



**UNIVERSIDAD DON BOSCO**  
**FACULTAD DE INGENIERÍA**  
**ESCUELA DE INGENIERÍA ELÉCTRICA**  
**ESCUELA DE INGENIERÍA ELECTRÓNICA**

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE SUPERVISIÓN LOCAL Y**  
**REMOTO APLICADO A SUBESTACIONES ELÉCTRICAS DE POTENCIA DE LA**  
**UNIVERSIDAD DON BOSCO**

TRABAJO DE GRADUACIÓN PREPARADO PARA LA FACULTAD DE INGENIERÍA

PRESENTADO POR:

**RAFAEL ANTONIO CÁCERES HERNÁNDEZ**

**FABRICIO JOSÉ MELGAR ACOSTA**

OCTUBRE DE 2004

SAN SALVADOR, EL SALVADOR, CENTRO AMÉRICA.

**UNIVERSIDAD DON BOSCO**  
**FACULTAD DE INGENIERÍA**  
**ESCUELA DE INGENIERÍA ELÉCTRICA**  
**ESCUELA DE INGENIERÍA ELECTRÓNICA**

**RECTOR:**  
**ING. FEDERICO MIGUEL HUGUET RIVERA**

**SECRETARIO GENERAL:**  
**LIC. MARIO RAFAEL OLMOS**

**DECANO:**  
**ING. GODOFREDO GIRÓN**

**ASESOR:**  
**ING. JULIO A. RIVERA PINEDA**

**JURADOS:**  
**ING. RICARDO GONZÁLEZ NÁJERA**  
**ING. CARLOS ALEXANDER LÓPEZ**  
**ING. JOSÉ ROBERTO SORIANO**

AGRADECIMIENTOS:

Nuestro principal agradecimiento va dirigido hacia EL CREADOR DEL UNIVERSO, Quien nos permitió iniciar, desarrollar y terminar esta tarea, y así cumplir una de nuestras metas más anheladas. Sin Su Luz ningún objetivo, por mas pequeño que fuera, se hubiese cumplido. Sin el apoyo del MAESTRO no hubiésemos podido terminar esta carrera. Gracias Infinitas DIOS ETERNO.

Agradecemos también a nuestras familias, quienes nos apoyaron incondicionalmente a lo largo de este camino duro, pero transitable gracias a su confianza y respaldo.

Nuestras mas sinceras muestras de gratitud al Asesor y a los Jurados de este trabajo, ya que han sido una verdadera guía para alcanzar los objetivos planteados.

Gracias a todas aquellas personas que de una u otra manera nos mostraron su respaldo a lo largo de nuestras carreras.

A todos ellos, les damos nuestras mas sinceras muestras de gratitud y aprecio.

*Fabricio Melgar*

*Rafael Cáceres*

*"Los problemas trascendentes a los que nos enfrentamos no pueden resolverse con el mismo nivel del pensamiento del que los creó"*

***Albert Einstein***

## ÍNDICE.

<b>Contenido.</b>	<b>Página</b>
<b>Capítulo I: Esquema General del Proyecto. ....</b>	<b>2</b>
1.1 Introducción.....	2
1.2 Justificaciones. ....	4
1.3 Objetivos Realizados. ....	6
1.3.1 Objetivo general. ....	7
1.3.2 Objetivos específicos.....	7
1.4 Descripción General del Proyecto.....	9
1.5 Criterios de Diseño. ....	9
1.5.1 Creación de Pantallas en Cirnet para Windows.....	9
1.5.1.1 Aplicación basada en MS Access. ....	10
1.5.1.2 Importancia de las alarmas.....	12
1.5.1.3 Datos trifásicos en pantalla principal de cada nodo. ....	12
1.5.1.4 Secuencia de datos por fase de cada subestación. ....	13
1.5.1.5 Gráficos de tendencias históricas. ....	13
1.5.1.6 Datos en tiempo real.....	14
1.5.1.7 Tablas numéricas del data logger y energía. ....	15
1.5.1.8 Red Circutor CVMk y el equipo CVM 96.....	15
1.5.1.9 Verificación de datos a nivel remoto (Internet). ....	15

<b>Capítulo II: Marco Teórico Básico.....</b>	<b>18</b>
2.1 Generalidades sobre los Software SCADA. ....	18
2.1.1 Definición de sistema SCADA. ....	18
2.1.2 Necesidad de un sistema SCADA.....	20
2.1.3 Funciones.....	21
2.1.4 Ventajas de los sistemas SCADA. ....	21
2.1.5 Desventajas de los sistemas SCADA. ....	22
2.1.6 Prestaciones de un SCADA. ....	22
2.1.7 Requisitos de un sistema SCADA.....	23
2.1.8 Módulos de un SCADA.....	23
2.2 Bases de Datos.....	24
2.2.1 Generalidades sobre Bases de Datos. ....	25
2.2.2 Orígenes y Antecedentes de las bases de datos. ....	26
2.3 Teoría de Redes.....	27
2.4 Clasificación de las Redes. ....	27
2.4.1 Según su tamaño y extensión.....	27
2.4.2 Según la tecnología de transmisión.....	28
2.4.3 Topologías de red .....	29
2.4.4 Redes LAN Ethernet.....	36
2.4.5 Formato de trama Ethernet .....	38
2.4.6 Tipos de redes Ethernet .....	41
2.4.7 Redes LAN Token Ring.....	43
2.4.8 Redes LAN FDDI.....	47
<b>Capítulo III: Funcionamiento del Software.....</b>	<b>53</b>
3.1 Manual de Usuario de la Aplicación SCADA en Cirnet .....	53
3.1.1 Acceso a la aplicación supervisoria.....	54
3.1.2 Función de Secciones.....	55
3.1.3 Pantalla de Ayuda.....	56

3.1.4 Pantalla de Alarma.....	57
3.1.5 Función de Energía. Barras y tablas.....	58
3.1.6 Función de Energía. Lineales y tablas.....	62
3.1.7 Ejemplo de Pantalla de Control.....	66
3.1.8 Diagrama Unifilar.....	68
3.1.9 Sistema de Alarmas.....	72
3.1.10 Tarifación.....	73
3.1.11 Configuración de equipos en la red SCADA.....	74
3.2 Composición de las Bases de Datos.....	77
3.3 Manual de Usuario de la Aplicación en Microsoft Access.....	79
3.3.1 Modo de ingreso al Sistema.....	79
3.3.2 Descripción de Formularios.....	81
3.3.3 Registros de las Bases de Datos.....	88
3.3.4 Manual de Usuario de la Aplicación para Internet.....	89
<b>Capitulo IV: Manual de Referencia.....</b>	<b>97</b>
4.1 Sistema Operativo.....	97
4.1.1 Definiciones de Sistemas de Archivos. ....	97
4.1.2 Tabla de asignación de archivos (FAT). ....	98
4.1.3 Sistema de Archivos FAT32. ....	98
4.1.4 Sistema de archivos NTFS. ....	98
4.1.5 Elección entre NTFS, FAT y FAT32. ....	99
4.2 Características de Software y Hardware. ....	101
4.2.1 Software CirNET para Windows. ....	101
4.2.2 Bases de Datos. ....	103
4.3 Características de la Red Seleccionada. ....	104
4.4 Metodos de E/S serie para comunicación entre terminales. ....	105
4.5 Metodo de comunicación serie asincrono. ....	105
4.6 Reglas para la recepción serie. ....	106
4.7 Interfaz RS-232. ....	106

4.7.1 Norma RS-232 ó V.24. ....	107
4.7.2 Características de la norma RS-232. ....	108
4.8 Interfaz RS-485. ....	109
4.8.1 Características de la RS-485 (EIA ó V11/CCITT). ....	111
4.9 Requisitos del Sistema. Hardware, Software y Conexiones. ....	112
4.10 Ventajas del Sistema. ....	114
4.11 Protocolos y Forma de Conexión de Dispositivos. ....	114
4.12 Modulo de Expansión CVM/RED. ....	115
4.13 Modo Maestro-Eslavos. ....	118
4.13.1 Conexión Física de Dispositivos. ....	118
4.13.2 Protocolos de Comunicación. ....	118
4.14 Modulo Convertidor Inteligente. ....	120
4.15 Regleta de Conexionado. ....	121
4.16 Esquema de Conexión del CVMk. ....	122
4.17 Conexión RED RS-485 a un Ordenador PC (Interfaz serial RS-232).....	123
4.18 Especificaciones Técnicas de los analizadores de redes CIRCUTOR.....	123
4.18.1 Descripción de analizador de redes CVMk.....	124
4.18.2 Mediciones Eléctricas.....	125
4.19 Costos del Proyecto.....	125
<b>Conclusiones. ....</b>	<b>127</b>
Recomendaciones. ....	128
Glosario. ....	130
Bibliografía.....	134
<b>Anexos.....</b>	<b>135</b>
a. PHP con algunas aplicaciones comunes.....	136
b. Código de la Aplicación en Access.....	153
c. Código de la Aplicación en PHP.....	166
d. Acerca de HTML.....	185

### **1.1 Introducción.**

Este documento presenta el desarrollo, utilización, descripción, y otras características de un sistema de monitoreo. La supervisión tiene lugar de forma local y remota (red de área local e Internet). La aplicación creada, con las características de los sistemas SCADA, incluye almacenamiento, procesamiento y verificación de la información a nivel local y remoto. Esto permite controlar en gran medida el estado del sistema al cual se ha aplicado esta herramienta. Se han podido desarrollar todas las funciones con que cuenta el software *Cirnet* para Windows, incluyendo la tarifación, aunque esta última es limitada por el tipo de tarjetas de red o módulos de expansión utilizados. Todo el sistema se encuentra soportado por otra herramienta creada en Microsoft Access para resolver el problema de que *Cirnet* no puede trabajar en todos los sistemas operativos, esto con el objetivo de observar el estado del sistema a todo nivel, y sin ninguna barrera, esta herramienta extra está vinculada con las bases de datos que van siendo almacenadas de forma automática, mes a mes, para poder llevar a cabo la verificación de los parámetros eléctricos almacenados de forma simple, es decir, procesando la información y presentándola de manera más adecuada.

La aplicación desarrollada incluye el monitoreo de dos nodos SCADA, representados por dos subestaciones del Centro de Investigación y Transferencia de Tecnología (CITT) de la Universidad Don Bosco.

Se verifica en este sistema la aplicación de todos los conocimientos adquiridos a lo largo de los años de estudio, los cuales son tan diversos como las aplicaciones que pueden tener los sistemas de monitoreo o sistemas SCADA, aunque en este caso se basan en la supervisión de los sistemas de potencia, concretamente de subestaciones eléctricas.

Actualmente los sistemas de supervisión y monitoreo tienen una gran importancia en la industria y están aplicados a un sin número de diversas naturalezas. Actualmente es muy fácil encontrar en la industria, empresas que dependen totalmente de estos sistemas, a tal grado que existe una frase muy común en el



ámbito de creación y utilización de los sistemas SCADA: un sistema de supervisión, control y adquisición de datos se ha convertido en los ojos de todo proceso automatizado. Esto implica que por lo general el sistema SCADA se utiliza completamente para controlar desde todo ángulo un proceso o sistema determinado. Generalmente los sistemas controlados por SCADA tienen sistemas de control semiautomáticos, por si acaso le fallara su vista (por lo tanto necesita tener un mecanismo de respaldo para poder ser utilizado en caso de que falle el software, interfaces, redes, etc.). Ya que los SCADA no son infalibles, por ejemplo si falla la alimentación de respaldo, en algunos procesos críticos (generación eléctrica), el sistema debe poder controlarse de otras maneras. Para ello se tiene sistemas de respaldo de estos sistemas, nodos independientes pero que pueden utilizarse de forma emergente. En general el grado de confianza que generan estos sistemas es muy grande y están teniendo una gran difusión. En procesos críticos existen nodos SCADA auxiliares que proveen el respaldo a los equipos y maquinaria a la que están acoplados.

El desarrollo de la aplicación presentada en este documento, esta enfocada a beneficiar en primer lugar la entidad a la cual se le ha acoplado, supliendo una necesidad que se tiene hace mucho tiempo en cuanto al control en el consumo energético y de variables relacionadas con el mismo. En definitiva se identifico la necesidad de controlar sistemas y equipos, y se adecuaron los conocimientos y las herramientas disponibles para lograr este fin, tal y como fue planteado desde el inicio de este proyecto.

## **1.2 Justificaciones.**

1. El monitoreo del estado de un sistema eléctrico por medio de las variables que mejor lo describen son características del tipo de herramienta que se describen en este trabajo. En todo proceso que involucra a la potencia eléctrica están implicadas las subestaciones de transformación de energía eléctrica. Estos sistemas son una parte esencial en la industria en general. Por esta razón debe existir un control de todas las variables, o al menos de las variables mas importantes que describen a estos sistemas. Con la aplicación de nuevas tecnologías han quedado atrás el censo de variables en el lugar geográfico en que se ejecuta el proceso o se encuentra el sistema a monitorear y se están automatizando todas las labores de recolección de información para ser utilizada instantánea e históricamente, todo esto con el objetivo de diagnosticar el estado del proceso o sistema que se verifica y establecer soluciones y análisis en casos de falla o malfuncionamiento de dicho sistema a partir de los datos que se han adquirido.
2. Utilizar los recursos disponibles en la Universidad Don Bosco para el desarrollo herramientas de monitoreo y control de variables, y profundizar en el aprendizaje de dichas herramientas, no solo para el beneficio de quienes los desarrollan e implementan, sino que también para el beneficio de la misma Universidad.
3. El desarrollo del esquema de pantallas está basado en la importancia que tienen las variables que están siendo supervisadas. Las pantallas principales presentan los datos trifásicos y sus correspondientes gráficos de tendencia en tiempo real. Estos muestran el estado global del sistema en un periodo determinado. En un plano secundario pero también relevante se muestran los datos por fase los cuales establecen el estado del sistema pero de forma extendida, lo cual permite establecer la carga que experimenta cada fase con respecto a las variables fundamentales (corrientes, voltajes, etc.), así como las variables derivadas relacionadas con la potencia (activa y reactiva)

y consumo de energía eléctrica. El diseño de las diferentes pantallas con las que cuenta el sistema SCADA esta basado en la verificación de otros sistemas de supervisión de sistemas de potencia (generación eléctrica y subestaciones de direccionamiento de potencia o switcheo); otro aspecto tomado para el diseño es la selección de variables de características similares y/o que se relacionan entre sí, por ejemplo la presentación de todas las variables de carácter trifásico en la pantalla principal de la aplicación. También se han tomado en cuenta opiniones relacionadas con el diseño a lo largo del desarrollo del sistema SCADA.

### **1.3 Objetivos Realizados.**

En el planteamiento inicial de los objetivos se estableció que el sistema sería capaz de monitorear al menos dos nodos SCADA, lo cual se ha cumplido. Se ha adicionado una aplicación monofásica al sistema SCADA el cual consta de un analizador de redes monofásicas, para efectos de prueba de la red SCADA, comprobando de esta forma la topología de red planteada de forma teórica en las presentaciones preliminares de este proyecto.

Hasta la fecha se ha logrado realizar el sistema de monitoreo a nivel local, centralizado en una sola Terminal para tres puntos, dos nodos SCADA que poseen cada uno las características de control supervisorio, adquisición y almacenamiento de datos (según el planteamiento inicial), basados en equipos analizadores de redes CVMk. El sistema de supervisión también permite monitorear un tercer equipo CVM 96 SP (Single Phase), el cual se constituye en un tercer nodo o punto de red, el cual se ha denominado de "*prueba*" debido a que solo es parte de la red local del sistema de monitoreo para verificación de su funcionamiento y probar que la red SCADA puede ser extendida hasta el número de equipos máximo permisible de la topología RS-485 (32 equipos por puerto serie ó Comm).

En el monitoreo de los subestaciones, no se ha discriminado ninguna de las variables que permite medir el dispositivo Circutor (en ambos nodos), esto debido a que todas variables que permiten monitorear estos equipos poseen una importancia gran importancia.

Se adquirió un solo modulo de expansión (CVM/RED), el cual permite la conexión de un equipo Circutor CVMk mediante la ayuda de un convertidor RS-232/RS-485, hacia un puerto serie (Comm 2) de la Terminal (PC), el cual también permite conectar equipos de la familia CVM, en topología de red estrella. En el caso de la conexión del equipo en el nodo de la subestación 1 (nodo inicial), se establece comunicación por medio de un modulo RS-232, y este a su vez se conecta, para realizar la comunicación, con la Terminal por medio del otro puerto serie disponible (Comm 1).

### *1.3.1 Objetivo general.*

- Desarrollar una herramienta de supervisión para sistemas de potencia eléctrica, con las características de los sistemas SCADA (Control Supervisorio y Adquisición de Datos), aplicado a subestaciones eléctricas; herramienta que permitirá evaluar los parámetros de funcionamiento y determinar el estado del sistema, sobre la base de la verificación de las variables eléctricas más sobresalientes a nivel local y remoto. Esto también permitirá la aplicación de los conocimientos adquiridos en manejo de sistemas de monitoreo y mediciones eléctricas en los sistemas de potencia, así como herramientas relacionadas con infraestructura de redes informáticas.

### *1.3.2 Objetivos específicos.*

- Monitorear por medio de los equipos CIRCUTOR y su interfaz con el programa CIRNET para Windows, todos los parámetros eléctricos posibles en los diferentes nodos que conforman la red SCADA, a continuación se especifican los nombres de estos parámetros:
  - a) Día y hora de la medición
  - b) Energía activa, fase 1 (tarifa 1)
  - c) Energía reactiva L, fase 1 (tarifa 1)
  - d) Energía reactiva C, fase 1 (tarifa 1)
  - e) Frecuencia
  - f) Factor de Potencia (trifásico)
  - g) Intensidades de corriente por fase
  - h) Intensidad promedio trifásica
  - i) Potencia activa por fase
  - j) Potencia reactiva L por fase
  - k) Potencia reactiva C por fase
  - l) Tensiones por fase
  - m) Tensiones de línea
  - n) Tensión promedio trifásica

- o) Demanda máxima (tarifa 1)
  - p) Energía consumida (kWh), fase 1 (tarifa 1)
- Almacenar por medio de bases de datos, la información correspondiente a periodos de medición o de monitoreo convenientes, para tenerlos disponibles a fin de poder analizar formal y periódicamente, de forma local o a larga distancia (Internet). Para cumplir con este objetivo se ha diseñado una herramienta sencilla que puede utilizarse para poder realizar el análisis y la supervisión de los sistemas de potencia en estudio.

## **1.4 Descripción General del Proyecto.**

El desarrollo de la herramienta de monitoreo que se ha diseñado e implementado, se ha basado en la verificación de otros sistemas de supervisión (generación y distribución eléctrica). La mayoría de los sistemas de supervisión o monitoreo de variables cumplen con un esquema lógico de creación y desarrollo similar al que se ha implementado. Junto con este hecho se ha querido fusionar otro aspecto, esto es obtener los mayores beneficios de los recursos con los que se cuenta para el desarrollo de esta herramienta SCADA.

## **1.5 Criterios de Diseño.**

### *1.5.1 Creación de Pantallas en Cirnet para Windows.*

Todo desarrollo de un sistema de monitoreo inicia con la identificación del proceso o sistema que se quiere supervisar, identificación de procesos o sistemas auxiliares y la identificación de otros procesos o sistemas que no intervienen directamente en el proceso principal pero que a la vez se desean monitorear. Junto con la identificación inicial de estos aspectos, se verifican las variables a monitorear, por lo general son todas las variables involucradas en dicho proceso pero con un orden ó jerarquía con respecto su forma de presentación. Esto quiere decir que hay ciertas variables que se presentaran en todas o en la mayoría de pantallas de las cuales esta formada el sistema de monitoreo, debido a su gran importancia y relación con el proceso. Para el caso existen variables que dependen directamente del sistema que se quiere monitorear y otras que dependen de los sistemas auxiliares que intervienen en determinada fase del proceso. Esto quiere decir que son variables que se reflejan en el sistema a monitorear pero que no pueden ser controladas por el propietario del proceso ó sistema, por ejemplo, la frecuencia es una variable que se refleja en todo sistema de potencia que involucra transporte y distribución de energía eléctrica, pero esta no puede ser controlada dentro de estos sistemas, sino solo en un sistema de generación. En cambio la corriente ó el voltaje son variables importantes en estos dos sistemas y en alguna medida controlables en ambos, pero la frecuencia pasa a un plano de variable de

verificación y no de control. Variables locales como temperatura, vibraciones, etc., están presentes únicamente en ciertos sistemas. Es decir que debe ser considerada la importancia que posee la variable dentro del sistema y su ubicación dentro del mismo al momento de decidir su monitoreo y trascendencia.

De la misma manera se hará énfasis en las variables que pueden ser controladas por el cliente al cual le pertenece el sistema de monitoreo y aquellas sobre las cuales puede ejercer cierto control. Las otras solo serán ubicadas como variables de verificación.

Los esquemas o gráficos que se utilizan en los diferentes sistemas SCADA, que corresponden a partes del proceso o sistema, cumplen una función decorativa o estética dentro de estos, y su función es únicamente ubicar al usuario geográficamente dentro de dicho proceso. Un grupo de variables eléctricas agrupadas en una pantalla, con sus respectivas unidades y elementos descriptivos de pertenencia, tienen mayor importancia que un motor girando dentro de un proceso determinado y su ubicación geográfica exacta dentro de la planta. Pero ambos aspectos suponen una herramienta poderosa de verificación y supervisión, ya que no solo presenta la descripción eléctrica del proceso, sino que adiciona la ubicación geográfica del mismo y su función dentro del proceso.

La descripción anterior corresponde a que este tipo de sistemas de supervisión tiene como función enfatizar el estado del sistema, por lo tanto la presentación de las variables se vuelve repetitivo y en cierta medida monótona, pero esto con el objetivo de tener presentes las variables controlables por el cliente, para este caso el factor de potencia se convierte en una variable de esta naturaleza.

A continuación se presentan, en forma detallada, algunos criterios de diseño usados para el desarrollo del sistema SCADA:

#### ***1.5.1.1 Aplicación basada en MS Access.***

La aplicación supervisora esta fundamentada sobre una herramienta creada especialmente para resolver el problema que presenta *Cirnet*, al no poder operar en cualquier sistema operativo. El sistema SCADA esta soportado por Microsoft



Access, la cual es una herramienta sencilla de manejo de bases de datos, las cuales en *Cirnet* tienen un carácter simple pero muy extenso, debido a que estas agrupan periodos de medición por mes; periodos en los cuales el sistema, según configuración, obtiene promedios de los subperiodos que se le han asignado (minutos, cuartos de hora, etc.). Si la aplicación SCADA no puede operar por causa del sistema operativo, perfectamente puede operar la herramienta de MS Access, la cual es compatible con todos los sistemas operativos Windows, del 95 a la fecha. Con la aplicación creada en MS Access incluso pueden verificarse variables en tiempo real, ya que se han utilizado las características de Intercambio Dinámico de datos (DDE) que poseen los programas de Microsoft montados sobre plataformas Windows, esta simplemente vincula las bases de datos que están actualizándose en un momento determinado y las refresca. Al inicio del desarrollo del sistema de monitoreo, se configuró la aplicación para que pudiera crear los correspondientes archivos de bases de datos (de extensión .dbf), los cuales pueden ser visualizados en MS Excel. Estos archivos, creados mes a mes automáticamente por el sistema, pueden ser importados hacia Access, donde se muestran de forma sencilla, interactiva y menos cargada de como los presenta Excel. En Access los archivos de bases de datos, pueden ser manipulados para obtener resultados tales como informes, los cuales pueden ser impresos según el cliente lo requiera. En Access los datos numéricos están debidamente identificados, por el contrario de lo que sucede en el formato **.dbf**, donde simplemente aparece como referencia el código de la variable adquirida.

Las variables han sido agrupadas por afinidad y se presentan de forma numérica y según lo permite la manipulación de variables de la base de datos. Esta manipulación de variables solo puede realizarse de dos maneras: importando las variables o vinculándolas. Cuando las variables se importan al programa MS Access no hay opción de que estas sean refrescadas, mientras que si se vinculan, al mismo tiempo que la base de datos se actualiza, la aplicación en Access captura dicha actualización y permite que las variables sean refrescadas.

Para la presentación de las variables por pantalla se han identificado las variables por fase y en algunos caso se han involucrado variables de línea y fase, junto con variables comunes como: fecha, hora, frecuencia y factor de potencia. Se quieren representar variables globales como la demanda máxima del sistema y la energía total consumida pero esto depende del tipo de modulo de comunicación. Para este caso los módulos de comunicación solo permiten la presentación de las variables de consumo o demanda de la fase 1. Puede lograrse la verificación de las variables en bruto obtenidas de la base de datos, y esto es posible en la pantalla de registros, la cual presenta los registro de los últimos meses.

#### ***1.5.1.2 Importancia de las alarmas.***

En todo sistema de monitoreo las alarmas cumplen con una función muy importante. Estas deben proporcionar información crítica de las variables supervisadas, acerca de valores de las mismas que sobrepasan los niveles nominales de operación proporcionados, tanto por las especificaciones técnicas de los equipos (transformadores de las subestaciones), limites de operación en la red de distribución de energía eléctrica, etc. En caso de necesitar cambiar los limites de los valores de alarma existe la posibilidad de lograrlo por medio de una entrada por teclado al iniciar el programa o la aplicación SCADA. La operación del sistema de alarmas depende de las variables consideradas criticas que necesitan un monitoreo más intenso, tales como: voltajes y corrientes de fase, factor de potencia, etc. Se ha trabajado en la presentación de las alarmas especificando exactamente la variable que ha entrado en crisis, lo cual facilita el rastreo de la misma cuando esta se ha detectado.

#### ***1.5.1.3 Datos trifásicos en pantalla principal de cada nodo.***

El estado general del sistema (subestación eléctrica) puede ser conocido a partir de las variables globales que el sistema verifica y almacena. Estas variables se presentan en la pantalla principal de cada uno de los nodos del sistema SCADA.

Los datos trifásicos involucran las potencias, frecuencia y factor de potencia. En general, estos valores describen el estado general, junto con voltajes y corriente se complementa la información más importante de cada nodo o subestación.

#### ***1.5.1.4 Secuencia de datos por fase de cada subestación.***

Se han incluido pantallas en las que se presentan datos por fase de cada subestación (nodo). Esto se debe a que en los sistemas por lo general las variables pueden tener diferentes valores, ya que alimentan a cargas que normalmente no son de naturaleza trifásica. Debido a esto las variables en cuestión pueden alcanzar valores superiores a los nominales, sin ser reflejados en las variables trifásicas que presenta el sistema. Por esta razón se han diseñadas pantallas en las que existen diferentes variables críticas para cada fase. Estas pantallas se describirán mas adelante en este documento.

#### ***1.5.1.5 Gráficos de tendencias históricas.***

Los datos históricos están relacionados con la configuración de la base de datos, es decir que debe realizarse la configuración y selección de las variables que quieren almacenarse en la base de datos para que el sistema permita presentarlos en forma histórica. En este caso *Cirnet* para Windows permite presentar gráficos y tablas de tendencia histórica para cualquier variable que se seleccione. La selección del tipo de gráfico no es una opción de usuario, sino una limitación del sistema, es decir que este solo permite presentar los datos históricos en forma de histograma y diagramas lineales, y esto lo ha definido previamente el fabricante.

Los datos históricos son una herramienta que debe seleccionarse por el usuario del sistema y por esta razón han de presentarse de forma obligatoria dentro de un sistema de monitoreo, para ser utilizados cuando sea necesario obtener conclusiones relacionadas con el comportamiento de las variables para determinar que ha provocado la falla o el malfuncionamiento del sistema.

Las variables incluidas en los gráficos de tendencia histórica, por lo general, están relacionados con el estudio de ciertas condiciones a las que ha sido sometido el

sistema. Estas condiciones tienen que ver con el análisis de una cierta falla, un periodo de estudio para determinar como ha variado algún parámetro durante ese rango. Para este caso en particular los gráficos históricos importantes son los que el software permite mostrar y estos están relacionados con las diferentes variables que supervisa la aplicación, por esta razón la forma de presentación de estos gráficos históricos esta relacionada con la limitación que a este respecto posee el software y que no permite presentar diferentes tipos de grafico, sino solo los predeterminados por el fabricante. Se puede escoger entre toda la gama de variables que se permite supervisar Cirnet y los equipos Circutor, dependiendo del modulo de comunicación, pero no se pueden variar los graficos que el sistema puede presentar, tampoco pueden ser manipulados, por que dependen de las tablas o bases de datos ya creadas y que no pueden variarse, ya que son archivos de solo lectura.

