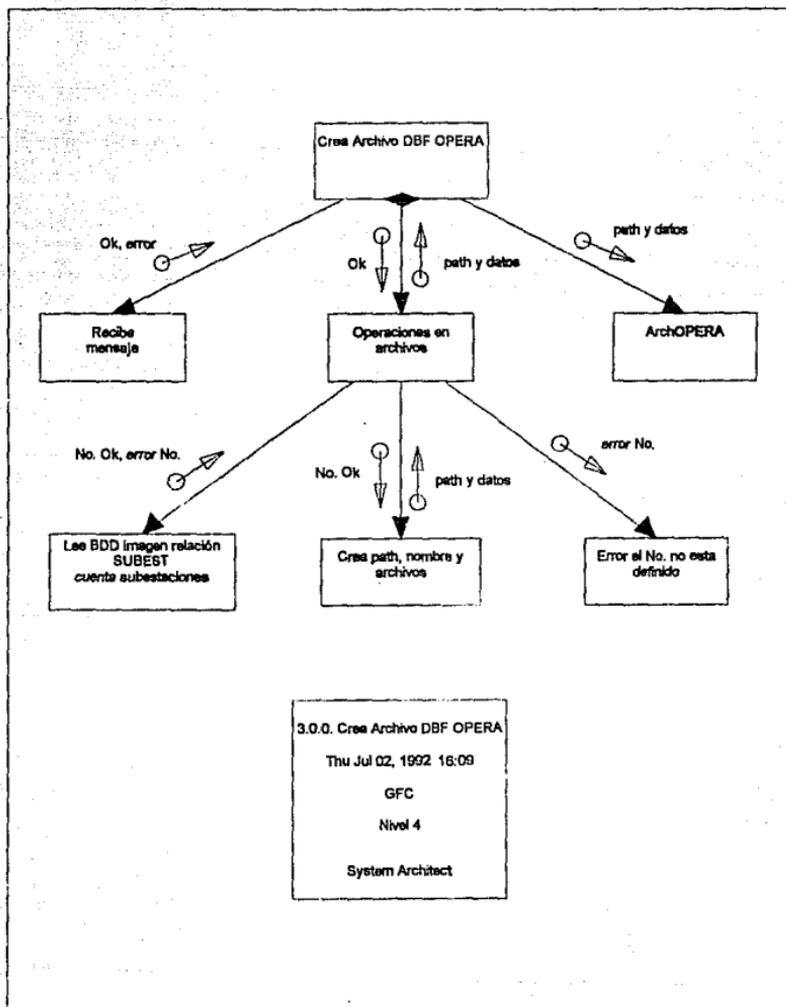


DIAGRAMA 3.18 Crea archivo DBF OPERA

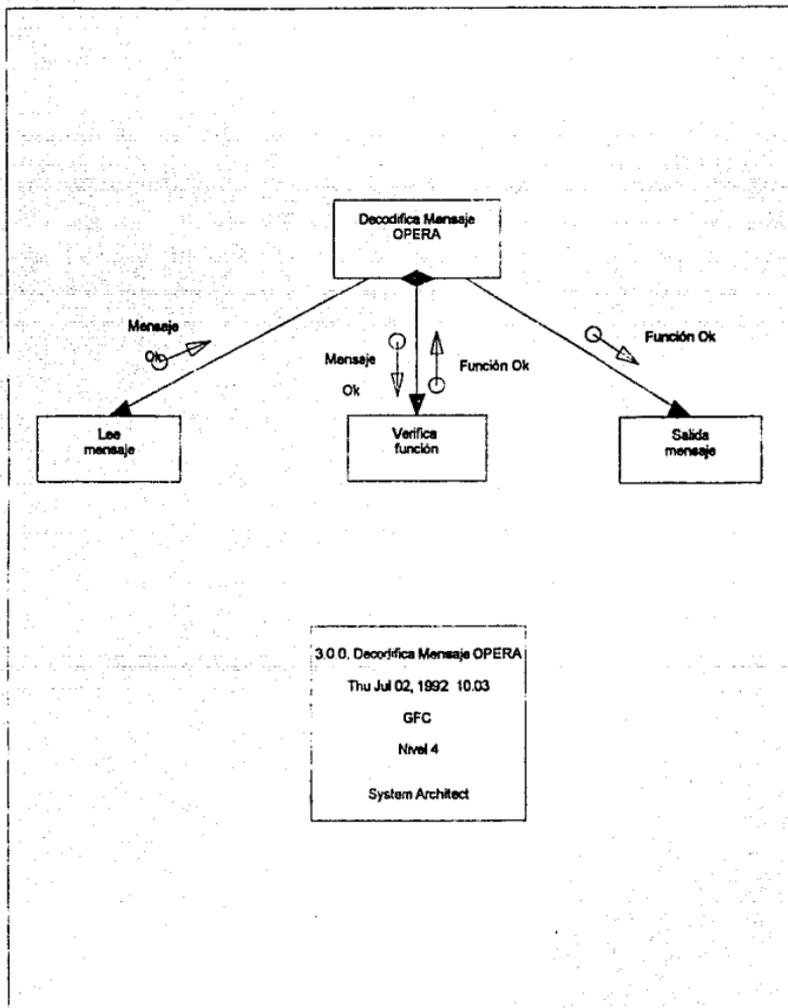


DIAGRAMA 3.19 Decodifica mensaje OPERA

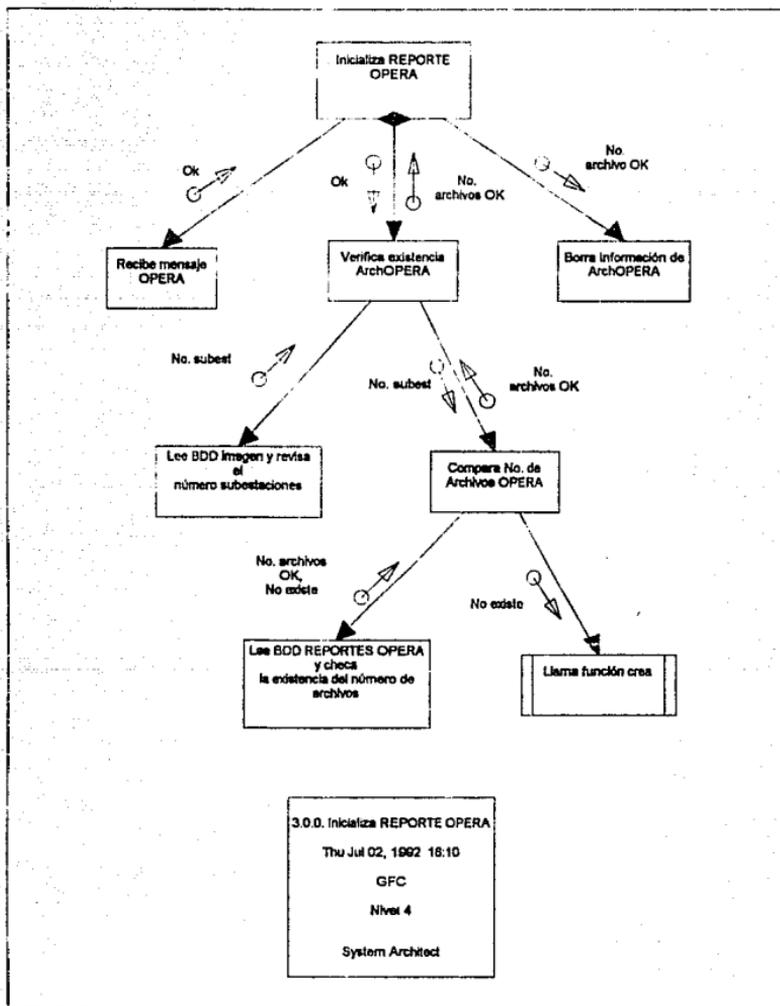


DIAGRAMA 3.20 Inicializa reporte OPERA

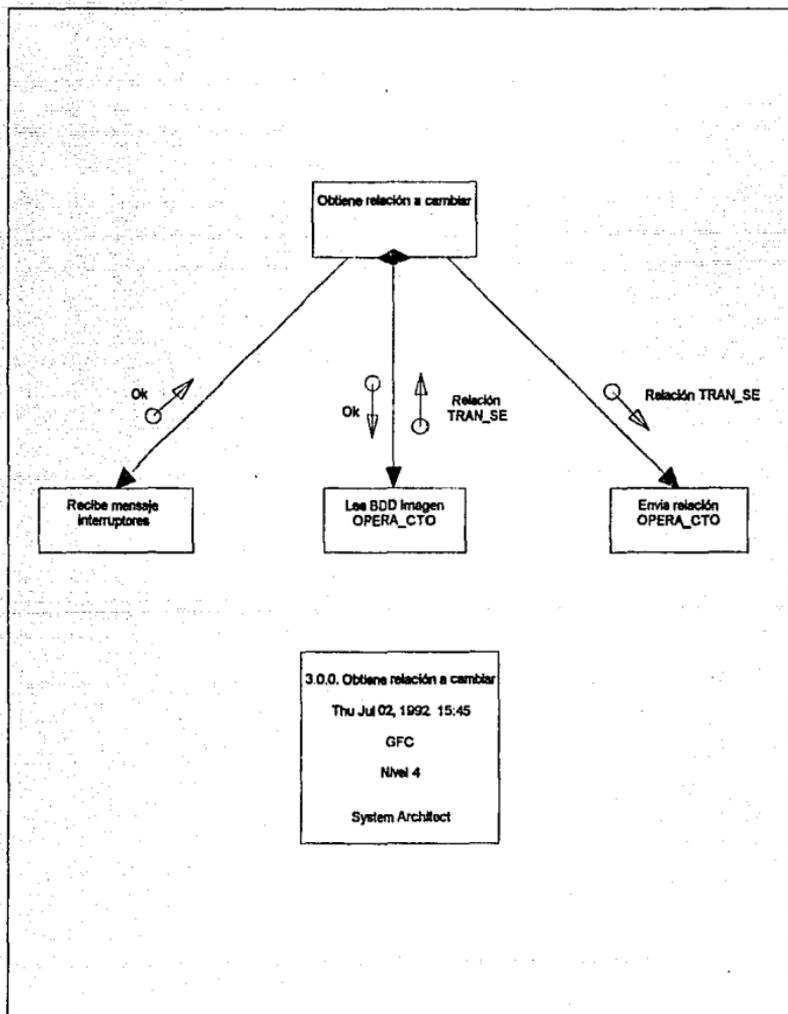


DIAGRAMA 3.21 Obtiene relación a cambiar

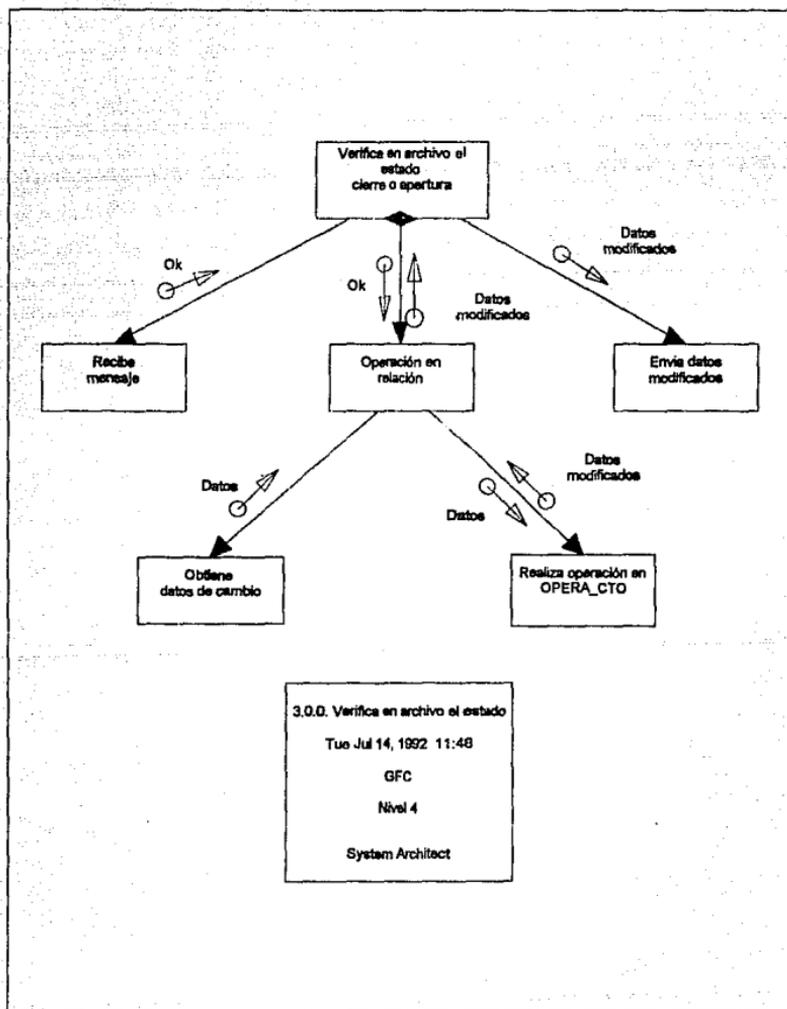


DIAGRAMA 3.22 Verifica en archivo el estado

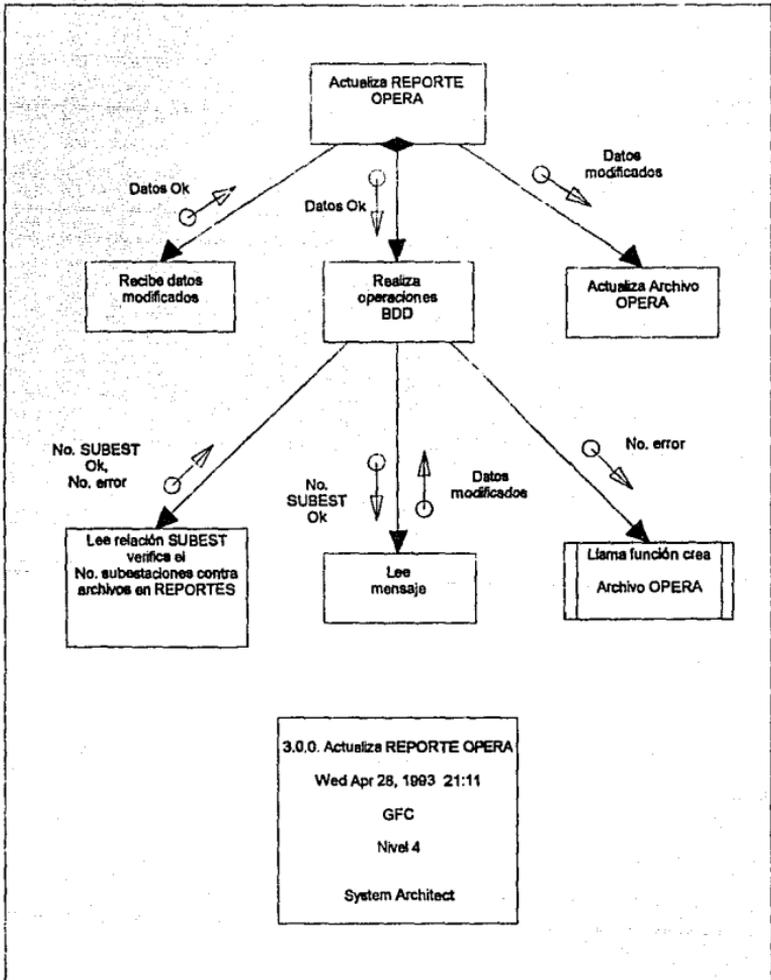


DIAGRAMA 3.23 Actualiza reporte Opera

**Proceso HISTORICO**

Se presenta en el diagrama 3.24 los módulos que conforman a este proceso, así como una breve descripción de cada uno ellos.

**Actualiza reporte HISTORICO**

Este proceso actualiza el archivo del HISTORICO cada 24 horas, con un período de muestreo fijo de 15 minutos, teniendo 96 muestras máximas si el reporte comenzó a la cero horas, sino solo el número que sea muestreado al terminar el día. El reporte genera siete subdirectorios de acuerdo al día de la semana que es generado el reporte, al comenzar la semana nuevamente se reescribe en el primer subdirectorio, la información se almacena en disco para tener los datos históricos de los puntos analógicos durante una semana. Ejecuta la actualización en el archivo en forma inmediata modificando la información de los siguiente campos:

<i>NOMBRE</i>	<i>TIPO</i>	<i>DESCRIPCION</i>
NOM	Caracter (10)	Nombre del punto analógico
NOM SE	Entero (8)	Nombre de la subestación
UNIDADES	Entero (8)	Unidades de ingeniería
VALOR 01	Entero (16)	Valor de muestra 01
VALOR 02	Entero (16)	Valor de muestra 02
...	...	...
...	...	...
VALOR 95	Entero (16)	Valor de muestra 95
VALOR 96	Entero (16)	Valor de muestra 96

**Crea archivo DBF HISTORICO**

Proceso que crea el archivo HISTORICO al iniciar el día, creando el nuevo subdirectorio del día en curso. La estructura utilizada fue definida en el punto anterior y esos son los elementos del reporte Histórico.