#### ***1.5.1.6 Datos en tiempo real.***

El monitoreo de los datos en tiempo real es de vital importancia para conocer el estado del sistema en un momento determinado (presente). Por esta razón existen varias formas de presentarlos, entre ellas las que más se utilizan en este proyecto son: datos numéricos, gráficos de tendencia, gráficos porcentuales, esquemas análogos, etc.

Los datos numéricos son utilizados en la mayoría de pantallas, de tal forma que se constituyen en la forma más común de presentación de datos y la que obligatoriamente el sistema debe presentar. Los datos mostrados en los gráficos de tendencia se han ubicado con relación a la importancia de los mismos y de acuerdo a la necesidad de tener un tiempo de evaluación de la variable para verificar su comportamiento en un rango corto de tiempo. Los datos presentados en forma porcentual también son obligatorios debido a que están relacionados con el valor nominal o máximo que posee dicha variable y los limites dentro de los que se debe mantener para estar controlada. La presentación de datos en forma analógica esta relacionada solo con una forma adicional de presentación de datos y

se constituye en una forma opcional en que el usuario puede verificar el estado del sistema.

#### ***1.5.1.7 Tablas numéricas del data logger y energía.***

Otra manera de obtener información a partir de los datos adquiridos y almacenados por el sistema, con cierto grado de procesamiento son las tablas de datos numéricos que permite visualizar e imprimir *Cirnet*. Esto se puede realizar por periodos de medición predeterminados inicialmente. En las tablas se presentan los datos que conforman los diferentes gráficos que se han mencionado en el criterio anterior, con la diferencia de que estos están disponibles para otro tipo de análisis, o simplemente para verificar como se ha comportado el sistema en momento determinados dentro de un periodo de medición.

#### ***1.5.1.8 Red Circutor CVMk y el equipo CVM 96.***

La aplicación SCADA incluye la conexión física de equipos analizadores de redes CVMk, y se ha incluido el equipo monofásico de verificación de calidad de energía, solo como equipo de prueba en la red, para establecer la forma en que se conectan los diferentes componentes de un sistema SCADA y sus diferentes nodos. El equipo CVM 96 no ha sido incluido en el desarrollo del sistema por que no era parte de los objetivos del proyecto, pero si puede ser parte de la prueba del mismo. Otra razón fundamental para la no-inclusión de este equipo es el hecho de que *Cirnet* no permite la adquisición de datos para estos analizadores CVM 96; solamente es permitida la conexión del equipo para el monitoreo del CVM 96, es decir que este se muestra conectado a la red. Para esta topología de equipos existen otros programas que permiten la adquisición y análisis de los datos que el equipo CVM 96 verifica. La observación de los datos que el CVM 96 obtiene puede realizarse a nivel local, en el aparato, y debido a que es monofásico, las variables que permite monitorear son limitadas.

### **1.5.1.9 Verificación de datos a nivel remoto (Internet).**

Las nuevas tecnologías que se basan en los Servicios Web (estas consisten en relacionar o vincular información sobre infraestructura de estándares o plataformas abiertas e infraestructura común), por medio de tres áreas fundamentales: protocolos de comunicación, descripción de servicios y plataforma abierta, cada uno de los cuales esta especificado por estándares abiertos. Esto permitirá integrar de forma versátil y segura información distribuida en diferentes equipos, físicamente alejados, pero enlazados por medio del Internet, para poder fusionar la información y que esta pueda ser verificada por un usuario remoto y geográficamente alejado del sistema que se está supervisando, pudiendo acceder a la información y estado del sistema en tiempo real, según se muestra en el siguiente esquema:

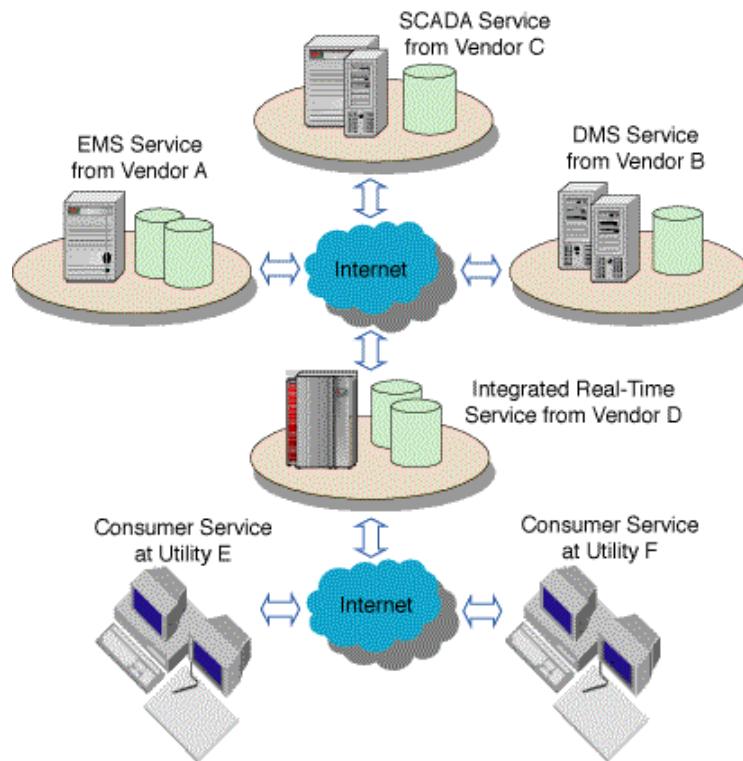


Figura 1.1 Web Services en un sistema de información.

Con relación a lo expuesto anteriormente se ha desarrollado un sistema de monitoreo que puede operar de forma remota a través de Internet. Esta herramienta posee características similares a las de la herramienta desarrollada en

MS Access en cuanto a la sencillez con que las variables pueden ser verificadas, y añaden la versatilidad de poseer información a distancia. El criterio utilizado está planteado en los objetivos iniciales y lo respalda la necesidad de mantener informado al responsable del sistema incluso a larga distancia (a través del Internet). Esta herramienta de monitoreo a distancia posee características de seguridad, ya que ninguna persona no autorizada puede verificar la información que ha estado almacenando y vinculando en Internet. Además, no se pueden realizar modificaciones a ninguna aplicación o herramienta, sin previa autorización.

## 2.1 Generalidades sobre los Software SCADA.

### 2.1.1 Definición de sistema SCADA.

SCADA es el acrónimo de Supervisory Control And Data Acquisition. Un sistema SCADA esta basado en computadores que permiten supervisar y controlar a distancia una instalación, proceso o sistema de características variadas. A diferencia de los Sistemas de Control Distribuido, el lazo de control es generalmente cerrado por el operador. Los Sistemas de Control Distribuido se caracterizan por realizar las acciones de control en forma automática. Hoy en día es fácil hallar un sistema SCADA realizando labores de control automático en cualquiera de sus niveles, aunque su labor principal sea de supervisión y control por parte del operador. En la Tabla 1 se muestra un cuadro comparativo de las principales características de los sistemas SCADA y los Sistemas de Control Distribuido (DCS) (Estas Características no son limitantes para uno u otro tipo de sistemas, sino que por el contrario son típicas).

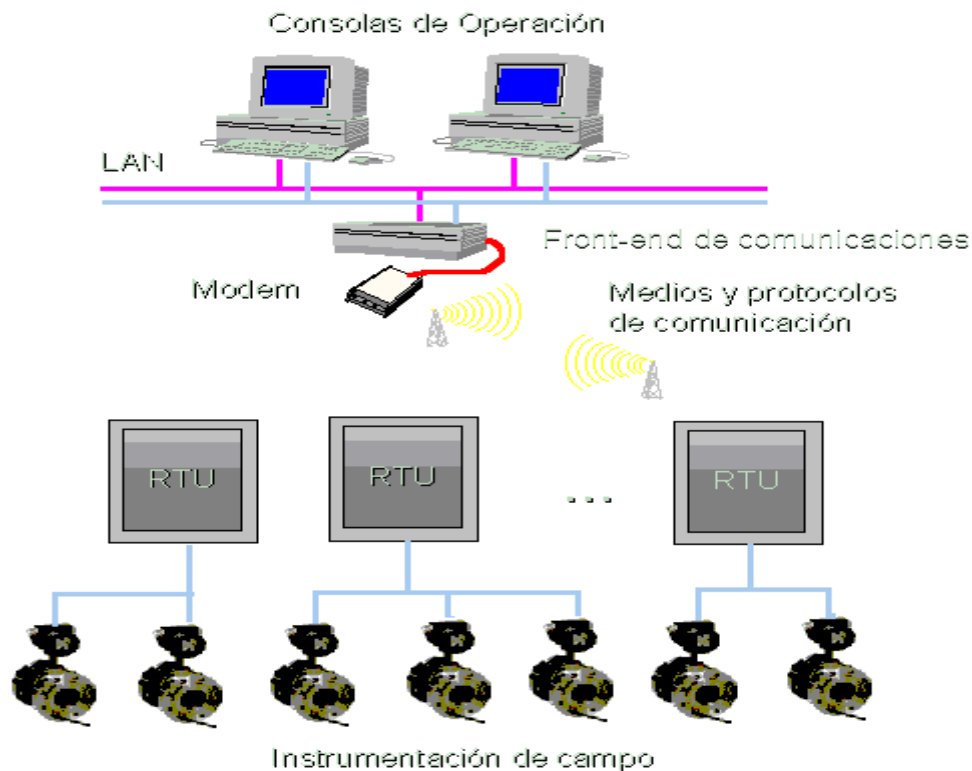


Figura 2.1 Forma de interacción de los sistemas SCADA.



<b>ASPECTO</b>	<b>SCADA</b>	<b>DCS</b>
TIPO DE ARQUITECTURA	CENTRALIZADA	DISTRIBUIDA
TIPO DE CONTROL PREDOMINANTE	SUPERVISORIO: Lazos de control cerrados por el operador. Adicionalmente: control secuencial y regulatorio.	REGULATORIO: Lazos de control cerrados automáticamente por el sistema. Adicionalmente: control secuencial, batch, algoritmos avanzados, etc.
TIPOS DE VARIABLES	DESACOPLADAS	ACOPLADAS
ÁREA DE ACCIÓN	Áreas geográficamente distribuidas.	Área de la planta.
UNIDADES DE ADQUISICIÓN DE DATOS Y CONTROL	Remotas, PLC´s.	Controladores de lazo, PLC´s.
MEDIOS DE COMUNICACIÓN	Radio, satélite, líneas telefónicas, conexión directa, LAN, WAN.	Redes de área local, conexión directa.
BASE DE DATOS	CENTRALIZADA	DISTRIBUIDA

Tabla 2.1 Algunas diferencias típicas entre sistemas SCADA y DCS.

El flujo de la información en los sistemas SCADA es como se describe a continuación: el fenómeno físico lo constituye la variable que deseamos medir. Dependiendo del proceso, la naturaleza del fenómeno es muy diversa: presión, temperatura, flujo de potencia, intensidad de corriente, voltaje, etc. Este fenómeno debe traducirse a una variable que sea inteligible para el sistema SCADA, es decir, en una variable eléctrica. Para ello, se utilizan los sensores o transductores. Los sensores o transductores convierten las variaciones del fenómeno físico en variaciones proporcionales de una variable eléctrica. Las variables eléctricas más utilizadas son: voltaje, corriente, carga, resistencia o capacitancia. Sin embargo, esta variedad de tipos de señales eléctricas debe ser

procesada para ser entendida por el computador digital. Para ello se utilizan acondicionadores de señal, cuya función es la de referenciar estos cambios eléctricos a una misma escala de corriente o voltaje. Además, provee aislamiento eléctrico y filtraje de la señal con el objeto de proteger el sistema de transientes y ruidos originados en el campo. Una vez acondicionada la señal, la misma se convierte en un valor digital equivalente en el bloque de conversión de datos. Generalmente, esta función es llevada a cabo por un circuito de conversión analógico/digital. La computadora (PC) almacena esta información, la cual es utilizada para su análisis y para la toma de decisiones. Simultáneamente, se muestra la información al usuario del sistema, en tiempo real. Basado en la información, el operador puede tomar la decisión de realizar una acción de control sobre el proceso. El operador comanda al computador a realizarla, y de nuevo debe convertirse la información digital a una señal eléctrica. Esta señal eléctrica es procesada por una salida de control, el cual funciona como un acondicionador de señal, la cual la transforma de escala para manejar un dispositivo dado: bobina de un relé, setpoint de un controlador, etc.

### *2.1.2 Necesidad de un sistema SCADA.*

Para evaluar si un sistema SCADA es necesario para manejar una instalación dada, el proceso a controlar debe cumplir las siguientes características:

- a) El número de variables del proceso que se necesita monitorear es alto.
- b) El proceso está geográficamente distribuido. Esta condición no es limitativa, ya que puede instalarse un SCADA para la supervisión y control de un proceso concentrado en una localidad.
- c) La información del proceso se necesita en el momento en que los cambios se producen en el mismo, o en otras palabras, la información se requiere en tiempo real.
- d) La necesidad de optimizar y facilitar las operaciones de la planta, así como la toma de decisiones, tanto gerenciales como operativas.

- e) Los beneficios obtenidos en el proceso justifican la inversión en un sistema SCADA. Estos beneficios pueden reflejarse como aumento de la efectividad de la producción, de los niveles de seguridad, etc.
- f) La complejidad y velocidad del proceso permiten que la mayoría de las acciones de control sean iniciadas por un operador. En caso contrario, se requerirá de un Sistema de Control Automático, el cual lo puede constituir un Sistema de Control Distribuido, PLC's, Controladores a Lazo Cerrado o una combinación de ellos.

### *2.1.3 Funciones de un Sistema SCADA.*

Dentro de las funciones básicas realizadas por un sistema SCADA están las siguientes:

1. Recabar, almacenar y mostrar información, en forma continua y confiable, correspondiente a la señalización de campo: estados de dispositivos, mediciones, alarmas, etc.
2. Ejecutar acciones de control iniciadas por el operador, tales como: abrir o cerrar válvulas, arrancar o parar bombas, etc.
3. Alertar al operador de cambios detectados en la planta, tanto aquellos que no se consideren normales (alarmas) como cambios que se produzcan en la operación diaria de la planta (eventos). Estos cambios son almacenados en el sistema para su posterior análisis.
4. Aplicaciones en general, basadas en la información obtenida por el sistema, tales como: reportes, gráficos de tendencia, historia de variables, cálculos, predicciones, detección de fugas, etc.

### *2.1.4 Ventajas de los sistemas SCADA.*

- No se requiere de personal para realizar labores de lectura de medidores ya que estos son leídos y enviados a terminales a través de la red.

- Es mucho más confiable que los sistemas tradicionales.
- Sistemas de bajo costo de mantenimiento
- Sistema más rápido de medición

#### *2.1.5 Desventajas de un sistema SCADA.*

- Se requiere de una red confiable ya que no se podrá realizar las mediciones en caso de no contar con la red
- Altos costos iniciales ya que hay que adquirir equipos e implantar la solución.
- Se requiere además realizar gastos en conexión a la red de datos.

#### *2.1.6 Prestaciones de un SCADA.*

Un paquete SCADA debe estar en disposición de ofrecer las siguientes prestaciones:

- a. Posibilidad de crear paneles de alarma, que exigen la presencia del operador para reconocer una parada o situación de alarma, con registro de incidencias.
- b. Generación de históricos de señal de planta, que pueden ser volcados para su proceso sobre una hoja de cálculo.
- c. Ejecución de programas, que modifican la ley de control, o incluso anular o modificar las tareas asociadas al autómatas, bajo ciertas condiciones.
- d. Posibilidad de programación numérica, que permite realizar cálculos aritméticos de elevada resolución sobre la CPU del ordenador.

Con ellas, se pueden desarrollar aplicaciones para ordenadores (tipo PC, por ejemplo), con captura de datos, análisis de señales, presentaciones en pantalla, envío de resultados a disco e impresora, etc.

Además, todas estas acciones se llevan a cabo mediante un paquete de funciones que incluye zonas de programación en un lenguaje de uso general (como C, Pascal, o Basic), lo cual confiere una potencia muy elevada y una gran versatilidad. Algunos SCADA ofrecen librerías de funciones para lenguajes de uso general que

permiten personalizar de manera muy amplia la aplicación que desee realizarse con dicho SCADA.

#### *2.1.7 Requisitos de un sistema SCADA.*

Un SCADA debe cumplir varios objetivos para que su instalación sea perfectamente aprovechada:

Deben ser sistemas de arquitectura abierta, capaces de crecer o adaptarse según las necesidades cambiantes de la empresa.

Deben comunicarse con total facilidad y de forma transparente al usuario con el equipo de planta y con el resto de la empresa (redes locales y de gestión).

Deben ser programas sencillos de instalar, sin excesivas exigencias de hardware, y fáciles de utilizar, con interfaces amigables con el usuario.

#### *2.1.8 Módulos de un SCADA.*

Los módulos o bloques software que permiten las actividades de adquisición, supervisión y control son los siguientes:

- **Configuración:** permite al usuario definir el entorno de trabajo de su SCADA, adaptándolo a la aplicación particular que se desea desarrollar.
- **Interfaz gráfica del operador:** proporciona al operador las funciones de control y supervisión de la planta. El proceso se representa mediante sinópticos gráficos almacenados en el ordenador de proceso y generados desde el editor incorporado en el SCADA o importados desde otra aplicación durante la configuración del paquete.
- **Módulo de proceso:** ejecuta las acciones de mando preprogramadas a partir de los valores actuales de variables leídas.
- **Gestión y archivo de datos:** se encarga del almacenamiento y procesado ordenado de los datos, de forma que otra aplicación o dispositivo pueda tener acceso a ellos.

- **Comunicaciones:** se encarga de la transferencia de información entre la planta y la arquitectura hardware que soporta el SCADA, y entre ésta y el resto de elementos informáticos de gestión.

## **2.2 Bases de Datos.**

Las bases de datos son las herramientas más utilizadas por los sistemas SCADA. Estas son las encargadas de almacenar toda la información que se está recopilando a través de los diferentes equipos de medición, periféricos, etc. Estas permiten obtener datos históricos que resultan útiles para la consulta en caso de anomalías en los equipos supervisados. Los datos recopilados además de proporcionar información acerca del comportamiento del sistema supervisado, también permiten realizar análisis a partir de comparación de resultados, por ejemplo, el comportamiento del sistema en un mes con respecto a otro y a partir de los datos de operación deducir si este está en buen estado. Por esta razón todo sistema SCADA gira alrededor de su base de datos, en cuanto a la utilidad que estas proporcionan como herramienta de supervisión y análisis.

El sistema de supervisión y adquisición de datos que se presenta en este trabajo no es la excepción, es decir que también gira alrededor de las bases de datos que se almacenan, no solo en cuanto a información instantánea, sino también en cuanto a información histórica.

### *2.2.1 Generalidades sobre Bases de Datos.*

#### **2.2.1.1 Definición de Bases de Datos.**

Un conjunto de información almacenada en memoria auxiliar que permite acceso directo y un conjunto de programas que manipulan esos datos

Base de Datos es un conjunto exhaustivo no redundante de datos estructurados y organizados independientemente de su utilización, y su implementación en máquinas accesibles en tiempo real y compatibles con usuarios concurrentes con necesidad de información diferente y no predecible en tiempo.

### *2.2.2 Orígenes y Antecedentes de las bases de datos.*

La historia de las bases de datos surge desde mediados de los años sesenta. En 1970 **Codd** propuso el modelo relacional, este modelo es el que ha marcado la línea de investigación por muchos años, ahora se encuentran los modelos orientados a objetos.

#### **2.2.2.1 Ventajas de las bases de datos.**

- Independencia de datos y tratamiento.
- Cambio en datos no implica cambio en programas y viceversa (menor costo de mantenimiento).
- Coherencia de resultados.
- Reduce redundancia.
- Acciones lógicamente únicas.
- Se evita inconsistencia.
- Mejora en la disponibilidad de datos
- No hay dueño de datos (No igual es igual a ser públicos).
- Ni aplicaciones ni usuarios.
- Guardar descripción (Da la idea de catálogos).
- Cumplimiento de ciertas normas.
- Restricciones de seguridad.
- Accesos (Usuarios de datos).
- Operaciones (Operaciones sobre datos).
- Más eficiente gestión de almacenamiento.

#### **2.2.2.2 Componentes de un sistema de base de datos.**

Un sistema de bases de datos estará formado por:

- Datos
- Programas
- Personas
- Máquinas

Las funciones de cada componente de un sistema de base de datos son las siguientes:

a. **Datos.**

Es lo que se conoce como base de datos propiamente dicha. Para manejar estos datos se utilizan una serie de programas.

b. **Programas.**

Son los encargados de manejar los datos, son conocidos como DBMS (Data Base Management System) o también SGBD (Sistema Gestor de Base de Datos). Los DBMS tienen dos funciones principales que son:

- La definición de las estructuras para almacenar los datos.
- La manipulación de los datos.

c. **Personas.**

Encargados de manipular programas y datos soportados en las bases de datos.

d. **Equipos.**

Infraestructura física que puede manejar los programas y datos, y son manejados a su vez por las personas para verificación y tratamiento de las bases de datos.

### ***2.2.2.3 Clasificación de las bases de datos por su uso y aplicación.***

a. Bases de datos referenciales.

Son aquellas que ofrecen registros que su vez son representaciones de documentos primarios. Dentro de éstas distinguen, as su vez, entre:

- Bibliográficas: aquellas cuyo contenido son registros de tipo bibliográfico.
- Directorios: aquellas cuyo contenido está referido a la descripción de otros recursos de información, como por ejemplo un directorio de bases de datos.

b. Bases de datos fuente.

Son aquellas bases de datos que ofrecen el documento completo, no una representación del mismo. También se distinguen entre ellas:

Numéricas: contienen información de tipo numérico, como un censo o indicadores cuantitativos.



- Textuales: ofrecen el texto completo de un documento.
- Mixtas: combinan ambos tipos de información, como por ejemplo informes económicos de empresas, datos geo-económicos, etc.

## **2.3 Teoría de Redes.**

Las redes hoy en día forman parte del desarrollo de toda entidad. Algunos ejemplos de estas son la red Internet. Esta red ha revolucionado las formas de comunicación, de almacenamiento y consulta de la información, etc. Esta red internacional permite compartir desde archivos hasta bases de datos y correos electrónicos. Existe otro tipo de redes que están limitadas por las fronteras de una organización o compañía. Esas redes reciben el nombre de Intranet. Poseen las mismas características que la Internet en cuanto a compartir información, pero esto se realiza de forma segura, ya que las únicas personas que pueden comunicarse, compartir y almacenar información, son aquellas que laboran para la empresa en la cual esta situada dicha red interna (Intranet).

Se denomina red de computadoras a una serie de host autónomos y dispositivos especiales intercomunicados entre sí. Sin embargo, este concepto genérico de red incluye multitud de tipos diferentes de redes y posibles configuraciones de las mismas, por lo que desde un principio surgió la necesidad de establecer clasificaciones que permitieran identificar estructuras de red concretas.

Las posibles clasificaciones de las redes pueden ser muchas, atendiendo cada una de ellas a diferentes propiedades, las mas comúnmente aceptadas se describen en el siguiente apartado.

## **2.4 Clasificación de las Redes.**

### *2.4.1 Según su tamaño y extensión.*

**Redes LAN.** Las redes de área local (Local Area Network) son redes de ordenadores cuya extensión es del orden de entre 10 metros a 1 kilómetro. Son redes pequeñas, habituales en oficinas, colegios y empresas pequeñas, que generalmente usan la tecnología de broadcast, es decir, aquella en que a un sólo

cable se conectan todas las máquinas. Como su tamaño es restringido, el peor tiempo de transmisión de datos es conocido, siendo velocidades de transmisión típicas de LAN las que van de 10 a 100 Mbps (Megabits por segundo).

**Redes MAN.** Las redes de área metropolitana (Metropolitan Area Network) son redes de ordenadores de tamaño superior a una LAN, soliendo abarcar el tamaño de una ciudad. Son típicas de empresas y organizaciones que poseen distintas oficinas repartidas en un mismo área metropolitana, por lo que, en su tamaño máximo, comprenden un área de unos 10 kilómetros.

**Redes WAN.** Las redes de área amplia (Wide Area Network) tienen un tamaño superior a una MAN, y consisten en una colección de host o de redes LAN conectadas por una subred. Esta subred está formada por una serie de líneas de transmisión interconectadas por medio de routers, aparatos de red encargados de rutear o dirigir los paquetes hacia la LAN o host adecuado, enviándose éstos de un router a otro. Su tamaño puede oscilar entre 100 y 1000 kilómetros.

**Redes Internet.** Una Internet es una red de redes, vinculadas mediante ruteadores gateways. Un gateway o pasarela es un computador especial que puede traducir información entre sistemas con formato de datos diferentes. Su tamaño puede ser desde 10000 kilómetros en adelante, y su ejemplo más claro es Internet, la red de redes mundial.

**Redes inalámbricas.** Las redes inalámbricas son redes cuyos medios físicos no son cables de cobre de ningún tipo, lo que las diferencia de las redes anteriores. Están basadas en la transmisión de datos mediante ondas de radio, microondas, satélites o infrarrojos.

#### *2.4.2 Según la tecnología de transmisión.*

**Redes de Broadcast.** Aquellas redes en las que la transmisión de datos se realiza por un sólo canal de comunicación, compartido entonces por todas las máquinas de la red. Cualquier paquete de datos enviado por cualquier máquina es recibido por todas las de la red.

**Redes Point-To-Point.** Aquellas en las que existen muchas conexiones entre parejas individuales de máquinas. Para poder transmitir los paquetes desde una máquina a otra a veces es necesario que éstos pasen por máquinas intermedias, siendo obligado en tales casos un trazado de rutas mediante dispositivos routers.

Clasificación de las redes según el tipo de transferencia de datos que soportan:

**Redes de transmisión simple.** Son aquellas redes en las que los datos sólo pueden viajar en un sentido.

**Redes Half-Duplex.** Aquellas en las que los datos pueden viajar en ambos sentidos, pero sólo en uno de ellos en un momento dado. Es decir, sólo puede haber transferencia en un sentido a la vez.

**Redes Full-Duplex.** Aquellas en las que los datos pueden viajar en ambos sentidos a la vez.

### *2.4.3 Topologías de red*

Hemos visto en el tema sobre el modelo OSI y la arquitectura TCP/IP que las redes de ordenadores surgieron como una necesidad de interconectar los diferentes host de una empresa o institución para poder así compartir recursos y equipos específicos.

Pero los diferentes componentes que van a formar una red se pueden interconectar o unir de diferentes formas, siendo la forma elegida un factor fundamental que va a determinar el rendimiento y la funcionalidad de la red.

La disposición de los diferentes componentes de una red se conoce con el nombre de **topología de la red**. La topología idónea para una red concreta va a depender de diferentes factores, como el número de máquinas a interconectar, el tipo de acceso al medio físico que deseemos, etc.

Podemos distinguir tres aspectos diferentes a la hora de considerar una topología:

1. La topología física, que es la disposición real de las máquinas, dispositivos de red y cableado (los medios) en la red.

2. La topología lógica, que es la forma en que las máquinas se comunican a través del medio físico. Los dos tipos más comunes de topologías lógicas son broadcast (Ethernet) y transmisión de tokens (Token Ring).
3. La topología matemática, mapas de nodos y enlaces, a menudo formando patrones.

La topología de broadcast simplemente significa que cada host envía sus datos hacia todos los demás hosts del medio de red. Las estaciones no siguen ningún orden para utilizar la red, sino que cada máquina accede a la red para transmitir datos en el momento en que lo necesita. Esta es la forma en que funciona Ethernet.

En cambio, la transmisión de tokens controla el acceso a la red al transmitir un token eléctrico de forma secuencial a cada host. Cuando un host recibe el token significa que puede enviar datos a través de la red. Si el host no tiene ningún dato para enviar, transmite el token hacia el siguiente host y el proceso se vuelve a repetir.

Vamos a ver a continuación los principales modelos de topología.

Modelos de topología

Las principales modelos de topología son:

#### ***2.4.3.1 Topología de bus***

La topología de bus tiene todos sus nodos conectados directamente a un enlace y no tiene ninguna otra conexión entre nodos. Físicamente cada host está conectado a un cable común, por lo que se pueden comunicar directamente, aunque la ruptura del cable hace que los hosts queden desconectados.

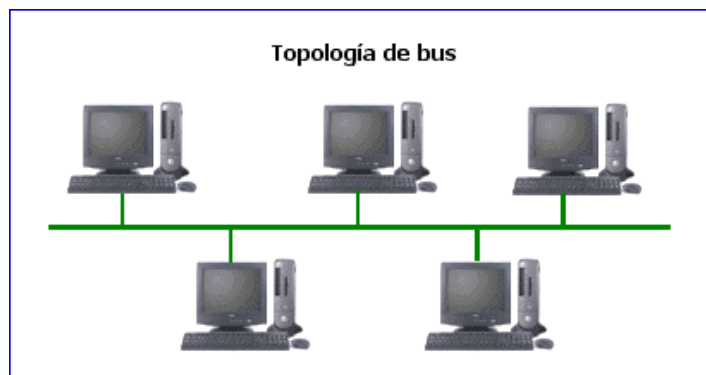


Figura 2.2 Topología de Bus.

La topología de bus permite que todos los dispositivos de la red puedan ver todas las señales de todos los demás dispositivos, lo que puede ser ventajoso si desea que todos los dispositivos obtengan esta información. Sin embargo, puede representar una desventaja, ya que es común que se produzcan problemas de tráfico y colisiones, que se pueden paliar segmentando la red en varias partes. Es la topología más común en pequeñas LAN, con hub o switch final en uno de los extremos.

#### **2.4.3.2 Topología de anillo**

Una topología de anillo se compone de un solo anillo cerrado formado por nodos y enlaces, en el que cada nodo está conectado solamente con los dos nodos adyacentes.



Figura 2.3 Topología en Anillo.

Los dispositivos se conectan directamente entre sí por medio de cables en lo que se denomina una cadena margarita. Para que la información pueda circular, cada estación debe transferir la información a la estación adyacente.

### ***2.4.3.3 Topología de anillo doble***

Una topología en anillo doble consta de dos anillos concéntricos, donde cada host de la red está conectado a ambos anillos, aunque los dos anillos no están conectados directamente entre sí. Es análoga a la topología de anillo, con la diferencia de que, para incrementar la confiabilidad y flexibilidad de la red, hay un segundo anillo redundante que conecta los mismos dispositivos.

La topología de anillo doble actúa como si fueran dos anillos independientes, de los cuales se usa solamente uno por vez.

### ***2.4.3.4 Topología en estrella***

La topología en estrella tiene un nodo central desde el que se irradian todos los enlaces hacia los demás nodos. Por el nodo central, generalmente ocupado por un hub, pasa toda la información que circula por la red.

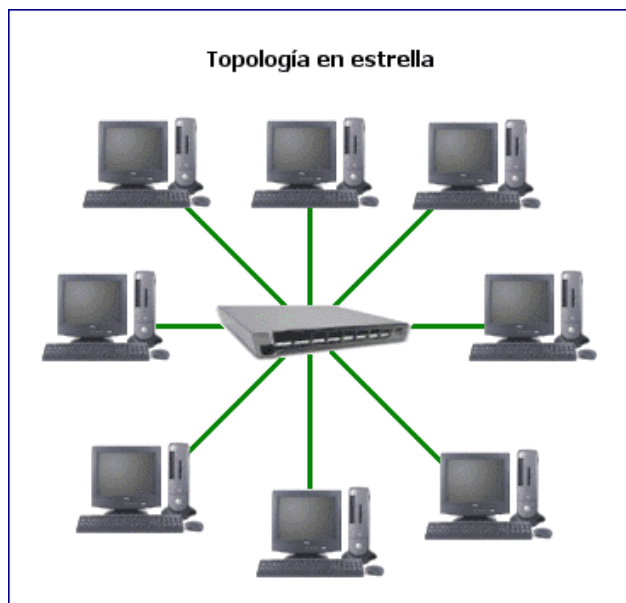


Figura 2.4 Topología en Estrella.

La ventaja principal es que permite que todos los nodos se comuniquen entre sí de manera conveniente. La desventaja principal es que si el nodo central falla, toda la red se desconecta.

#### ***2.4.3.5 Topología en estrella extendida:***

La topología en estrella extendida es igual a la topología en estrella, con la diferencia de que cada nodo que se conecta con el nodo central también es el centro de otra estrella. Generalmente el nodo central está ocupado por un hub ó un switch, y los nodos secundarios por hubs.

La ventaja de esto es que el cableado es más corto y limita la cantidad de dispositivos que se deben interconectar con cualquier nodo central.

La topología en estrella extendida es sumamente jerárquica, y busca que la información se mantenga local. Esta es la forma de conexión utilizada actualmente por el sistema telefónico.

#### ***2.4.3.6 Topología en árbol***

La topología en árbol es similar a la topología en estrella extendida, salvo en que no tiene un nodo central. En cambio, un nodo de enlace troncal, generalmente ocupado por un hub o switch, desde el que se ramifican los demás nodos.



Figura 2.5 Topología en Árbol.

El enlace troncal es un cable con varias capas de ramificaciones, y el flujo de información es jerárquico. Conectado en el otro extremo al enlace troncal generalmente se encuentra un host servidor.

#### ***2.4.3.7 Topología en malla completa***

En una topología de malla completa, cada nodo se enlaza directamente con los demás nodos. Las ventajas son que, como cada todo se conecta físicamente a los demás, creando una conexión redundante, si algún enlace deja de funcionar la información puede circular a través de cualquier cantidad de enlaces hasta llegar a destino. Además, esta topología permite que la información circule por varias rutas a través de la red.

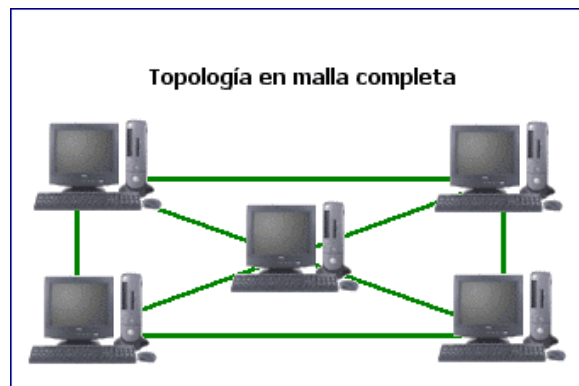


Figura 2.6 Topología en Malla completa.

La desventaja física principal es que sólo funciona con una pequeña cantidad de nodos, ya que de lo contrario la cantidad de medios necesarios para los enlaces, y la cantidad de conexiones con los enlaces se torna abrumadora.

#### ***2.4.3.8 Topología de red celular***

La topología celular está compuesta por áreas circulares o hexagonales, cada una de las cuales tiene un nodo individual en el centro.



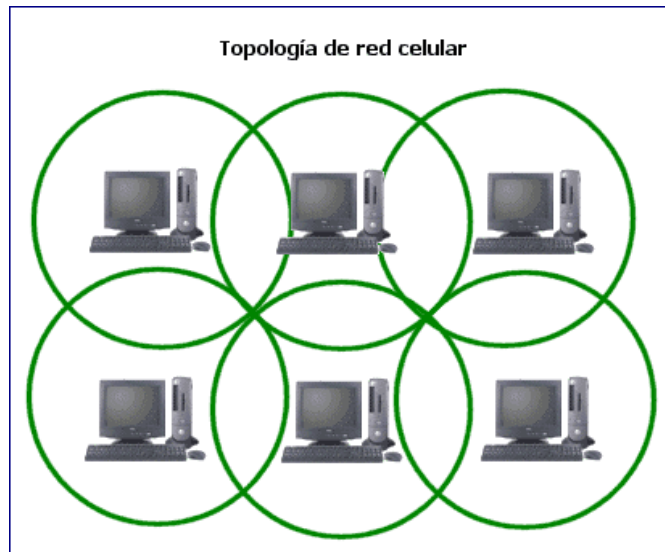


Figura 2.7 Topología de Red celular.

La topología celular es un área geográfica dividida en regiones (celdas) para los fines de la tecnología inalámbrica. En esta tecnología no existen enlaces físicos; sólo hay ondas electromagnéticas.

La ventaja obvia de una topología celular (inalámbrica) es que no existe ningún medio tangible aparte de la atmósfera terrestre o el del vacío del espacio exterior (y los satélites). Las desventajas son que las señales se encuentran presentes en cualquier lugar de la celda y, de ese modo, pueden sufrir disturbios y violaciones de seguridad.

Como norma, las topologías basadas en celdas se integran con otras topologías, ya sea que usen la atmósfera o los satélites.

#### ***2.4.3.9 Topología irregular***

En este tipo de topología no existe un patrón obvio de enlaces y nodos. El cableado no sigue un modelo determinado; de los nodos salen cantidades variables de cables. Las redes que se encuentran en las primeras etapas de construcción, o se encuentran mal planificadas, a menudo se conectan de esta manera.

Las topologías LAN más comunes son:

- **Ethernet:** topología de bus lógica y en estrella física o en estrella extendida.
- **Token Ring:** topología de anillo lógica y una topología física en estrella.
- **FDDI:** topología de anillo lógica y topología física de anillo doble.

A continuación se explican detenidamente.

#### *2.4.4 Redes LAN Ethernet*

Ethernet es la tecnología de red LAN más usada, resultando idóneas para aquellos casos en los que se necesita una red local que deba transportar tráfico esporádico y ocasionalmente pesado a velocidades muy elevadas. Las redes Ethernet se implementan con una topología física de estrella y lógica de bus, y se caracterizan por su alto rendimiento a velocidades de 10-100 Mbps.

El origen de las redes Ethernet hay que buscarlo en la Universidad de Hawai, donde se desarrolló, en los años setenta, el **Método de Acceso Múltiple con Detección de Portadora y Detección de Colisiones, CSMA/CD** (Carrier Sense and Multiple Access with Collision Detection), utilizado actualmente por Ethernet. Este método surgió ante la necesidad de implementar en las islas Hawai un sistema de comunicaciones basado en la transmisión de datos por radio, que se llamó Aloha, y permite que todos los dispositivos puedan acceder al mismo medio, aunque sólo puede existir un único emisor en cada instante. Con ello todos los sistemas pueden actuar como receptores de forma simultánea, pero la información debe ser transmitida por turnos.

El centro de investigaciones PARC (Palo Alto Research Center) de la Xerox Corporation desarrolló el primer sistema Ethernet experimental en los años 70, que posteriormente sirvió como base de la especificación 802.3 publicada en 1980 por el Institute of Electrical and Electronic Engineers (IEEE).

Las redes Ethernet son de carácter no determinista, en la que los hosts pueden transmitir datos en cualquier momento. Antes de enviarlos, escuchan el medio de transmisión para determinar si se encuentra en uso. Si lo está, entonces esperan.

En caso contrario, los host comienzan a transmitir. En caso de que dos o más host empiecen a transmitir tramas a la vez se producirán encontronazos o choques entre tramas diferentes que quieren pasar por el mismo sitio a la vez. Este fenómeno se denomina **colisión**, y la porción de los medios de red donde se producen colisiones se denomina **dominio de colisiones**.

Una colisión se produce pues cuando dos máquinas escuchan para saber si hay tráfico de red, no lo detectan y, acto seguido transmiten de forma simultánea. En este caso, ambas transmisiones se dañan y las estaciones deben volver a transmitir más tarde.

Para intentar solventar esta pérdida de paquetes, las máquinas poseen mecanismos de detección de las colisiones y algoritmos de postergación que determinan el momento en que aquellas que han enviado tramas que han sido destruidas por colisiones pueden volver a transmitir.

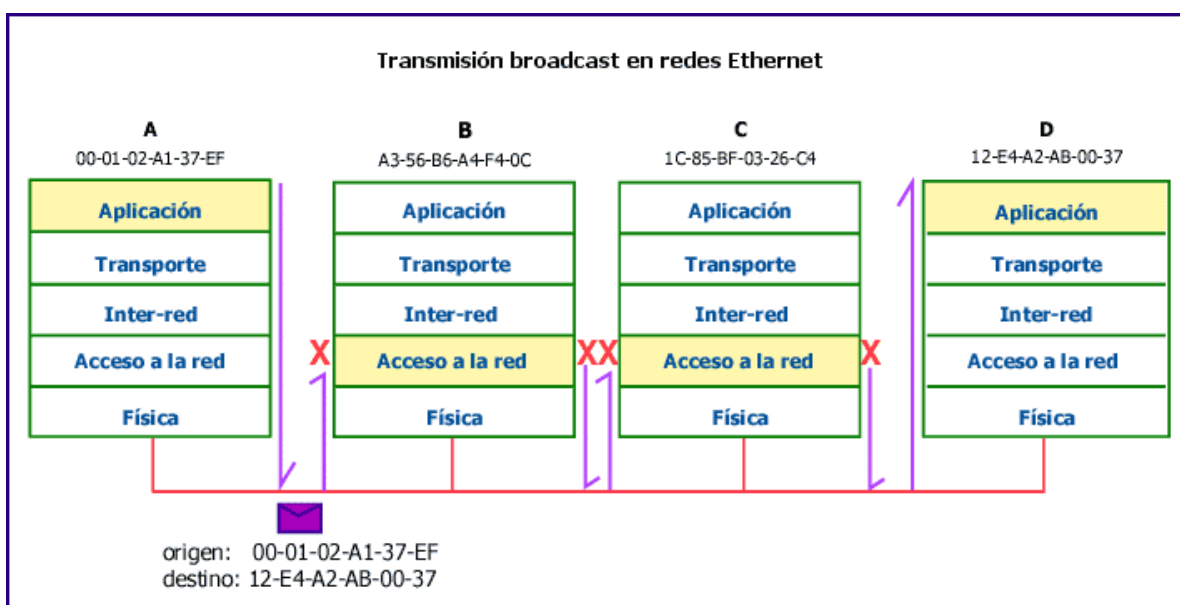


Figura 2.8 Transmisión Broadcast en Redes Ethernet.

Existen dos especificaciones diferentes para un mismo tipo de red, Ethernet y IEEE 802.3. Ambas son **redes de broadcast**, lo que significa que cada máquina puede ver todas las tramas, aunque no sea el destino final de las mismas. Cada máquina examina cada trama que circula por la red para determinar si está destinada a ella.

De ser así, la trama pasa a las capas superiores para su adecuado procesamiento. En caso contrario, la trama es ignorada.

Ethernet proporciona servicios correspondientes a las capas física y de enlace de datos del modelo de referencia OSI, mientras que IEEE 802.3 especifica la capa física y la porción de acceso al canal de la capa de enlace de datos, pero no define ningún protocolo de Control de Enlace Lógico.

Ethernet es una tecnología de broadcast de medios compartidos. El método de acceso CSMA/CD que se usa en Ethernet ejecuta tres funciones:

1. Transmitir y recibir paquetes de datos.
2. Decodificar paquetes de datos y verificar que las direcciones sean válidas antes de transferirlos a las capas superiores del modelo OSI.>
3. Detectar errores dentro de los paquetes de datos o en la red.

Tanto Ethernet como IEEE 802.3 se implementan a través de la **tarjeta de red** o por medio de circuitos en una placa dentro del host.

#### *2.4.5 Formato de trama Ethernet*

Según se ha visto, los datos generados en la capa de aplicación pasan a la capa de transporte, que los divide en segmentos, porciones de datos aptas para su transporte por red, y luego van descendiendo por las sucesivas capas hasta llegar a los medios físicos. Conforme los datos van bajando por la pila de capas, paso a paso cada protocolo les va añadiendo una serie de cabeceras y datos adicionales; necesarios para poder ser enviados a su destino correctamente. El resultado final es una serie de unidades de información denominadas tramas, que son las que viajan de un host a otro.

La forma final de la trama obtenida, en redes Ethernet, es la siguiente:

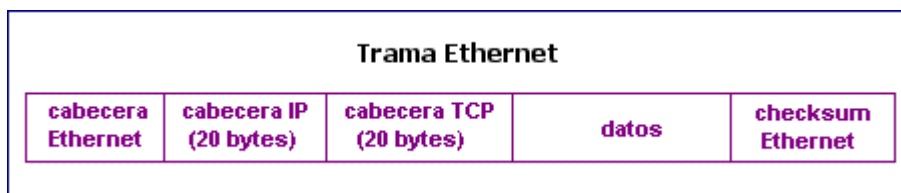


Figura 2.9 Trama Ethernet.

Y los principales campos que la forman son:

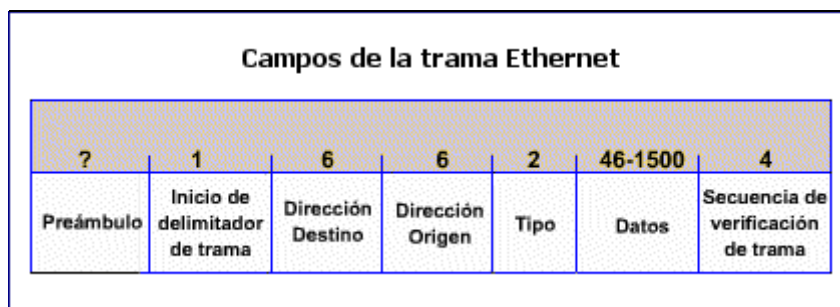


Figura 2.10 Campos de la Trama.

- **Preámbulo:** Patrón de unos y ceros que indica a las estaciones receptoras que una trama es Ethernet o IEEE 802.3. La trama Ethernet incluye un byte adicional que es el equivalente al campo Inicio de Trama (SOF) de la trama IEEE 802.3.
- **Inicio de trama (SOF):** Byte delimitador de IEEE 802.3 que finaliza con dos bits 1 consecutivos, y que sirve para sincronizar las porciones de recepción de trama de todas las estaciones de la red. Este campo se especifica explícitamente en Ethernet.
- **Direcciones destino y origen:** Incluye las direcciones físicas (MAC) únicas de la máquina que envía la trama y de la máquina destino. La dirección origen siempre es una dirección única, mientras que la de destino puede ser de broadcast única (trama enviada a una sola máquina), de broadcast múltiple (trama enviada a un grupo) o de broadcast (trama enviada a todos los nodos).
- **Tipo (Ethernet):** Especifica el protocolo de capa superior que recibe los datos una vez que se ha completado el procesamiento Ethernet.

- Longitud (IEEE 802.3): Indica la cantidad de bytes de datos que sigue este campo.
- Datos: Incluye los datos enviados en la trama. En la especificación IEEE 802.3, si los datos no son suficientes para completar una trama mínima de 64 bytes, se insertan bytes de relleno hasta completar ese tamaño (tamaño mínimo de trama). Por su parte, las especificaciones Ethernet versión 2 no especifican ningún relleno, Ethernet espera por lo menos 46 bytes de datos.
- Secuencia de verificación de trama (FCS): Contiene un valor de verificación CRC (Control de Redundancia Cíclica) de 4 bytes, creado por el dispositivo emisor y recalculado por el dispositivo receptor para verificar la existencia de tramas dañadas.

Cuando un paquete es recibido por el destinatario adecuado, les retira la cabecera de Ethernet y el checksum de verificación de la trama, comprueba que los datos corresponden a un mensaje IP y entonces lo pasa a dicho protocolo para que lo procese. El tamaño máximo de los paquetes en las redes Ethernet es de 1500 bytes.

#### 2.4.6 Tipos de redes Ethernet

Existen por lo menos 18 variedades de Ethernet, relacionadas con el tipo de cableado empleado y con la velocidad de transmisión.

Tipo	Medio	Ancho de banda máximo	Longitud máxima de segmento	Topología Física	Topología Lógica
10Base5	Coaxial grueso	10 Mbps	500 m	Bus	Bus
10Base-T	UTP Cat 5	10 Mbps	100 m	Estrella; Estrella Extendida	Bus
10Base-FL	Fibra óptica multimodo	10 Mbps	2.000 m	Estrella	Bus
100Base-TX	UTP Cat 5	100 Mbps	100 m	Estrella	Bus
100Base-FX	Fibra óptica multimodo	100 Mbps	2.000 m	Estrella	Bus
1000Base-T	UTP Cat 5	1000 Mbps	100 m	Estrella	Bus

Figura 2.11 Variedades de Redes Ethernet.

Las tecnologías Ethernet más comunes y más importantes las son:

- **Ethernet 10Base2.** Usa un cable coaxial delgado, por lo que se puede doblar más fácilmente, y además es más barato y fácil de instalar, aunque los segmentos de cable no pueden exceder de 200 metros y 30 nodos. Las conexiones se hacen mediante *conectores en T*, más fáciles de instalar y más seguros.
- **Ethernet 10Base5.** También llamada Ethernet gruesa, usa un cable coaxial grueso, consiguiendo una velocidad de 10 Mbps. Puede tener hasta 100 nodos conectados, con una longitud de cable de hasta 500 metros. Las conexiones se hacen mediante la técnica denominada *derivaciones de vampiro*, en las cuales se inserta un polo hasta la mitad del cable, realizándose la derivación en el interior de un transceiver, que contiene los elementos necesarios para la detección de portadores y choques. El transceiver se une al computador mediante un cable de hasta 50 metros.

- **Ethernet 10Base-T.** Cada estación tiene una conexión con un hub central, y los cables usados son normalmente de par trenzado. Son las LAN más comunes hoy en día. Mediante este sistema se paliaron los conocidos defectos de las redes 10Base2 y 10Base5, a saber, la mala detección de derivaciones no deseadas, de rupturas y de conectores flojos. Como desventaja, los cables tienen un límite de sólo 100 metros, y los hubs pueden resultar caros.
- **Ethernet 10Base-FX.** Basada en el uso de fibra óptica para conectar las máquinas, lo que la hace cara para un planteamiento general de toda la red, pero idónea para la conexión entre edificios, ya que los segmentos pueden tener una longitud de hasta 2000 metros, al ser la fibra óptica insensible a los ruidos e interferencias típicos de los cables de cobre. Además, su velocidad de transmisión es mucho mayor.
- **Fast Ethernet.** Las redes 100BaseFx (IEEE 802.3u) se crearon con la idea de paliar algunos de los fallos contemplados en las redes Ethernet 10Base-T y buscar una alternativa a las redes FDDI. Son también conocidas como redes Fast Ethernet, y están basadas en una topología en estrella para fibra óptica. Con objeto de hacerla compatible con Ethernet 10Base-T, la tecnología Fast Ethernet preserva los formatos de los paquetes y las interfaces, pero aumenta la rapidez de transmisión hasta los 100 Mbps. En las redes Fast Ethernet se usan cables de cuatro pares trenzados de la clase 3, uno de los cuales va siempre al hub central, otro viene siempre desde el hub, mientras que los otros dos pares son conmutables. En cuanto a la codificación de las señales, se sustituye la codificación Manchester por señalización ternaria, mediante la cual se pueden transmitir 4 bits a la vez. También se puede implementar Fast Ethernet con cableado de la clase 5 en topología de estrella (100BaseTX), pudiendo entonces soportar hasta 100 Mbps con transmisión full dúplex.



### 2.4.7 Redes LAN Token Ring

Las redes Token Ring son redes de tipo determinista, al contrario de las redes Ethernet. En ellas, el acceso al medio está controlado, por lo que solamente puede transmitir datos una máquina por vez, implementándose este control por medio de un token de datos, que define qué máquina puede transmitir en cada instante. Token Ring e IEEE 802.5 son los principales ejemplos de redes de transmisión de tokens.

Las redes de transmisión de tokens se implementan con una topología física de estrella y lógica de anillo, y se basan en el transporte de una pequeña trama, denominada token, cuya posesión otorga el derecho a transmitir datos. Si un nodo que recibe un token no tiene información para enviar, transfiere el token al siguiente nodo. Cada estación puede mantener al token durante un período de tiempo máximo determinado, según la tecnología específica que se haya implementado.

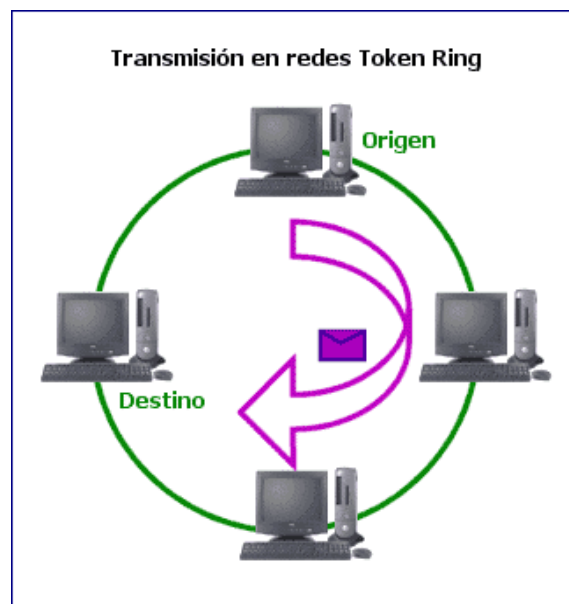


Figura 2.12 Transmisión en Redes Token Ring.

Cuando una máquina recibe un token y tiene información para transmitir, toma el token y le modifica un bit, transformándolo en una secuencia de inicio de trama. A continuación, agrega la información a transmitir a esta trama y la envía al anillo, por el que gira hasta que llega a la estación destino.

Mientras la trama de información gira alrededor del anillo no hay ningún otro token en la red, por lo que ninguna otra máquina puede realizar transmisiones.

Cuando la trama llega a la máquina destino, ésta copia la información contenida en ella para su procesamiento y elimina la trama, con lo que la estación emisora puede verificar si la trama se recibió y se copió en el destino.

Como consecuencia de este método determinista de transmisión, en las redes Token Ring no se producen colisiones, a diferencia de las redes CSMA/CD como Ethernet. Además, en las redes Token Ring se puede calcular el tiempo máximo que transcurrirá antes de que cualquier máquina pueda realizar una transmisión, lo que hace que sean ideales para las aplicaciones en las que cualquier demora deba ser predecible y en las que el funcionamiento sólido de la red sea importante.

La primera red Token Ring fue desarrollada por la empresa IBM en los años setenta, todavía sigue usándose y fue la base para la especificación IEEE 802.5 (método de acceso Token Ring), prácticamente idéntica y absolutamente compatible con ella. Actualmente, el término Token Ring se refiere tanto a la red Token Ring de IBM como a la especificación 802.5 del IEEE.

Las redes Token Ring soportan entre 72 y 260 estaciones a velocidades de 4 a 16 Mbps, se implementan mediante cableado de par trenzado, con blindaje o sin él, y utilizan una señalización de banda base con codificación diferencial de Manchester.

#### Tokens

Los tokens están formados por un byte delimitador de inicio, un byte de control de acceso y un byte delimitador de fin. Por lo tanto, tienen una longitud de 3 bytes.

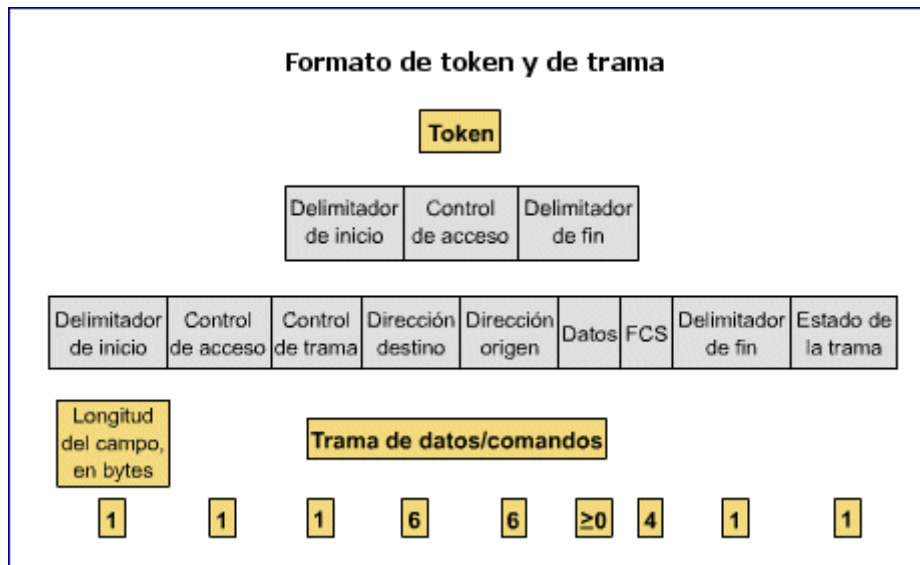


Figura 2.13 Formato de Token y de Trama.

El delimitador de inicio alerta a cada estación ante la llegada de un token o de una trama de datos/comandos. Este campo también incluye señales que distinguen al byte del resto de la trama al violar el esquema de codificación que se usa en otras partes de la trama.

El byte de control de acceso contiene los campos de prioridad y de reserva, así como un bit de token y uno de monitor. El bit de token distingue un token de una trama de datos/comandos y un bit de monitor determina si una trama gira continuamente alrededor del anillo.

El delimitador de fin señala el fin del token o de una trama de datos/comandos. Contiene bits que indican si hay una trama defectuosa y una trama que es la última de una secuencia lógica.

El tamaño de las tramas de datos/comandos varía según el tamaño del campo de información. Las tramas de datos transportan información para los protocolos de capa superior, mientras que las tramas de comandos contienen información de control y no poseen datos para los protocolos de capa superior.

En las tramas de datos o instrucciones hay un byte de control de trama a continuación del byte de control de acceso. El byte de control de trama indica si la trama contiene datos o información de control. En las tramas de control, este byte especifica el tipo de información de control.