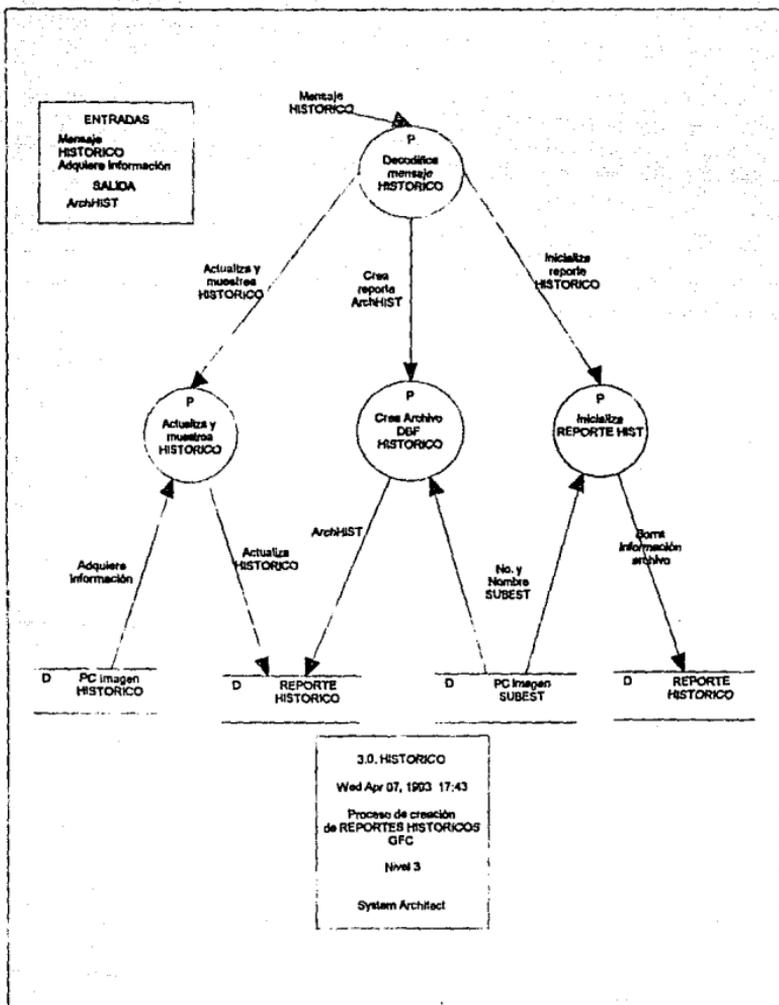


DIAGRAMA 3.24 Proceso de creación del reporte Histórico de Mediciones

### **Decodifica mensaje HISTORICO**

Proceso que enruta el mensaje recibido al módulo encargado de procesarlo. Las opciones del mensaje son las siguientes:

- 1.- Actualiza y muestrea reporte HISTORICO
- 2.- Crear archivo reporte HISTORICO
- 3.- Inicializar base de datos reporte HISTORICO

### **Inicializa reporte HISTORICO**

Proceso que borra la información que contiene el archivo HISTOTRICO de sus 96 muestras si existe, en caso de no existir llama a la función "Crea Archivo DBF HISTORICO".

### **Muestrea y actualiza reporte HISTORICO**

Proceso que envía un mensaje de reconocimiento de activación para actualizar los datos del reporte HISTORICO en forma directa en el archivo, en caso de encontrarse en estado apagado no actualiza los datos ni genera el reporte

## **Capítulo 4**

# **ENLACE Y CONVERSION DE BDD MAC-5000 CON PC**

En este capítulo se realiza un breve análisis de la forma en que debe efectuarse el enlace para la correcta comunicación entre el sistema de cómputo de la EM MAC - 5000 y la computadora personal en que se realizaron los programas para el subsistema de reportes. El objetivo original de este trabajo de tesis no contemplaba la realización de este enlace, por lo que únicamente se muestra la parte teórica.

Además, describe los formatos manejados en el sistema MAC - 5000 y la manera de realizarse la conversión de la BDD a un formato comercial (dBase III plus).

Estos dos puntos se consideran como parte fundamental para la integración total del sistema, ya que de ellos depende el correcto manejo de la información entre la computadora personal y la EM MAC - 5000.

#### 4.1 ENLACE PARA COMUNICACION PC MAC - 5000

Se cuenta con varias opciones para realizar este enlace dependiendo de la manera que se desee implementar. La primera es un enlace punto a punto full-duplex entre la tarjeta MAC-1186 y el puerto serie COM1 o COM2 de la computadora personal. Esto implica la realización de los programas de enlace tanto en la tarjeta MAC-1186 como en la PC. La segunda es utilizar en la canasta MAC una tarjeta de comunicaciones, tal como la MAC-851, en la que se aprovecharía algún protocolo ya definido y probado como lo es el PAC [IBA 89], teniendo únicamente que realizar un programa para la tarjeta MAC-1186 que envíe a este puerto de comunicaciones los datos para su transmisión según las reglas del protocolo. Con esta opción se cuenta con dos alternativas en el lado de la PC; se puede colocar una tarjeta PC-851 con enlace a PC BUS que este ejecutando el protocolo PAC, lo cual descargaría al programa del subsistema de reportes las labores de comunicación con la EM MAC - 5000 y únicamente lo atendería cuando tuviera que enviar o recibir un mensaje. Esta configuración tiene la ventaja de que solo se necesita realizar los programas de direccionamiento al puerto de comunicación sin preocuparse del protocolo y la desventaja de que se necesitan dos tarjetas adicionales para poder realizarse la comunicación. La otra alternativa sería adaptar el protocolo PAC para poderlo ejecutar desde el puerto COM1 o COM2 de la PC, aunque esto pudiera alentar al sistema pues el procesador de la PC tendría que ocuparse de todas las funciones y recordemos que el MS-DOS no es un sistema multitareas, por lo que tiene que finalizar una tarea antes de realizar otra.

Como se observa la mejor opción parece ser mediante el protocolo PAC colocando en la canasta MAC una tarjeta MAC-851 y en la PC una tarjeta PC-851 como se muestra en la figura 4.1. A continuación se da una breve explicación de las características de comunicación en la tarjeta MAC-1186, en la PC, y se explica el protocolo PAC (que soporta el estándar físico RS - 485) con MAC-851 y PC-851 para englobar todas las opciones de comunicación dadas.

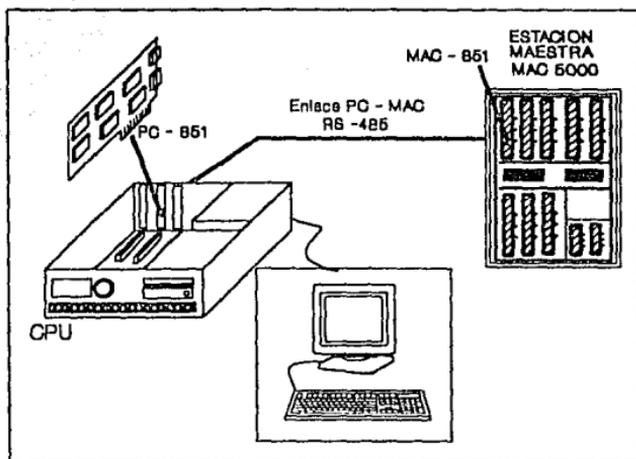


FIGURA 4.1 Enlace PC - MAC usando las tarjeta MAC-851 y PC-851

#### 4.1.1 Norma RS - 485

La norma RS - 485 es un estándar específico de las características eléctricas de transmisores y receptores utilizados en sistemas digitales balanceados en multipunto .

Este estándar especifica las características eléctricas de los transmisores y receptores que son empleados cuando se especifican para el intercambio de señales binarias en una interconexión multipunto de equipos digitales. Cuando se implementa dentro de las recomendaciones de este estándar, se pueden colocar múltiples transmisores y receptores a un cable de interconexión común.

Un sistema de intercambio incluye la conexión de uno o más transmisores por un cable de interconexión balanceado a uno o más receptores y resistores terminales. Las características eléctricas del circuito se especifican en términos de voltaje requerido, la corriente y valores de las resistencias obtenidas por la medición de los puntos interconectados de transmisores y receptores. Las cargas de los circuitos se definen como

el número de o las fracciones de "unidades de carga" que presenta el circuito. Las recomendaciones están dadas con respecto a las limitaciones en la señalización impuestas por los parámetros de: longitud del cable, balance, y las terminaciones para las instalaciones individuales [EIA 83].

Los valores de los parámetros especificados en este estándar son similares aquellos del RS - 422A. Estos parámetros permiten que los transmisores y receptores diseñados puedan utilizarse y cumplan los requerimientos de ambos estándares. Este estándar no especifica otras características tales como: calidad de la señal, tiempos, protocolos, asignamiento a los pines, etc., que son esenciales para la operación adecuada de los equipos interconectados. Se presupone que este estándar tendrá referencias y especificaciones adicionales necesarias para asegurar la interfaz entre los equipos.

La figura 4.2 muestra una aplicación para interconectar transmisores y receptores que cumplan con los parámetros eléctricos. Los elementos en la aplicación son transmisores (T), receptores (R), cables de transmisión y resistencias de terminación ( $R_t$ ).

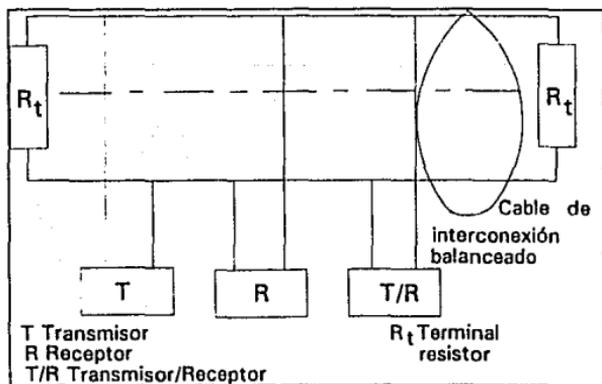


FIGURA 4.2 Interconexión entre transmisores, receptores

#### 4.1.2 Protocolo PAC

A continuación se describe detalladamente el protocolo PAC, se especificarán las características en los diferentes niveles del modelo OSI (Interconexión de Sistemas Abiertos), de la Organización Estándar Internacional (ISO, de sus siglas en inglés) y el modelo propuesto por el Comité de la IEEE 802.

##### *Nivel físico (INAF)*

Este es el nivel más bajo del modelo. Define las características mecánicas, eléctricas, temporales y funcionales para activar, mantener y desactivarse las conexiones físicas para la transmisión de bits. Como ejemplo de normas a este nivel, están los estándares EIA RS-232 y RS-485. Para PAC, las especificaciones del nivel físico para la comunicación, utilizando el estándar EIA RS-485, tienen las características básicas [IBA 89]:

- Comunicación serie, asíncrona byte por byte
- Velocidad de transmisión y recepción de 38 400 bauds
- Datos de 8 bits, bit de arranque, bit de paro, 1 bit indicador de dirección (1) o dato (0)
- Multipunto
- Diagrama de temporización figura 4.3 (a)
- Diagrama de conector figura 4.3 (b)
- Características eléctricas de la red, correspondientes a la norma EIA RS-485.
- Bit menos significativo primero
- Canal de respaldo

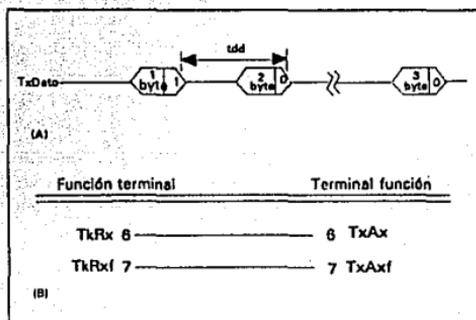


FIGURA 4.3 (a) Diagrama de temporización, (b) Diagrama de conector

### Nivel de Acceso al Medio (NAM)

Este nivel es el encargado de controlar el acceso a la red cuando la topología lo requiera. Por ejemplo, existen protocolos "token Pass", "token ring", maestro-esclavo con "polling", etc. Esta función no existe en el modelo OSI, así que fue la principal causa de la creación de este modelo del Comité IEEE 802. En seguida se definen las funciones del nivel de acceso para el protocolo de adquisición y control.

**Direcciones.** En este nivel se forma la dirección destino del mensaje DIRD, es decir, se le inserta el noveno bit a 1 para indiar a las estaciones receptoras que un mensaje viaja por la red. De inmediato, regresa el bit a, o para el envío de datos.

**Acceso a la red.** Se utiliza una configuración maestro-esclavo, donde el maestro con muestreo (polling) controla el acceso a la red a los esclavos. Estos no pueden iniciar una comunicación, sin haber recibido el acceso a la red. Los comandos para lograr estas funciones tienen el formato que se muestra en la figura 4.4.

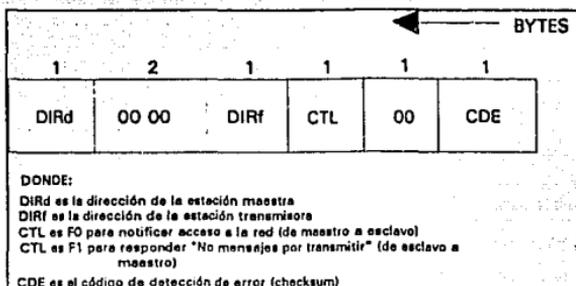


FIGURA 4.4 Comandos de acceso a red

### *Nivel de control de enlace lógico (NEL)*

Este debe ser un nivel único, en el que pueden conectarse varias formas del nivel NAM, dependiendo de los medios. Controla el enlace lógico de las estaciones, estableciendo o terminando el enlace. Sus funciones son empaquetar la información en el formato del mensaje, insertar la dirección destino, formar los campos de los códigos de detección de error apropiados. Asimismo, en recepción reconoce su dirección, desempaqueta la información, y valida la transmisión por medio de los códigos de detección de error. En caso de existir, maneja las retransmisiones, o envía los reconocimientos apropiados. Por último, este es el nivel encargado de conectarse con niveles más altos del usuario/aplicación. Para el protocolo PAC, las funciones NEL son:

- Formato del mensaje (figura 4.5)
- Dirección destino. En caso de redes multipunto, este campo tiene la dirección de la estación receptora
- Longitud/tipo. Estos bytes indican el tipo de información:

Long./tipo = 0 significa mensaje de control cuyo código en el campo SEC/CTL  
 Long./tipo <> 0 indica el número de bytes del campo info que tiene el mensaje

DIRd	LONG/TIPO	DIRf	SEC/CTL	REC/MOD	INFO	CDE
<p><b>DONDE:</b></p> <p>DIRd es dirección de estación destino (1 byte)</p> <p>LONG/TIPO es longitud de mensaje/tipo de mensaje (2 bytes)</p> <p>DIRf es dirección de estación fuente (1 byte)</p> <p>SEC/CTL es número secuencial de mensaje / control (1 byte)</p> <p>REC/MOD es reconocimiento / modificar (1 byte)</p> <p>INFO es bytes de datos (0 - 2048 bytes)</p> <p>CDE es código de detección de errores (1 byte)</p>						

FIGURA 4.5 Formato de mensaje

**Secuencia/control.** Cuando la información es un mensaje, este campo tiene el número secuencial de mensajes que cada estación transmisora ha enviado. La numeración es módulo 256. Cuando la información es un control, el campo tiene el código del control.

**Reconocimientos.** Cuando la información es un mensaje, este campo se utiliza para reconocer positivamente los mensajes recibidos. Cuando la información es control, el campo tiene un argumento modificador de algunos códigos de control.

**Código de detección de error.** Se utiliza el código de detección de suma (checksum), es decir el complemento a dos de la suma módulo 256 de todos los bytes del mensaje.

**Reconocimientos.** Estos pueden ser positivos o negativos. Los primeros pueden enviarse con el formato de comandos o pueden ser parte del mensaje que el receptor tenga que enviar. Los negativos deben enviarse exclusivamente en formato comando. Cualquiera de estos reconocimientos deben enviarse dentro de un tiempo máximo ( $T_{m\acute{a}x}$ ).

**Comandos.** Los códigos de control que gobiernan el sistema de comunicaciones (campo CTL en el formato de comandos) son números entre uno y 255 cuya función está clasificada en tres grupos:

- Códigos para las funciones del nivel NAM (ya mencionados).
- Códigos para las funciones NEL que incluyen sincronía entre las estaciones, funciones de tolerancia a fallas, tales como entrada de la red de respaldos, reconocimientos, etc.
- Códigos con los que los niveles superiores pueden acceder al nivel NEL, como configuración de la red, arranque o paro de la red, informe de fallas, etc.

**Manejo de errores.** Se basa en retransmisiones del mensaje, y tiene las características:

- En caso de vencerse el tiempo fuera ( $T_{m\acute{a}x}$ ) sin recibir nada, el transmisor enviará de nuevo el mensaje (hasta tres veces) antes de considerar fuera de servicio el receptor.
- En caso de detectar un error en el mensaje, o por falta de *buffer*, el receptor enviará reconocimiento negativo. En este caso, el transmisor enviará el mensaje tantas veces como sea necesario.
- En caso de detectar un mensaje incompleto, el receptor regresará un comando de reconocimiento negativo por mensaje incompleto. Al recibir este comando, la estación transmisora envía comandos de integridad de red, hasta recibir un reconocimiento de integridad de receptor, es decir, el aviso de recuperación de sincronía entre mensajes.
- Cuando la estación maestra detecte la falla de un esclavo, sincronizará a toda la red por el canal de respaldo y, en caso de que este se encuentre en buenas condiciones, se envía a todas las estaciones un comando tipo *broadcast* para indicar el cambio de canal. Este modelo puede utilizarse, además, para describir la arquitectura del hardware utilizado en el sistema de comunicaciones. La figura 4.6 muestra la

relación entre los niveles del modelo IEEE con las tarjetas de la familia MAC, diseñadas en el Departamento de Electrónica, del IIE. Realizan los tres niveles inferiores, las tarjetas MAC que a su vez, pueden descomponerse en partes, según su función.

Controla el nivel físico, el circuito inteligente DUART (MC2682) junto con los manejadores RS-485, DS3696N; estos soportan contención y se apegan a las normas EIA RS-485. Las funciones de acceso al medio y enlace lógico son realizadas por la inteligencia del microprocesador 8039 o 8031 (versiones sin ROM del 8048 o 8051), con el que sería posible implantar otros tipos de protocolos en los niveles NAM y/o NEL, ver figura 4.6:

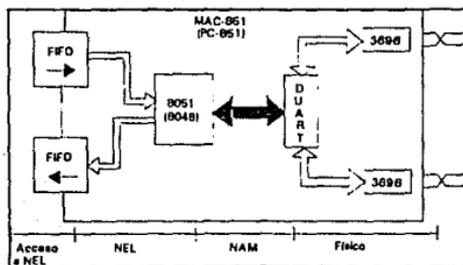


FIGURA 4.6 Tarjeta MAC-851, PC-851

Un aspecto interesante en la implantación del sistema son los puntos de acceso al nivel NEL, ya que, dependiendo de su eficacia, los niveles superiores y la aplicación pueden recibir un servicio eficiente de la red. Para esto, las tarjetas cuentan con dos memorias FIFO (primero en entrar, primero en salir) de 512 bytes de doble acceso, uno desde IBUS III y el otro, desde el procesador interno.

## 4.2 CONVERSION BDD MAC - PC

Para la conversión de las relaciones manejadas en formato MAC para la PC en formato dBase, se llevo a cabo en tres partes: formato MAC, formato dBase y la implementación de conversión de formatos.

#### 4.2.1 Formato MAC (meta\_r y meta\_a)

La base de datos (BDD) de un sistema de control supervisorio es el centro de información más importante, ya que ahí es donde se almacena el estado actual del sistema a controlar, en este caso, una red eléctrica. De ahí que el acceso, control y manipulación de la BDD de una manera eficiente y segura, sea de suma importancia para el desarrollo de las aplicaciones de supervisión que desde luego se realizan bajo un ambiente de tiempo real [GON 88].

El presente punto describe las características más importantes del Manejador de Base de Datos de Tiempo Real (MBAD\_TR), así como una breve introducción al modelo relacional y a sus operadores. Se describen las características particulares del diccionario de datos (DdD) y de cada una de las primitivas que ofrece MBAD\_TR para la manipulación de la BDD. Además, se presentan los formatos que deben seguir los programas de aplicación para acceder a MBAD\_TR. EL MBAD\_TR corre bajo el sistema operativo para procesos concurrentes (SOPCO 86) [MEJ 90] y cuenta con un monitor encargado de controlar la concurrencia de procesos lectores o escritores a la Base de Datos (MONLYE), además es el encargado de la manipulación correcta de los datos en su formato interno (MAC mediante el meta\_r y meta\_a).

##### *Modelo relacional*

La información que se almacena en computadoras se divide, usualmente, en grupos. Dicha información se encuentra en tres grupos comúnmente llamados "archivos".

La manera de agrupar datos, columnas y renglones es el fundamento en el cual se basa el modelo relacional para BDD.

Al conjunto de atributos que se agrupan con el propósito de dar información relevante se le denomina "*relación*". Los atributos de una *relación* puede tomar distintos valores en distintos momentos, al conjunto de valores que puede llegar a tomar un atributo se le denomina "*dominio*" del atributo.

Si tomamos un renglón cualquiera de las tablas mostradas, observamos que el renglón completo nos describe un ente en un momento específico. A los renglones de una relación se les conoce con el nombre de "*tuple*", otro término utilizado también es el de "*registro*".

Un tuple puede identificarse inequívocamente a partir del valor de uno o unos de sus atributos. A el (los) atributo(s) que forman tal identificación se le llama "*llave*" de la relación. En una base de datos Modelo Relacional, se tiene como regla que: "*El valor de la llave dentro de la relación debe ser único*", esto es no se permite que haya tuples repetidos.

La decisión de como agrupar y organizar la información de una aplicación dada es dictada normalmente por la propia aplicación y por las previsiones de información del Administrador de la Base de Datos. Esto es, la información se almacena de tal manera que más tarde pueda accederse en una forma sencilla y ágil con el fin de responder a preguntas de los usuarios.

#### *Diccionario de datos*

El diccionario de datos en un sistema de BDD es el lugar en el cual se encuentra *información referente a los datos* que van a ser manipulados. El diccionario de datos bajo el modelo relacional debe proveer información a cerca de las relaciones y atributos que forman la BDD.

En MBAD\_TR el diccionario de datos se compone de dos conjuntos de información, uno referente a las relaciones y otro a los atributos que componen a esas relaciones. Al primero de ellos se le denomina META\_RELACION y al otro META\_ATRIBUTO. El sufijo META se le dá debido a que dichos conjuntos contienen información de otros datos y además se les manipula como datos mismos, esto es, tienen dos puntos de vista.

La información que provee cada conjunto se describe en las dos secciones siguientes:

**Meta\_relación**

La entidad META\_RELACION contiene los descriptores de todas las relaciones del sistema, además de el descriptor de ella misma y el de la entidad META\_ATRIBUTO.

Los atributos que definen a esta RELACION de RELACIONES, son básicamente todas las características que definen a cualquier relación, además de otros que son necesarios para la implementación física del DICCIONARIO.

IDREL	Identificador a cada relación por medio de un número.
NOMREL	Identificador a cada relación por su nombre.
NUMATR	Indicar el orden de la relación (número de atributos).
NUMTUP	Indicar la cardinalidad de la relación (número de tuples)
LONTUP	Indicar la longitud del tuple en BYTES
DARCH	Apuntar al archivo de datos
DATRIB	Apuntar al inicio del arreglo de descriptores de atributos de la relación.
ORDLLA	Indicar si la llave es continua, discontinua o desordenada.
ATRLLA	Indicar el número de atributos que forman la llave
LONLLA	Indicar la longitud de la llave en bytes.

**Meta\_atributo**

La entidad META\_ATRIBUTO contiene los descriptores de los atributos de todas las relaciones del sistema, incluyendo los de ella misma y los de la entidad META\_RELACION.

Los atributos que definen a esta *relación de atributos* son, básicamente, todas las características que definen a cualquier atributo, además de otros que son necesarios para la implementación física del diccionario:

IDREL	Ligar a un atributo con su relación.
IDATR	Indicar la posición del atributo en la relación.
NOMATR	Identificar a un atributo por su nombre.
DRELAC	Apuntar al inicio del descriptor de su relación.
TIPDAT	Identificar el tipo de datos que maneja.
LONATR	Indicar la longitud del atributo en BYTES.
IDLLA	El atributo es parte de la llave
OFFSET	Distancia al inicio de la relación en BYTES.

#### *Estado inicial del diccionario de datos*

El diccionario de datos para una BDD bajo MBAD\_TR debe tener inicialmente 2 relaciones y 18 atributos.

Lo anterior obedece a que el *diccionario de datos* es auto-describible, la primer relación en el diccionario debe corresponder a la entidad META-RELACION la cual se auto-describe y describe a la entidad META-ATRIBUTO. Enseguida la entidad META-ATRIBUTO que describe a los atributos de la entidad META-RELACION y los suyos propios. Posteriormente el diccionario de datos va evolucionando conforme se agregan las relaciones de la aplicación.

#### **4.2.2 Formato dBase III plus**

Code base [SEQ 90] es una biblioteca del lenguaje C para la administración de las bases de datos. Su compatibilidad con dBase III plus le permite acceder archivos de cualquier otro producto compatible, ya que el code base utiliza las mismas formas de acceder y crear los archivos.

Usando el code base, es posible almacenar y acceder cientos de megabytes de información en la PC. Esta información se almacena en el disco en archivos de base de datos. Es posible tener varios archivos DBF abiertos a la vez, sin embargo, sólo uno se accede a la vez. Las rutinas trabajan con la base seleccionada.

La razón de tener varios archivos abiertos a la vez, es que abrir y cerrar bases de datos implica un gran tiempo de procesamiento, pero seleccionar entre bases de datos abiertas es casi instantáneo.

Para cada base abierta existe un área en memoria llamada el buffer del registro. Para acceder y modificar este registro existen rutinas que trabajan sobre los campos y cuando una rutina modifica el registro, los cambios se escriben directamente al disco.

Cada base de datos tiene un conjunto de índices abiertos, cada archivo índice corresponde a una base de datos específica y cuando la base se modifica, todos los archivos índices son actualizados automáticamente.

Para entender como opera el code base es útil conocer las bases de su estructura interna. Las cuales se describen a continuación:

#### *Formato del buffer del registro*

Un registro del code base usa el mismo formato de almacenamiento que los registros almacenados en disco. Inician con una bandera que indica que si el registro esta borrado lógicamente, esta bandera normalmente es un espacio y es un asterisco (\*) cuando el registro esta marcado para borrarse. Después de esta bandera están todos los campos de la base de datos en orden consecutivo, no hay caracteres especiales delimitando los datos.

Todos los tipos de campos se almacenan como ASCII. A continuación se muestran los tipos de campos y la forma en que se almacenan:

Tipo de campo	Formato de almacenamiento
Numérico	ASCII justificado a la derecha
Punto flotante	Igual que el numérico
Tipo fecha	CCYYMMDD (Century, Year, Month, Day)
Lógicos	T, t, 'Y' y 'y' para verdadero
Tipo memo	10 caracteres de ASCII representando una posición en el archivo memo correspondiente.

#### Número de referencia de campo

La estructura de la base de datos, de tipo "base", contiene un puntero a un arreglo a estructura de tipo "campo". Este arreglo se define por la instrucción *dUse*, la cual contiene toda la información del campo asociado con la base de datos. La estructura del *campo* es definida de la siguiente manera:

```

type struct
{
    char nombre(11); /* Nombre del campo */
    int tipo; /* Tipo del campo */
    char ancho; /* Ancho del campo */
    int decimales; /* Número de decimales */
    int offset; /* Offset dentro del buffer del registro */
} FIELD;

```

#### 4.2.3 Implementación de conversión de formatos

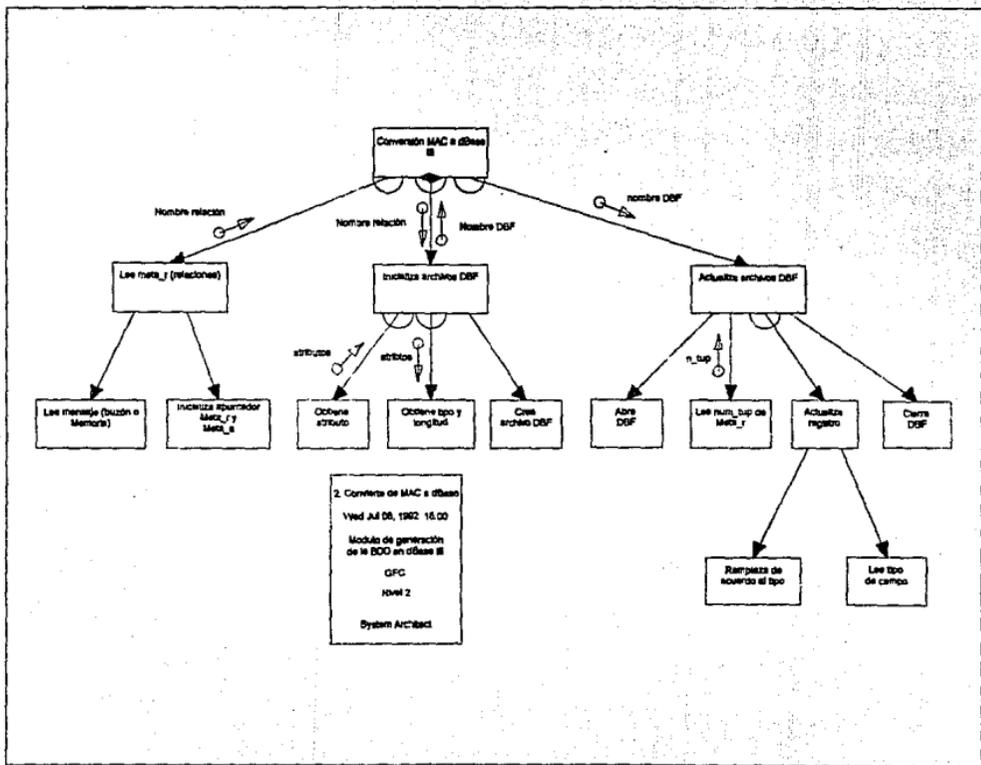
En el diagrama 4.1 se muestra a detalle el módulo de conversión de formata MAC a dBase, este diagrama de estructura se realizó a gran detalle, de manera que cada módulo pueda implementarse de manera rápida a cualquier lenguaje.

Se describe a continuación los tres módulos principales, ya implementados, con el nombre de las funciones del código fuente.

En el siguiente módulo se convierte los archivos de formato MAC a dBase a partir de las relaciones contenidas en las estructuras *meta\_r* y *meta\_a*, implícitamente crea los archivos DBF, los nombres de los campos de los nuevos archivos y actualiza los valores con los registrados en las relaciones.

Se realizaron tres funciones para la conversión de formato MAC a dBase:

- La función *conv\_dbf* crea todos los archivos DBF en base a los datos de la relación *meta\_r*. Se realiza un ciclo para convertir todas las relaciones especificadas en *meta\_r*, exceptuando *meta\_r* y *meta\_a* en archivos DBF, de *meta\_r* obtiene las direcciones donde se encuentran los atributos, el número de atributos que tiene esa relación y el nombre de la relación para crear un nuevo archivo pero con extensión DBF. Realiza una serie de llamadas a otras rutinas para convertir de relación en relación y actualizar sus datos de la misma manera.
- La función *conv\_relación* convierte una relación a un archivo DBF a partir de los datos que se pasan como parámetros (dirección de los atributos y número de atributos que la forman) y de los datos contenidos en *meta\_a*. En esta función se tienen las equivalencias de los tipos de datos utilizados en el formato MAC y los correspondientes en el Code Base para el formato dBase.
- Por último, la función *actual\_dbf* extrae los datos contenidos en memoria y apuntados por *meta\_r* y *meta\_a* para actualizar los nuevos archivos DBF creados.



## **Capítulo 5**

# **PRUEBAS E INTEGRACION DEL SISTEMA**

En este capítulo se presentan las pruebas tanto por módulos, como de la integración del sistema para la completa verificación de los siguientes puntos:

- Pruebas de los módulos principales.
- Módulo de conversión del formato MAC a formato dBase.
- Activación de los reportes.
- Reloj.
- Verificación de operaciones en los reportes.
- Almacenamiento cada 24 horas.
- Solicitud de impresión.
- Creación mensual de los archivos de reportes .
- Estabilidad del sistema.
- Graficación del reporte Histórico de Mediciones.
- Presentación en pantalla.

## 5.1. PRUEBAS DE LAS FUNCIONES PRINCIPALES

La siguiente prueba se realizó para cada módulo al concluir su codificación, las pruebas realizadas fueron de funcionalidad y estabilidad por función.

Las funciones a las que se le realizaron las pruebas son las siguientes: Crea, Muestra, Actualiza, Inicializa, Mensaje de interruptores.

### CREA

Para esta función se asignaron nombres correspondientes para cada archivo de los reportes, el cual para crearlo se definieron las estructuras con los campos necesarios para cada reporte, esto esta definido en el módulo "Const.h", a continuación se define un ejemplo de estas estructuras:

```
FIELD opin_ff] = {
/*NOMBRE      , TIPO , WIDTH , DEC , OFFSET*/
{"NOM_SE"     , 'N' , 16 , 0 , 0 },
{"NOM  "      , 'N' , 11 , 0 , 0 },
{"DIA   "     , 'N' , 3 , 0 , 0 },
{"FALLAS "    , 'N' , 10 , 0 , 0 },
{"MANIOBRAS , 'N' , 10 , 0 , 0 }
};
```

En este caso se define la estructura para el archivo de Operación de Interruptores, una vez definida la estructura se verifica si el archivo esta creado en el subdirectorio correspondiente, para lo cual se realizó una función que checa el nombre del directorio OPIN para este caso en particular (VMMV, VMMI, OPERA, HISTORICO y COME para los otros), y verificando, si el subdirectorio de mes actual se ha creado y que contenga todos los archivos con los nombres de las subestaciones definidas para ese reporte.

La prueba de este módulo consiste simular, la creación de estos archivos solicitandolo manualmente, posteriormente se incluyó esta función en el módulo de "Reloj" donde esto era verificado periódicamente.

## MUESTREO

Esta función consiste de recolectar datos de las relaciones ACUMULADO, ANALOGICO y DIGITAL definidas para almacenar la información adquirida de campo. Para realizar la prueba de esta función se necesito que la BDD de las relaciones tuviera algunos datos para realizar por medio del módulo de "Reloj" y una función temporal "Cont\_valor" que incrementaba y decrementaba un valor base, almacenando este valor en las relaciones ANALOGICO y ACUMULADO que son necesarias para el muestreo de los reportes: VMMV, VMMI, COME, OPIN y OPERA.

Una vez inicializadas las relaciones con el valor base comienza la función a verificar los datos de las relaciones definidas anteriormente con sus respectivas relaciones asociadas: BARRA, CORRI\_SUB, INT\_SE y TRAN\_SE respectivamente y almacenar los datos necesarios en estas para su posterior uso en la función de *actualización*.

## ACTUALIZA REPORTES

Para realizar esta función se requirió antes la simulación del muestreo para cada una de las relaciones correspondientes para cada archivo de los reportes, una vez que en las relaciones se almacena la información, se solicitaba manualmente la actualización de cada reporte y posteriormente en el módulo de "Reloj" se incluyo una función que asigna una variable para cada reporte la cual verificaba que se cumpliera un tiempo predefinido y realizaba satisfactoriamente esta función.

## INICIALIZA

Esta función sólo requirió que los archivos de los reportes hubieran sido actualizados, conteniendo información esta función lo único que hace es borrarla de todos los campos definidos para cada reporte: VMMV, VMMI, COME, OPIN, OPERA e HISTORICO.

## 5.2. CONVERSION DEL FORMATO MAC A FORMATO DBASE

La prueba de conversión del formato a dBase se realizó mediante la simulación del formato MAC dentro de la PC realizando el siguiente procedimiento:

1. Se crearon dos estructuras de datos, la primera de ellas tiene campos idénticos a los del formato Meta\_r con la misma longitud para cada uno de los campos correspondientes. Esta estructura se inicializó con siete registros en los cuales se describen las cinco relaciones que se desean convertir, además del Meta\_r y el Meta\_a, se definieron cinco de las relaciones que normalmente se utilizan en el subsistema de reportes.
2. En la estructura Meta\_a se inicializaron los campos correspondientes a todos y cada una de las estructuras definidas en el Meta\_r.
3. Se definió un encabezado de mensaje, el cual contenía la longitud correspondiente a la suma de Meta\_a, Meta\_r y el encabezado mismo. Además de la función y los demás parámetros del mensaje.
4. Se colocó en una área de memoria reservada los datos correspondientes al encabezado, Meta\_r y Meta\_a, lo que simula el mensaje recibido por el módulo de enlace y se le proporciona la dirección de inicio al programa de conversión de formato, el cual se encarga de crear las bases de datos en formato dBase.
5. Una vez creada la base de datos en formato dBase, se verificó que los campos y los datos contenidos en ellas correspondieran a lo especificado en el formato MAC. Esta prueba se realizó utilizando el dBase III plus y el Xtree Gold para verificar que los formatos fueran 100% compatibles.