A continuación del byte de control de trama hay dos campos de dirección que identifican las estaciones destino y origen. Como en el caso de IEEE 802.5, la longitud de las direcciones es de 6 bytes. El campo de datos está ubicado a continuación del campo de dirección. La longitud de este campo está limitada por el token de anillo que mantiene el tiempo, definiendo de este modo el tiempo máximo durante el cual una estación puede retener al token.

Y a continuación del campo de datos se ubica el campo de secuencia de verificación de trama (FCS). La estación origen completa este campo con un valor calculado según el contenido de la trama. La estación destino vuelve a calcular el valor para determinar si la trama se ha dañado mientras estaba en tránsito. Si la trama está dañada se descarta. Como en el caso del token, el delimitador de fin completa la trama de datos/comandos.

#### Sistema de prioridad

Las redes Token Ring usan un sistema de prioridad sofisticado que permite que determinadas estaciones de alta prioridad usen la red con mayor frecuencia. Las tramas Token Ring tienen dos campos que controlan la prioridad: **el campo de prioridad** y el **campo de reserva**.

Sólo las estaciones cuya prioridad es igual o superior al valor de prioridad que posee el token pueden tomar ese token. Una vez que se ha tomado el token y éste se ha convertido en una trama de información, sólo las estaciones cuyo valor de prioridad es superior al de la estación transmisora pueden reservar el token para el siguiente paso en la red. El siguiente token generado incluye la mayor prioridad de la estación que realiza la reserva. Las estaciones que elevan el nivel de prioridad de un token deben restablecer la prioridad anterior una vez que se ha completado la transmisión.

#### Mecanismos de control

Las redes Token Ring usan varios mecanismos para detectar y compensar los fallos de la red. Uno de estos mecanismos consiste en seleccionar una estación de la red Token Ring como el monitor activo. Esta estación actúa como una fuente centralizada de información de temporización para otras estaciones del anillo y

ejecuta varias funciones de mantenimiento del anillo. Potencialmente cualquier estación de la red puede ser la estación de monitor activo.

Una de las funciones de esta estación es la de eliminar del anillo las tramas que circulan continuamente. Cuando un dispositivo transmisor falla, su trama puede seguir circulando en el anillo e impedir que otras estaciones transmitan sus propias tramas; esto puede bloquear la red. El monitor activo puede detectar estas tramas, eliminarlas del anillo y generar un nuevo token.

La topología en estrella de la red Token Ring de IBM también contribuye a la confiabilidad general de la red. Las **MSAU** (unidades de acceso de estación múltiple) activas pueden ver toda la información de una red Token Ring, lo que les permite verificar si existen problemas y, de ser necesario, eliminar estaciones del anillo de forma selectiva.

Otro mecanismo de control de fallos de red es el conocido como **Beaconing**. Cuando una estación detecta la existencia de un problema grave en la red (por ejemplo, un cable roto), envía una **trama de beacon**. La trama de beacon define un dominio de error. Un dominio de error incluye la estación que informa acerca del error, su vecino corriente arriba activo más cercano (NAUN) y todo lo que se encuentra entre ellos.

Entonces el beaconing inicia un proceso denominado **autoreconfiguración**, en el que los nodos situados dentro del dominio de error automáticamente ejecutan diagnósticos. Este es un intento de reconfigurar la red alrededor de las áreas en las que hay errores. Físicamente, las MSAU pueden lograrlo a través de la reconfiguración eléctrica.

#### *2.4.8 Redes LAN FDDI*

Las redes FDDI (Fiber Distributed Data Interface - Interfaz de Datos Distribuida por Fibra) surgieron a mediados de los años ochenta para dar soporte a las estaciones de trabajo de alta velocidad, que habían llevado las capacidades de las tecnologías Ethernet y Token Ring existentes hasta el límite de sus posibilidades.

Están implementadas mediante una física de estrella (lo más normal) y lógica de anillo doble de token, uno transmitiendo en el sentido de las agujas del reloj (anillo principal) y el otro en dirección contraria (anillo de respaldo o back up), que ofrece una velocidad de 100 Mbps sobre distancias de hasta 200 metros, soportando hasta 1000 estaciones conectadas. Su uso más normal es como una tecnología de backbone para conectar entre sí redes LAN de cobre o computadores de alta velocidad.

El tráfico de cada anillo viaja en direcciones opuestas. Físicamente, los anillos están compuestos por dos o más conexiones punto a punto entre estaciones adyacentes. Los dos anillos de la FDDI se conocen con el nombre de primario y secundario. El anillo primario se usa para la transmisión de datos, mientras que el anillo secundario se usa generalmente como respaldo.

Se distinguen en una red FDDI dos tipos de estaciones: las estaciones **Clase B, o estaciones de una conexión (SAS)**, se conectan a un anillo, mientras que las de **Clase A, o estaciones de doble conexión (DAS)**, se conectan a ambos anillos.

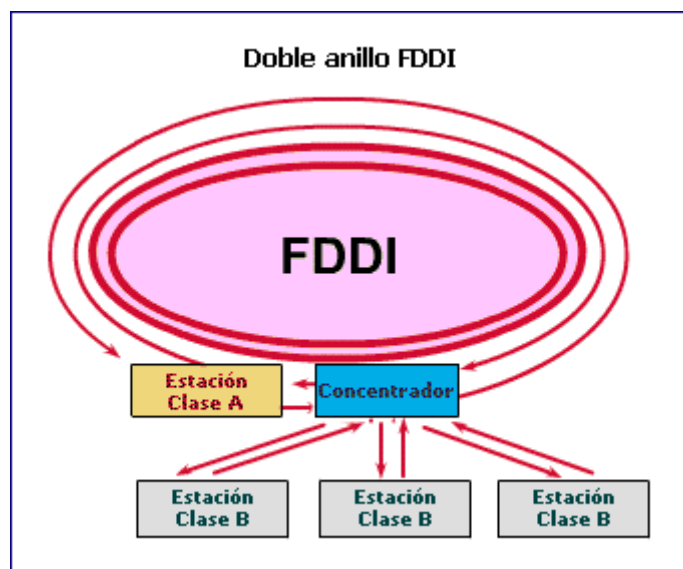


Figura 2.14 Doble Anillo FDDI .

Las SAS se conectan al anillo primario a través de un concentrador que suministra conexiones para varias SAS. El concentrador garantiza que si se produce una falla o interrupción en el suministro de alimentación en algún SAS determinado, el anillo

no se interrumpa. Esto es particularmente útil cuando se conectan al anillo PC o dispositivos similares que se encienden y se apagan con frecuencia.

Las redes FDDI utilizan un mecanismo de transmisión de tokens similar al de las redes Token Ring, pero, además, acepta la asignación en tiempo real del ancho de banda de la red, mediante la definición de dos tipos de tráfico:

**Tráfico Síncrono:** Puede consumir una porción del ancho de banda total de 100 Mbps de una red FDDI, mientras que el tráfico asíncrono puede consumir el resto.

**Tráfico Asíncrono:** Se asigna utilizando un esquema de prioridad de ocho niveles. A cada estación se asigna un nivel de prioridad asíncrono.

El ancho de banda síncrono se asigna a las estaciones que requieren una capacidad de transmisión continua. Esto resulta útil para transmitir información de voz y vídeo. El ancho de banda restante se utiliza para las transmisiones asíncronas

FDDI también permite diálogos extendidos, en los cuales las estaciones pueden usar temporalmente todo el ancho de banda asíncrono.

El mecanismo de prioridad de la FDDI puede bloquear las estaciones que no pueden usar el ancho de banda síncrono y que tienen una prioridad asíncrona demasiado baja.

En cuanto a la codificación, FDDI no usa el sistema de Manchester, sino que implementa un esquema de codificación denominado **esquema 4B/5B**, en el que se usan 5 bits para codificar 4. Por lo tanto, dieciséis combinaciones son datos, mientras que las otras son para control.

Debido a la longitud potencial del anillo, una estación puede generar una nueva trama inmediatamente después de transmitir otra, en vez de esperar su vuelta, por lo que puede darse el caso de que en el anillo haya varias tramas a la vez.

Las fuentes de señales de los transceptores de la FDDI son LED's (diodos electroluminiscentes) o láseres. Los primeros se suelen usar para tendidos entre máquinas, mientras que los segundos se usan para tendidos primarios de backbone.

Tramas FDDI

Las tramas en la tecnología FDDI poseen una estructura particular. Cada trama se compone de los siguientes campos:

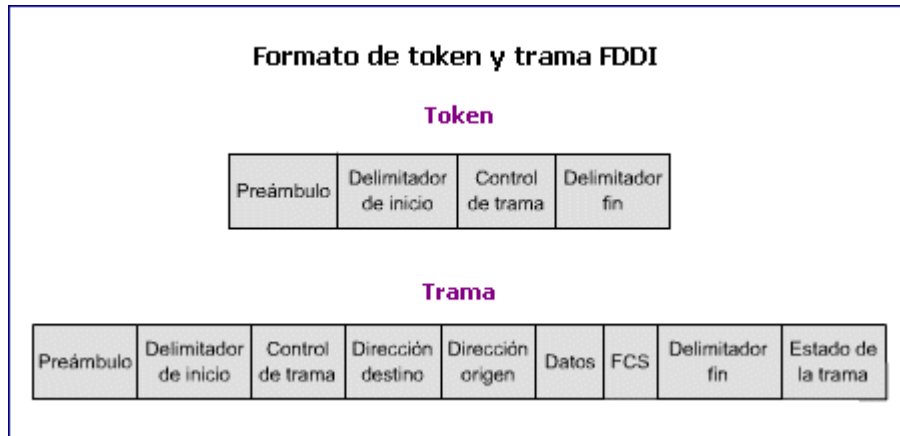


Figura 2.15 Formato de Token y Trama FDDI.

Preámbulo, que prepara cada estación para recibir la trama entrante.

Delimitador de inicio, que indica el comienzo de una trama, y está formado por patrones de señalización que lo distinguen del resto de la trama.

Control de trama, que contiene el tamaño de los campos de dirección, si la trama contiene datos asíncronos o síncronos y otra información de control.

Dirección destino, que contiene la dirección física (6 bytes) de la máquina destino, pudiendo ser una dirección unicast (singular), multicast (grupal) o broadcast (cada estación).

Dirección origen, que contiene la dirección física (6 bytes) de la máquina que envió la trama.

Secuencia de verificación de trama (FCS), campo que completa la estación origen con una verificación por redundancia cíclica calculada (CRC), cuyo valor depende del contenido de la trama. La estación destino vuelve a calcular el valor para determinar si la trama se ha dañado durante el tránsito. La trama se descarta si está dañada.

Delimitador de fin, que contiene símbolos que indican el fin de la trama.

Estado de la trama, que permite que la estación origen determine si se ha producido un error y si la estación receptora reconoció y copió la trama.

Medios en las redes FDDI



FDDI especifica una LAN de dos anillos de 100 Mbps con transmisión de tokens, que usa un medio de transmisión de fibra óptica.

Aunque funciona a velocidades más altas, FDDI es similar a Token Ring. Ambas configuraciones de red comparten ciertas características, tales como su topología (anillo) y su método de acceso al medio (transferencia de tokens).

Una de las características de FDDI es el uso de la fibra óptica como medio de transmisión. La fibra óptica ofrece varias ventajas con respecto al cableado de cobre tradicional, por ejemplo:

Seguridad: la fibra no emite señales eléctricas que se pueden interceptar.

Confiabilidad: la fibra es inmune a la interferencia eléctrica.

Velocidad: la fibra óptica tiene un potencial de rendimiento mucho mayor que el del cable de cobre.

Existen dos clases de fibra: monomodo (también denominado modo único); y multimodo. La fibra monomodo permite que sólo un modo de luz se propague a través de ella, mientras que la fibra multimodo permite la propagación de múltiples modos de luz. Los **modos** se pueden representar como haces de rayos luminosos que entran a la fibra en un ángulo determinado.

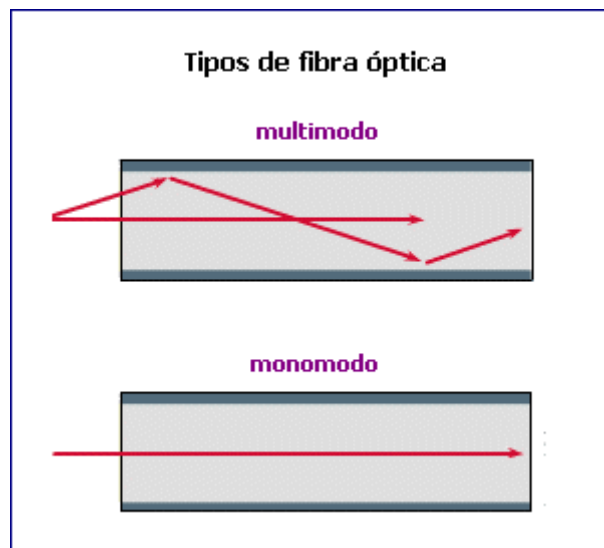


Figura 2.16 Tipos de Fibra óptica.

Cuando se propagan múltiples modos de luz a través de la fibra, éstos pueden recorrer diferentes distancias, según su ángulo de entrada. Como resultado, no

llegan a su destino simultáneamente; a este fenómeno se le denomina **dispersión modal**.

La fibra monomodo puede acomodar un mayor ancho de banda y permite el tendido de cables de mayor longitud que la fibra multimodo. Debido a estas características, la fibra monomodo se usa a menudo para la conectividad entre edificios mientras que la fibra multimodo se usa con mayor frecuencia para la conectividad dentro de un edificio. La fibra multimodo usa los LED como dispositivos generadores de luz, mientras que la fibra monomodo generalmente usa láser.

### 3.1 Manual de Usuario de la Aplicación SCADA en *Cirnet*.

El Sistema para el monitoreo de Subestaciones Eléctricas estará encargado de realizar un control supervisorio y almacenamiento de datos de estos sistemas. Esta aplicación también será capaz de capturar información en tiempo real para su representación numérica y gráfica, de la siguiente naturaleza:

- Gráficos lineales.
- Gráficos de barra.
- Medidores analógicos.
- Barras de estado (gráficos porcentuales).

De esta manera poder tener un control efectivo sobre el estado del sistema que se esta supervisando. Este control supervisorio se realizará a través de diversos formularios o pantallas en las cuales el usuario podrá buscar información del proceso, estado de las alarmas, etc. Estas y otras acciones complementarias permitirán obtener conclusiones con respecto a la situación del mismo.

#### 3.1.1 Acceso a la aplicación supervisoría.

Cuando se inicia la aplicación se presenta una pantalla, en la cual se tendrá que introducir una contraseña, de esta manera tendrán acceso solo el personal responsable del monitoreo, siendo esta una medida de seguridad para el sistema. Al introducirse la contraseña de forma incorrecta se presentara un mensaje de error, la cual le pedirá al usuario que reintente introducir la contraseña correcta.



Figura 3.1 Esquema de pantalla del Password.

Después de introducir correctamente la contraseña se tendrá acceso a la pantalla de modificación de los rangos de alarmas. Antes de iniciar esta pantalla se presenta una advertencia, ya que las manipulaciones inadecuadas pueden provocar el malfuncionamiento de las alarmas, dejando fuera de rango las variables críticas que el sistema presenta.

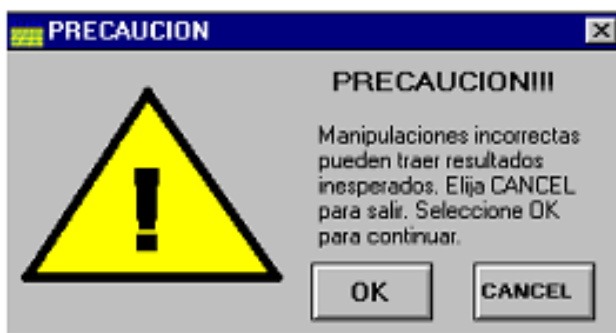


Figura 3.2 Advertencia ante cambios en los rangos de alarma.

En esta pantalla se presentan todas las variables que poseen un comando de alarma asignado. Esta pantalla solo debe manipularse si se quieren registrar cambios en los valores críticos para cada variable indicada. Si se desconoce su uso o los valores entre los cuales las variables pueden oscilar según los cambios a realizar, lo mejor es evacuar esta etapa, ya que al provocar cambios inadecuados se podría fuera de rango la variable y cuando esta realmente este en un punto crítico no se podrá percibir.

DETERMINE LOS RANGOS PARA LAS ALARMAS:		
TENSION:	INTENSIDAD:	FAC. POT.:
VMIN: <input type="text"/>	IMAX: <input type="text"/>	F.P. : <input type="text"/>
VMAX: <input type="text"/>	FAC. POT. III: <input type="text"/>	FP III: <input type="text"/>

Figura 3.3 Determinación de nuevos rangos de alarma.

Al dejar la pantalla de nueva selección de rangos de alarma, se presenta la pantalla de inicio del sistema, la cual presenta todos los valores trifásicos del sistema, para ambos nodos, según se seleccione. En esta pantalla también se ha

dispuesto el acceso a las diferentes funciones mediante una botonera, estas se presentan a continuación:

- Función de Secciones.
- Función Unifilar.
- Función de Selección.
- Función de Ayuda.
- Función de Facturación.
- Función de Data Logger.
- Función de Energía.

Esta pantalla principal servirá de base o plataforma a otras pantallas que serán cargadas según la necesidad del momento o la acción a realizar. Esta pantalla contiene la hora y fecha del sistema como se muestra a continuación:



Figura 3.4 Esquema de Pantalla Principal.

### 3.1.2 Función de Secciones.

En esta función se presentara un menú de selección de las diferentes subestaciones eléctricas (nodos supervisados) que compone el sistema a monitorear. Cada una de estas subestaciones se subdividirá por fases, de esta manera se tendrá un mejor manejo de los datos.



Figura 3.5 Menú por Fases según Nodo.

Al seleccionar el botón de FASE1 de Subestación1, se tendrá acceso a una pantalla que mostrara los datos tanto en procesos gráficos (lineales y porcentuales) como en formato numérico. De esta manera al seleccionar otro botón de la pantalla del Menú de Selección se presentara los datos de dicha fase correspondiente con el mismo formato de la pantalla siguiente:

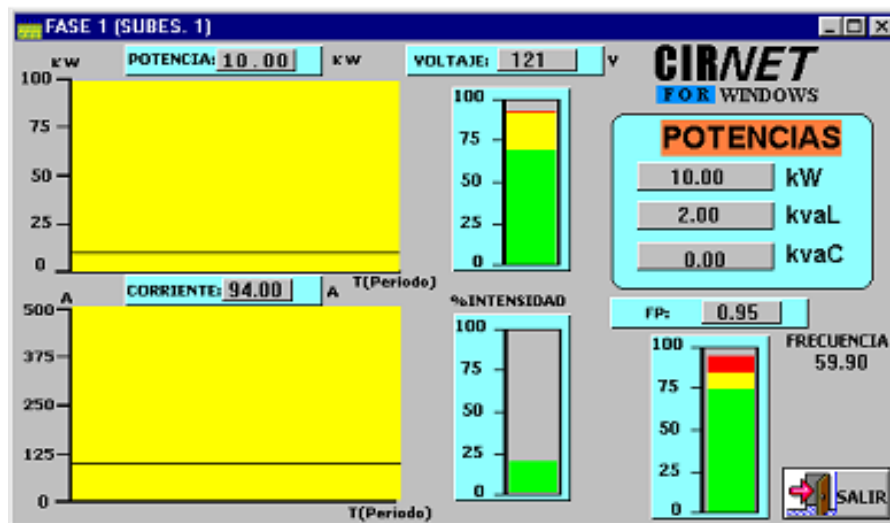


Figura 3.6 Esquema de Pantalla de una Fase

### 3.1.3 Pantalla de Ayuda.

Esta pantalla de ayuda podrá interactuar el usuario, ya que posee la función de entrada por teclado, en la cual podrá seleccionarse la pantalla de ayuda que se

deseo (1-6 pantallas), otra forma de selección de estas pantallas será mediante los botones de avance hacia delante o atrás.

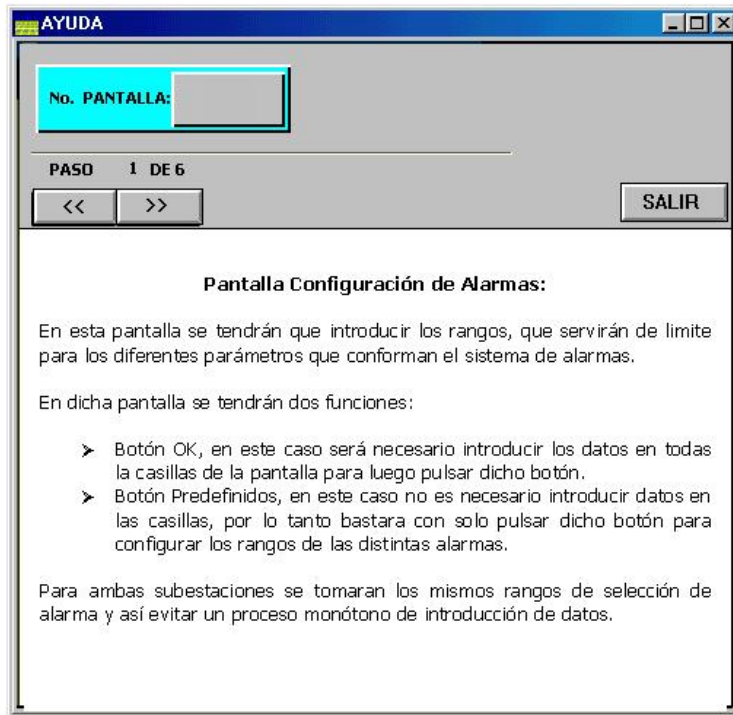


Figura 3.7 Esquema de la Pantalla de Ayuda.

### 3.1.4 Pantalla de Alarma.

En este formulario se tendrá que escribir el nombre del archivo de alarmas que se desea verificar, teniendo también opción de selección del archivo mediante el desplazamiento de barra. El archivo de alarma mostrará los diferentes procesos de activación de las todas las alarmas que fueron detectadas por el sistema.

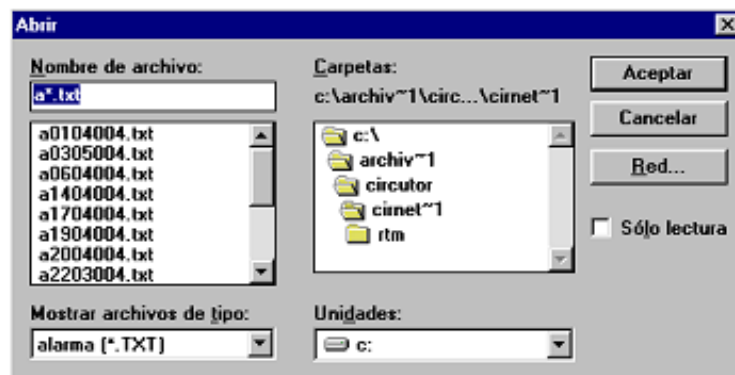


Figura 3.8 Abriendo un Archivo de Ayuda.

Si se selecciona a0305004.txt, se mostrara una pantalla con formato de texto del proceso ocurrido según el mensaje de alarma, en la cual se indicara el día de ocurrencia y su respectiva hora, junto con un texto corto indicando que parámetro se ha visto afectado por el sistema. La letra con que comienza el archivo "a" indica que es un archivo de alarma los siguientes números 03 indicara el día en que se ha guardado dicho formato, el 05 indicara el mes en que se han producido y los siguientes números indican el año. Es decir, que cada día se irán generando un archivo del proceso de alarmas, la cual se podrá tener acceso mediante esta ventana.

```

C:\ARCHIVO\CIRCUITOP\CIRNET\1\A0305004.TXT
0 ACT 03/05/2004 09/49/58 BAJA POTENCIA
0 ACT 03/05/2004 09/49/58 BAJO FACTOR DE POTENCIA 1
0 ACT 03/05/2004 09/49/58 BAJO FACTOR DE POTENCIA 2
0 ACT 03/05/2004 09/49/58 BAJO FACTOR DE POTENCIA 3
1 ACT 03/05/2004 09/49/58 VOLTAJE1
1 ACT 03/05/2004 09/49/58 VOLTAJE2
1 ACT 03/05/2004 09/49/58 VOLTAJE3
0 REC 03/05/2004 09/50/15 BAJA POTENCIA
0 REC 03/05/2004 09/50/15 BAJO FACTOR DE POTENCIA 1
0 REC 03/05/2004 09/50/15 BAJO FACTOR DE POTENCIA 2
0 REC 03/05/2004 09/50/15 BAJO FACTOR DE POTENCIA 3
1 REC 03/05/2004 09/50/15 VOLTAJE1
1 REC 03/05/2004 09/50/15 VOLTAJE2
1 REC 03/05/2004 09/50/15 VOLTAJE3
0 ACT 03/05/2004 10/20/50 BAJA POTENCIA
0 ACT 03/05/2004 10/20/50 BAJO FACTOR DE POTENCIA 1
0 ACT 03/05/2004 10/20/50 BAJO FACTOR DE POTENCIA 2
0 ACT 03/05/2004 10/20/50 BAJO FACTOR DE POTENCIA 3
1 ACT 03/05/2004 10/20/50 VOLTAJE1
  
```

Figura 3.9 Estado de las Alarmas.

Esta opción es el equivalente al registro histórico de mensajes de alarma que se presenta en el sistema según los parámetros especificados. En este aparecen: estado de la alarma (actuando o reconocida), fecha, hora y la variable en dicho estado.

### 3.1.5 Función de Energía. Gráficos de barra y Tablas

Mediante este botón se accede a las funciones de tratamiento de energía de la base de datos. Este modo permite hacer análisis de consumo energéticos diarios, semanales o mensuales, por equipo o por servicios. La presentación se puede hacer tanto en diagrama de barras como en tablas numéricas, que en ambos casos pueden ser impresas. Mediante este análisis resulta de suma facilidad verificar la distribución de consumo de energía eléctrica en cualquier periodo de tiempo; a



partir de aquí se pueden extraer conclusiones para costes de secciones, distribución de cargas entre tarifas, posibles fraudes, etc. En definitiva suministra una amplia información para llevar a cabo cualquier acción que tenga como objetivo un ahorro energético o simplemente una cuantificación exacta de la energía consumida.

Mediante la siguiente pantalla se puede configurar el periodo de tiempo que se incluirá en la visualización de los datos. Es importante mencionar que cuando se selecciona un aparato es diferente al funcionamiento cuando se visualiza un servicio. Siendo los servicios agrupaciones de distintos equipos en los que se obtiene una energía conjunta fruto de sumar o restar la de cada uno de ellos. Ya que también en esta pantalla se puede seleccionar tanto los aparatos que conforman la red, como los servicios. La primera pantalla que aparece es la que posee la configuración de energía para todos los módulos conectados a la red.

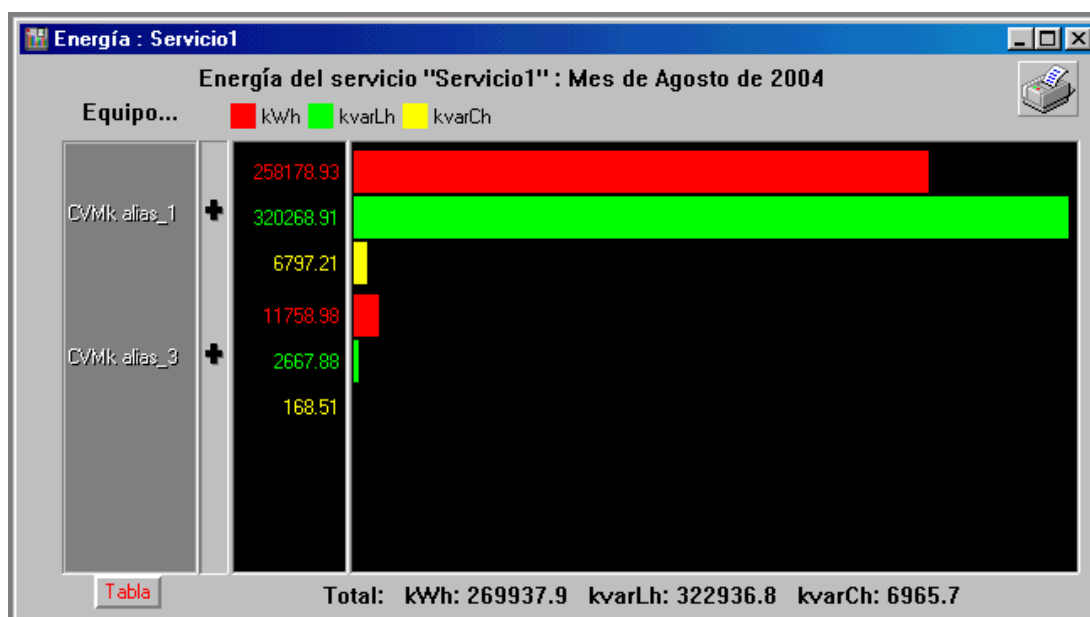


Figura 3.10 Datos de Energía en forma de Grafico de Barras.