A continuación se muestran las estructuras Meta\_r y Meta\_a utilizadas para esta prueba, y los valores con los que se inicializaron.

```

struct META_R {
    unsigned char idrel;
    char nomrel[9];
    unsigned char numatr;
    int numtup;
    int lontup;
    char *darch;
    char *datrib;
    unsigned char ordlla;
    unsigned char atrlla;
    unsigned char lonlla;
} meta_r[MAXREL];

```

```

struct META_A {
    unsigned char idrel;
    unsigned char idatr;
    char nomatr[9];
    char *drelac;
    unsigned char lipdat;
    unsigned char lonatr;
    unsigned char idlla;
    int offset;
} meta_a[MAXATR] = {

```

```

{ 1,1,"idrel ",(char *) &meta_r[0],1,1,1,0 },
{ 1,2,"nomrel ",(char *) &meta_r[0],0,9,1,1 },
{ 1,3,"numatr ",(char *) &meta_r[0],1,1,1,10 },
{ 1,4,"numtup ",(char *) &meta_r[0],2,2,1,11 },
{ 1,5,"lontup ",(char *) &meta_r[0],2,2,1,13 },
{ 1,6,"darch ",(char *) &meta_r[0],5,4,1,15 },
{ 1,7,"datrib ",(char *) &meta_r[0],5,4,1,19 },
{ 1,8,"ordllav ",(char *) &meta_r[0],1,1,1,23 },
{ 1,9,"atrilav ",(char *) &meta_r[0],1,1,1,24 },
{ 1,10,"lonllav ",(char *) &meta_r[0],1,1,1,25 },

```

```
{ 2,1,"idrel ",(char *) &meta_r[1],1,1,1,0 },
{ 2,2,"idatr ",(char *) &meta_r[1],1,1,1,1 },
{ 2,3,"nomatr ",(char *) &meta_r[1],0,9,1,2 },
{ 2,4,"drelac ",(char *) &meta_r[1],5,4,1,11 },
{ 2,5,"tipdat ",(char *) &meta_r[1],1,1,1,15 },
{ 2,6,"lonatr ",(char *) &meta_r[1],1,1,1,16 },
{ 2,7,"idlla ",(char *) &meta_r[1],1,1,1,17 },
{ 2,8,"offset ",(char *) &meta_r[1],2,2,1,18 },

{ 3,1,"numutr ",(char *) &meta_r[2],1,1,1,0 },
{ 3,2,"nomutr ",(char *) &meta_r[2],0,10,1,1 },
{ 3,3,"status ",(char *) &meta_r[2],1,1,1,11 },
{ 3,4,"protoc ",(char *) &meta_r[2],1,1,1,12 },
{ 3,5,"veloci ",(char *) &meta_r[2],1,1,1,13 },
{ 3,6,"numsub ",(char *) &meta_r[2],1,1,1,14 },
{ 3,7,"diagno ",(char *) &meta_r[2],3,4,1,15 },
{ 3,8,"ibarrí ",(char *) &meta_r[2],1,1,1,19 },
{ 3,9,"ninten ",(char *) &meta_r[2],1,1,1,20 },

{ 4,1,"numsub ",(char *) &meta_r[3],1,1,1,0 },
{ 4,2,"nomsus ",(char *) &meta_r[3],0,10,1,1 },
{ 4,3,"locali ",(char *) &meta_r[3],0,20,1,11 },
{ 4,4,"numuni ",(char *) &meta_r[3],1,1,1,31 },

{ 5,1,"numpto ",(char *) &meta_r[4],2,2,1,0 },
{ 5,2,"nompto ",(char *) &meta_r[4],0,10,1,2 },
{ 5,3,"tipole ",(char *) &meta_r[4],1,1,1,12 },
{ 5,4,"proala ",(char *) &meta_r[4],1,1,1,13 },
{ 5,5,"tipdig ",(char *) &meta_r[4],1,1,1,14 },
{ 5,6,"tiplic ",(char *) &meta_r[4],1,1,1,15 },
{ 5,7,"inhibi ",(char *) &meta_r[4],1,1,1,16 },
{ 5,8,"piori ",(char *) &meta_r[4],1,1,1,17 },
{ 5,9,"edoact ",(char *) &meta_r[4],1,1,1,18 },
{ 5,10,"numsub ",(char *) &meta_r[4],1,1,1,19 },
{ 5,11,"edofun ",(char *) &meta_r[4],1,1,1,20 },
```

```

{ 5,12,"ndiatab ",(char *) &meta_r[4],1,1,1,21 },
{ 5,13,"visual ",(char *) &meta_r[4],1,1,1,22 },
{ 5,14,"bdarec ",(char *) &meta_r[4],1,1,1,23 },
{ 5,15,"edonor ",(char *) &meta_r[4],1,1,1,24 },
{ 5,16,"nomlar ",(char *) &meta_r[4],1,1,1,25 },
{ 5,17,"logica ",(char *) &meta_r[4],1,1,1,26 }
), *atributo;

```

```
char *Utr,*Subest,*Digital;
```

```
void inicializa()
```

```

{
meta_r[0].idrel = 1;           meta_r[1].idrel = 2;
strcpy(meta_r[0].nomrel,"meta_r ");  strcpy(meta_r[1].nomrel,"meta_a ");
meta_r[0].numatr = 10;       meta_r[1].numatr = 8;
meta_r[0].numtup = 5;        meta_r[1].numtup = 48;
meta_r[0].lontup = 26;       meta_r[1].lontup = 20;
meta_r[0].darch = (char *) &meta_r[0]; meta_r[1].darch = (char *) &meta_a[0];
meta_r[0].datrib = (char *) &meta_a[0]; meta_r[1].datrib = (char *) &meta_a[10];
meta_r[0].ordfla = 0;        meta_r[1].ordfla = 0;
meta_r[0].atrla = 1;         meta_r[1].atrla = 1;
meta_r[0].lonlla = 1;        meta_r[1].lonlla = 1;

meta_r[2].idrel = 3;         meta_r[3].idrel = 4;
strcpy(meta_r[2].nomrel,"utr ");  strcpy(meta_r[3].nomrel,"subest ");
meta_r[2].numatr = 9;        meta_r[3].numatr = 04;
meta_r[2].numtup = 18;       meta_r[3].numtup = 15;
meta_r[2].lontup = 21;       meta_r[3].lontup = 32;
meta_r[2].darch = &Utr[0];    meta_r[3].darch = &Subest[0];
meta_r[2].datrib = (char *) &meta_a[18]; meta_r[3].datrib = (char *) &meta_a[27];
meta_r[2].ordfla = 0;        meta_r[3].ordfla = 0;
meta_r[2].atrla = 1;         meta_r[3].atrla = 1;
meta_r[2].lonlla = 1;        meta_r[3].lonlla = 1;

```

```

meta_r[4].kirel = 5;
strcpy(meta_r[4].nomrel,"digital ");
meta_r[4].numatr = 17;
meta_r[4].numtup = 10;
meta_r[4].iontup = 32;
meta_r[4].darch = &Digital[0];
meta_r[4].datnib = (char *) &meta_a[31];
meta_r[4].ordlla = 1;
meta_r[4].airlla = 1;
meta_r[4].ionlla = 1;
}

```

A continuación, se muestran las estructuras para las bases de datos en dBase obtenidas mediante el dBase III plus.

```

Structure for database:   C:\subest.dbf
Number of data records:  15
Date of last update :   04/28/93

```

Field	Field Name	Type	Width	Dec
1	NUMSUB	Numeric	3	
2	NOMSUB	Character	10	
3	LOCALI	Character	20	
4	NUMUNI	Numeric	3	
** total **			37	

```

Structure for database:   C:\utr.dbf
Number of data records:  18
Date of last update :   04/28/93

```

Field	Field Name	Type	Width	Dec
1	NUMUTR	Numeric	3	
2	NOMUTR	Character	10	
3	STATUS	Numeric	3	
4	PROTOC	Numeric	3	
5	VELOCI	Numeric	3	
6	NUMSUB	Numeric	3	

7	DIAGNO	Numeric	10
8	TBARR	Numeric	3
9	NINTEN	Numeric	3
** total **			42

Structure for database: C:\digital.dbf

Number of data records: 10

Date of last update : 04/28/93

Field	Field Name	Type	Width	Dec
1	NUMPTO	Numeric	5	
2	NOMPTO	Character	10	
3	TIPELE	Numeric	3	
4	PROALA	Numeric	3	
5	TIPDIG	Numeric	3	
6	TIPLIC	Numeric	3	
7	INHBI	Numeric	3	
8	PRIORI	Numeric	3	
9	EDOACT	Numeric	3	
10	NUMBSUB	Numeric	3	
11	EDOFUN	Numeric	3	
12	INDTAB	Numeric	3	
13	VISUAL	Numeric	3	
14	BDARR	Numeric	3	
15	EDONOR	Numeric	3	
16	NUMLAR	Numeric	3	
17	LOGICA	Numeric	3	
** total **			61	

### 5.3. ACTIVACION DE LOS REPORTES

Esta prueba se realizó con la ayuda del módulo de "Reloj", en el cual se verifican periódicamente las variables de activación de cada uno de los reportes. La revisión de estos valores en pantalla se hizo utilizando la función "Revisión BDD" donde se obtienen los valores de la relación "Reportes".

Efectuado esto, se procede a cambiar el estado de activación de cada reporte utilizando la función "Muestreo" en su campo *activación/desactivación*, realizando uno a uno estos cambios se puede verificar que al llegar su tiempo se activen o permanezcan sin ningún cambio. Existe un botón para cada reporte que indica visualmente cuando se activa, además de tener una señal audible. El orden en que se realizó la prueba es el siguiente:

OPIN	OPERA
VMMV	HISTORICO
VMMI	COME

La prueba efectuada implicó a las funciones de: muestreo, inicialización, actualización e impresión de los reportes, ya que una vez desactivado no se realiza ninguna de las funciones anteriores, además se almacena la hora y fecha de desactivación en el archivo "Períodos", para tener un control de estas desactivaciones de los reportes.

### 5.4. RELOJ

En esta prueba se pretende conocer la eficiencia del módulo de "Reloj", ante periodos de activación de los reportes. Para esta prueba se utilizaron unas bases de datos semejantes a las utilizadas en la EM MAC - 5000 que opera actualmente en la Ciudad de Morelia.

Se probaron diferentes tiempos de activación de los reportes, haciendo este cambio en forma manual mediante la función "Muestreo", además se simuló la recepción del mensaje de interruptores, obteniendo las siguientes conclusiones:

1. Para bases de datos tan extensas como las utilizadas, el tiempo de muestreo debe ser mayor de un minuto para todos los reportes, ya que con un minuto, el programa queda en un ciclo infinito, lo cual deshabilita la interfaz de usuario, porque al finalizar un reporte, se activa inmediatamente el siguiente reporte que se encuentra en la cola de espera, lo que no da oportunidad a volver a tener control sobre el sistema.
2. Aunque este periodo de muestreo depende en gran parte del tamaño de la BDD y la velocidad de la PC, es aconsejable utilizar periodos de muestreo mayores de dos minutos.
3. Una observación importante, es que el periodo de muestreo se realiza para cada reporte, aunque el tiempo definido se haya cumplido, este entra en una cola de espera, la cual asegura que se ejecutará el muestreo cuando el procesador este disponible.
4. La secuencia utilizada para cada uno de los reportes es la siguiente:
  - a) Verifica si existe mensaje de interruptor para actualizar el reporte de Operación de Circuitos e incrementar el contador de la relación INTE\_SE en la cuenta de maniobras.
  - b) Verifica si el tiempo de muestreo ha llegado a su periodo estipulado para realizar los reportes en el siguiente orden (o dependiendo del periodo de muestreo):

Reporte VMM  
Reporte VMMI

Reporte COME  
Reporte Histórico

## 5.5. VERIFICACION DE OPERACIONES EN LOS REPORTES

Para la realización de esta prueba, se presentan unas tablas que proporcionan la información sobre la que se efectuaron las operaciones, indicando la operación y los resultados obtenidos:

### - Reporte VMMV

Para llevar a cabo esta operación se simuló la entrada de valores aleatorios, esto se realizó por medio de la función "Cont\_valor" que utiliza un valor base (1000) que es almacenado en la relación de ANALOGICO al arrancar el sistema por primera vez.

Al realizar el primer muestreo el valor es obtenido de la relación ANALOGICO y es comparado con el valor del 50% del voltaje nominal de una barra específica de la relación BARRA, si el valor es mayor o igual que este, será almacenado en la relación BARRA en los campos V\_MAX y V\_MIN por ser el primer valor de muestreo, de no ser así, no se almacena ningún valor. Para la siguiente muestra se lleva a cabo la función "Cont\_valor" que ahora incrementa en uno el valor base y lo almacena en la relación ANALOGICO siendo ahora este el nuevo valor para el siguiente muestreo, se repite la comparación con el valor del 50% del voltaje nominal de la barra específica, si este valor es mayor o igual pasará a la siguiente función en la cual se verifica si hay un valor almacenado en este registro, de ser así, es comparado primeramente con el valor almacenado en la relación BARRA en el campo V\_MAX, si es mayor se reescribe en este campo, en caso contrario se comparará con el campo V\_MIN de la misma barra, si el valor es menor que el almacenado, es sustituido por él, siendo este valor el mínimo; pero si el registro está vacío se almacena el valor en los dos campos como se hizo al iniciar la operación del sistema.

Esta operación es periódicamente realizada para la obtención de los valores máximo y mínimo del reporte.

La siguiente tabla ejemplifica esta operación para la barra V-110-B1:

No. Muestras	BARRA V-110-B1		
	Valor muestra	V MAX	V MIN
1	1000	1000	1000
2	1001	1001	1000
3	999	1001	999
4	1002	1002	999
5	998	1002	998

#### - Reporte VMMI

Como en el anterior reporte se utilizó la función "Cont\_valor" para simular la entrada de valores con un valor base de 1000 que es almacenado en la relación de ANALOGICO al poner en operación el sistema por primera vez.

Para realizar el primer muestreo el valor es obtenido de la relación ANALOGICO y es comparado con el valor 66, si es mayor o igual se almacena en la relación CORRI\_SUB en los campos I\_MAX y I\_MIN por ser el primer valor muestreado, de no ser así, se almacena el valor de 66 o 0.005 amperes. Para la siguiente muestra se lleva a cabo la función "Cont\_valor" que ahora incrementa en uno el valor base y lo almacena en la relación ANALOGICO siendo ahora este el nuevo valor para el siguiente muestreo, se repite la comparación con el valor 66, si este valor es mayor o igual pasará a la siguiente función en la cual se verifica si hay un valor almacenado en este registro de ser así, es comparado primeramente con el valor almacenado en la relación CORRI\_SUB en el campo I\_MAX, si es mayor se reescribe en este campo, en caso contrario se comparará con el campo I\_MIN del mismo circuito, si el valor es menor que el almacenado, es sustituido por el de 66, siendo este valor el mínimo; pero si el registro esta vacío se almacena el valor en los dos campos como se hizo al iniciar la operación del sistema.

Esta operación es repetida periódicamente para la obtención de los valores máximo y mínimo de corriente de los reportes.

La siguiente tabla ejemplifica esta operación para el circuito 73230:

No. Muestras	CORRI_SUB 73230		
	Valor muestra	I MAX	I MIN
1	1000	1000	1000
2	1001	1001	1000
3	999	1001	999
4	1002	1002	999
5	998	1002	998

#### - Reporte OPIN

Para llevar a cabo esta operación se simuló la entrada del mensaje de interruptores por medio de la función "REPORTES" en su campo *MSJE INT* en el cual se introduce el número del punto y el control completo (SI/NO), es decir, si es control completo se realiza una maniobra en caso contrario es una falla.

El reporte de Operación de Interruptores es el encargado de contabilizar estas dos operaciones, para llevar a cabo la cuenta de estas operaciones se realizó una función, en la cual, una vez verificado el control completo, busca con el punto digital el valor asociado en la relación ACUMULADO para obtener el valor de la cuenta para ese interruptor, una vez obtenido el valor, la función lo incrementa en uno y lo almacena en esta relación y también en la relación INTE\_SE verificando antes si esta vacío el campo, de no ser así lo incrementa.

En caso de no ser control completo sólo incrementa el campo de "cuenta" en la relación ACUMULADO, con el fin de, obtener a las 24 hrs la cuenta total del día para los campos de maniobras y fallas, siendo este último obtenido de la resta de los siguientes campos: "cuenta total" de la relación de ACUMULADO menos "maniobras" de la relación INTE\_SE igual a las fallas ocurridas durante el día. Para este fin se utilizó otra función que genera el mensaje de interruptores para un punto específico, llevando a cabo las cuatro posibles combinaciones para el cambio de estado de interruptor Si/No (apertura/cierre) y la causa de ello (maniobra/falla), utilizando el módulo "Reloj" para

generar periódicamente este mensaje, además de cambiar la hora por medio de la función "Muestreo" para poder generar el reporte a las 24 hrs y con esto checar si los datos eran procesados correctamente y obteniendo la cuenta correcta en ese día.

Para lo anterior se genero una tabla donde se muestran las cuatro posibilidades y el contador.

Muestra	Mensaje de interruptores Interruptor 73170	Causa INTE SE	Cuenta en ACUMULADO
1	Control completo/Apertura	maniobra	1
2	Sin control completo/Apertura	falla	2
3	Contro completo/Cierre	maniobra	3
4	Sin control completo/Cierre	falla	4

#### - Reporte COME

Para llevar a cabo esta operación se simuló la entrada de valores aleatorios, esto se realizó por medio de una función "Cont\_come" que utiliza un valor base (500) que es almacenado en la relación de ACUMULADO al arrancar el sistema por primera vez.

Al realizar el primer muestreo el valor es obtenido de la relación ACUMULADO y es almacenado en la relación TRAN\_SE en la cual se verifica la unidad correspondiente a este valor (KWH o KVARH), una vez verificado esto, se checa si existe un valor para ese transformador registrado, si existe se realiza una suma con valor anterior y el actual, siendo el resultado el valor finalmente que se almacena en la realación, de no ser así sólo es almenado. Estas operaciones se repiten ahora para la otra unidad definida.

Para el siguiente muestreo la función "Cont\_come" incrementa con uno el valor base y repite la operación anterior periódicamente para la obtención de los valores de los transformadores de este reporte mensualmente.

La siguiente tabla ejemplifica esta operación para el transformador 42010 y 42020:

No. Muestras	TRAN_SE 42010 y 42020		
	Valor muestra	KWH	KVARH
1	500	500	500
2	501	1001	1001
3	502	1503	1503
4	503	2006	2006
5	504	2510	2510

#### - Reporte OPERA

Para llevar a cabo esta operación se simuló la entrada del mensaje de interruptores por medio de la opción REPORTES en su campo *Msjc INT*, los valores de las corrientes al efectuar una apertura son obtenidos de la relación ANALOGICO en la cual el valor base es 1000 que es el utilizado en todos nuestros reportes.

El reporte de Operación de Circuitos es el encargado de llevar el control de los cambios de estados sufridos por cada interruptor, por lo cual se definió una estructura para el reporte la cual es la siguiente:

- El reporte verifica que el archivo este creado a la hora de que es mandado el mensaje, de no estar lo crea.
- Se verifica en el archivo existente el que el punto ha ser registrado no existe, en caso de existir verifica si el registro esta completo o no, es decir, para cada punto se requieren los siguientes campos: la apertura y la hora de ocurrencia, la corriente antes de la apertura, la causa, el cierre y hora de ocurrencia.
- Para verificar que tipo de cambio de estado es el mensaje enviado define si es control completo, de serlo la causa es una maniobra, en caso contrario es falla.

- Además por medio de una variable se define si el registro esta completo, es decir si se encuentra de un estado de apertura pasará a un estado de cierre, el lo cual completa este registro y para el siguiente es agregado otro donde puede esperar un cierre o apertura.

## 5.6. ALMACENAMIENTO CADA 24 HORAS

Para esta prueba, el sistema se ejecutó durante 72 horas, comenzando el primer día a las 9:00 hrs, con lo cual se simuló una recolección de datos en la BDD realizando el muestreo periódico en el módulo "Reloj" con los siguientes tiempos:

Reporte VMMV	2 min
Reporte VMMI	3 min
Reporte COME	4 min
Reporte HISTORICO	5 min

Al llegar las 00:00 hrs, el módulo "Reloj" manda el mensaje de actualización a cada uno de los reportes en el siguiente orden:

HISTORICO	COME
VMMV	OPIN
VMMI	

La prueba fue realizada satisfactoriamente, observandose los siguientes resultados:

- Para los reportes VMMV, VMMI, y COME se inicializaron correctamente cada uno de ellos en las relaciones BARRA, CORRI\_SUB y TRAN\_SE respectivamente, después de la actualización en la BDD reportes de la información.
- Para los reportes OPIN y OPERA se simuló el cambio de estado de un interruptor en especial, activando este cambio en una función temporal en el módulo "Reloj", la cual simula el cambio de estado de apertura/cierre y cierre/apertura, además falla/maniobra y maniobra/falla. Realizados estos cambios, se procede a esperar el tiempo asignado para la actualización (24:05 hrs) del reporte OPIN. El reporte de

OPERA, se actualizó durante el desarrollo de esta simulación ya que al recibir el mensaje se almacena la información en el registro correspondiente, siendo de esta manera la actualización para este reporte.

3. Para el reporte Histórico al cumplirse las 00:00 hrs, es creado un nuevo archivo correspondiente al día en el subdirectorio HISTORICO. La información es actualizada cada período de muestreo en el archivo correspondiente de forma inmediata.

## 5.7. SOLICITUD DE IMPRESION

Para esta prueba el sistema requirió almacenar datos en los archivos de los reportes, una vez creado el reporte y actualizado la información en el archivo, se seleccionaba en el menu principal la opción REPORTE en el campo de IMPRESION, presentandose un submenu de con el nombre de las subestaciones, seleccionada una subestación y verificado que la impresora estuviera en línea se realiza la impresión del reporte, en caso que la impresora no estuviese en línea aparece un mensaje para que intente nuevamente la impresión. Esta solicitud de impresión de reportes es manual y automática. En la siguiente figura se muestra el formato de los reportes:

### REPORTE DE OPERACION DE CIRCUITOS

SUBESTACION: SANTIAGUITO    FECHA: 28-ABRIL-93 16:22    PAG# 1  
 PERIODO: 1-ABRIL-93 00:05    28-ABRIL-93

CIRCUITO	DIA_HORA_A	DIA_HORA_C	CORRIENTE	CAUSA
STG-73230	25 21:48:22	26 03:45:67	100.000	Maniobra
STG-73170	26 15:15:47	26 23:47:00	101.000	Falla
STG-73140	27 01:23:55		114.400	Falla
STG-72010	27: 17:05:08	28 15:34:56	118.870	Maniobra

## 5.8. CREACION MENSUAL DE ARCHIVOS DE REPORTES

En esta prueba se verificó el correcto funcionamiento de los reportes VMMV, VMMI, COME y OPIN al efectuarse el cambio de mes. Los reportes deben cumplir con lo siguiente:

1. Al efectuarse el cambio de mes, en el subdirectorio correspondiente para cada Reporte se crea un nuevo subdirectorio con el nombre del mes actual. Esta prueba se efectuó cambiando en la función de "Muestreo" la fecha y hora repetidamente para simular que se realizaba el reporte en varios días y con ello simular el cambio de mes.
2. Al iniciar el mes actual se almacena la fecha y hora en un archivo llamado "Periodos", con el fin de tener un control de cuando el reporte es activado, en caso de no estar activado no efectúa la función de creación de los archivos para este mes hasta no ser activado por el usuario otra vez.
3. La creación por mes del Reporte OPERA se realiza hasta que ocurra un cambio de estado en algún interruptor en otro mes. Para probar esta función se requirió auxiliarse de dos funciones la de cambio de fecha y hora en "Muestreo" y el envío del mensaje del interruptor en "REPORTE".

## 5.9. PRUEBAS DE ESTABILIDAD DEL SISTEMA

Para establecer la estabilidad del sistema se asignó a cada reporte un tiempo de muestreo corto, con el fin de verificar sino entraba en un ciclo infinito, con lo cual se pierde el control completo del sistema, para probar que el sistema funciona a su mayor capacidad, se dejó ejecutando durante 100 hrs, en las cuales realizó las siguientes funciones:

- Creación de archivos de reportes mensuales.
- Muestreo de los reportes.
- Inicialización de los archivos de las relaciones utilizadas.
- Actualización a las 24 hrs.
- Actualización del reporte de Operación de circuitos por un mensaje

- Creación diaria del reporte Histórico de Mediciones.
- Cambio de la frecuencia de muestreo.
- Cambio de fecha y hora.
- Elaboración de reportes en impresora.
- Presentación de la información de los reportes en pantalla.
- Graficación del archivo Histórico en pantalla.

Se observo que durante este periodo de prueba todas las funciones se ejecutaron sin ningun problema.

## 5.10. GRAFICACION DEL REPORTE HISTORICO

Se realizó la prueba de graficación en pantalla para cada uno de los puntos declarados en el archivo Histórico de Mediciones, el comportamiento de los puntos va de acuerdo a la simulación realizada en el proceso de "Reloj". Esta gráfica se puede observar no importando el número de muestra en el Histórico, ya que los valores no encontrados son considerados como ceros. A continuación se dan los pasos para mostrar una gráfica de algún punto.

1. Se selecciona la subestación sobre la que se desea conocer el comportamiento de algún punto.
2. Del menú de puntos se selecciona de esa subestación uno en especial.
3. A partir de este momento se muestra en pantalla una ventana con los valores gráficos y tabulares teniendo como acotación de la gráfica el valor mínimo y máximo encontrado en ese punto (valor muestra Vs tiempo).
4. El sistema de graficación contiene una característica de poder cambiar la escala de los valores de la gráfica mostrada, lo cual es importante cuando se desea conocer a mayor detalle alguna región de valores.

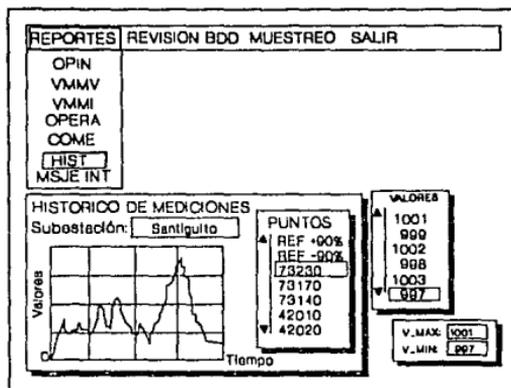


FIGURA 5.1 Menú de graficación de Histórico de mediciones

## 5.11. PRESENTACION EN PANTALLA

En este punto se presenta la descripción de la manera en que se realizó la interfaz de usuario para el subsistema de Reportes.

El sistema consiste de un menú principal con cuatro opciones: Reportes, BDD revisión, Muestreo y Salir, cada una de estas opciones a su vez presenta un menú desplegable, estos menus son manejados con las flechas del cursor o mediante el ratón posicionandolo en la opción deseada y presionando el botón izquierdo. A continuación se explican cada una de las opciones de los menús:

### REPORTES

En este menú se despliegan los nombres de los reportes y una opción para el envío del mensaje de interruptor para la simulación de este, además para cada una de estas opciones existe otro submenú desplegable donde se presentan las funciones que se realizan a cada uno de ellos. A continuación se muestra el menu principal desplegando la opción de REPORTES y como ejemplo el submenú del VMMV, se debe mencionar que los

submenús de las opciones VMMI y COME son iguales a este, por lo cual una vez definido este submenú quedan descritas las dos opciones anteriores.



FIGURA 5.2 Menú principal opción VMMV

En la opción OPIN se muestra el submenú correspondiente a las funciones que se utilizan para efectuar el reporte.



FIGURA 5.3 Menú opción OPIN

En la opción OPERA se despliega el siguiente submenú mostrando las funciones que se utilizan para realizar este reporte.



FIGURA 5.4 Menú opción OPERA

En el reporte HISTORICO se despliega una ventana donde se muestra un campo llamado subestación donde se selecciona la subestación para visualizar la información de los puntos analógicos de esta.

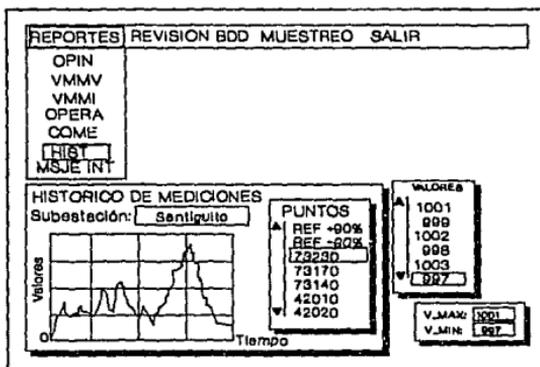


FIGURA 5.1 Menú de graficación de Histórico de mediciones

En la opción Msje INT se muestra una ventana que sirve para la simulación de la entrada de un mensaje de cambio de estado de interruptor indicando el número de interruptor, el tipo de operación (apertura/cierre), y si es control completo o no.

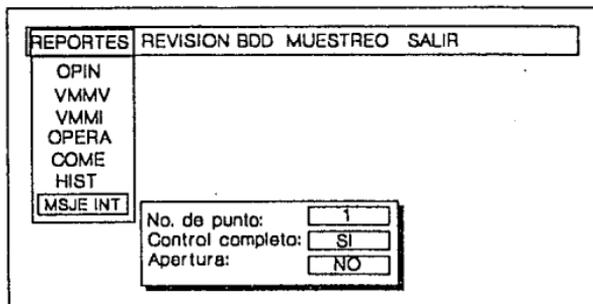


FIGURA 5.5 Menú opción Msje INT

Para las otras opciones se dará una breve explicación y se ejemplifica con una opción en general, ya que el comportamiento de esos menús es muy similar a lo que ya se ha explicado.

### REVISION BDD

Esta menú presenta solamente un submenú BDD la cual presenta todos los archivos que se encuentren en el subdirectorio C:\IMAGEN\RELACIONES con extensión DBF, una vez solicitada esta función y desplegado el submenú, se puede seleccionar cualquier archivo, del cual se obtienen los campos que lo forman y la opción de poder editar, borrar o modificar cualquiera de sus campos. Esta opción se utiliza sólo para revisar los archivos de trabajo y los cuales son temporales.

### MUESTREO

En esta opción se realiza el cambio de tiempo periódico de cada reporte, no cualquier persona puede tener acceso a esta función por lo cual al seleccionar alguna de las siguientes opciones: VMMV, VMMI, COME, HIST, OPERA y OPIN; presenta una

ventana que despliega el mensaje de PASSWORD sin el cual no entrará al siguiente nivel de menús. Una vez que se ha elegido el reporte a modificar y tecleado el password correcto, aparece otra ventana con los campos de "Tiempo de muestreo", que presenta en pantalla el valor actual de muestreo y la otra opción es " Reporte activo", mostrando el valor definido de "SI". Al entrar en esta ventana el usuario puede modificar la frecuencia con la cual se realizará el muestreo de los reporte y activarlo o desactivarlo.

Otra opción de este menú es la de fecha en la cual también aparece la ventana del password y una vez que es tecleado, se presenta otra ventana mostrando la hora y fecha actual, siendo posible en este nivel modificar estos parámetros.

### SALIR

En esta opción es la salida al sistema operativo.

Se presenta a continuación y a manera de ejemplo el camino seguido para obtener una gráfica del reporte histórico de mediciones, los valores del reporte de Operación de Circuitos (OPERA).

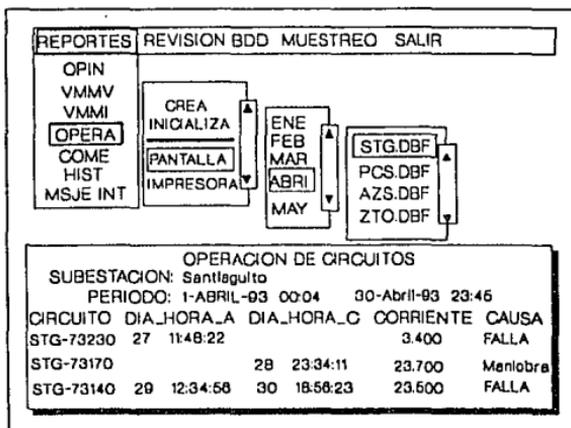


FIGURA 5.6 Menú opción OPERA

## CONCLUSIONES

En este trabajo se desarrollaron los programas del subsistema de reportes para la estación maestra MAC-500 en una computadora personal, estos informes presentan en una forma tabular o gráfica las lecturas periódicas de los puntos del sistema (digitales o analógicos), para conocer y analizar el comportamiento de la red eléctrica, sin interferir con el sistema de control y adquisición de datos de la estación maestra. Las principales conclusiones obtenidas de la realización de este trabajo de tesis son :

- Fácil comprensión de reportes, debido a que la información sobre la operación y consumo de energía de los circuitos de la red eléctrica se presentan en un medio ambiente amigable y fácil de utilizar.
- Su arquitectura permite la adaptación de otros módulos de reportes, teniendo únicamente que enlazarlos al programa principal.

- Flexibilidad de uso, ya que al ser sus archivos compatibles con dBase III se pueden emplear otras herramientas que también utilicen este formato (dBase, FoxPro, Lotus, Xtree Gold, etc.) para facilitar la impresión de reportes y su análisis posterior en otros departamentos de la compañía eléctrica.
- Una vez realizado el enlace a la estación maestra, permitirá descargarla de estas tareas permitiéndole efectuar el control y la adquisición de datos de una manera más eficiente.
- Por otra parte el uso de una metodología de ingeniería de software (Data Flow Ward & Mellor para tiempo real) permitió la especificación y diseño del subsistema de reportes de una manera clara y ordenada, lo que hizo que el proceso de codificación y pruebas se realizará de una forma sencilla.

En lo personal el colaborar en un proyecto real del IIE, me permitió obtener como experiencia el conocimiento de las técnicas utilizadas en el desarrollo de este tipo de proyectos, especialmente en área de control supervisorio.

Por último, se presentan opciones a futuro interesantes para la estación maestra MAC-5000, a partir de la integración total del sistema. Algunas de ellas son las siguientes:

1. La posibilidad de integrar el uso de paquetes comerciales para mejorar el despliegue y visualización de los reportes. Tal es el caso de poder presentar los reportes en gráficas a colores y por medio de ventanas.
2. Ya no existirá la limitante para hacer crecer los reportes actuales o generar otros nuevos, de acuerdo a las necesidades del usuario. Un ejemplo inmediato, es el reporte de Histórico de Mediciones, el cual actualmente se genera para una sola subestación seleccionada por el operador, sin embargo el usuario ya está requiriendo que se pueda generar para todas las subestaciones que esté supervisando la MAC-5000, lo cual implica tener una gran cantidad de memoria para poder almacenar toda la información.