En la parte izquierda del grafico se indica el equipo del cual se han tomado los registros de datos, correspondiente al mes en que han sido tomadas las mediciones. Estos gráficos pueden ser considerados como datos procesados, ya que el sistema presenta de forma visual una comparación del consumo energético de cada nodo del SCADA. Por lo tanto el grafico de barras posee información

recopilada y agrupada de forma secuencial y lógica. En caso de necesitar imprimir dichos gráficos, la pantalla les permite a través de un icono de impresora realizar tal acción.

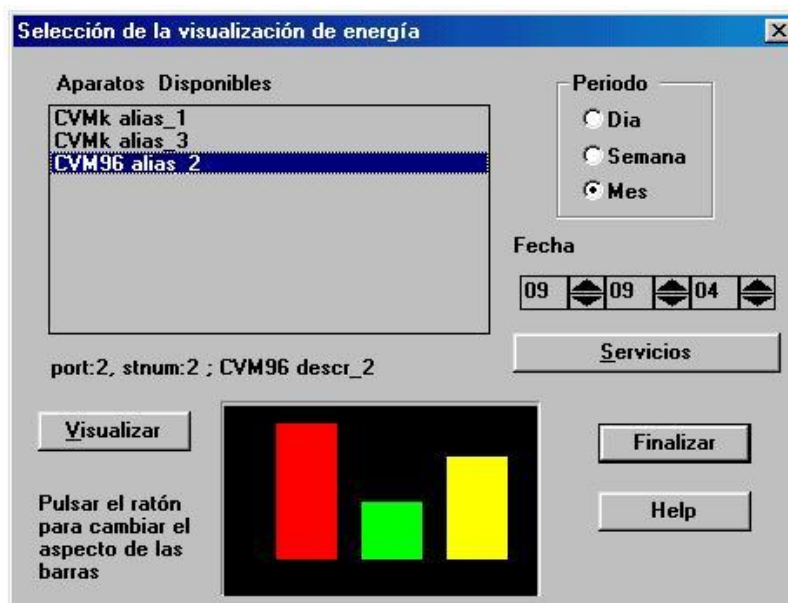


Figura 3.11 Configuración para la visualización de Datos de Energía.

En dicha pantalla se tiene opción a un apartado de fecha, mediante este dialogo se selecciona la fecha de los datos de energía que se van a visualizar. Si la visualización es diaria se tiene en cuenta la fecha completa; cuando se visualiza una semana se toma aquella que incluye el día seleccionado; finalmente cuando selecciona el periodo mensual el programa no tiene en cuenta el día y toma como dato el mes seleccionado.

Una vez realizada estas configuraciones se tendrá que seleccionar el botón de "Visualizar", el cual permitirá mostrar una ventana con el grafico de energía, pero la ventana de selección queda en primer plano pudiendo efectuar nuevas selecciones. Cuando se pulsa el botón "Finalizar" desaparece la ventana de Selección de la visualización de energía, quedando solamente la ventana del grafico de energía.

La visualización de la energía registrada por el aparato CVMk presenta el siguiente aspecto:

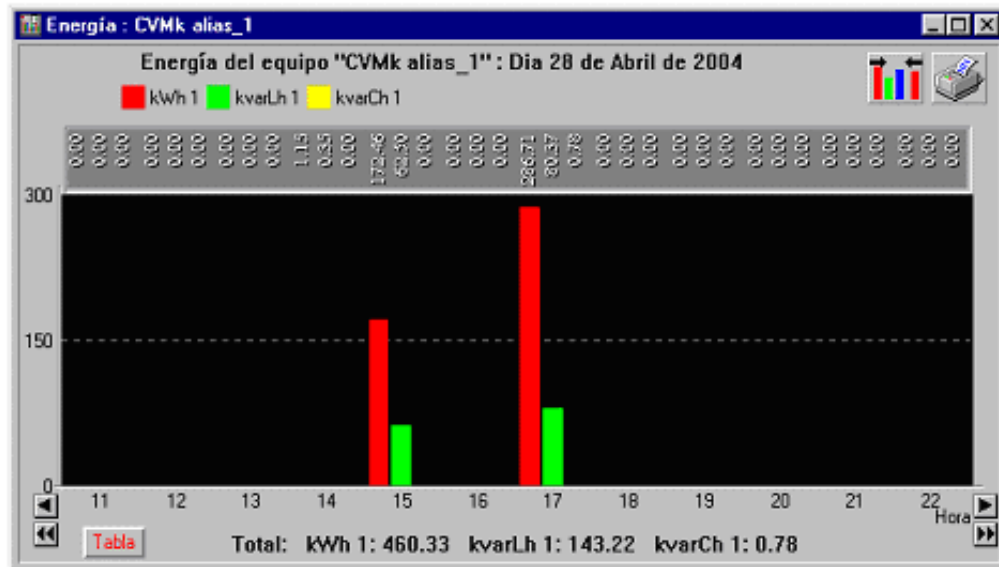


Figura 3.12 Datos de Energía en forma de Grafico de Barras.

En la parte inferior del grafico se indica la hora, día de la semana ó día del mes a que hacen referencia las barras que sobre el se representan.

Sobre cada una de las barras se visualiza el valor numérico de la energía para ese periodo, en el caso mostrado se visualiza cada hora. En caso de ser una semana o un mes el mostrado, se verificaría un día en los intervalos de estos periodos.

En la parte inferior del grafico se muestra el total consumido, en este caso para un día, pero si se visualiza semana es el total para una semana, de la misma forma para el mes.

En la esquina inferior derecha e izquierda se muestran unos botones de desplazamiento, el superior es para desplazarse hora por hora (según el periodo seleccionado) y así visualizar el periodo siguiente, y el de abajo es para desplazarse hasta el final del grafico.

En la esquina superior derecha se tendrá acceso a impresión del grafico, como también el botón para comprimir el grafico, es decir que todas las barras del periodo de tiempo se podrán visualizar con el fin de efectuar una rápida visión de la evolución de las energías.

También posee un botón "Tabla", con el cual se conseguirá la visualización de todos los valores de la gráfica en formato numérico, es decir mediante la representación en una tabla, como se muestra a continuación:

Hora	kWh 1	kvarLh 1	kvarCh 1	Cos.Fi.Acum
12:00	0.00	0.00	0.00	
13:00	0.00	0.00	0.00	
14:00	1.15	0.35	0.00	0.96
15:00	172.46	62.50	0.00	0.94
16:00	0.00	0.00	0.00	
17:00	286.71	80.37	0.78	0.96
18:00	0.00	0.00	0.00	
19:00	0.00	0.00	0.00	
20:00	0.00	0.00	0.00	
21:00	0.00	0.00	0.00	
22:00	0.00	0.00	0.00	
23:00	0.00	0.00	0.00	

Total: kWh 1: 460.33 kvarLh 1: 143.22 kvarCh 1: 0.78

Figura 3.13 Datos de Energía en forma de Tablas.

En dicha pantalla se muestra la hora en la cual se realizó el censo de la Energía y del Factor de Potencia. El botón que antes era "Tabla" ahora cambia a "Gráf", indicando de esta manera que se puede regresar a la pantalla de Gráficos (pantalla anterior, gráfico de barras).

### 3.1.6 Función de Energía. Gráficos lineales y Tablas.

El botón LOGGER da acceso a las funciones de tratamiento de todos los parámetros de la base de datos por ejemplo: tensiones, intensidades, potencias, etc. Aunque es posible visualizar energías, la representación que se efectuara será del valor del contador al final de cada periodo, con lo cual se visualizará un gráfico monótono creciente. Mediante este botón se puede visualizar la evolución diaria de cualquiera de los parámetros que se han seleccionado para grabar en la base de datos (predefinido en la fase de diseño).

Al pulsar el botón de "LOGGER" se muestra el siguiente diálogo en el que se definen las variables que deseen mostrar y la configuración de la visualización.

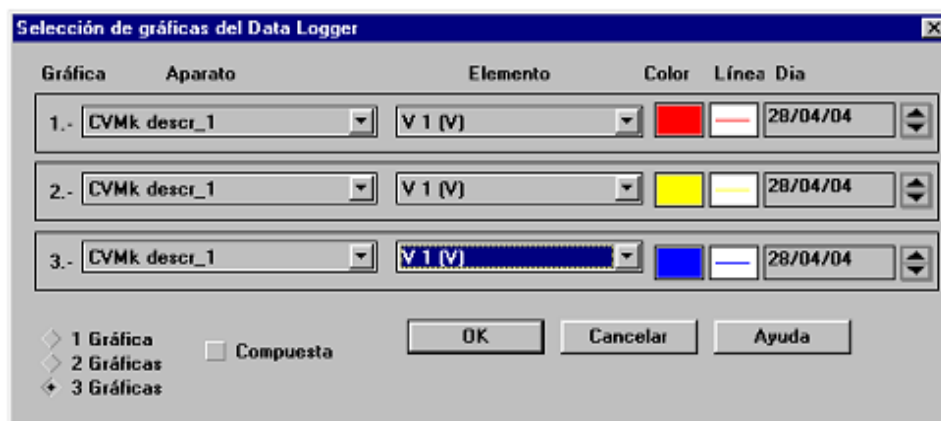


Figura 3.14 Configuración del Data Logger.

El programa permite visualizar simultáneamente un máximo de tres gráficos distintos por ventana. Es decir, se puede hacer selección mediante el menú desplegable de los diferentes equipos implementado en la red del cual se mostraran los datos. La selección de equipo de medida es independiente para cada uno de los tres gráficos a elegir.

Una vez seleccionado el equipo del cual se mostraran los datos, el siguiente paso es elegir la variable a mostrar. Esto se hace mediante un menú desplegable.

Posee también un área en el que se puede seleccionar el color con que se visualizara la evolución de la variable correspondiente. Pulsando sobre esta área aparece el menú estándar de Windows para cambiar el color, teniendo también la capacidad de cambiar el tipo de línea que ha de servir para verificar los valores. Al pulsar sobre esta área aparece un menú que nos permite seleccionar el tipo de línea y grosor con el que se representara el grafico de esta variable. En donde el grosor se puede introducir numéricamente (mediante una casilla).

Posee un campo para selección del día del cual se visualizara la evolución de la variable correspondiente. Mediante las flechas se puede ir aumentando y disminuyendo día a día la fecha, aunque también se puede editar.

Otro campo disponible es el de selectores de gráficas, en el cual se pueden seleccionar tanto una, dos o tres gráficas en la misma ventana. Junto a este campo se encuentra un selector determinado como Compuesta, en el cual al habilitarla se visualizaran las variables eléctricas seleccionadas en el mismo grafico,

es decir, utilizando el mismo eje de tiempos "X" y de valores "Y". Esta posibilidad resulta de gran utilidad para comparar la evolución de distintos parámetros.

Una vez terminada la configuración, se tendrá que pulsar el botón de "Aceptar" en la cual aparecerá una ventana de espera indicando que se están procesando los datos y tras esto aparece el grafico y el Descriptor de LOGGER: Es una ventana en la que se incluyen una serie de botones mediante los cuales se pueden efectuar modificaciones en la visualización de valores. A continuación se muestra un grafico en el que se representan las tres tensiones:



Figura 3.15 Grafico Lineal de Datos de Tensión.

A continuación se definirán los diferentes botones del Descriptor del LOGGER, de izquierda a derecha:

- El primer botón permite mostrar un día posterior al que se esta visualizando en ese momento.
- El segundo botón, permite pasar a mostrar un día anterior al que se esta visualizando en ese momento.
- Botón "MENU", hace aparecer el Menú de selección del LOGGER.
- Botón de Impresión, permite obtener una copia impresa de lo que se esta visualizando en ese instante en el LOGGER. Al seleccionar esta opción

aparecerá el cuadro de dialogo de la impresora con las distintas opciones que proporciona Windows.

- Botón de "TABLA", con este botón se pueden visualizar los datos en formato numérico, es decir su representación será en tablas numéricas, en ese momento el mismo botón cambia de aspecto en forma de un grafico, lo cual indica que permitirá pasar a la visualización gráfica anterior. El aspecto que presentan estas tablas se muestran a continuación:

V 1 (V) 29/04/04 CMk aias_1											
Hora	V1(V)	Hora	V1(V)	Hora	V1(V)	Hora	V1(V)	Hora	V1(V)	Hora	V1(V)
◀		10:49	122.0	10:53	122.0	10:57	122.0	11:01	122.0	11:05	122.0
10:46	122.0	10:50	122.0	10:54	122.0	10:58	122.0	11:02	122.0	11:06	122.0
10:47	122.0	10:51	122.0	10:55	122.0	10:59	122.0	11:03	122.0	11:07	122.0
10:48	122.0	10:52	122.0	10:56	122.0	11:00	122.0	11:04	122.0		

V 2 (V) 29/04/04 CMk aias_1											
Hora	V2(V)	Hora	V2(V)	Hora	V2(V)	Hora	V2(V)	Hora	V2(V)	Hora	V2(V)
00:00	....	00:04	....	00:08	....	00:12	....	00:16	....	00:20	....
00:01	....	00:05	....	00:09	....	00:13	....	00:17	....	00:21	....
00:02	....	00:06	....	00:10	....	00:14	....	00:18	....	00:22	....
00:03	....	00:07	....	00:11	....	00:15	....	00:19	....		

V 3 (V) 29/04/04 CMk aias_1											
Hora	V3(V)	Hora	V3(V)	Hora	V3(V)	Hora	V3(V)	Hora	V3(V)	Hora	V3(V)
◀		10:24	122.0	10:28	123.0	10:32	122.0	10:36	122.0	10:40	123.0
10:21	123.0	10:25	123.0	10:29	122.0	10:33	122.0	10:37	....	10:41	....
10:22	123.0	10:26	122.0	10:30	122.0	10:34	123.0	10:38	122.0	10:42	....
10:23	123.0	10:27	122.0	10:31	122.0	10:35	123.0	10:39	....		

Figura 3.16 Datos de Tensión en forma de Tablas.

- Botón "ZOOM", permite realizar un acercamiento en el eje Y de los gráficos que se están visualizando.
- El ultimo botón despliega una ventana con la descripción de cada uno de los gráficos: color, equipo, variable, unidades y la fecha.

Otro gráfico adicional que puede mostrarse, es el que presenta los datos históricos en formato lineal y sobrepuestos, de los diferentes equipos. En este pueden observarse los gráficos de voltaje o tensión para la fase 1 de ambas subestaciones (el color refiere al nodo del sistema, según se ha diseccionado: nodo 1 = sub 1 = rojo; nodo 2 = sub 2 = amarillo) en una fecha y horas determinados. Presenta también datos analizados a partir del establecimiento de máximos y mínimos, así como la obtención de la variable media o promedio. Esto se muestra a continuación en el gráfico:

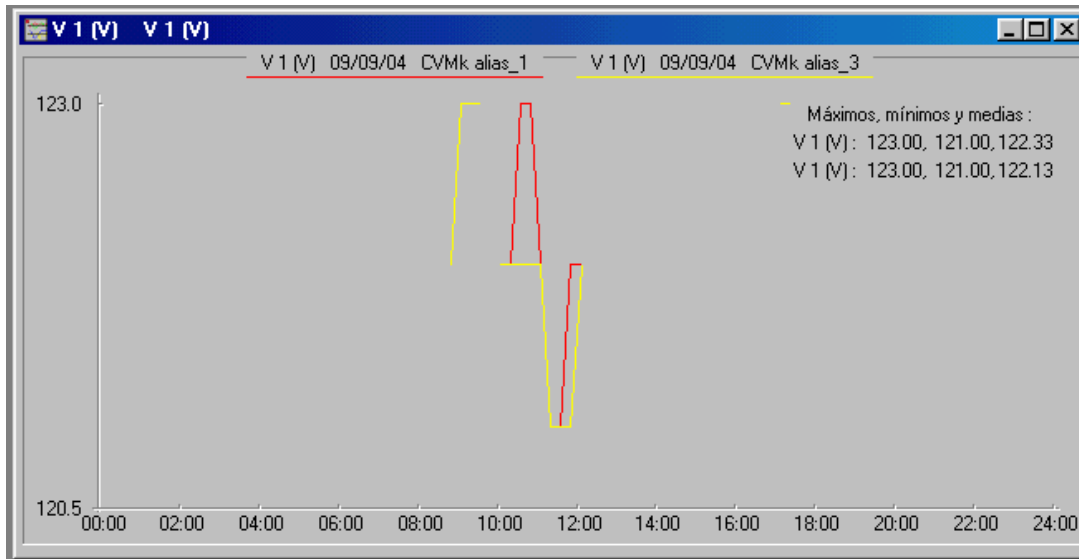


Figura 3.17 Datos de Tensión en forma grafico lineal.

### 3.1.7 Ejemplo de Pantalla de Control.

Mediante esta opción, se mostrará un cuadro de dialogo de precaución, ya que al seguir con este procedimiento se trata de prevenir al usuario que podría realizar un procedimiento inadecuado en caso de que el sistema (subestación) pudiera estar controlada, ya sea para conectarla o desconectarla de la red de distribución eléctrica. Esto obviamente esta directamente relacionado con la operación de la Subestación Eléctrica de Potencia y debe ser una decisión justificada. Para el ejemplo solo se presenta una simulación de dicho control. Como se ha propuesto en el inicio esto es trabajo de posteriores proyectos y no de presente.

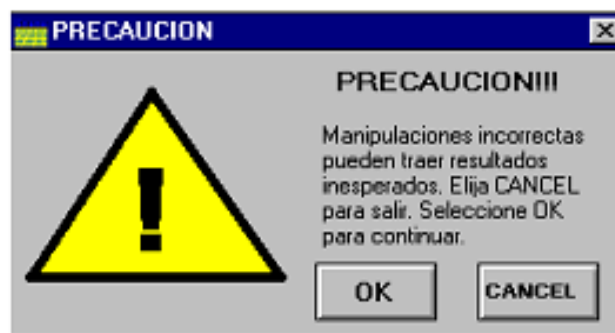


Figura 3.18 Esquema de Llamado de Atención

Al presionar el botón de OK se presentara una pantalla con el titulo de "Tabla de Interruptores", mediante la cual se presenta una visualización animada de



elementos como: líneas, interruptores, etc. Transformando de esta manera al sistema supervisorio en un versátil cuadro sinóptico de una instalación eléctrica. Ya que al presionar el interruptor general se mostrara un cambio de color de la Línea Primaria (De color rojo se transformara a color azul). Esta acción solo se presenta como muestra del sistema, ya que cada uno de los interruptores pueden enviar una señal a las salidas analógicas del equipo de control y medición, ya sea de forma trifásica o monofásica. Lo anterior solo puede lograrse con tarjetas especiales o módulos de expansión y periféricos especiales que poseen este tipo de salidas analógicas para el control de relevadores. Con los equipos que se ha trabajado en el desarrollo de esta aplicación, no se tiene la posibilidad de realizar tales acciones. Si se posee una infraestructura de control como la descrita anteriormente se podría controlar al sistema al grado de que los interruptores mostrados operarían directamente sobre el sistema, ya sea de forma completa (trifásica) o por fase, desactivando al sistema o a una de sus fases ante fallas inminentes, deshabilitar carga por sobrecarga de la misma. Este control físico no se llevara a cabo por consideraciones hechas en el Anteproyecto pero con esta aplicación se dejan las bases para su futura implementación, teniendo de esta manera el control total del sistema no solo con relación al monitoreo de datos gráfica y numéricamente, sino que también en cuanto a conexión y desconexión de equipos físicamente.

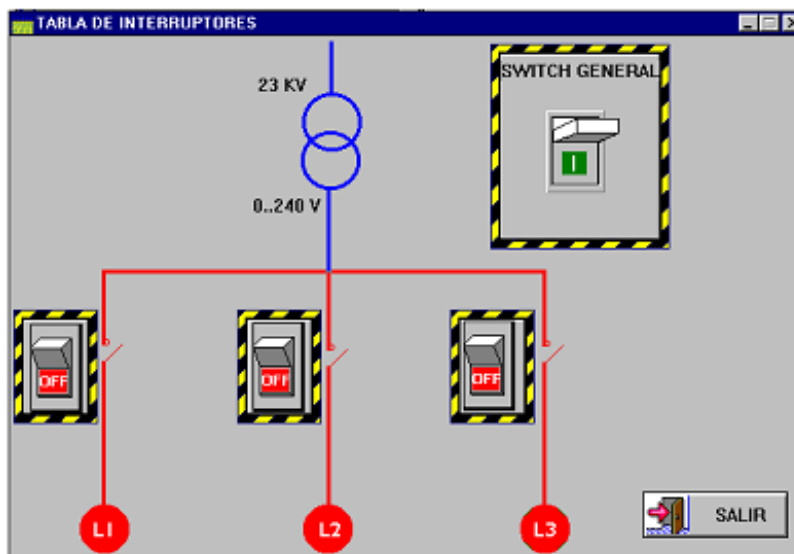


Figura 3.19 Ejemplo de Pantalla de Control.

Al activar el interruptor general se podrá activar cualquiera de las tres Fases. Si se presiona sobre el interruptor de Fase 1 (se presentara un cambio de color en la línea) se realizara la alimentación de la línea a la carga, presentando así una pantalla con información de los datos correspondiente a dicha fase como se presento en el apartado de "Función de Selección", en la cual se escogió el botón de Fase1.

### *3.1.8 Diagrama Unifilar.*

Al escoger este botón se tendrá acceso a una pantalla determinada UNIFILAR, en la cual se presenta un diagrama unifilar por cada subestación que estaría presente en dicha Red. Es decir, en la Subestación 1 se tendrá acceso para cada fase correspondiente a dicha Subestación, de esta manera se tendrá un control más efectivo ya que se visualizaran los datos individuales por fase, siendo este un ambiente más fácil para la búsqueda de ciertos parámetros que determinaran el estado de la planta, siendo un factor muy importante para establecer conclusiones sobre el consumo energético, de variaciones de parámetros como: Frecuencia, Tensión, Intensidad, etc., en los cuales se realizara el constante monitoreo para tener control sobre el proceso, ya que al haber variaciones de parámetros o en alguno casos si se sobrepasan ciertos rangos de operaciones de ciertos parámetros se tendrá información sobre esa anomalía.

Esta pantalla posee también ambientación gráfica, ya que al seleccionar el botón de Fase1 de la Subestación1 (que se encuentra debajo de la correspondiente fase y figura del medidor CVMk) se cambiara el color de la línea y el botón tendrá un cambio del texto.

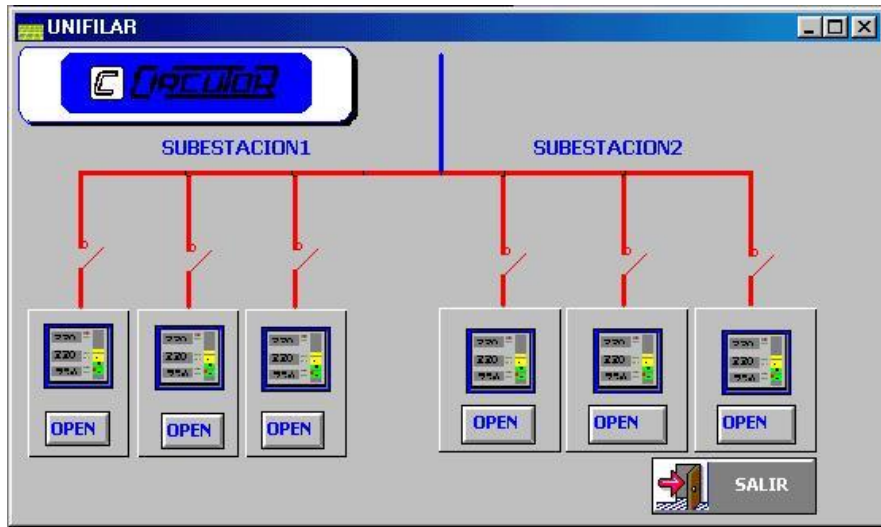


Figura 3.20 Diagrama Unifilar.

Al presionar dicho botón se presentara la siguiente pantalla, que corresponde a los datos visuales (en formato numérico) de los parámetros de la Subestación1 de Fase1:

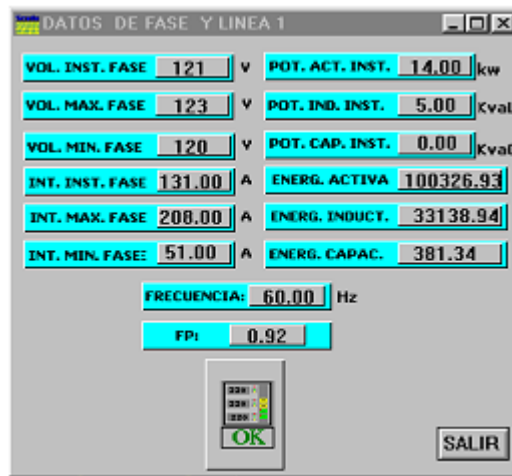


Figura 3.21 Acceso a Datos Instantáneos por el Diagrama Unifilar.

Al final de esta pantalla se presenta un botón con el dominio de un icono CVMk, el cual presentara dos funciones:

1. La primera función de este icono CVMk se podrá llevar a cabo manteniendo pulsado el botón derecho del ratón, el cual aparecerá la parte frontal del CVMk, como se muestra a continuación:



Figura 3.22 Esquema del Equipo de Medición respectivo (Tiempo Real).

Manteniendo pulsado el botón derecho del ratón y pulsando con el izquierdo sobre el icono de *Display* se pueden ir cambiando la pagina que se esta visualizando. La primera pagina siempre visualiza las tensiones simples de cada una de las fases, la segunda pagina visualiza las tres intensidades y la tercera, kW trifásicos, factor de potencia trifásico y frecuencia (siendo esta la que se presenta en la pantalla).

2. La segunda función de este icono CVMk se podrá llevar a cabo pulsado el botón izquierdo del ratón, en el cual se abre una ventana en la que se muestra tensión por fase (fase – neutro), intensidad por fases, potencia activa y reactiva por fase. Todos estos datos serán representados en modo analógico y numérico. También se muestran los contadores de energía y la máxima demanda:

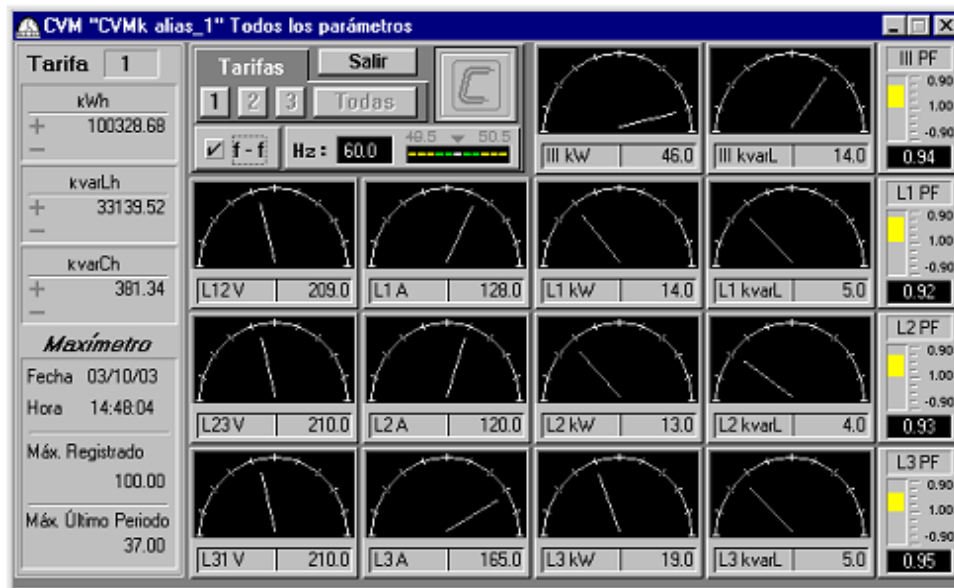
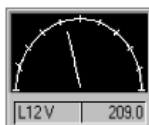


Figura 3.23 Mediciones en Tiempo Real presentados en forma Analógica.



Cada uno de los parámetros se muestran en modo analógico y en modo numérico. En los ángulos de la visualización analógica se muestra el valor inicial y final de la escala con el fin de poder rápidamente conocer el valor aproximado de una variable. En la zona inferior izquierdo se muestra el parámetro que se representa y las unidades.