3. Con la conexión directa a la PC y la transferencia continua de datos en tiempo real, se genera para la MAC-5000 un camino de apertura hacia el exterior, que amplía grandemente su potencialidad de utilización, ya que podrían desarrollarse con relativa facilidad las siguientes funciones:
  - transferencia de toda la base de datos en tiempo real a la PC para su explotación por paquetes especializados por ejemplo de despacho de energía, de análisis de disturbios, localización de fallas, etc.
  - transferencia de la información entre PC's por medio de enlaces tipo red, facilitando el uso y el acceso a los datos en tiempo real, a los diferentes usuarios de red eléctrica supervisada.
4. Con la eliminación del sistema operativo CP/M y del disco duro, y realizando el arranque de la MAC-5000 desde la PC, se reducirá considerablemente el tiempo de arranque de la misma, facilitando el cargado de nuevas versiones de programación y reducirá las posibilidades de falla de la estación maestra MAC-5000.

#### **Usuarios Potenciales.**

Los usuarios potenciales del presente trabajo de tesis son básicamente las dos subáreas en las que actualmente se encuentra operando la MAC-5000: Morelia y Toluca.

Para ello, se requiere que el traslado de los reportes a la PC, sea prácticamente transparente con respecto a la versión actual, a fin de que sea fácil y rápidamente instalada en campo, puesto que, debido a la gran responsabilidad que tiene cada una de las maestras, no es permitido que tengan que dejar de operar por un periodo largo, a fin de instalar el nuevo manejo de los reportes.

## REFERENCIAS

- [ACK 92] Ackerman W., Wayne R., "Understanding Supervisory Systems", *Computer Applications in Power*, October 1992.
- [EIA 83] Electronic Industries association, *EIA Standard RS - 485*, April 1983.
- [GAR 91] Garcia J. *Turbo C manejo de archivos*, Macrobit 1991.
- [GAU 87] Gaushell Dennis J., "Supervisory Control and Data Acquisition", *Proceedings of the IEEE*, Vol. 75, No. 12, December 1987.
- [GHE 91] Ghezzi C., Jazayeri M., *Fundamentals of software engineering*, Prentice Hall, INC., Englewood Cliffs.
- [GON 88] González H., González F., "MBad-TR, manual de usuario", *Reporte interno del Departamento de electrónica IIE*, Mayo 1988.

- [GUT 90] Gutiérrez G. "Estación Maestra para control supervisorio desarrollada en el IIE", *Boletín IIE Marzo/Abril 1990*, Vol. 14, No. 2.
- [GUT 91] Gutiérrez G. "Manual de Ingeniero de Sistema Revisión 1.0", *Reporte interno del Departamento de Electrónica IIE*, Mayo 91.
- [HID 92] Hidalgo A., "Especificación y diseño de la migración del subsistema de reportes de la estación maestra MAC - 5000 a una computadora personal", *Reporte interno del Departamento de Electrónica IIE*, Julio 1992.
- [IBA 89] Ibarguengoytia P. "Comunicaciones para controladores", *Boletín IIE Septiembre/Octubre 1989*.
- [IEE 91] IEEE Tutorial Course *Fundamental of supervisory systems*, IEEE Power Engineering Society 1991.
- [MEJ 90] Mejía P., "Referencia de programación del sistema operativo para procesos concurrente", *Reporte interno del Departamento de Electrónica IIE*, Mayo 1990.
- [MIR 89] Mireles M., "Manual de usuario MAC - 1186", *Reporte interno del Departamento de Electrónica IIE*, Junio 89.
- [MOL 89] Molina P., "Referencia técnica MAC - 1186", *Reporte interno del Departamento de Electrónica IIE*, Julio 89.
- [MOS 91] Motorola, MDLC and MOSCAD, "MOSCAD technical notes", Motorola Inc. 1991.
- [OAK 89] Oakland Group, Inc. *The C-scape Interface Management Systems*, 1989.
- [PRES 88] Pressman R., *Software engineering*, 2da. ed. McGraw-Hill International editions, Computer Science Serie, 1988.

- 
- [SAU 92] Saucedo E., Valdivia R., "Metodología propuesta para el desarrollo de programas de cómputo en tiempo real", *Reporte interno del Departamento de Electrónica IIE*, Septiembre 1992.
- [SCH 88] Schild H., *Programación en Turbo C*, McGraw-Hill/Interamericana de España, S. A., 1988.
- [SEQ 90] Sequiter Software Inc., *Code Base 4.2 User's Guide*, 1990.
- [SOM 88] Sommerville I., *Ingeniería de software*, 2da ed. Addison Wesley Iberoamericana 1988.
- [SYS 90] System Architect, *User's Guide*, Popkin software & systems Inc., 1990.

## ***Apéndice A***

# **CARACTERISTICAS DE LAS HERRAMIENTAS DE DESARROLLO**

En este apéndice se describen las características de las herramientas utilizadas para el desarrollo de los programas del subsistema de reportes para la estación maestra MAC - 5000, las cuales son:

1. Turbo C versión 2.0      Ambiente de desarrollo y compilador en lenguaje C para la creación del código ejecutable a partir de los códigos fuentes.
2. Code Base 4.2            Biblioteca de rutinas para administración y manejo de bases de datos compatibles con dBase III plus.
3. C-scape 3.0              Bibliotecas de rutinas para controlar la interfaz de usuario de los programas desarrollados.

#### 4. System Architect

Herramienta CASE para la especificación y diseño del subsistema de reportes bajo la metodología "Data Flow Ward & Mellor". Además, la herramienta soporta las siguientes metodologías:

- Diagrama de flujo de datos Gane & Sarson
- Diagrama de flujo de datos DeMarco/Yourdon
- Diagramas de estructura
- Diagramas de Entidad - Relación
- Diagramas de flujo
- Diagramas de Descomposición
- Diagramas de transición de estados
- Transición de estados Ward & Mellor
- Diagrama de flujo de datos Ward & Mellor

#### A.1 LENGUAJE 'C' (TURBO C VERSION 2.0)

El lenguaje C es sencillo, compacto, elegante y veloz, por lo que resulta de gran utilidad para aquellos programadores que pretendan un software versátil y al mismo tiempo efectivo.

No es por tanto de extrañar la gran difusión que está logrando en sus últimos años, sobre todo después de la aparición del 'Turbo C' en 1987, siendo una de sus mayores virtudes la compatibilidad y similitud con el código máquina, lo que le convierte también en una herramienta necesaria en sustitución del lenguaje ensamblador.

Sin embargo, los mejores atributos, provienen de la posibilidad de realizar una programación y un diseño modular, de su estructura compacta y de la portabilidad y eficiencia.

El lenguaje C, es pues, un lenguaje informático muy portable, que se traduce fácilmente a otros lenguajes con sólo realizar algunas modificaciones mínimas en las líneas de encabezado.

Además, es un lenguaje amistoso y versátil que soluciona problemas, ya que fue diseñado precisamente para reemplazar la codificación en lenguaje ensamblador y permitir al usuario tener una interfaz hombre-máquina, por medio de comandos mucho más fáciles que el ensamblador.

Sin lugar a duda el lenguaje C, a través de su corta existencia, ha demostrado ser uno de los más versátiles, completos y estructurados, que posee como gran ventaja destacar, la transportabilidad, es decir, la posibilidad de utilizarlo tanto como en macroordenadores, miniordenadores y microordenadores.

Son cada vez más numerosos los programas en lenguaje C, como los siguientes: El sistema operativo UNIX y la mayoría de los programas relativos con las bases de datos, como el dBase II, dBase III plus y dBase IV [GAR 91].

Otro aspecto importante con el lenguaje C es que sólo tiene 38 palabras clave (32 que definió el estándar ANSI y 5 que añadió Borland para permitir usar mejor algunos aspectos especiales del entorno PC) [SCH 88]. Estas palabras clave son las órdenes que constituyen el lenguaje C.

## A.2 UTILERIA CODE BASE 4.2

Code Base es una biblioteca de rutinas en "C" para la administración y manejo de bases de datos. Su compatibilidad con dBase III le permite trabajar con cualquier otro producto compatible con dBase. Ya que utiliza las mismas convenciones del dBase es fácil de aprender. Quien ha programado en el lenguaje "C" y entienda los fundamentos de las bases de datos no tiene problemas para utilizar el Code Base.

Code Base soporta directamente Turbo C, Quick C, Microsoft C, Zortech C++, Watcon C 386 y los sistemas operativos Xenix y SCO Unix. Existe una biblioteca preconstruída (T4.LIB) la cual es utilizada con turbo C. Estas bibliotecas se construyeron usando el modelo de memoria *large*, con emulación de punto flotante, usando el administrador de pantallas del Code Base, utilizando archivos índices dBase NDX, y utilizando el sistema operativo MS-DOS [SEQ 90].

Para utilizar esta biblioteca del Code Base en el ambiente interactivo del Turbo C, se necesita un archivo "project". Como ejemplo el archivo "d4simple.pj", puede ser usado para crear el programa "d4simple.exe" y debe contener las siguientes líneas:

```
d4simple.c
T4.lib
```

Para el desarrollo del subsistema de reportes se utilizó la versión en línea del Turbo C, ya que el ambiente interactivo no soportaba al código fuente, utilizando para compilar y enlazar las siguientes instrucciones:

Compilación:

```
tcc -c -ml -DEGA_14 -I\cs\include;\codebase\include nombre.c
```

Ligado:

```
tlink @reporte2.rsp
```

En donde el archivo reporte2.rsp contiene las siguientes instrucciones:

```
\t\lib\c0l reporte genera general repo2 graph imprime
reporte
\t\lib\emu \t\lib\mathl \t\lib\cl \t\lib\graphics \cs\lib\t2lowl +
cs\lib\t2lscap \codebase\lib\t4
```

A continuación se muestra una lista de las rutinas del Code Base organizadas por categorías. Esta lista contiene únicamente las rutinas más utilizadas.

<i>Añadir, escribir e insertar</i>	d4append, d4write, x4insert.
<i>Acceso a campos</i>	f4char, f4double, f4int, f4long, f4ncpy, f4ptr, f4str, f4record
<i>Reemplazo de campos</i>	f4r char, f4r double, f4r int, f4r long, f4r str.
<i>Información del campo</i>	f4decimals, f4j_ref, f4num_fields, f4name, f4record_width, f4ref, f4type, f4width.
<i>Filtros</i>	x4filter, x4filter reset, x4filter do, i4filter.

<b><i>Abrir, cerrar y crear archivos</i></b>	d4create, d4use, d4use_excl, i4open, i4index, u4open.
<b><i>Archivos memo</i></b>	m4edit, m4exist, m4read, m4write.
<b><i>Borrado lógico, Borrado físico y reindexado</i></b>	d4pack, d4zap, i4reindex.
<b><i>Posición del registro</i></b>	d4bottom, d4go, d4seek_double, d4seek_str, d4skip, d4top, x4bottom, x4go, x4seek, x4skip, x4top.
<b><i>Relación entre bases</i></b>	x4relate, x4relate_do, x4relate_reset.

Para cada base de datos abierta existe un área de memoria el buffer del registro. Muchas de las funciones de alto nivel que administra la base de datos trabajan con este registro, como por ejemplo, la rutina *d4go* lee un registro de la base de datos del disco a este buffer del registro. Similarmente, las rutinas *d4write* y *d4append* escriben explícitamente el buffer al archivo de la base de datos.

Para acceder y modificar este buffer existen rutinas que trabajan con los campos, cuando una de ellas modifica el buffer, los cambios se escriben al disco automáticamente.

Cada base de datos puede tener un conjunto de archivos índices. Cada base de datos puede tener su archivo índice seleccionado. Las rutinas de alto nivel de administración de la base de datos tales como: *d4seek* y *d4skip* utilizan el archivo índice seleccionado de la base de datos en uso.

Los archivos memo para guardar texto de longitud variable. Para eficiencia, cada campo se salva con una cantidad de espacio fijo dentro del registro de la base de datos, el campo memo ocupa exactamente 10 bytes de espacio en el registro dentro de la base de datos y una cantidad variable dentro de un archivo memo separado. Un archivo memo tiene el mismo nombre que el archivo de la base de datos correspondiente excepto en que la extensión se cambia de DBF a DBT. El mismo archivo memo se usa para todos los campos memos de una base de datos en particular.

### A.3 UTILERIA C-SCAPE 3.0

C-scape es una herramienta que ayuda a controlar la interfaz de usuario de los programas en C. Específicamente, es una biblioteca de rutinas en C con las que se puede virtualmente crear y modificar cualquier tipo de pantalla de texto o de entrada de datos. Se puede diseñar rápidamente pantallas con rutinas análogas a la función estándar *printf* y se proporcionan funciones de más alto nivel para crear sistemas de menús y pantallas.

Se puede añadir características sofisticadas a los programas en C, tales como: ventanas, soporte de gráficos, ayudas sensitivas al contexto, edición de texto, listas que se desplazan y validación de datos.

El C-scape es flexible, ya que puede modificarse cualquier parte de él para acoplarse a las necesidades de la aplicación particular. Se pueden crear los tipos de campo, las rutinas de validación, los editores de texto, los bordes de las ventanas, pantallas de ayuda y sistemas de menús de acuerdo a la aplicación.

Las principales características del C-scape incluyen [OAK 89]:

- Un lenguaje de definición flexible, basado en la instrucción *printf*, para el diseño de pantallas.
- Un administrador de ventanas que controla automáticamente la apariencia y posición de las ventanas, administra la salida enviada a las ventanas que se encuentre sobre puestas por otras, y trabaja en modo texto y modo gráfico.
- Posee la característica de definir campos y enlazar funciones a ellos. Estas funciones de campos validan o regulan el contenido de los campos.
- La capacidad de las funciones de campo para determinar el tipo de datos que se mostrara en pantalla, como se editaran y teclearan los datos y como se validaran. C-scape tiene funciones predefinidas para la mayoría de las rutinas necesarias para la edición de estos campos.
- Una familia de rutinas para edición de texto. Estas rutinas pueden crear desde unas notas breves hasta editores de texto completos.
- Sofisticados bordes de ventanas, que varían desde simples rectángulos hasta bordes con título con barras de desplazamiento y ayuda en línea.

- Posee un sistema de ayuda sensitiva al contexto, que permite mostrar mensajes de ayuda cuando sean necesitados. Cada pantalla o campo puede tener su propio mensaje de ayuda individual. Se pueden personalizar las funciones que muestran los mensajes de ayuda para darle la vista que se desee en cualquier programa.
- Rutinas de alto nivel, que son construidas desde las rutinas de bajo nivel del C-scape. Estas rutinas incluyen varios sistemas de menús tales como: menús desplegables, menús anidados.

Como una ayuda complementaria al desarrollo de software, se puede utilizar el C-scape con el diseñador de pantallas *Look & feel*. Con *Look & feel*, se pueden crear pantallas y menús interactivos para convertirlos en código C que se puede compilar. Esto elimina la necesidad de teclear código para tales atributos de pantalla tales como: definiciones de campos, posiciones de los caracteres y los colores.

*Look & feel* puede crear y editar archivos con imagenes de pantalla. Un programa puede cargar y ejecutar estas pantallas y menús durante el tiempo de ejecución. Esto reduce el tamaño del archivo ejecutable y hace posible cambiar la apariencia del programa sin recompilarlo.

#### A.4 HERRAMIENTA CASE SYSTEM ARCHITECT

El paquete "System Architect" es una herramienta de CASE ("Computer Aided Software Engineering") que soporta múltiples metodologías de diseño de programas de cómputo y ayuda a realizar los trabajos de una manera ordenada y más eficiente. Se ejecuta bajo la extensión gráfica Microsoft Windows del sistema operativo MS-DOS y soporta metodologías manejadas por procesos o manejadas por datos. Una breve descripción de algunas de las metodologías que soporta son:

#### A.4.1. Modelado de procesos (Diagrama de Flujo de Datos)

El modelado de procesos describe las características funcionales de un sistema. Estas funciones se representan como transformaciones de información, dibujadas como círculos. Una o más entradas, mostradas como flechas etiquetadas, manejan la transformación para producir información de salida. Este modelo puede ser aplicado al sistema completo o solo al programa, la clave es representar la información alimentada o/y producida por la transformación.

Un diagrama de flujo de datos (DFD) es una técnica que representa el flujo de información y las transformaciones que se aplican mientras los datos se mueven de la entrada a la salida. El DFD se puede usar para representar un sistema o programa de cómputo a cualquier nivel de abstracción. De hecho, los DFD se pueden particionar en niveles que representan el flujo de información y detalles funcionales.

Los estilos de los DFD varían en la forma de presentación pero tienen las mismas facilidades, por ejemplo, los DFD de Yourdon/DeMarco usan círculos como símbolos de procesos y flechas arqueadas como flujo de datos. Mientras el estilo Gane & Sarson usa un rectángulo con las esquinas redondeadas como símbolo de proceso y una notación adicional de etiquetas.

Este método es simple y se puede presentar en varios niveles por lo que se facilita el detallado del sistema, pudiéndose llegar al grado de detalle deseado.

#### A.4.2. Modelo Entidad - Relación

Esta es una técnica para representar las entidades y sus relaciones asociadas, esencialmente los elementos estáticos de un sistema. Puede usarse como complemento del modelado de procesos para la definición de las bases de datos.

Una base de datos contiene entidades de información que están relacionadas por medio de organización y asociación. La arquitectura lógica de una base de datos se define por un esquema que representa las definiciones de relaciones con las entidades de información. La arquitectura física de la base de datos depende de la configuración del

equipo. Sin embargo, el esquema (descripción lógica) y la organización (descripción física) deben conjuntarse para satisfacer los requerimientos funcionales para acceso, análisis y reportes.

La normalización de los archivos de datos y los análisis subsecuentes de la capacidad de la base de datos se deben acoplar con una técnica efectiva para representar las relaciones entre objetos de datos. El diagrama de Entidad -Relación y sus subsecuentes extensiones se usan frecuentemente como una herramienta gráfica para representar las relaciones entre los datos.

#### A.4.3. Diagramas de estructura

El diseño estructurado usa una herramienta gráfica, llamada diagrama de estructura, para hacer entendible fácilmente el sistema. El diagrama de estructura es el resultado de mostrar el sistema particionándolo en módulos, y construyendo el diagrama con esos símbolos para mostrar jerarquía, organización y comunicación entre módulos. En lugar de representar la jerarquía por niveles (añadiendo un nuevo diagrama a cada símbolo, como en el modelado de procesos), la jerarquía en el diagrama de estructura forma un árbol de módulo representado en un diagrama de una página.