En la sección de Tarifa 1, se muestran todos los contadores con el prefijo k (kilo).

En la sección de maxímetro, se muestra la fecha, hora y valor del máximo registrado y finalmente el maxímetro del ultimo periodo.

Si el equipo llevara insertado él modulo CVM/REDMAX, se podrían visualizar cada uno de los grupos de contadores por separado o de forma conjunta, de manera contraria solamente estará activado la Tarifa 1. Con dicha tarjeta se activarían los tres botones de la sección de TARIFA y así ir conmutando entre las distintas tarifas. Cuando se pulsa el botón "Todas" aparecerá una ventana en la que se muestran simultáneamente las tres tarifas. El módulo con que se cuenta en la en el nodo remoto (subestación 2) es el CVM/RED, el cual al igual que el CVM/ER-232 solo permite la verificación de datos de Tarifa 1, lo cual se refiere a la fase 1. Esto aplica también al nodo 1 (subestación 1)

### 3.1.9 Sistema de Alarmas.

La aplicación que se ha generado con el sistema SCADA llevara asociado consigo un sistema de alarmas. Dicha ventana se abrirá cuando se cumpla una condición de alarma, al producirse dicha situación, el subsistema de alarmas puede registrar un mensaje: visualizado en la pantalla e impresora. Como se muestra a continuación:

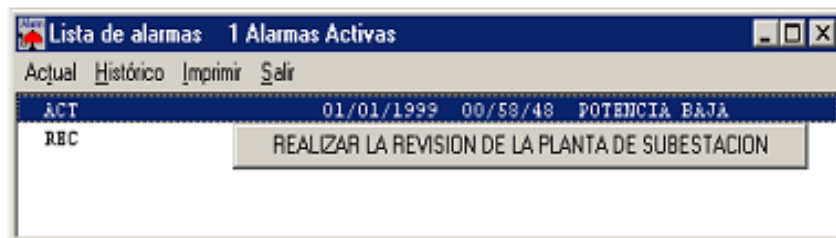


Figura 3.24 Sugerencias en Alarmas.

Dicha ventana tiene un texto determinado como ACK en el cual al posicionarse sobre ella y pulsando el botón izquierdo del ratón, se dará lugar al reconocimiento de las alarmas pendientes.

En esta ventana va apareciendo una línea de información por cada cambio de estado de las alarmas definidas.

Las tres letras de la izquierda, de la Lista de alarmas significan:

- **ACT:** Activación.
- **DES:** Desactivación.
- **REC:** Reconocimiento.

Junto a estas letras aparece la fecha y la hora correspondiente al momento en que se produjo el cambio de estado. El resto de la línea es el texto corto (breve descripción) de la alarma correspondiente.

Del menú de la Lista de alarma, se describirán las siguientes Opciones:

- **Opción Actual**, permite visualizar la lista de alarmas actuales en el caso de que estuviéramos visualizando alarmas de un histórico.
- **Opción Histórico**, permite abrir un fichero de alarmas que hayamos creado, esta opción es útil para revisar las activaciones y los reconocimientos de días pasados.
- **Imprimir**, permite imprimir la lista actual de alarmas, tanto si son históricos como actuales.
- **Salir**, permite salir de la lista de alarma siempre y cuando no haya ninguna alarma activada.

Si se estuviera visualizando un histórico en el menú, aparecería una opción mas "Por Grupo", esta opción nos permite ordenar las alarmas por grupos o por fechas. Pulsando encima de una alarma con el botón izquierdo del ratón, podemos visualizar el texto largo que hayamos configurado previamente (mencionado en la fase de diseño), el cual es una descripción mas completa de la alarma.

### *3.1.10 Tarifación.*

En esta pantalla se presenta la configuración que puede aplicarse para obtener un reporte de la energía consumida y su costo, ya sea por Potencia Contratada o por Facturación de Energía. Pueden seleccionarse ambas opciones o simplemente una de ellas, la que aplique con el sistema. La potencia contratada se refiere a cierta potencia que es contratada con una empresa distribuidora o generadora, según sea la cantidad de energía y un precontrato inicial, para un periodo determinado, esta se expresa en kilovatios (kW). La facturación de energía esta relacionada con la cantidad de energía que consume cierto cliente durante un determinado tiempo, y esto dependiendo de las horas de máxima y mínima demanda, esta facturación se expresa en kilovatio-hora (kWh). Al llenar todos los campos con la información correspondiente se podrá obtener un dato del costo de la potencia o energía consumida, según sea el caso. Los contadores son están relacionados con las fases que pueden facturarse. Estos equipos y sus módulos de comunicación (CVM/ER-232 y CVM/RED) solo pueden aplicar la tarifación a la fase 1, por esta razón solo esta habilitada la tarifa 1. Previamente se debe establecer el nodo al cual se le ha de aplicar la facturación. A continuación se presenta la pantalla de configuración de facturas:

Figura 3.25 Configuración de Tarifas.

### 3.1.11 Configuración de equipos en la red SCADA.


En la fase de diseño de la aplicación se deben seguir ciertas configuraciones para el buen funcionamiento del equipo. Al presionar sobre el botón de Puertos el sistema detecta automáticamente los puertos serie configurados previamente, y los dibuja, visualizando así los puertos libres a los cuales pueden conectarse los equipos. Pulsando sobre el interruptor  (Configuración de puertos), aparecerá una ventana con los drivers disponibles. Se pulsa sobre el elemento deseado, y se arrastra hasta el puerto al que lo tengamos conectado físicamente. Una vez escogido el aparato aparecerá el dialogo propio del driver. También dependiendo del mismo se podrán o no conectar mas de un equipo en el mismo puerto, esto depende de la red instalada: si es RS-232 solo puede haber un elemento; en caso contrario de ser RS-485 se pueden llegar a conectarse hasta 32 dispositivos CIRCUTOR en un mismo puerto.





Figura 3.26 Pantalla de Configuración de los Dispositivos conectados a la red.

A continuación se presenta la pantalla de dispositivos analizadores de red que pueden ser conectados y soportados por *Cirnet*. Debe notarse que aparece el CVM 96, pero el modelo trifásico. En esta aplicación se utiliza el monofásico, como modulo de prueba.

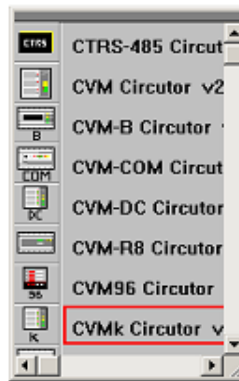


Figura 3.27 Pantalla de dispositivos que pueden conectarse en red.

Como se aprecia en la tabla de Drivers, se indican los nombre de los diferentes equipos y su respectiva versión. En esta pantalla se pulsa sobre el dispositivo deseado y se arrastra hasta el puerto que se desea instalar la red (Drag and Drop). Al pulsar sobre el icono del CVMk de la pantalla de Configuración de Drivers en la red aparece el siguiente dialogo:



Figura 3.28 Pantalla de parámetros de configuración de la red.

En esta ventana se pueden programar todos los parámetros de comunicación del equipo como son:

- La velocidad de transmisión del equipo.
- El tipo de paridad, cuando se selecciona una paridad, par o impar, la longitud de los paquetes asíncronos aumenta en un bit.
- Stop Bit, son los bits de parada, valores posibles 1 ó 2.
- Data Bit, es la longitud del dato de transmisión. Puede ser 7 u 8 bits. En CIRBUS (Protocolo propio de CIRCUTOR) solo son necesarios 7 bits, pero los equipos pueden trabajar a 8 bits para adaptarse a redes instaladas con protocolo de 8 bits. Cuando se selecciona MODBUS automáticamente se cambia a 8 bits. MODBUS es un modulo o protocolo de comunicación de plataforma abierta, lo cual indica que puede ser usado en otros programas de monitoreo que no tengan ninguna vinculo con *Cirnet*.

### 3.2 Composición de las Bases de Datos.

Las bases de datos que la aplicación SCADA posee están directamente relacionadas con las variables que mide directamente del sistema de potencia el cual esta supervisando. El equipo analizador de redes CIRCUTOR posee un numero de variables máximo que puede ser verificado, esto depende del tipo de tarjeta de red que tenga dicho equipo. El tipo de tarjeta de red con que se cuenta permite obtener los datos por medio de una interfaz RS-232, como ya se ha mencionado, y que va acoplada al puerto paralelo. Esta tarjeta de red permite supervisar un máximo de 29 variables. Otras tarjetas permiten aumentar el numero de variables verificadas (ejemplo: el modulo de expansión REDMAX permite un número de 43 variables). A continuación se presenta el total de las variables que pueden ser supervisadas con el equipo que se tiene, estas poseen un nombre característico y su código de trabajo en el sistema:

- Hora (Time)
- Fecha (Date)
- Tensión Simple Instantánea fase 1 (CV\_1)
- Tensión Compuesta Instantánea fases 12 (CV\_12)
- Intensidad Instantánea fase1 (CA\_1)
- Potencia activa por fase 1 (CKW\_1)
- Factor de Potencia fase 1 (CPF\_1)
- Tensión Simple Instantánea fase 2 (CV\_2)
- Tensión Compuesta Instantánea fases 23 (CV\_23)
- Intensidad Instantánea fase 2 (CA\_2)
- Potencia activa por fase 2 (CKW\_2)
- Factor de Potencia fase 2 (CPF\_2)
- Tensión Simple Instantánea fase 3 (CV\_3)
- Tensión Compuesta Instantánea fases 31 (CV\_31)
- Intensidad Instantánea fase 3 (CA\_3)
- Potencia activa por fase 3 (CKW\_3)
- Factor de Potencia fase 3 (CPF\_3)

- Tensión Promedio Trifásica (CVAV\_III)
- Tensión Compuesta Trifásica (CVC\_III)
- Intensidad Promedio Trifásica (CAAV\_III)
- Potencia Activa Trifásica (CKW\_III)
- Factor de Potencia Trifásico (CPF\_III)
- Potencia Aparente Trifásica (CKVA\_III)
- Frecuencia (CHZ)
- Energía Activa fase 1, tarifa 1 (CKWH\_1)
- Energía Reactiva Inductiva fase 1, tarifa 1 (CKVARLH\_1)
- Energía Reactiva Capacitiva fase 1, tarifa 1 (CKVARCH\_1)
- Demanda máxima fase1, Tarifa 1 (CMAX\_DEM\_1)
- Energía consumida fase1, Tarifa 1 (CKWH\_1)

Estos datos son presentados en formato de base de datos (el tipo de archivo tiene la forma siguiente: *nombre.dbf*), en una hoja electrónica, en Excel pero poseen la propiedad que es un formato de solo lectura. Estos archivos se van actualizando automáticamente con respecto a las mediciones que realizan en función del tiempo.

Para poder trabajar en con estos archivos en otras herramientas Microsoft, tal como se ha hecho en Access, se han importado los datos desde el formato de Excel y se puede realizar todo el manejo de la información según convenga. Esto ha permitido que puedan crearse ambientes de consulta más agradables y menos pesados para el usuario. Pero esto también implica que el usuario debe poseer conocimientos mínimos de MS Office para poder entender el uso de estas herramientas. En este sentido se ha tratado de crear una herramienta sencilla para realizar la consulta de toda la información almacenada, y así evitar que el ambiente difícil del sistema de supervisión sea un impedimento para su utilización.

### 3.3 Manual de Usuario de la Aplicación en Microsoft Access para el Sistema de Monitoreo Local para Subestaciones Eléctricas del CITT.

#### 3.3.1 Modo de ingreso al Sistema de Verificación de la Información Almacenada.

Para iniciar con la consulta de las bases de datos por medio de la herramienta realizada en Access, el sistema solicita una contraseña, la cual esta formada por cuatro dígitos numéricos (XXXX, similar a la que solicita el *Cirnet* antes de iniciar a operar). Aparece el siguiente cuadro de dialogo en la figura 1 en el cual se debe digitar la contraseña antes mencionada.

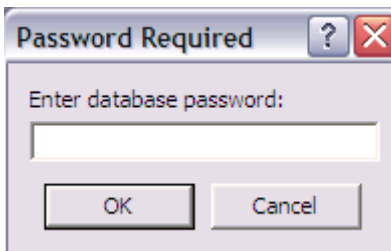


Figura 3.29. Contraseña de la aplicación en ACCESS

Al digitar correctamente la contraseña, el sistema permite elegir a continuación, desde su menú principal las opciones a las cuales se puede acceder.

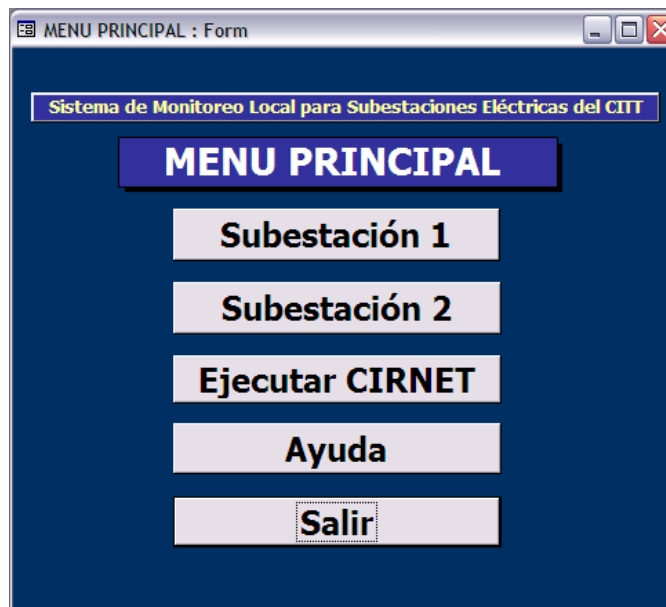


Figura 3.30 Menú Principal.

El menú principal muestra cuatro opciones, de las cuales las dos primeras (subestación 1 y 2) permiten verificar los diferentes parámetros eléctricos que se

encuentran agrupados en el submenú de cada subestación (voltajes de fase, de línea, potencias, energías, etc.). Estas dos primeras opciones están relacionadas con los dos nodos mínimos con que debía cumplir el sistema de supervisión. Para esta primera defensa solo se encuentran habilitados los formularios de enlace a las bases de datos de la subestación 1 (ubicada detrás del edificio de eléctrica, en el CITT), para realizar la verificación de los mismos. El submenú correspondiente a la subestación 2 ya se encuentra habilitado, como se planteó, y posee una perfecta comunicación con el software SCADA (esta se encuentra ubicada detrás del edificio de metrología, CITT). Los formularios que pertenecen a este submenú pueden ser consultados al igual que los de la subestación 1 (nodo 1).

Las opciones restantes del menú principal son: ejecutar la aplicación *Cirnet*, la cual permite acceder desde Access a la aplicación personalizada del sistema de monitoreo y adquisición de datos (SCADA). La cuarta opción se refiere a la ayuda para trabajar con la aplicación hecha en MS Access y *Cirnet*; ayuda que únicamente da la idea de cómo se conforman los formularios, que tipos de formularios posee la aplicación, etc. Se da por entendido que el usuario de la aplicación debe saber utilizar MS Access. La última opción u opción cinco es la de salir del sistema; esta opción permite cerrar la aplicación creada en Access y por consiguiente cualquier aplicación que estuviera siendo consultada al momento de ejecutar dicho comando. Pero cabe resaltar que se cancelan solo las aplicaciones relacionadas con Access, mientras que las aplicaciones relacionadas con *Cirnet* permanecen vigentes.

A continuación se mostrará la apariencia de los submenús correspondientes a cada subestación. La única diferencia entre ellos es el equipo o sistema de potencia al que están aplicados. Para el caso, el submenú relacionado con la subestación 1 es el siguiente:

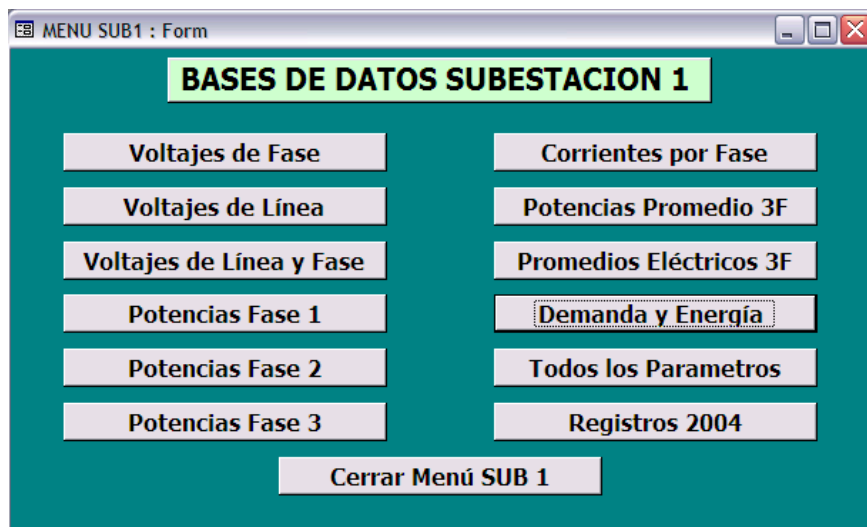


Figura 3.31 Menú de la Subestación 1 (Nodo 1).

De la misma forma, pero con la diferencia de que el Submenú 2 se refiere a la subestación 2 (ubicada a detrás del edificio que alberga los laboratorios de metrología)

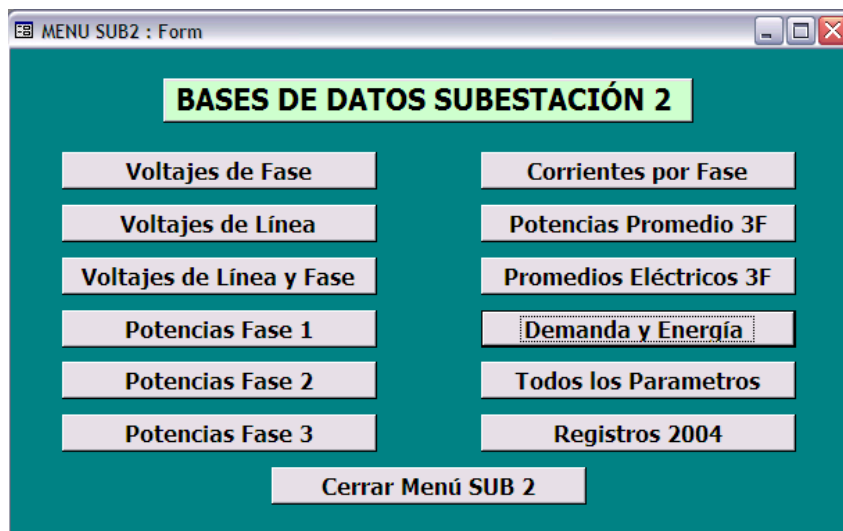


Figura 3.32 Menú de la Subestación 2 (Nodo 2).

### 3.3.2 Descripción de Formularios.

A continuación se realizará la descripción de los diferentes formularios que componen los submenús correspondientes a las subestaciones acopladas al sistema de monitoreo.

### **3.3.2.1 Voltajes de Fase.**

Cuando se selecciona este formulario en el menú correspondiente a los voltajes de fase, se presentan los diferentes valores de voltaje de fase correspondientes a la subestación 1. estos valores pueden ser consultados desde las diferentes bases de datos que han sido almacenadas en el sistema de supervisión. Es decir, actualmente se tienen almacenados datos de tres meses y cualesquiera de ellos se pueden consultar. La estructura de este formulario para voltajes de fase presenta en pantalla: fecha de la medición y su hora correspondiente, voltajes de cada una de las fases y la frecuencia de la red. Además posee el numero de datos totales que contiene la base de datos del mes correspondiente (mes que esta siendo verificado). Posee un botón de salida para poder evacuar el formulario. De forma comparativa pueden tenerse las mediciones para cada una de las fases para un momento determinado y un día en particular, esto permite que pueda tenerse una idea de cual era el comportamiento del sistema de potencia en dicho momento con respecto al voltaje.

FECHA DE MEDICION	HORA DE MEDICION
22/03/2004	09:52:00

VOLTAJES DE FASE	
Voltaje de Fase 1 (CV_1) [V]	124
Voltaje de Fase 2 (CV_2) [V]	124
Voltaje de Fase 3 (CV_3) [V]	124

FRECUENCIA DE RED	
FRECUENCIA (HZ):	60.1

Cerrar

Record: 1 of 302

Figura 3.33 Formulario de Voltajes de Fase.

### **3.3.2.2 Voltajes de Línea.**

Con la selección del botón correspondiente a los voltajes de línea se accede a un formulario que posee una estructura similar al de voltajes de fase. La diferencia está en que para este caso se verifican los voltajes de línea para la subestación 1,



también de forma comparativa con el resto de parámetros como en el caso anterior.

FECHA DE MEDICION	HORA DE MEDICION
22/03/2004	09:52:00

VOLTAJES DE LINEA	
Voltaje de Línea 12 (CV_12) [V]	213
Voltaje de Línea 23 (CV_23) [V]	213
Voltaje de Línea 31 (CV_31) [V]	213

FRECUENCIA DE RED	
FRECUENCIA (HZ):	60.1

Cerrar

Record: 1 of 302

Figura 3.34 Formulario de Voltajes de Línea.

### 3.3.2.3 Voltajes de Línea y Fase.

En este caso se resumen ambos voltajes de la subestación, es decir los voltajes de línea y fase, los cuales pueden ser analizados y comparados de forma compacta en este espacio. En este caso se puede acceder por medio del botón correspondiente en el submenú de la subestación 1.

FECHA DE MEDICION	HORA DE MEDICION
22/03/2004	09:52:00

VOLTAJES DE FASE		VOLTAJES DE LINEA	
Voltaje de Fase 1 (CV_1) [V]	124	Voltaje de Línea 12 (CV_12) [V]	213
Voltaje de Fase 2 (CV_2) [V]	124	Voltaje de Línea 23 (CV_23) [V]	213
Voltaje de Fase 3 (CV_3) [V]	124	Voltaje de Línea 31 (CV_31) [V]	213

FRECUENCIA DE RED	
FRECUENCIA (HZ):	60.1

Cerrar

Record: 1 of 302

Figura 3.35 Formulario de Voltajes de Línea y Fase.

### 3.3.2.4 Corriente Eléctrica por Fase.

Al tener acceso al formulario de corriente eléctrica por fase se pueden verificar, además de la fecha y hora de medición y la frecuencia del voltaje que le da origen a estas corrientes. De la misma forma que en los casos anteriores se puede tener acceso a las bases de datos que se han almacenado durante los periodos de medición. Debido a que la subestación 1 está conectada en estrella en el secundario, la corriente de fase es la misma que la de línea, por lo cual solo se tiene un formulario para la corriente. En el caso del voltaje se tiene uno para voltajes de línea y otro para los voltajes de fase. Puede apreciarse también en este formulario el número de mediciones realizadas durante el periodo en estudio y el botón para salir del formulario.

INTENSIDADES DE CORRIENTE POR FASE

**CORRIENTE ELECTRICA POR FASE SUB 1**

FECHA DE MEDICION: 22/03/2004      HORA DE MEDICION: 09:52:00

**INTENSIDADES DE CORRIENTE ELECTRICA**

Intensidad de Corriente Fase 1 (CA_1) [A]	149
Intensidad de Corriente Fase 2 (CA_2) [A]	131
Intensidad de Corriente Fase 3 (CA_3) [A]	171

**FRECUENCIA**

FRECUENCIA (HZ): 60.1

Cerrar

Record: 1 of 302

Figura 3.36 Formulario de Corrientes por Fase.

### 3.3.2.5 Potencias por Fase.

En este formulario se accede a la información relacionada con todas las potencias que describen a la fase 1: potencia activa, reactiva inductiva y capacitiva. Se presenta también el factor de potencia al cual estuvo trabajando el sistema en la fecha y horas señaladas al inicio del formulario. El resto de las fases del sistema también presentan un formulario específico para presentar sus datos, estos no se describirán por que en esencia son similares al de la fase 1, y pueden ser accedidos de la misma manera que en este caso: seleccionando el botón

respectivo en el submenú de la subestación que quiera verificarse, en ambas subestaciones (nodos 1 y 2).

FECHA DE MEDICION	HORA DE MEDICION
22/03/2004	09:52:00

POTENCIA ACTIVA Y REACTIVA	
Potencia Activa Fase 1 (CKW_1) [kW]	17
Potencia Reactiva Inductiva Fase 1 (CKVARL_1) [kVAR]	6
Potencia Reactiva Capacitiva Fase 1 (CKVARC_1) [kVAR]	0

FACTOR DE POTENCIA 1	
Factor de Potencia Fase 1 (CPF_1)	0.93

Cerrar

Record: 1 of 302

Figura 3.37 Formulario de Potencias en Fase 1.

### **3.3.2.6 Potencia Promedio Trifásica.**

Seleccionando este formulario se accede a las potencias promedios del sistema durante el mínimo periodo de tiempo en el cual se realizan las mediciones. Es decir, que si se almacena una base de datos mensualmente, y los periodos de medición son cada 15 minutos, el sistema obtendrá un promedio durante esos 15 minutos de las diferentes potencias medidas en ese lapso, y así sucesivamente durante cada periodo medición. Esto no es así con las demás mediciones, ya que solo se toma la medición instantánea al final de ese periodo. El programa sistema de supervisión solo obtiene el promedio en el caso de valores promedio de lo contrario obtiene una medición instantánea. El formulario presenta el respectivo factor de potencia promedio para el periodo de medición asignado.

POTENCIAS TOTAL SUBESTACION 1

**POTENCIAS PROMEDIO TRIFASICAS SUB 1**

FECHA DE MEDICION: 22/03/2004      HORA DE MEDICION: 09:52:00

**POTENCIA APARENTE, ACTIVA Y REACTIVA PROMEDIO TRIFASICA**

Potencia Aparente Promedio Trifásica (CKVA_III) [kVA]	55.5
Potencia Activa Promedio Trifásica (CKW_III) [kW]	51
Potencia Reactiva Inductiva Promedio Trifásica (CKVARL_III) [kVAR]	19
Potencia Reactiva Capacitiva Promedio Trifásica (CKVARC_III) [kVAR]	0

**FACTOR DE POTENCIA PROMEDIO TRIFASICO**

Factor de Potencia Promedio Trifásico (CPF_III)	0.92
---	------

Cerrar

Record: 1 of 302

Figura 3.38 Formulario de Potencias Trifásica.

### 3.3.2.7 Promedio de Parámetros Eléctricos.

Al acceder a este formulario se pueden notar, además de la fecha y hora de medición, los promedios trifásicos de los parámetros eléctricos de voltaje y corriente. Estas mediciones también han sido obtenidas conforme al promedio de mediciones en tiempo menor supervisado. Como se ha expuesto anteriormente, este promedio se obtiene sacando el promedio de los parámetros eléctricos instantáneos durante el periodo mínimo de medición (en minutos: 1, 15, 30, etc.). También se ha incluido la frecuencia de operación del sistema.

PORMEDIO DE PARAMETROS ELECTRICOS

**PROMEDIO TENSIONES Y CORRIENTES SUB1**

FECHA DE MEDICION: 22/03/2004      HORA DE MEDICION: 09:52:00

**PROMEDIO DE VOLTAJES Y CORRIENTES TRIFASICOS**

Voltaje Promedio Trifásico de Fase (CVAV_III) [V]	124
Corriente Promedio Trifásica (CAAV_III) [A]	150
Voltaje Promedio Trifásico de Línea (CVC_III) [V]	213

**FRECUENCIA DE RED**

Frecuencia (HZ):	60.1
------------------	------

Cerrar

Record: 1 of 302

Figura 3.39 Formulario de Voltajes y Corrientes promedio.

### 3.3.2.8 Demanda Máxima y Energía consumida por Fase.

En cuanto al contenido de este formulario se presentan tanto la energía activa como reactiva (inductiva y capacitiva) consumida durante el periodo de medición. También se presenta la demanda máxima en la fase 1. Debido a que la tarjeta de conexión a la PC instalada en el analizador de redes CIRCUTOR posee parámetros limitados, lo cual permite obtener solo datos de demanda y consumo para la fase 1, no se han creado formularios para el resto de las fases debido a que no se obtuvo ningún modulo que permitiera adicionar las fases restantes, pero con la instalación de una nueva tarjeta de red con parámetros ampliados podrían obtenerse mediciones para las tarifas 2 y 3, que corresponden a las fases respectivas.

La demanda máxima es un factor sumamente importante, ya que por medio de este se puede saber cual es el limite al que el sistema de potencia supervisado esta llegando en cuanto a demanda de potencia.