La conexión entre los módulos son "llamadas", y para cada módulo existe una descripción/especificación que puede definir rigurosamente la función interna del módulo.

El diagrama de estructura (y la documentación que lo acompaña en el diccionario de datos) sirve como guía a la implementación y como herramienta de documentación. El diagrama de estructura puede guiar la asignación de tareas de codificación, y el orden en el cual son escritas y probadas.

#### A.2.4. Diagramas de descomposición

Esta metodología permite un análisis por medio de diagramas de organización (organigramas). El uso de esta técnica no ofrece mucha ventaja al desarrollar un sistema en tiempo real.

### A.2.5. Sistemas de tiempo real

Es una extensión del modelado de procesos con adiciones y variaciones a la notación de los símbolos. Utiliza los diagramas de transición de estado para definir un tipo especial de proceso llamado "transformación de control"; tiene dos estilos, el primero usa círculos como estados y flechas como transiciones, mientras que el segundo es el estilo Ward & Mellor que es un poco más riguroso y completo que el anterior (posteriormente se describirá esta metodología). Asimismo se pueden crear más niveles de DFD como en el modelado de procesos.

El "System Architect" también incluye soporte para sistemas de transacción y sistemas en tiempo real. En el apéndice B se encuentran los fundamentos de la metodología empleada en el diseño, opcionalmente, se dispone de una metodología orientada a objetos utilizando la notación Grady Booch. [SYS 90]

El "System Architect" utiliza todo el conjunto de manejadores de Microsoft Windows, así como la interfaz de usuario. Esta interfaz incluye menús desplegables, cuadros de diálogo e íconos; la amplia variedad de manejadores permiten que el "System Architect" sea utilizado con una gran variedad de equipos comerciales.

Existe soporte interno de un diccionario, el cual incluye elementos de datos, expresiones de estructuras de datos, flujo de datos, almacén de datos, descripciones de procesos, banderas de datos, de control, descripciones de los módulos y descripción de los requerimientos, entre otros. Toda la información de la enciclopedia se mantiene y se integra dentro del diccionario, el cual incluye el almacenamiento de los símbolos gráficos, definiciones y diagramas.

El Diccionario/Enciclopedia se mantiene en un formato de "arquitectura abierta", lo que permite al usuario acceder toda la información del sistema. El Diccionario/Enciclopedia se almacena en formato dBase III Plus, dando flexibilidad adicional en el uso de estos datos, los cuales se pueden utilizar para comunicarse con otras herramientas CASE o para la generación personalizada de reportes. El diccionario puede ser importado y exportado a otros productos mediante el formato de archivo CSV ("Comma Separated Value").

Existe también soporte para revisión de reglas en los diagramas de flujo de datos, incluyendo las reglas de DeMarco/Yourdon y Gane & Sarson; cuenta con utilerías para reportar errores en línea y para realizar reportes del contenido del diccionario.

Se tiene una interfaz de usuario gráfica, la cual permite reducción, ampliación, vista de la página completa y otros modos de video más. Existe soporte para múltiples tipos de letras proporcionadas por otros fabricantes. Los tamaños de las plumas, de la cabezas de flechas y otros elementos de los diagramas se pueden personalizar. No existen limitaciones prácticas en el tamaño o el número de símbolos gráficos contenidos en un diagrama. Hojas de estilos permiten al usuario personalizar símbolos con respecto a muchas de sus características.

El sistema operativo Windows ofrece un medio ambiente gráfico de multi-ventanas manejado por eventos provenientes del mouse o del teclado. El sistema operativo permite al usuario cambiar de una aplicación a otra en cualquier momento sin ninguna pérdida de datos por el cambio de contexto, además, el "System Architect" permite que el texto y gráficas sean exportadas a otros programas compatibles con el ambiente Windows.

## **Apéndice B**

# **METODOLOGIA DE DESARROLLO**

Organizar el proceso de desarrollo de programas es una actividad crítica que va desde la administración de la gente hasta la administración de todos los productos realizados durante el ciclo de vida. También involucra la definición de los métodos apropiados y su combinación con las metodologías. De acuerdo al diccionario Webster's New World (1977), un método se define como la manera de hacer algo en una forma especialmente ordenada, y una metodología se define como un conjunto de métodos. Las metodologías frecuentemente se conjuntan con las organizaciones de software para realizar estándares para el proceso de desarrollo de programas.

En este apéndice se describe la metodología utilizada para el diseño de sistemas en tiempo real, basada en el método de diagramas de flujo de datos de Ward & Mellor y utilizando el paquete de CASE System Architect mencionado en el apéndice A.

## B.1 SISTEMAS DE INFORMACION

Una de las áreas de aplicación más grande y con un crecimiento rápido dentro de la computación es el almacenamiento y recuperación de datos, se ha llamado a esta clase de sistemas "Sistemas basados en información", o simplemente "Sistemas de información" porque el propósito primario del sistema es administrar la información. El corazón de tales sistemas es una *base de datos* a la que se le aplican transacciones para crear, recuperar, actualizar o eliminar elementos.

Los sistemas de información han ganado importancia por el valor de la información como recurso. Los datos que esos sistemas administran son usualmente los recursos más valiosos de una empresa. Los sistemas de información están orientados a los datos y se pueden caracterizar por el manejo que le dan a ellos.

Otra característica importante de los sistemas de información es la necesidad de proporcionar interacción con los usuarios que no tengan un respaldo técnico o que este sea mínimo. De esta manera los requerimientos de interacción hombre-máquina, al igual que un ambiente amigable para el usuario, son de importancia primordial en estos casos. En particular, se requiere que la interacción con la aplicación sea por medio de menús, los menús deben diseñarse uniformemente, y la navegación entre las diferentes funciones proporcionadas por la aplicación debe ser fácil. Los usuarios nunca deben sentir que se perdieron; siempre deben de estar en control con la interacción de la aplicación, y la aplicación debe ser segura contra posibles fallas de los usuarios.

## B.2 SISTEMAS DE TIEMPO REAL

Aparte de los sistemas de información existe otra clase de sistemas llamados *Sistemas de tiempo real*. La principal característica de esos sistemas es que deben de responder a eventos dentro de un periodo de tiempo predefinido y estricto. Por ejemplo, en un sistema de monitoreo de una fábrica, los programas necesitan responder a un inusitado incremento en la temperatura para inmediatamente activar ciertos switches o sonar una alarma.

Los sistemas de tiempo real han sido estudiados extensivamente. El corazón de un sistema en tiempo real es un ejecutivo (scheduler) que ordena las acciones del sistema. Existen dos tipos básicos de algoritmos para el ejecutivo: por tiempo muerto y por prioridad. En el método por prioridad cada acción tiene una prioridad asociada, la acción con la mayor prioridad es la que se ejecutará en el siguiente tiempo [GHE 91].

En el método por tiempo muerto, cada acción tiene un tiempo específico en el cual debe ser iniciado o finalizado, es responsabilidad del ejecutivo asegurar que las acciones se sucedan de tal manera que satisfagan todos los requerimientos.

Finalmente, otras cualidades de los programas pueden ser importantes en el caso de sistemas en tiempo real también. Se dijo anteriormente que los aspectos de interfaz hombre-máquina son relevantes en el caso de sistemas de información, pero también son relevantes para los sistemas de tiempo real. Por ejemplo, la interfaz externa de un sistema de control y monitoreo de una planta industrial debe ser diseñada de tal manera que el operador entienda perfectamente el estado del sistema bajo control y pueda operar la planta eficazmente.

### B.3 METODO DE ANALISIS ORIENTADO AL FLUJO DE DATOS

La idea de una metodología que guíe al programador en su trabajo durante todas las fases de desarrollo del programa llama la atención; incrementa la confianza en la gente sobre lo que esta haciendo, enseña a la persona con falta de experiencia como resolver problemas de una manera sistemática y fomenta una técnica uniforme y estándar para resolver problemas.

El diseño orientado al flujo de datos (y el diseño de software en general) tiene sus orígenes en anteriores conceptos de diseño que acentúan la modularidad, el diseño top-down y la programación estructurada. Sin embargo, el diseño orientado al flujo de datos extiende esas técnicas por medio de integrar explícitamente el flujo de información en el proceso de diseño. Stevens, Myers, y Constatine fueron pioneros en el diseño de software basado en el flujo de datos a través del sistema [SOM 88].

### B.3.1 Diagramas de flujo de datos (DFD)

Los diagramas de flujo de datos (DFD) es una notación bien conocida y ampliamente utilizada, para especificar las funciones de un sistema de información. Los DFD describen los sistemas como colecciones de datos que son manipulados por funciones. Los datos se pueden organizar de varias maneras: pueden ser almacenados en almacenes de datos, pueden moverse como flujo de datos y pueden ser transferidos hacia o desde un medio ambiente externo.

Una de las razones de la aceptación de los DFD es que pueden ser expresados por medio de notación gráfica atractiva que los hace fáciles de utilizar. Los elementos básicos de un DFD son [GHE 91]:

- *Burbujas*. Utilizadas para representar funciones, esto es, una transformación de los datos.
- *Flechas*. Utilizadas para representar flujos de datos, las flechas que llegan a las burbujas representan valores de entrada dentro del dominio de la función representada por la burbuja, mientras que las flechas que salen representan los resultados de la función.
- *Líneas paralelas*. Utilizadas para representar almacenes de datos. Las flechas que entran o salen de estas líneas paralelas representan datos que se insertan o que se extraen del almacén de datos respectivamente.
- *Rectángulos de E/S*. Utilizados para representar la adquisición de datos y la producción durante la interacción hombre-máquina.

La figura B.1 representa los símbolos gráficos anteriormente descritos:

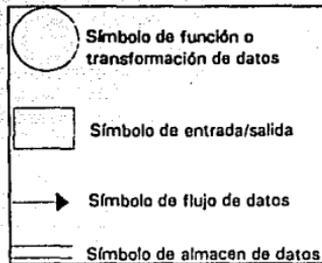


FIGURA B.1 Símbolos básicos para construir diagramas de flujos de datos

Para ejemplificar el uso de los DFD en la figura B.2 se describe un sistema de información simplificado para una biblioteca pública. Los datos y funciones mostrados no son necesariamente funciones y datos a efectuar en la computadora. El DFD describe objetos físicos, tales como libros y anaqueles. Con almacenes de datos que parecen ser, pero no necesariamente, archivos de computadora. Obtener un libro del estante puede ser hecho automáticamente por un robot o manualmente, pero en ambos casos, la acción de obtener un libro se representa por la función mostrada en la burbuja. La figura puede aún representar la organización de la biblioteca sin procedimientos computarizados.

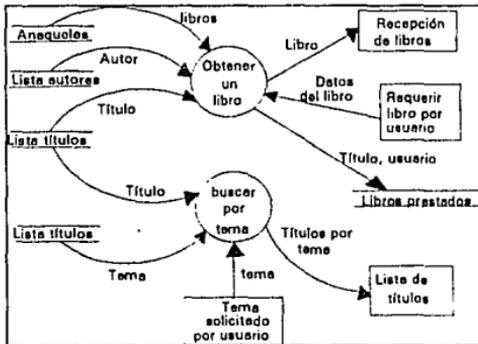


FIGURA B.2 DFD de sistemas de información de una biblioteca

La figura describe el hecho de que para obtener un libro es necesario lo siguiente: Un requerimiento de usuario explícito consistente del título y el nombre del autor del libro y el nombre del usuario; acceso a los anaques que contienen los libros; una lista de autores y una lista de títulos. Todo esto proporciona la información necesaria para encontrar el libro.

La forma en que el libro se obtiene, sin embargo, no se menciona en la figura. Si no hubieramos usado nuestra experiencia acerca de la forma que se presta un libro, no hubiera manera de deducir esta información de la figura. De esta manera, debemos considerar este DFD como la primera aproximación (nivel 1) de la descripción del sistema de información de la biblioteca.

Una descripción más a detalle de como se puede seleccionar un libro de los anaques de la biblioteca se da en la figura B.3 la cual es una refinación de una parte de la figura B.2. Este DFD aún es impreciso y se debe de refinar hasta el nivel de detalle que proporcione suficiente información para la correcta interpretación del sistema.

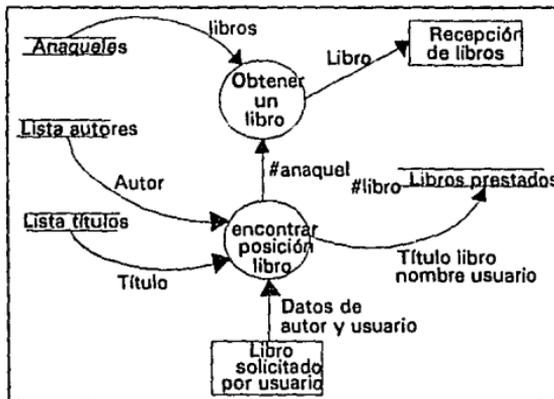


FIGURA B.3 Refinamiento parcial del DFD del sistema de información de una biblioteca

Como se menciono anteriormente cada burbuja (Transformación de Flujo de Datos) puede ser refinado en capaz para representar más detalle. La figura B.4 ilustra este concepto.

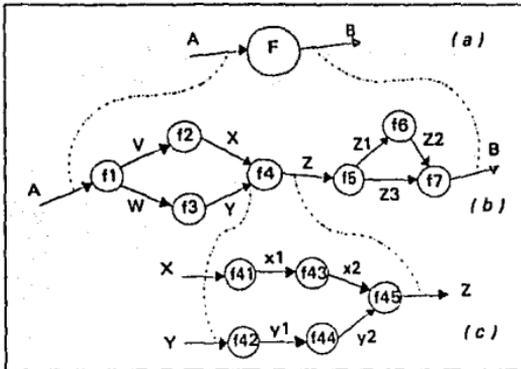


FIGURA B.4 Refinamientos del flujo de información

El modelo fundamental para el sistema F ilustrado en la figura B.4(a) indica la entrada primaria como A y la última salida B. Se refinó el modelo F en transformaciones f1 a f7, tal como se muestra en la figura B.4(b). Note que la continuidad del flujo de información debe ser mantenida, esto es, la entrada y salida del refinamiento debe ser la misma. Una refinación posterior de f4 (ver figura B.4(c)) representa detalles en la forma de las transformaciones f41 a f45. De nuevo, la entrada (x,y) y la salida (z) permanecen sin cambio.

Las herramientas anteriores pueden proporcionar descripciones del sistema a varios niveles de abstracción que pueden ser útiles para documentar los resultados del análisis realizado sobre el sistema a desarrollar.

### B.3.2. Extensiones del modelo de flujo de datos

Los métodos de diseño y análisis orientados al flujo de datos se han extendido para el análisis y diseño con una notación apropiada de sistemas en tiempo real. Aunque se han desarrollado varios esquemas basados en el modelo de flujo de datos, la técnica propuesta por Ward & Mellor está ganando gran aceptación, la cual se describe a continuación [PRE 88].

Para la extensión de la notación clásica de análisis y diseño orientados al flujo de datos y para acomodarse a las demandas impuestas por un sistema en tiempo real, se debe desarrollar una notación para representar lo siguiente:

- a) Flujo de información que es adquirido o producido en tiempo continuo.
- b) Control de información que pasa a través del sistema y el procesamiento asociado de control.
- c) Estados del sistema y el mecanismo que causa transición entre estados.

En un porcentaje significativo de aplicaciones de tiempo real, el sistema debe monitorear información continua en el tiempo generada por algún proceso de tiempo real. Por ejemplo, consideremos el control de una planta química. Los niveles de presión y temperatura se deben de monitorear por razones de seguridad, los sensores se instalan para generar las señales apropiadas cuando cualquiera de esos niveles excedan algunos valores predefinidos. La notación convencional de flujo de datos no hace distinción entre datos discretos y datos continuos en el tiempo. Esto es, un examen del DFD no provee indicación de la continuidad en el tiempo del flujo de datos. Una extensión a la notación convencional de flujo de datos, mostrado en la figura B.5, provee un mecanismo para representar flujo de datos continuos en el tiempo.

La flecha con doble cabeza se utiliza para representar flujo continuo mientras una flecha sencilla se usa para indicar flujo de datos discretos. En esta figura, la temperatura se mide continuamente mientras que se provee un solo valor para el punto de ajuste de temperatura. El proceso mostrado produce una salida continua en el tiempo, *valor de corrección de nivel*.

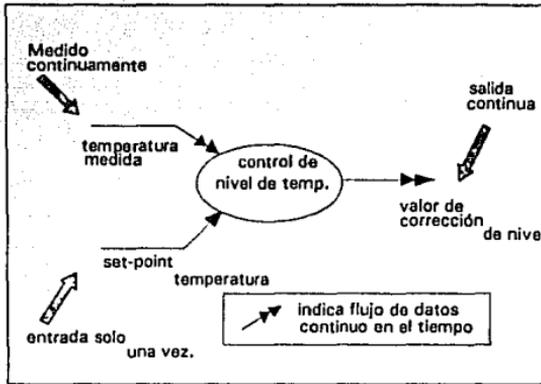


FIGURA B.5. Ejemplo de notación Ward & Mellor

Cuando el modelo físico o de implementación es creado, el diseñador debe establecer un mecanismo para la colección de datos continuos en el tiempo. Obviamente, el sistema digital adquiere datos en una manera cuasi-continua usando técnicas como muestreo cíclico ("polling") a alta velocidad. La notación indica donde se requerirá equipo convertidor analógico-a-digital y cuales transformaciones son las que demandarán programas de cómputo de alta funcionalidad.

En representaciones convencionales orientadas al flujo de datos, el control o flujo de eventos no se representa explícitamente. Esta exclusión es restrictiva cuando se consideran aplicaciones de tiempo real y por esta razón se ha desarrollado una notación especializada para representar flujo de eventos y procesamiento de control. Continuando con la convención establecida por los diagramas de flujo de datos, el flujo de datos se representa con una flecha sólida. El flujo de control, sin embargo, se representa con una flecha punteada o sombreada.