DEMANDA Y ENERGIA SUBESTACION 1	
<b>DEMANDA MAXIMA Y ENERGIA FASE 1 SUB 1</b>	
FECHA DE MEDICION	HORA DE MEDICION
2/03/2004	09:52:00
<b>ENERGIA ACTIVA Y REACTIVA FASE 1</b>	
Energia Activa Fase 1 (CKWH_1) [kWh]	81892.206
Energia Reactiva Inductiva Fase 1 (CKVARLH_1) [kVARh]	26701.403
Energia Reactiva Capacitiva Fase 1 (CKVARCH_1) [kVARh]	344.223
<b>DEMANDA MAXIMA FASE 1</b>	
Demanda Máxima Fase 1 (CMAX_DEM_1) [kW]	44
Cerrar	
Record: 1 of 302	

Figura 3.40 Formulario de demanda y Energía en Fase 1.

### 3.3.2.9 Todos los Parámetros Eléctricos.

Este es un formulario de consulta general, el cual abarca todas las variables eléctricas que el sistema puede capturar y presentar. En ambos nodos SCADA se cuenta con esta aplicación y es el único de los formularios que tiene la capacidad de ser impreso desde un comando en el formulario. El resto de formularios puede

imprimirse desde la barra de menú en el submenú *Archivo*. A continuación se muestra este formulario:

PARAMETROS ELECTRICOS SUB 1	
Fecha (DATA)	03/05/2004 [dd/mm/aaaa]
Hora (TIME)	10:31:00 [hh/mm/ss]
Tensión Simple Instantanea fase 1 (CV_1)	121 [V]
Tensión Compuesta Instantanea fase 12 (CV_12)	208 [V]
Intensidad Instantanea fase1 (CA_1)	74 [A]
Potencia Activa fase 1 (CKW_1)	7.75 [KW]
Potencia Reactiva Inductiva fase 1 (CKVARL_1)	0.5 [KVARL]
Potencia Reactiva Capacitiva fase 1 (CKVARC_1)	0 [KVARC]
Factor de Potencia fase 1 (CPF_1)	0.96
Tensión Simple Instantanea fase 2 (CV_2)	121 [V]
Tensión Compuesta Instantanea fase 23 (CV_23)	208 [V]
Intensidad Instantanea fase 2 (CA_2)	48.5 [A]
Potencia Activa fase 2 (CKW_2)	4.75 [KW]
Potencia Reactiva Inductiva fase 2 (CKVARL_2)	0.5 [KVARL]
Potencia Reactiva Capacitiva fase 2 (CKVARC_2)	0 [KVARC]
Factor de Potencia fase 2 (CPF_2)	0.97

Record: 1 of 99

Figura 3.41 Formulario de Todas las variables eléctricas.

### 3.3.3 Registros de las Bases de Datos.

En este formulario se seleccionan los datos completos, en bruto, como son obtenidos por medio de la interfaz con el analizador de redes. Se pueden encontrar todas las mediciones realizadas tanto en el pasado como las actuales. Debido a que se ha configurado para que se realicen por mes, aquí pueden verificarse por mes todas las mediciones que se han especificado en los formularios que hasta el momento se han descrito y accedido.

REGISTROS SUB1 2004 : Form

**SELECCION DE TABLAS CON PARAMETROS ELECTRICOS DE 2004 SUB 1**

ENERO

FEBRERO

MARZO

ABRIL

MAYO

JUNIO

JULIO

AGOSTO

SEPTIEMBRE

OCTUBRE

NOVIEMBRE

DICIEMBRE

Cerrar

Figura 3.41 Formulario de Registros de Datos.

Debe notarse que todos los registros de los datos obtenidos no pueden modificarse, es decir que estos datos solo poseen la característica de ser consultados y las bases de datos están protegidas de tal forma que los registros no puedan ser alterados. A continuación se presenta una base de datos en bruto pertenecientes al sistema de monitoreo. Puede notarse la diferencia entre una forma de visualización sencilla y más compacta y la forma de presentación por defecto que *Cirnet* realiza. Aquí se puede notar la importancia de la herramienta en Access.

	DATA	TIME	CV_1	CV_12	CA_1	CKW_1	CKVARL_1	CKVARC_1	CPF_1
▶	03/07/2004	10:20:01	121	208	272.333	30	11	0	
	03/07/2004	10:22:01	121	209	200	23	7	0	
	03/07/2004	10:23:01	121	208	201.286	23	7	0	
	03/07/2004	10:24:00	121	209	202.75	23.167	7	0	
	03/07/2004	10:25:01	121	208	205.7	23	7	0	
	03/07/2004	10:26:00	121	209	205.2	23.1	7	0	
	03/07/2004	10:27:00	121	209	220.6	24.667	8	0	
	03/07/2004	10:28:00	121	208	234.444	26.3	9	0	
	03/07/2004	10:29:00	121	208	236.667	26.778	9	0	
	03/07/2004	10:30:01	121	208	248.4	27.8	9.8	0	
	03/07/2004	10:31:01	121	208	266.9	29.8	10.3	0	
	03/07/2004	10:32:01	121	208	278	31	11	0	
	03/07/2004	10:33:00	121	208	277.111	31	11	0	
	03/07/2004	10:34:00	121	208	277	31	11	0	
	03/07/2004	10:35:00	121	208	240.8	27.2	9	0	
	03/07/2004	10:36:00	121	209	186.778	21	5.3	0	
	03/07/2004	10:37:00	121	209	185.111	21	5.222	0	
	03/07/2004	10:38:00	121	209	184.4	21	5.1	0	
	03/07/2004	10:40:00	121	208	184.474	21	5.053	0	
	03/07/2004	10:41:00	121	209	187.5	21.111	5.222	0	
	03/07/2004	10:42:00	121	208	215.1	24.1	6.8	0	
	03/07/2004	10:43:01	121	208	230.667	25.8	7.9	0	
	03/07/2004	10:44:01	121	208	229.889	26	8	0	
	03/07/2004	10:45:00	121	208	228.7	26	8	0	
	03/07/2004	10:46:01	121	208	231.111	26	8	0	
	03/07/2004	10:47:01	121	208	231	25.9	7.9	0	
	03/07/2004	10:48:01	121	208	217.7	24.778	7.111	0	
	03/07/2004	10:49:00	121	208	216.333	24.2	7.1	0	

Record: 1 of 452

Figura 3.42 Base de datos de las variables adquiridas en Julio.

Con esta pantalla termina el desarrollo de la aplicación en Access. Debe recordarse que existen muchas herramientas que pueden ser utilizadas para trabajar con bases de datos, pero se ha querido presentar una forma sencilla de trabajar con ellas, tanto para los desarrolladores de la aplicación, como para el que será el usuario final.



### 3.4 Manual de Usuario de la Aplicación para la Supervisión Remota por medio de Internet.

Como se ha planteado en los objetivos y en los conceptos teóricos básicos, se dejó para esta última fase del proyecto, el monitoreo remoto a través de Internet. Este monitoreo consiste en la verificación de las variables más importantes, de las dos subestaciones incluidas en el sistema SCADA (nodos). Debe destacarse que para poder tener acceso a través de Internet, debe poseerse un servidor en el cual se soporta la aplicación, de lo contrario no se podrá tener acceso a la herramienta. Dentro de las características de esta herramienta también se hace énfasis en la forma segura de su acceso y la facilidad que implica trabajar con ella, lo cual simplifica su uso y permite el manejo de la misma por usuarios que posean conocimientos básicos de Internet.

A continuación se presenta una breve descripción de la forma de manipulación de dicha herramienta.

Por medio de una dirección electrónica, establecida por el proveedor de Servicios Web se tendrá acceso a la siguiente página.

En ella puede observarse que se solicita un nombre de usuario y un password. Sin ellos no puede ingresarse a la aplicación y mucho menos se tiene acceso a los datos del sistema. Antes un mensaje de advertencia pone en sobre aviso al usuario para que ingrese su Identificación, previamente asignada por el administrador.



Figura 3.43 Página de acceso a la aplicación Web.



Figura 3.44 Pagina de acceso a la aplicación Web.

El funcionamiento de la pagina de inicio es parecido al de cualquier sistema de correos electrónicos en los cuales se solicita dicha información.

Si se digita correctamente la clave de acceso, se permitirá ingresar al sistema y un mensaje de advertencia, previo a la presentación de la pagina que permite introducir toda la información que detalla el mensaje:



Figura 3.45 Pagina de verificación de la subestación 1 (nodo 1).

...: Sistema SCADA de Monitoreo en Línea ...:



Agregar usuario:

---

Datos del usuario:

Login (ID):	<input type="text"/>	Perfil:	Administrador
Password:	<input type="text"/>	Reescriba el Password:	<input type="text"/>
Nombres:	<input type="text"/>	Apellidos:	<input type="text"/>
Telefono de casa:	<input type="text"/>	Celular:	<input type="text"/>
Email:	<input type="text"/>		

Figura 3.46 Pagina de introducción de datos para ser admitido como usuario predeterminado.

En la pagina de la figura 3.46 se validan los datos del usuario, ya sea para administrar o solamente para monitorear al sistema.



Figura 3.47 Mensaje de error.

Si se han digitado los datos incorrectamente, el sistema no reconocerá al usuario y aparecerá el mensaje de error que aparece en la figura 3.47.

Los usuarios poseen características diferentes, por ejemplo, el administrador posee privilegios diferentes al de los supervisores; entre esos ellos, puede realizar cambios en las formas de acceso en cuanto a seguridad, verificar código fuente, etc. Mientras que un supervisor o monitor solamente puede verificar los valores o variables que pueden ser verificadas con este sistema.

Si se han introducido los datos correctamente y al "Validar y enviar", en el botón inferior izquierdo, la aplicación permite verificar en los botones Subestación 1 ó Subestación 2, entonces el usuario ha sido admitido y se podrá supervisar al sistema con las paginas que se expondrán mas adelante. Si no se han introducido los datos correctamente se tendrá que "Limpiar", con el botón del centro, los datos digitados incorrectamente y volver a digitarlos. Sino se tiene una clave y un usuario asignado, simplemente no se podrá acceder al sistema de monitoreo remoto.

Si la clave de acceso ha sido admitida y los botones de Subestación 1 y Subestación 2 responden a un "click", entonces aparecerán pantallas como las mostradas a continuación, las cuales poseen las mismas características en cuanto a construcción:

::: Sistema SCADA de Monitoreo en Línea :::

Fecha: 1/01/1999

**Subestación 1**

Parámetro	Valor
Potencia Activa Trifasica [KW]	
Potencia Inductiva Trifasica [KVARL]	
Potencia Capacitiva Trifasica [KVARC]	
Factor de potencia Trifasico	

Parámetro	F1	F2	F3
Voltaje [V]			
Corriente [A]			
Factor de potencia			
Frecuencia [Hz]			

Parámetro	Valor
Energia Activa [KWH]	
Energia Inductiva [KVALH]	
Energia Capacitiva [KVARCH]	

Figura 3.48 Pagina de verificación de la subestación 1 (nodo 1).

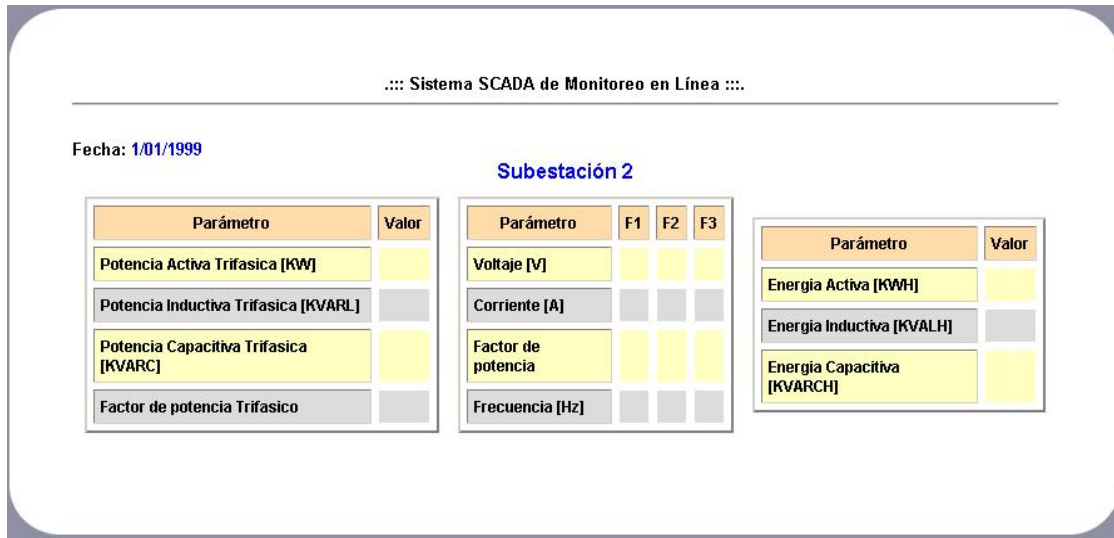


Figura 3.49 Pagina de verificación de la subestación 2 (nodo 2).

Puede observarse que las características de ambas paginas son las mismas, la diferencia está en que ambas pertenecen a nodos diferentes. Son comunes las variables que pueden ser verificadas por medio de ellas. Debe hacerse notar que estas paginas se refrescan automáticamente cada tres segundos, lo cual permite tener datos casi en tiempo real acerca del estado del sistema.

Las paginas muestran potencias trifásicas (activa, reactiva inductiva, reactiva capacitiva) y el factor de potencia trifásico, en el recuadro de la izquierda. En el cuadro del centro se encuentran parámetros por fase (Voltaje, corriente, factor de potencia y frecuencia nominal). Y en el recuadro de la derecha se muestran las diferentes energías (activa, reactivas inductiva y reactiva capacitiva); todas estas variables se presentan con sus respectivas unidades.

Debe notarse que debido a que esta aplicación interactúa directamente con las bases de datos creadas por *Cirnet*, solo pueden encontrarse datos de energía correspondientes a la tarifa 1 ó fase 1. esto se debe a que las tarjetas que poseen los equipos Circutor solo realizan esta función, y no permiten verificar las otras fases.



Figura 3.50 Pagina de cierre de sesión.

Si da un "click" en el botón salir se podrá ver la pagina mostrada en la figura 3.50 la cual muestra el fin de la sesión, con lo cual se sale del sistema.

## CAPITULO IV: MANUAL DE REFERENCIA.

El manual de referencia es una guía que tiene como objetivo informar acerca de las características que deben tener las diferentes herramientas software, hardware, periféricos, interfaces y red en las cuales la aplicación SCADA desarrollada en *Cirnet* para Windows trabaja de forma optima.

### **4.1 Sistema Operativo.**

Para que la aplicación SCADA trabaje correctamente, las terminales deben cumplir con ciertas características. Entre estas tenemos que dichas terminales deben poseer un sistema operativo que maneje un sistema de archivos FAT ó FAT32. Esto se debe a que *Cirnet* ha sido diseñado para trabajar soportado por un sistema operativo que maneje este tipo de archivos. Los sistemas operativos que manejan archivos FAT son las versiones de Windows Millenium hacia atrás (incluidas las versiones del 95 al 98). Algunas versiones más nuevas que estas, como por ejemplo Windows 2000, no permiten la operación de *Cirnet*. Aunque Windows XP podría permitirlo, debido a que puede seleccionarse el sistema de archivos que se requiere manejar (FAT, FAT32 o NTSF). En caso de poseer versiones de Windows en las que *Cirnet* no pueda trabajar, deberá adquirirse una versión actualizada de *Cirnet* para Windows, que pueda operar en sistemas operativos que manejan bloques de archivos de tipo NTSF.

#### *4.1.1 Definiciones de Sistemas de Archivos.*

Todos los sistemas de archivos que utiliza Windows organizan los discos duros en clústeres (cantidad mínima de espacio en disco que se puede asignar a un archivo). Estos están formados por uno o varios sectores contiguos. Cuanto menor sea el tamaño del clúster, más eficaz será el almacenamiento de la información en disco. Si no se especifica ningún tamaño de clúster durante el formato, Windows elegirá valores predeterminados en función del tamaño del volumen. Estos valores predeterminados se seleccionan para reducir la cantidad de espacio perdido y la

cantidad de fragmentación del volumen. Un clúster recibe también el nombre de unidad de asignación.

En relación con las redes, grupo de equipos independientes que funcionan juntos para proporcionar un conjunto común de servicios y presentan la imagen de un único sistema a los clientes. El uso de un clúster mejora la disponibilidad de los servicios y la escalabilidad y capacidad de administración del sistema operativo que los proporciona.

Existen tres sistemas de archivos, los cuales se definirán mas adelante.

#### *4.1.2 Tabla de asignación de archivos (FAT).*

Sistema de archivos utilizado por MS-DOS y otros sistemas operativos basados en Windows para organizar y administrar los archivos. La tabla de asignación de archivos (FAT, *File Allocation Table*) es una estructura de datos que Windows crea cuando se da formato a un volumen mediante el sistema de archivos FAT o FAT32. Windows almacena información acerca de cada archivo en la tabla de asignación de archivos, de forma que pueda recuperar el archivo posteriormente.

#### *4.1.3 Sistema de Archivos FAT32.*

Derivado del sistema de archivos Tabla de asignación de archivos (FAT). FAT32 admite tamaños de clúster más pequeños y volúmenes más grandes que FAT, lo que permite una asignación más eficaz del espacio en los volúmenes FAT32.

#### *4.1.4 Sistema de archivos NTFS.*

Sistema de archivos avanzado que proporciona características de rendimiento, seguridad, confiabilidad y avanzadas que no se encuentran en ninguna versión de FAT. Por ejemplo, NTFS garantiza la coherencia del volumen mediante técnicas estándar de registro de transacciones y recuperación. Si se producen errores en un sistema, NTFS utiliza el archivo de registro y la información de punto de comprobación para restaurar la coherencia del sistema de archivos. En



Windows 2000 y Windows XP, NTFS proporciona también características avanzadas como permisos para archivos y carpetas, cifrado, cuotas de disco y compresión.

#### 4.1.5 Elección entre NTFS, FAT y FAT32.

Se puede elegir entre tres sistemas de archivos diferentes para las particiones de disco en un equipo en que se ejecute Windows XP: NTFS, FAT y FAT32. A continuación se presenta la forma de comparar los sistemas de archivos.

En la mayoría de aplicaciones se recomienda el sistema de archivos NTFS, con excepción de la versión de *Cirnet* para Windows en la cual se ha desarrollado la aplicación SCADA, esto por las razones siguientes:

NTFS es más eficaz que FAT o FAT32 e incluye funciones necesarias para actuar como host de Active Directory así como otras funciones de seguridad importantes. Únicamente puede utilizar características como Active Directory y la seguridad basada en dominios si elige NTFS como sistema de archivos.

Puede convertir particiones a NTFS fácilmente. El programa de instalación simplifica la conversión, tanto si las particiones utilizaban FAT o FAT32 como si usaban la versión anterior de NTFS. Este tipo de conversión mantiene intactos los archivos, a diferencia de cuando se da formato a una partición. Si no necesita mantener intactos los archivos y dispone de una partición FAT o FAT32, se recomienda que se aplique *formato* a la partición con NTFS en lugar de *convertirla* desde FAT o FAT32. Si se da formato a una partición, se borrarán todos los datos y se empezará con una unidad limpia.

A fin de mantener el control de acceso en archivos y carpetas y admitir cuentas limitadas, se debe utilizar NTFS. Si se usa FAT32, todos los usuarios tendrán acceso a todos los archivos del disco duro, independientemente de su tipo de cuenta (administrador, limitado o estándar).

El sistema de archivos que funciona mejor con discos de gran tamaño es NTFS. (El siguiente sistema de archivos que funciona mejor con discos grandes es FAT32.)

Existe una situación en la que es posible que desee seleccionar FAT o FAT32 como sistema de archivos. Si es necesario disponer de un equipo que a veces ejecute un

sistema operativo de una versión anterior de Windows y otras veces ejecute Windows XP, deberá tener una partición FAT o FAT32 como partición principal (o de inicio) en el disco duro. La mayoría de las versiones anteriores de Windows no pueden tener acceso a una partición si utiliza la última versión de NTFS. Las dos excepciones son Windows 2000 y Windows NT 4.0 con Service Pack 4 o posterior. Windows NT versión 4.0 con Service Pack 4 o posterior tiene acceso a particiones con la última versión de NTFS, pero con algunas limitaciones: No se puede tener acceso a archivos que se han almacenado mediante características de NTFS que no existían cuando se publicó Windows NT 4.0.

Sin embargo, para cualquier otra situación en la que no existan varios sistemas operativos, el sistema de archivos recomendado es NTFS.

Una vez que se ha convertido una unidad o partición a NTFS, no puede volver a convertirla a FAT o FAT32 simplemente. Se tendrá que volver a formatear la unidad o partición que borrará todos los datos de la partición, incluidos los programas y los archivos personales.

La tabla siguiente describe la compatibilidad de cada sistema de archivos con varios sistemas operativos.

<b>NTFS</b>	<b>FAT</b>	<b>FAT32</b>
Un equipo en que se ejecuta Windows XP o Windows 2000 puede tener acceso a los archivos de una partición NTFS. Un equipo en que se ejecuta Windows NT 4.0 con Service Pack 4 o posterior puede tener acceso a algunos archivos. Otros sistemas operativos no permiten el acceso.	Es posible tener acceso a través de MS-DOS, todas las versiones de Windows, Windows NT, Windows 2000, Windows XP y OS/2.	Solo es posible tener acceso a través de Windows 95 OSR2, Windows 98, Windows Millennium Edition, Windows 2000 y Windows XP.

Tabla 4.1 Diferencias entre Sistemas de archivos.

La tabla siguiente compara los posibles tamaños de archivo y disco con cada sistema de archivos.

NTFS	FAT	FAT32
El tamaño mínimo de volumen recomendado es aproximadamente de 10 MB. Son posibles volúmenes superiores a 2 terabytes (TB). No puede utilizarse en disquetes.	Volúmenes desde el tamaño de un disquete hasta 4 GB. No admite dominios.	Volúmenes entre 512 MB y 2 TB. En Windows XP, únicamente puede dar formato a un volumen FAT32 hasta 32 GB. No admite dominios.
Tamaño del archivo limitado únicamente por el tamaño del volumen.	El tamaño máximo de archivo es 2 GB.	El tamaño máximo de archivo es 4 GB.

Tabla 4.2 Comparación de volúmenes de archivo manejados.

***Nota:*** Algunos programas más antiguos (esto incluye la versión de *Cirnet* para Windows con que se cuenta en la actualidad) podrían no funcionar en un volumen NTFS, por lo que se deben determinar los requisitos actuales del software antes de eliminarlo o instalar otro sistema operativo, lo cual implica una mayor inversión de recursos.

## 4.2 Características de Software y Hardware.

### 4.2.1 Software CirNET para Windows.

El programa CIRNET para Windows, es un programa **SCADA** muy potente que permite crear sistemas de medida y control tan sofisticados como el usuario desee.

El programa es totalmente abierto, configurable y ampliable.

Es una verdadera aplicación Windows, por lo que es muy intuitivo y fácil de aprender. Utiliza todas las ventajas y propiedades del entorno Windows, como son las **MDI (Multiple Documents Interface)** para una mejor visualización de los procesos.

Enlaza mediante el **protocolo DDE (Dynamic Data Exchange)** con otras aplicaciones Windows.

Permitiendo enviar a otras aplicaciones como **Excel, Word, Access**, etc. los datos que está obteniendo en tiempo real y realizar gráficos on-line, enviar los datos que desee a otro ordenador, etc.

Todo esto sin dejar de ejecutar la aplicación del programa **SCADA**.

El **módulo SCADA**, permite la visualización de una o más ventanas con dibujos de fondo, incorporar a las mismas, herramientas de animación en función de valores obtenidos, realizar gráficos en tiempo real, crear zonas sensibles al ratón, crear botones personalizados con funciones como: ejecución de scripts, aperturas de subpantallas, creación de recetas, etc.

Incluye también un **módulo de BASIC** para realizar programas y ejecutarlos en función de condiciones.

El **módulo de alarmas**, indica si una condición de alarma ha sido activada, reconocida o restaurada, permitiendo emitir avisos acústicos los cuales pueden ser mensajes de voz o simples señales de alarma.

Se puede hacer que se ejecute un script o código de programación y que se guarde en un fichero en formato Dbase o texto la descripción de la alarma para un posterior tratamiento.

También se pueden asignar textos largos a condiciones de alarmas para que la persona a cargo del sistema tenga información adicional para solventar el problema, o incluso una secuencia de vídeo, en la que se muestra como resolver el problema.

Con el **módulo de base de Datos** se puede elegir que parámetros se desean guardar de cada equipo, con ello se optimiza el espacio requerido para guardar las bases de datos.

También se pueden crear gráficos de la evolución de todos los parámetros que se han obtenido, como por ejemplo, obtener un gráfico sobre los consumos de una sección en particular, en el periodo de tiempo que desee, etc.

La filosofía del programa sigue una regla básica: la modularidad.

Esto significa que la mayoría de las posibilidades que ofrece el programa son fácilmente modificables o ampliables.

Por ejemplo, a medida que se vayan realizando nuevos drivers de comunicaciones de equipos nuevos, bastará con añadirlos junto al programa, y automáticamente serán reconocidos por este.

Al estar realizado bajo entorno Windows, el programa puede ser ejecutado simultáneamente con otras aplicaciones Windows, sin que se deje de adquirir datos.

Para generar aplicaciones más fácilmente se han implementado sistemas como **"arrastrar y soltar" (Drag & Drop)**.

El programa soporta múltiples formatos gráficos (mapa de bits, DIB, PCX, TGA) e incluso admite los gráficos del portapapeles.

Así mismo puede utilizar todas las impresoras, pantallas, redes de comunicación o ratones que sean compatibles con Windows.

#### *4.2.2 Bases de Datos.*

Las bases de datos son las herramientas más utilizadas por los sistemas SCADA. Estas son las encargadas de almacenar toda la información que se esta recopilando a través de los diferentes equipos de medición, periféricos, etc. Estas, también, permiten obtener datos históricos que resultan útiles para la consulta en caso de anomalías en los equipos supervisados. Los datos recopilados además de proporcionar información acerca del comportamiento del sistema supervisado, también permite realizar análisis a partir de comparación de resultados, por ejemplo, el comportamiento del sistema en un mes con respecto a otro y a partir de los datos de operación deducir si este esta en buen estado. Por esta razón todo sistema SCADA gravita alrededor de su base de datos, en cuanto a la utilidad que estas proporcionan como herramienta de supervisión y análisis.

El sistema de supervisión y adquisición de datos que se presenta en este trabajo no es la excepción, es decir que también gira alrededor de las bases de datos que

se almacenan, no solo en cuanto a información instantánea, sino también en cuanto a información histórica.

### **4.3 Características de la Red Seleccionada.**

La red dentro de la cual opera el sistema de monitoreo es una Red LAN ETHERNET con topología en estrella, que posee un servidor como eje (nodo) y varias terminales de acceso.

Esta topología tiene un nodo central desde el que se irradian todos los enlaces hacia los demás nodos. Por el nodo central, generalmente ocupado por un hub, pasa toda la información que circula por la red.

La ventaja principal es que permite que todos los nodos se comuniquen entre sí de manera conveniente. La desventaja principal es que si el nodo central falla, toda la red se desconecta.

Los componentes de la red se comunican por medio de cable UTP.

Esta configuración se seleccionó sobre la base de la existencia de esa red en la Universidad Don Bosco.

Entre los inconvenientes encontrados está el hecho de que sí bien es cierto la red estaba montada, esta posee un sistema operativo Windows NT Professional en cada una de sus terminales, plataforma bajo la cual *Cirnet* no puede operar. Para ello se acopló una computadora con otro Sistema Operativo (Windows 98), donde *Cirnet* funciona a la perfección. Por esta razón no todas las aplicaciones pueden operar en todas las terminales que componen la red. Por ejemplo, la aplicación en *Cirnet* opera solo en su correspondiente *Terminal*, mientras que las demás herramientas que conforman el paquete operan en todas las computadoras que conforman la red.

A continuación se presentan las características técnicas de la red identificada, dentro de la cual opera la aplicación SCADA que se describe en este documento.

Características Técnicas de la red:

Tipo	Medio	Ancho de banda máximo	Longitud máxima de segmento	Topología Física	Topología Lógica
10Base-T	UTP Cat 5	10 Mbps	100 m	Estrella; Estrella Extendida	Bus

Tabla 4.3 Características Técnicas de la Red Utilizada.

#### 4.4 Metodos de E/S serie para comunicación entre terminales.

Se utiliza el nombre genérico de Terminal para designar a los sistemas que se comunican utilizando un procedimiento serie de E/S. Una Terminal puede ser: una computadora un periférico etc. La comunicación entre terminales se hace utilizando líneas o canales de transmisión que puede ser:

- **Simplex:** Cuando un dispositivo solo es capaz de transmitir información en un solo sentido (cuando la información se transmite en un solo sentido).



Figura 4.1 Circuito de Transmisión Simplex.