#### B.4 METODOLOGIA UTILIZADA EN EL DESARROLLO DEL SUBSISTEMA DE REPORTES DE LA EM MAC - 5000

El diseño -la descomposición del sistema en módulos- se basa directamente de los DFD y se documenta usando diagramas de estructura. Estos diagramas son una estructura en la que cada nodo representa módulos. Cada módulo representa una abstracción funcional que será implementada posteriormente por un subprograma. De esta manera, en la figura B.6 si los módulos  $M_1$ ,  $M_2$  y  $M_3$  están subordinados al módulo  $M$ , entonces  $M$  llama a  $M_1$ ,  $M_2$  y  $M_3$ .

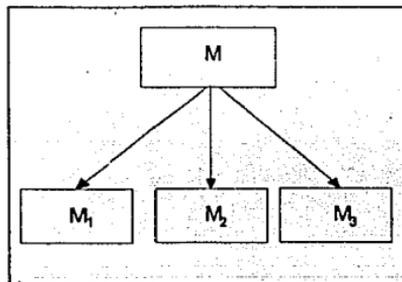


FIGURA B.6 Diagrama estructurado

El proceso en el cual un diagrama de estructura se deriva del DFD debe pretender obtener módulos con buen acoplamiento y cohesión. Por ejemplo, se deben evitar conexiones patológicas como las siguientes:

- Un salto dentro de un módulo subordinado en lugar de una llamada a ese módulo.
- Una referencia de un módulo a datos privados de otro módulo en lugar de pasar los datos al segundo módulo por medio de parámetros.

Los diagramas de estructura pueden ser más expresivos decorándolos con notaciones adicionales como se muestra en la figura B.7. La figura ilustra cuales parámetros fluyen entre los módulos:  $M$  pasa  $B$  a  $M_1$  y recibe  $A$ ;  $M$  también recibe a  $C$  de  $M_2$ . La figura también ilustra los patrones de control que gobiernan las llamadas de los

módulos subordinados: M puede llamar a  $M_1$  o repetidamente llamar a  $M_2$ . La selección se representa mediante el símbolo del diamante, y las iteraciones se representan por una flecha circular que agrupa a los módulos cuya ejecución se itera.

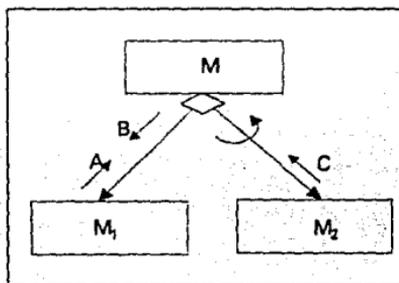


FIGURA B.7 Diagrama de estructura decorado

Los DFD se transforman en diagramas de estructura manualmente, pero la metodología proporciona algunos lineamientos a seguir. A continuación se muestran los pasos de la metodología utilizada para el diseño del sistema desarrollado [SAU 92].

1. *Desarrollar el nivel 01 DFD (Diagrama de Flujo de Datos) que incluya todos los datos primarios y flujo de eventos, que muestre todas las fuentes de información, destinos y almacenes de datos. Para algunas aplicaciones en tiempo real, el flujo de eventos puede ser más importante que el flujo de datos en la caracterización del sistema. Por esta razón, todos los flujos de eventos deberán representarse en el nivel más alto.*

2. *Crear una lista de eventos que describa los principales eventos asociados con el sistema. La lista de eventos podrá incluir actividades orientadas al control tales como "activación del switch de control", o "la temperatura alcanzó el punto de ajuste" y eventos orientados a comandos como "petición del operador del status actual", o "el usuario proporciona el valor de calibración". La lista de eventos se usa para refinar el nivel 01 del modelo de flujo y como fuente de información para la creación de diagramas de estado.*

3. *Refinar el modelo de flujo al nivel 02 y 03 y detallar.* Las recomendaciones del DFD se aplican igualmente para el refinamiento de sistemas en tiempo real. Se debe notar, sin embargo, que se refinan el flujo de datos y el flujo de eventos.

4. *Crear una representación diagramática de estados.* La lista de eventos puede asistir en la creación del diagrama de estados. Como en el modelo de flujo, el diagrama de estados se puede expandir a varios de niveles, con cada nivel subsecuente elaborado con la información contenida en el nivel precedente.

5. *Revisar el modelo de flujo y diagramas de estado para asegurar que cada uno represente la función y flujo del sistema consistentemente.* Los modelos de flujo para un sistema en tiempo real implican la estructura del diagrama de estados y, respectivamente, el diagrama de estado implica el flujo del sistema. Las representaciones para cada uno deben de ser consistentes.

6. *Usar notación convencional para describir objetos de datos y contenido de procedimientos.* Se puede proveer una descripción detallada de los objetos de datos con un diccionario de datos. Una narrativa de procesos para transformaciones de datos y control se puede desarrollar usando árboles de decisión o tablas de decisión.

7. *La arquitectura del programa se puede derivar usando mapeo de transformación y transacción.* Esos mapeos, se pueden aplicar a modelos de flujo de datos a cualquier nivel razonable de detalle. Cuando se presentan múltiples tareas o procesos, el mapeo se aplica al modelo de flujo para cada tarea o proceso individualmente.

8. *Revisar todas las representaciones e iterar cuando se requiera.*

#### **B.4.1. Consideraciones en el proceso de diseño**

El diseño orientado al flujo de datos permite una transición conveniente de representación de información a un diseño de la descripción de la estructura de un programa. La transición del flujo de información a la estructura se ejecuta como parte de un proceso de 5 pasos:

- 1) Se establece el tipo de flujo de información
- 2) Se indican las fronteras del flujo
- 3) Se proyecta a un programa estructurado el DFD
- 4) Se define por descomposición la jerarquía de control
- 5) Se refina la estructura resultante usando medidas de diseño y heurística.

## Apéndice C

# ESTRUCTURA DE LOS MENSAJES DE REPORTES

Este apéndice describe la estructura de los mensajes utilizados en el proceso de "Genera Reportes", el cual puede considerar cualquiera de estas opciones: crear, actualizar, inicializar, muestrear o el cambio de estado del interruptor, además de los reportes. El mensaje se define por medio de dos estructuras *MsjInt* y *MsjSol*, en los cuales se define si el mensaje es de cambio de estado de algún interruptor, por operador o los que son generados periódicamente, estos mensajes seleccionan el tipo de reporte y la operación que se efectuará.

Cada mensaje tiene la función de seleccionar el reporte, la estructura *MsjSol* puede tener seis diferentes opciones del tipo de reporte a escoger y puede realizar cuatro operaciones diferentes en cada reporte, estas opciones se definen en los siguientes puntos del apéndice.

Además se definen los mensajes que son utilizados para cada uno de los módulos de los reportes como son: *VMMV*, *VMMI*, *OPERA*, *HIST*.

## C.1. ESTRUCTURA MSJINT

La estructura *MsjInt* define si el mensaje es asociado a las operaciones efectuadas en los procesos que involucran a los interruptores como son Operación de Circuitos y Operación de Interruptores, de no ser así esta estructura define los que son generados periódicamente o por operador. A continuación se describen los elementos que forman a esta estructura:

<b>Longitud:</b>	Longitud del mensaje
<b>Función:</b>	En este campo se indica la función que se desea ejecutar:
	<b>NO_MSJE:</b> Define que el mensaje no es por cambio de estado del interruptor.
	<b>MSJE_INT:</b> Define cambio de estado en el interruptor
<b>Error:</b>	Código que regresa el mensaje al ser erróneo/correcto
<b>n_msje:</b>	Número de mensaje
<b>proc:</b>	Procedimiento que realiza el mensaje
<b>n_pto:</b>	Número del punto asociado al interruptor
<b>edo_fun:</b>	Estado funcional del interruptor, el cual se define por: <i>control completo:</i> Definido por SI (1) para maniobras <i>Sin control completo:</i> Definido por NO (0) por fallas
<b>edo_vis:</b>	Estado de cambio del interruptor, definido por: <i>Apertura:</i> Cuando el interruptor se abre se recibe un 1. <i>Cierre:</i> Cuando el interruptor se cierra se recibe un 0.

## C.2. ESTRUCTURA MSJSOL

Esta estructura describe que reporte se va a realizar y el tipo de operación a efectuar, definiéndose de la siguiente manera:

<b>Tipo_rep:</b>	Indica el reporte donde se realiza la operación, y puede ser activado para los seis procesos de reportes por las siguientes variables: <i>Msje_VMMV:</i> Manda al proceso VMMV <i>Msje_VMMI:</i> Manda al proceso VMMI <i>Msje_OPIN:</i> Manda al proceso OPIN
------------------	---

*Msje\_OPERA*: Manda al proceso OPERA

*Msje\_COME*: Manda al proceso COME

*Msje\_HIST*: Manda al proceso HIST

**operac**: Tipo de operación, el cual se define por los siguientes elementos:

*Crea*: Operación de crear un archivo en la BDD REPORTES dependiendo del tipo de reporte.

*Inicializa*: Operación de borrar la información contenida en el reporte seleccionado.

*Actualiza*: Actualiza el reporte seleccionado en la BDD REPORTES.

*Muestrea*: Operación que se realiza periódicamente para recolectar la información de la BDD y almacenarla en las relaciones de trabajo.

### C.3. MENSAJES DEL SISTEMA

A continuación se definen los mensajes para cada uno de los reportes.

#### Proceso VMMV

##### Crea archivo de reporte VMMV

MsjInt.función	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_VMMV	"Reporte VMMV"
MsjSol.operac	=	Crea	"Crea archivo VMMV"

##### Muestrea relación BARRA

MsjInt.función	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_VMMV	"Reporte VMMV"
MsjSol.operac	=	Muestrea	"Muestrea relación BARRA"

**Actualiza archivo de reporte VMMV**

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_VMMV	"Reporte VMMV"
MsjSol.operac	=	Actualiza	"Actualiza archivo VMMV"

**Inicializa archivo de reporte VMMV**

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_VMMV	"Reporte VMMV"
MsjSol.operac	=	Inicializa	"Inicializa reporte VMMV"

**Proceso VMMI**

**Crea archivo de reporte VMMI**

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_VMMI	"Reporte VMMI"
MsjSol.operac	=	Crea	"Crea archivo VMMI"

**Muestra relación CORRI\_SUB**

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_VMMI	"Reporte VMMI"
MsjSol.operac	=	Muestra	"Muestra relación CORRI_SUB"

**Actualiza archivo de reporte VMMI**

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_VMMI	"Reporte VMMI"
MsjSol.operac	=	Actualiza	"Actualiza archivo VMMI"

**Inicializa archivo de reporte VMMI**

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_VMMI	"Reporte VMMI"
MsjSol.operac	=	Inicializa	"Inicializa reporte VMMI"

**Proceso OPERA**

**Crea archivo de reporte OPERA**

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_OPERA	"Reporte OPERA"
MsjSol.operac	=	Crea	"Crea archivo OPERA"

**Actualiza archivo de reporte OPERA**

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_OPERA	"Reporte OPERA"
MsjSol.operac	=	Actualiza	"Actualiza archivo OPERA"

**Inicializa archivo de reporte OPERA**

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_OPERA	"Reporte OPERA"
MsjSol.operac	=	Inicializa	"Inicializa reporte OPERA"

**\* Mensaje de interruptores**

MsjInt.funcion	=	Msje_Int	"Mensaje por interrupción"
MsjInt.npto	=	npto	"número de interruptor"
MsjInt.edo_fun	=	1 / 0	"Control completo/no control"
MsjInt.edo_vis	=	1 / 0	"Apertura/Cierre"

**Proceso COME**

**Crea archivo de reporte COME**

MsjInt.función	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_COME	"Reporte COME"
MsjSol.operac	=	Crea	"Crea archivo COME"

**Muestra relación TRAN\_SE**

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_COME	"Reporte COME"
MsjSol.operac	=	Muestra	"Muestra relación TRAN_SE"

**Actualiza archivo de reporte COME**

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_COME	"Reporte COME"
MsjSol.operac	=	Actualiza	"Actualiza archivo COME"

**Inicializa archivo de reporte COME**

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_COME	"Reporte COME"
MsjSol.operac	=	Inicializa	"Inicializa reporte COME"

**Proceso HISTORICO**

**Crea archivo de reporte HISTORICO**

MsjInt.función	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_HIST	"Reporte HISTORICO"
MsjSol.operac	=	Crea	"Crea archivo HISTORICO"

**Muestra relación ANALOGICO**

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_HIST	"Reporte HISTORICO"
MsjSol.operac	=	Muestra	"Muestra relación ANALOGICO"

**Actualiza archivo de reporte HISTORICO**

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_HIST	"Reporte HISTORICO"
MsjSol.operac	=	Actualiza	"Actualiza archivo HISTORICO"

**Inicializa archivo de reporte HISTORICO**

MsjInt.funcion	=	NO_MSJE	"Mensaje por operador"
MsjSol.tipo_rep	=	Msje_HIST	"Reporte HISTORICO"
MsjSol.operac	=	Inicializa	"Inicializa reporte HISTORICO"

## **Apéndice D**

# **GLOSARIO DE TERMINOS**

- Barrido Lento** Después de un cierto número de intentos de comunicación del CIC hacia la UTR, si esta no contesta, el CIC la coloca en estado de "barrido lento", lo cual significa que la frecuencia de exploración hacia esa UTR tendrá un período más largo.
- BDD** Base de Datos. Se utiliza esta abreviación para hacer alguna referencia a la información almacenada en la estructura de datos manejada internamente por el programa.
- Canal** El canal de comunicación indica el medio de transmisión que se esta empleando para la comunicación con la UTR. El medio de comunicación puede ser por radio, teléfono, etc.
- CASE** "Computer Aided Software Engineering". Nombre dado al conjunto de reglas que automatizan las actividades que involucra el seguir alguna metodología de las existentes en la ingeniería de software.

<b>COME</b>	Consumo Mensual. Reporte que registra diariamente la medición del consumo de potencia de todos los transformadores en KWH y KVARH, agrupados por subestación.
<b>DFD</b>	Diagrama de Flujo de Datos. Es una técnica gráfica para presentar el flujo de información y las operaciones de transformación que sufren los datos cuando circulan de entrada a salida.
<b>EIA RS-485</b>	Estándar que especifica las características eléctricas de transmisores y receptores para intercambio de señales binarias en una conexión multipunto de equipos digitales.
<b>EM</b>	Estación Maestra. Equipo electrónico que permite supervisar desde un centro de control el estado de los sistemas eléctricos de potencia, en este caso.
<b>Entrada analógica</b>	Estos valores se obtienen normalmente de transductores de entrada que convierten la cantidad medida (Watts, Var, Volts, Amperes) a señales de baja energía (por ejemplo: 0-1 mA o 4-20 mA) las cuales son fácilmente procesadas con equipo electrónico.
<b>Entrada digital</b>	Los datos de la entrada digital se obtienen de la supervisión del estado de contactos externos (abierto/cerrado). Existen cuatro tipos de datos digitales, o información de la posición de los contactos: <ul style="list-style-type: none"> <li>• Estado Actual</li> <li>• Estado actual con memoria, esto es, el número de cambios de contactos desde el último reporte a la maestra.</li> <li>• Secuencia de eventos.</li> <li>• Acumuladores de pulsos.</li> </ul>
<b>HISTORICO</b>	Histórico de mediciones. Reporte que registra los valores medidos durante el día en todos los puntos analógicos, agrupado por subestación.

<b>ISO</b>	Organización Internacional de Estándares.
<b>Interfaz - usuario</b>	Proceso que permite al usuario tener una comunicación directa con la máquina y sus periféricos en cualquier sistema, para el manejo de la información tanto adquirida como procesada por el sistema.
<b>MAC</b>	Maestra de Adquisición y Control. Siglas utilizadas para la identificación de tarjetas y componentes de la estación maestra MAC - 5000.
<b>MAC - 1186</b>	Es una tarjeta de multiprocesamiento de adquisición y control, opera bajo el sistema operativo SOPCO 86 de la estación maestra MAC - 5000.
<b>MBAD-TR</b>	Manejador de base de datos en tiempo real que se ejecuta bajo el sistema Operativo SOPCO 86 y define las estructuras utilizadas en el formato MAC.
<b>OPERA</b>	Operación de Circuitos. Reporte que lleva un registro diario de las operaciones de apertura y cierre realizadas sobre los circuitos agrupando la información por subestación.
<b>OPIN</b>	Operación de Interruptores. Reporte que lleva un registro diario de todas las operaciones (falla o maniobras) realizadas en los interruptores y agrupada por subestación.
<b>OSI</b>	Modelo de referencia conocido como "Interconexión de Sistema Abiertos", utilizados en redes de comunicación de un sistema abierto.
<b>Reportes gerenciales</b>	Sistemas de informes de la red eléctrica, de diversas variables para análisis de fallas, información histórica o de tipo gerencial (consumos).

<b>SCADA</b>	Sistema de control supervisorio y adquisición de datos. Adquiere y concentra la información en tiempo real de los dispositivos de campo para su supervisión y telecontrol por medio de un operador o de manera automática.
<b>UTR</b>	Unidad Terminal Remota. Reciben la información directamente de los elementos de campo y se encargan de darle un formato para ser enviado a la estación maestra mediante un protocolo de comunicación, asimismo, reciben comandos de control de la estación maestra para accionar los dispositivos de campo como interruptores, válvulas, etc.
<b>VMMI</b>	Valor Máximo y Mínimo de Corriente en circuitos. Reporte de la medición diaria de los valores máximos y mínimos de corriente medida en todos los circuitos, agrupados por subestación.
<b>VMMV</b>	Valor Máximo y Mínimo de Voltaje en barras. Reporte de la medición diaria de los valores máximos y mínimos de voltaje en barras ordenados por subestación.
<b>WARD &amp; MELLOR</b>	Metodología de Ingeniería de Software para el diseño de sistemas en tiempo real.