- **Semiduplex:** En este tipo de canal los terminales son capaces de transmitir información en ambos sentidos pero no de forma simultanea.



Figura 4.2 Circuito de Transmisión Semiduplex.

- **Fullduplex:** En este tipo de canal ambos terminales son capaces de transmitir simultáneamente información en ambos sentidos.



Figura 4.3 Circuito de Transmisión Fullduplex.

#### **4.5 Metodo de comunicación serie asincrono.**

En el método asíncrono la transmisión se controla por medio de bits que enmarcan o definan cada carácter transmitido estos bits son los denominados bits de start y stop, y son utilizados por el Terminal receptor para sincronizar su reloj con el del transmisor para la recepción de cada carácter.

La transmisión serie asíncrona se basa en las siguientes reglas:

1. Cuando no se envían datos por la línea esta se mantiene en estado uno (marcas)
2. Cuando se desee transmitir un carácter se cambia primero en bits de inicio el cual pone la línea a cero lógico durante el tiempo de unos bits.
3. A continuación se envían todos los bits del carácter a transmitir cuyos intervalos son marcados por el reloj de transmisión.
4. A continuación del ultimo bit del carácter se envía el bit de señal o de stop el cual hace que la línea se ponga a "uno" por lo menos durante el tiempo de 1 bits.

#### **4.6 Reglas para la recepción serie.**

Los datos codificados según las reglas anteriores pueden ser detectados por el receptor de la siguiente forma:

1. Esperar en la línea de transmisión una transición activa en bajo en la señal recibida.
2. Luego de detectarse la transmisión debe de activarse un reloj de frecuencia y fase igual al del transmisor.
3. Se procede a mostrar la señal recibida por medio del reloj de recepción con el fin de formar el carácter de datos. (se va contando el número de bits)
4. Se lee un bit de la línea de transmisión y se comprueba si este es uno (bits de stop) para confirmar que no existe error de sincronización.

El bit de stop tiene la misión de llevar la línea de transmisión a estado "1" lógico para que el bit de inicio del próximo carácter a transmitir provoque la transición de uno a cero que permita al receptor sincronizar el siguiente carácter.



## 4.7 Interfaz RS-232.

### 4.7.1 Norma RS-232 o V.24.

Los sistemas de comunicación serie tienen a su disposición un conjunto de recomendaciones o normas elaboradas por asociaciones o institutos de normalización los cuales especifican con precisión todos los caracteres del sistema de comunicaciones.

Las normas para comunicaciones serie están clasificadas por niveles, para este caso solamente se estudia el nivel (1) el cual hace referencia a lo siguiente:

- A) Las características eléctricas de la señal.
- B) Las características mecánicas de la interfaz.
- C) Descripción funcional de la interfaz.

Una de las normas más ampliamente aceptadas para las comunicaciones serie es: **EIA RS232 – C**. La cual define las características funcionales, eléctricas y mecánicas de la interfaz entre un Terminal y un equipo de comunicaciones, por ejemplo la tarjeta instalada en el equipo CIRCUTOR, para establecer la comunicación entre este equipo Terminal y la computadora.

#### **DTE** (*equipo Terminal de datos*)

Es todo equipo que convierte señales del mundo real o señales eléctricas de datos codificados o viceversa.

#### **DCE** (*equipo de comunicaciones de datos*)

Es el equipo encargado del encaminamiento de la información a través del medio de información, este realmente es un módem.

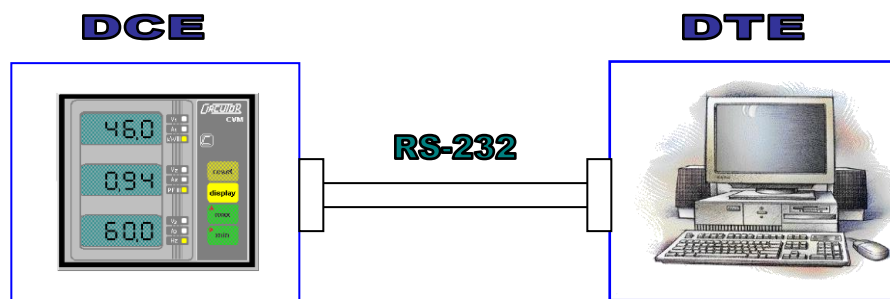


Figura 4.4 Comunicación CIRCUTOR - Terminal.

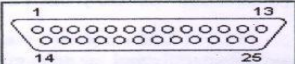
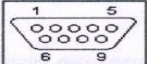
#### 4.7.2 Características de la norma RS-232.

##### A) CARACTERISTICAS MECANICAS.

El conector estándar puede ser de 25 pines DB 25 o de 9 pines DB 9.

##### B) El conector tiene forma trapezoidal.

E incluye dos tornillos de seguridad para asegurar su buena conexión.

Conector de 25 pin (en el PC):		Conector de 9 pin (en el PC):	
			
PIN-9	PIN-25	DESCRIPCION	
1	8	Carrier Detect (DCD/CD); Detección de portadora.	
2	3	Receive Data (RXD/RD); Entrada de datos.	
3	2	Transmit Data (TXD/TD); Salida de datos.	
4	20	Data Terminal Ready (DTR); Indica que el puerto está preparado.	
5	7	Signal Ground (GND/SG); Proporciona la referencia cero de voltaje.	
6	6	Data Set Ready (DSR); Datos preparados, listo para recibir.	
7	4	Request to Send (RTS); Petición de envío de datos.	
8	5	Clear to Send (CTS); Indica que se desea transmitir.	
9	22	Ring Indicator (RI); Anuncia una llamada al otro dispositivo.	
-	9-19	No empleadas	
-	1	Masa del chasis (GND)	

Voltajes máximos entre -15V y +15V. Salidas binarias entre +5V a +15V y -5V a -15V. Entradas binarias entre +3V a +15V y -3V a -15V. Voltajes de entrada entre -3V a +3V y voltajes de salida entre -5V a +5V son indefinidos. Voltajes positivos indican ON o ESPACIO, voltajes negativos indican OFF o MARCA.

Figura 4.5 Especificaciones del Pines de la RS-232.

#### 4.8 Interfaz RS-485.

Para lograr comunicación con equipos ubicados a distancias medianas (1200 a 1300 metros), se utiliza el cableado de comunicación multipunto: un mayor número de dispositivos pueden ser conectados por medio de un cable de tres hilos, de forma similar que en las redes ETHERNET. La mayor parte de sistemas conectados por medio de RS-485 utilizan una arquitectura Master/Slave (Maestro/Esclavo), donde cada unidad esclava tiene una sola dirección y responde solamente a paquetes dirigidos hacia esa unidad. Esos paquetes son generados por un Maestro (PC), que enlaza a todas las unidades Esclavas conectadas.

El principio de operación eléctrico está basado en el uso de transmisores diferenciales con voltajes alternantes entre 0 y 5 V. La RS-232 utiliza dos cables separados TP.

ESPECIFICACIONES		RS232	RS423	RS422	RS485
Modo de Operación		SINGLE -ENDED	SINGLE -ENDED	DIFERENCIAL	DIFERENCIAL
Total de dispositivos de manejo y recepción sobre una línea (un driver activo a la vez para redes RS-485)		1 DRIVER 1 RECVR	1 DRIVER 10 RECVR	1 DRIVER 10 RECVR	32 DRIVER 32 RECVR
Máxima longitud del cable		50 FT.	4000 FT.	4000 FT.	4000 FT.
Máximo rango de datos (40ft. - 4000ft. para RS422/RS485)		20kb/s	100kb/s	10Mb/s- 100Kb/s	10Mb/s- 100Kb/s
Máxima Salida de Voltaje del Driver		+/-25V	+/-6V	-0.25V to +6V	-7V to +12V
Nivel de Señal de Salida del Driver (carga mínima)	Carga completa	+/-5V a +/-15V	+/-3.6V	+/-2.0V	+/-1.5V
Nivel de Señal de Salida del Driver (máxima descarga)	Descargado	+/-25V	+/-6V	+/-6V	+/-6V
Impedancia máxima del Driver (Ohms)		3k a 7k	>=450	100	54
Max. Driver Current in High Z State	Activo	N/A	N/A	N/A	+/-100uA
Max. Corriente en el Driver, Estado Z Alto	Inactivo	+/-6mA @ +/-2v	+/-100uA	+/-100uA	+/-100uA
Slew Rate (Max.)		30V/uS	Adjustable	N/A	N/A
Rango de Entrada de voltaje del Receptor		+/-15V	+/-12V	-10V to +10V	-7V to +12V
Sensibilidad de Entrada del Receptor		+/-3V	+/-200mV	+/-200mV	+/-200mV
Resistencia de Entrada del Receptor (Ohms), (1 Carga Estándar para RS485)		3k a 7k	4k min.	4k min.	>=12k

Tabla 4.4 Comparación de características técnicas entre interfaces.

El estándar RS-485 fue creado para permitir la conexión de varios dispositivos a través de una línea común. Cada dispositivo representa una Unidad de Carga (UL: Unit Load) y se denomina nodo. Una red RS-485 puede contener hasta 32 UL, donde cada UL representa una carga de 12 K. Utilizando receptores de alta impedancia, la cantidad de nodos puede aumentarse hasta 256.

La transmisión se hace en modo diferencial: el cable (usualmente par trenzado) lleva la señal y su inverso. El receptor detecta la diferencia entre ambas. Como la mayor parte del ruido se acopla a los dos cables por igual, en el receptor se cancela. Esto le otorga más inmunidad al ruido que otras interfaces, como la RS-232.

El cable puede tener una longitud de hasta 1300 metros (4000 pies) y la velocidad de transmisión puede llegar hasta 10 Megabits/s. Sin embargo, estos límites no pueden alcanzarse simultáneamente: a mayor longitud del cable, menor velocidad. Por ejemplo, con 1300 metros de cable, la velocidad máxima es de 90 Kbit/s y con unos 16 metros (50 pies) la velocidad llega hasta 10 Mbit/s.

La transmisión de los datos se hace generalmente de manera asíncrona, como en La RS-232, con un bit de start, bits de datos, un bit de paridad opcional y un bit de stop. Aunque el bus RS-485 es bidireccional (pueden fluir datos en ambos sentidos) la transmisión es half-dúplex, es decir, en un sentido por vez y no simultáneamente.

La RS-485 no define ningún protocolo para controlar el flujo de datos en el bus. Lo más habitual es que un nodo, denominado master, interroge al resto, denominados slaves. El master envía a través del bus un requerimiento que incluye un número o código que identifica a un slave (ID), que al recibirlo, contesta con la información solicitada. También es posible una configuración multi-master, en la que la comunicación pueda ser iniciada por más de un nodo en la red. Sin embargo deben tomarse ciertas precauciones para evitar colisiones cuando dos master intentan transmitir simultáneamente.

La norma RS485 opera sobre líneas balanceadas y no es necesario un conductor entre las tierras de los equipos para establecer la comunicación. No resultan

necesarios blindajes especiales, si el cable dispone de una pantalla se le puede conectar a tierra en uno solo de los extremos. Los terminales de GND del lado A y B se conectarán a tierra solamente.

La interfaz RS-485 es una versión mejorada de RS-422, que permite que hasta 32 dispositivos compartan el bus serie en una configuración "daisy-chain", creando una red en la que un único dispositivo puede controlar dicho bus en un momento dado.

Se utiliza con frecuencia para interconectar las unidades de la Terminal alejada (RTU) o los reguladores programables de la lógica (PLC) en redes utilitarias de la subestación SCADA .

#### *4.8.1 Características de la RS-485 (EIA ó V11/CCITT).*

En resumen, las características sobresalientes de la RS-485 son:

- Alta inmunidad al ruido.
- Largo máximo de los conductores: 1300 metros (depende de la velocidad).
- Alta velocidad (hasta 10 Mega bits/s).
- Capacidad de conectar hasta 32 nodos.
- Capacidad de configuración en modo multi-Master.
- Tensión de emisión: nivel «0»: +1.5 V a +5 V, nivel «1»: -1.5 V a -5 V.
- Nivel de recepción: nivel «0»: > +0.2 V, nivel «1»: < -0.2 V.
- Puerto serie DB9 para consola de configuración y monitorización.

#### **4.9 Requisitos del Sistema. Hardware, Software y Conexiones.**

El propósito de establecer los requisitos del sistema es plantear una guía de utilización rápida de equipos y conexionado de los diferentes dispositivos usados en esta aplicación, no solo para poder realizar una programación y configuración adecuada de los mismos, sino que proponer una forma o procedimiento a seguir para obtener el mayor provecho de la herramienta desarrollada. Esto permitirá la mejor supervisión local y remota de los analizadores de redes CVMk, y por consiguiente de los sistemas de potencia a los que estos equipos están conectados, y centralizados en terminales o computadoras (PC), de las cuales puede aprovecharse su capacidad de procesamiento, almacenamiento, diferentes formas de conexión en red (red local: LAN y remota: Internet), etc. Esto con el objetivo de establecer resultados a partir de los datos analizados y establecer el estado del sistema a partir de dicho análisis.

A continuación se presentan algunos de los requisitos de para el montaje de estos sistemas, que permiten obtener un control supervisorio basado en analizadores de redes CVM (CIRCUTOR):

1. Colocar en los diferentes equipos que se quieren monitorear los analizadores de redes CVMk, los cuales deben poseer un módulo de comunicación RS-485.
2. Realizar el cableado a lo largo de toda la planta, pasando por las ubicaciones de los diferentes equipo, adicionando un máximo de 32 nodos, por puerto de comunicación serie (Comm), no permitiendo que la distancia máxima de conexión de un equipo sea mayor a los 1200 metros. El cableado deberá llegar finalmente hasta el lugar que se requiere como centro de supervisión (ubicación de la PC, Computador Maestro o Servidor).
3. Se requiere el uso de convertidores inteligentes, los cuales funcionan como convertidores de señales entre las interfaces RS-232 á RS-485. Estos convertidores también realizan la función de direccionar la información

desde los diferentes dispositivos, a partir de la dirección que se le ha establecido al equipo de medición CVM.

4. Es necesario contar con una computadora que hace las veces de servidor, Terminal o dispositivo maestro, en la cual debe estar instalado el software CIRNET para Windows, que es el encargado de manejar los protocolos de comunicación, previa configuración de equipos y la infraestructura que permite la creación de pantallas SCADA, visualizaciones, tendencias históricas, diferentes tipos de gráficos, etc.
5. Poseer una aplicación SCADA, basada en *Cirnet* para Windows, para integrar todos los componentes de hardware y software, para obtener el sistema de monitoreo requerido. Se debe destacar que para esta aplicación también se requieren las siguiente herramientas de Microsoft:

***Hoja de Cálculo: Excel.*** Puede soportar archivos de bases de datos (.dbf), los cuales son automáticamente creados por la aplicación, previa configuración, para la visualización de las diferentes variables programadas para ser mostradas.

***Manipulador de Bases de Datos: Access.*** Permite importar y establecer un puente con los archivos de bases de datos, creados por la aplicación en Excel, para ser visualizados y manipulados de forma sencilla y versátil. En este caso se convierte en la base de toda la aplicación, ya que desde Access se ejecuta la herramienta SCADA.

***Navegador en Internet: Internet Explorer.*** Permite visualizar la aplicación para el monitoreo remoto; Páginas Web creadas con vínculo en las Bases de Datos para poder tener acceso y visualizar las diferentes variables históricas y en tiempo real que el sistema adquiere y almacena, de forma segura.

Estos son elementos propuestos por los creadores de la aplicación, debido a que con cada uno de estos programas se ha trabajado y se han experimentado problemas mínimos en cuanto a intercambio dinámico de datos (DDE), ya que todos estos componentes pueden interactuar casi de forma perfecta entre ellos.

#### **4.10 Ventajas del Sistema.**

- 1) Reducción de costos por cableado al instalar sólo un par de cables (mas el cable de tierra).
- 2) Mayor inmunidad al ruido al usar comunicación digital, que es inherentemente más inmune que la análoga.
- 3) Permite un máximo de 32 dispositivos conectados en red por puerto de comunicación serie. Estos pueden ser instalados en cualquier instante, ya sea que se conecten con la red inicial o se vayan adicionado a la red medida que el sistema crece.
- 4) Permite enviar señales a largas distancias sin que sea necesario utilizar equipos que regeneren la señal, ya que su distancia máxima soportada es de 1200 metros. Siendo esta una ventaja muy importante para poder conformar redes de sistemas distantes.

#### **4.11 Protocolos y Forma de Conexión de Dispositivos.**

La tarjeta de salida RS-485 entrega 2 líneas aisladas galvánicamente, llamadas A y B. En el protocolo RS-485, estas líneas funcionan como par diferencial.

El 1 lógico se representa por la condición en que el voltaje en la línea A, es mayor que el de la línea B, y al contrario para el 0 lógico. Típicamente  $A - B = 5$  voltios para 1 lógico y  $A - B = -5$  voltios para 0 lógico.

La comunicación RS-485 es bidireccional Half duplex, a diferencia del RS-422 que usa los mismos niveles de voltaje pero con 2 pares de líneas para comunicación Full duplex.



#### 4.12 Modulo de Expansi3n CVM/RED.

El CVMk puede ser ampliado mediante m3dulos de expansi3n, las cuales permiten agregar prestaciones adicionales. Estas tarjetas se conectan en la parte posterior del CVMk, quedando en el interior del mismo. Las caracteristicas que este m3dulo proporciona son:

- Contadores de Energía ( kWh, kvarLh y kvarCh ).
- Permite la conexi3n de hasta 32 dispositivos CVMk a una distancia m3xima de 1200 metros.
- Este modulo tipo RED presenta dos Salidas integradas dentro del mismo conector DB9:
  - COM1, RS-485: Esta salida es la principal, permitiendo as3 la comunicaci3n serie con dispositivos ordenadores como un PC e incluso un PLC. Siendo sus terminales los pines: 1,2,5.
  - COM2, RS-485: Esta segunda salida es para poder realizar conexiones con los diferentes perif3ricos. Siendo sus terminales los pines: 3,4,5

Determinaci3n de Pines de la Norma RS-485:

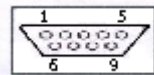


Figura 4.6 Conector de 9 pines, DB9

<b>PIN</b>	<b>Descripción</b>
1	Tx -
2	Tx +
5	GND
<b>Segunda Salida Serie para conectar los periféricos</b>	
3	Tx -
4	Tx +
5	GND

Tabla 4.5 Descripción del Pinado

Los módulos de expansión utilizados son el CVM/RED, su forma física se presenta a continuación:

- **Modulo CVM/RED.** Modulo de energía + 2 salidas RS-485:
  - Comm 1: RS-485 para computadora personal (PC) ó autómeta.
  - Comm 2: RS-485 para periféricos familia CVM.



Figura 4.7 Modulo de Expansión CVM/RED.

Esta tarjeta de expansión es la que se utiliza en el nodo 2 (subestación 2).

A continuación se describen algunas características de los módulos de expansión compatibles con el sistema, bajo las condiciones previamente establecidas.

- **Modulo CVM/232:** Comunicación con la norma RS-232 para establecerla con PC.



Figura 4.8 Modulo de CVM/232.

Este módulo es el que posee el nodo 1 (subestación 1). Solo permite la conexión de un dispositivo a uno de los puertos de comunicación. Con esta configuración actual de dispositivos, el sistema SCADA solo tiene capacidad para soportar 33 equipo en red, 1 equipo con RS-232 y 32 equipos con RS-485. Este módulo es instalado de fabrica en cada uno de los equipos Circutor (CVMk).

Otros módulos de expansión compatibles con el sistema desarrollado, se muestran a continuación:

- Tarjeta ER-232: Comunicación con la norma RS-232 para establecerla con PC o PLC.



Figura 4.9 Modulo de CVM/ER-232.

- Tarjeta ER-485: Comunicación con la norma RS-485 para la interconexión con periféricos.



Figura 4.10 Modulo de CVM/ER-485.

- A continuación se presentan Módulos de Expansión que presenta salidas analógicas como de Relé:

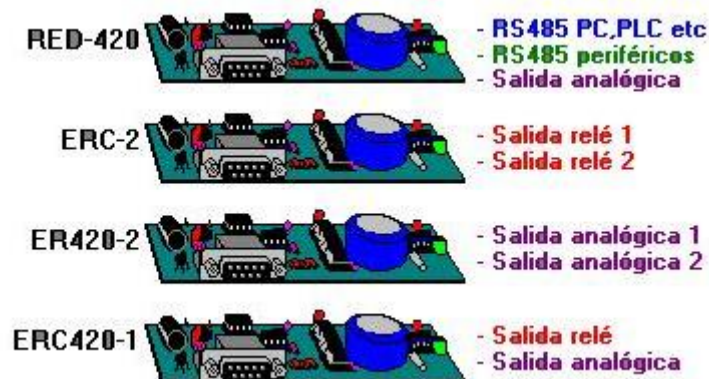


Figura 4.11 Módulos de la familia CVM y características.

## **4.13 Modo Maestro-Eslavos.**

### *4.13.1 Conexión Física de Dispositivos.*

Este modo de operación permite tener varios instrumentos conectados con un computador tipo PC, mediante un par de cables en forma de "bus" lineal. Los instrumentos se conectan al cable según se muestra en la Figura 4.9, pasando el cable de un dispositivo al otro.

El aislamiento galvánico se da desde la computadora Terminal (PC) al bus (mediante el módulo aislador convertidor inteligente) y del bus a cada los equipos que conformaran la red, logrando así, total protección de los equipos a transientes de alto voltaje y corto circuitos accidentales.

A veces es conveniente colocar una resistencia (1/4 Watt) en cada uno de los dos extremos de la línea de transmisión (cable) de modo de acoplar las impedancias y evitar rebotes de la señal. Estas resistencias deben ser de la misma resistencia (Ohms) de la impedancia característica del cable, según la especifica el fabricante.

El cable del bus debe ser adecuado para transmisión digital de baja velocidad (menos de 10 Mbit/seg) de los que hay muchos en el mercado.

Se recomiendan las siguientes características (se especifican también los términos en Ingles por que casi todos los catálogos de cable están en ese Idioma):

- 1) Un Par de cables enrollados individualmente. (Individually Twisted Pair)
- 2) Preferentemente blindado. (Shielded)
- 3) Preferencia 22 AWG ó 24 AWG.
- 4) Multifilar ó Sólido. (Stranded, Solid)

### *4.13.2 Protocolos de Comunicación.*

El computador o PC realiza la labor de maestro (coordina la comunicación), en tanto que los instrumentos colocados en la planta son esclavos y sólo responden a las preguntas ó comandos del maestro. Es decir, uno o varios aparatos CVMk (Esclavos) pueden conectarse a un ordenador ó PC (Maestro). Mediante este sistema puede lograrse, además del funcionamiento habitual de cada uno de ellos, la centralización de datos en un solo punto. Esto se logra ya que el equipo CVMk,

permite la interconexión con el módulo CVM/RED 485, dicha tarjeta posee salida para la comunicación RS-485. Si se conectan mas de un aparato a una sola línea serie, será preciso asignarle a cada uno de ellos un numero o dirección definido como: Numero de Periférico el cual podrá ser nombrado del 01 al 99, a fin que el ordenador PC central envíe a dichas direcciones las consultas de datos, debido a esta característica se pueden configurar una red tipo Bus. Dicha comunicación incluye dos mensajes:

- Requisición del Maestro.
- Respuesta del Esclavo.

Al utilizar la *Aplicación de supervisión, visualización y control de datos de Cirnet* para Windows (denominada como *SCADA 6.APL*. Este número corresponde a la versión de actualización y se encuentra en el menú de inicio "*Programas*"), se crearan reporte de datos que se almacenaran automáticamente, en el cual no necesita mayor información de protocolos de comunicaciones (esto debido a que todo este procedimiento de configuración se realizo en la parte de diseño), pues éste lo maneja la misma aplicación en forma transparente para el usuario.

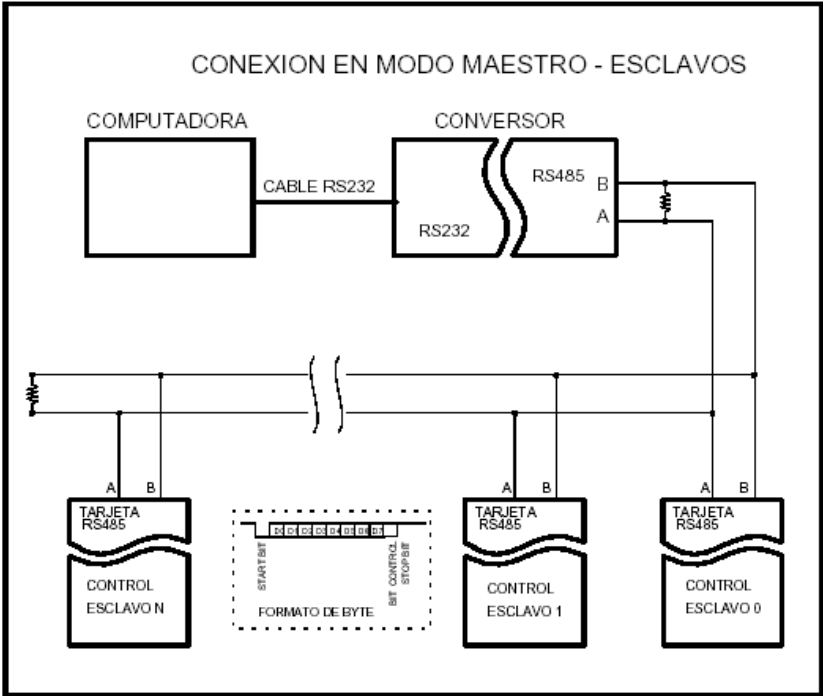


Figura 4.12 Conexionado de dispositivos en un bus RS-485

#### 4.14 Modulo Convertidor Inteligente.

El módulo aislador convertidor inteligente cumple una doble función:

1. Aislar galvánicamente las tierras de computador (PC) y el cable.
2. Convertir los niveles de voltaje del RS-232 a RS-485 y viceversa.

Debe conectarse al PC por un cable RS232 de 9 pines para ambos terminales. En el caso del convertidor inteligente, no será necesario controlar la señal RTS, lo cual es muy útil para usuarios que comuniquen con autómatas o se desarrollen el software de comunicación.

El conversor inteligente es necesario para el software de entorno Windows.

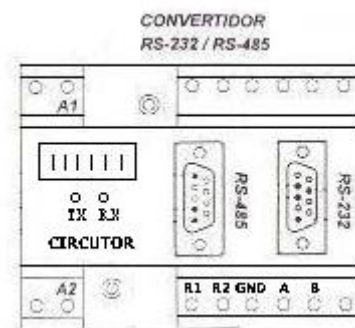


Figura 4.13 Carátula del Convertidor Inteligente RS-232/RS-485.

El convertidor dispone de los siguientes elementos:

- Dos bornes para alimentar el equipo (A1, A2): 220 V  $\pm$  10 %, alimentador de CA.
- Entrada – Salida Serie RS-232 (conector DB-9 hembra).
- Entrada – Salida Serie RS-485 (conector DB-9 macho).

Siendo las características de este elemento:

- El consumo es de 3 VA.
- El rango de velocidad puede oscilar entre: 4800 a 38400 baudios.
- Su estructura de longitud de datos: 10 – 11 bits.
- Se puede puentear entre los terminales R1 y R2 para conectar una resistencia final de línea para poder acoplar las impedancias y evitar así rebotes de la señal.

- Posee también dos bornes para poder conectar directamente el cable de la RS-485 que proviene del equipo CVMK, siendo el Terminal A que corresponde a TX+ y el Terminal B que corresponde a TX-.

Este convertidor posee un MICROSWITCH:



Figura 4.14 Microswitch del Convertidor Inteligente RS-232/RS-485.

Estos interruptores tienen las siguientes funciones:

- A. Longitud de Datos (Se debe seleccionar uno solo)
  - Interruptor 1 (Posición de ON): Selecciona 10 bits.
  - Interruptor 2 (Posición de ON) : Selecciona 11 bits.
- B. Velocidad en Baudios (Se debe seleccionar uno solo):
  - Interruptor 3 (Posición de ON): Selecciona la velocidad de 4800.
  - Interruptor 4 (Posición de ON): Selecciona la velocidad de 9600.
  - Interruptor 5 (Posición de ON): Selecciona la velocidad de 19200.
  - Interruptor 6 (Posición de ON): Selecciona la velocidad de 38400.

En dicho ejemplo se selecciono la longitud de datos de 10 bits, a una velocidad de 9600 baudios.

#### 4.15 Regleta de Conexionado.

El CVMK dispone de una regleta de conexiones en el lateral del aparato para poder conectar la alimentación del equipo y las señales de medición de la red.

Dicha regleta consta de los siguientes bornes:

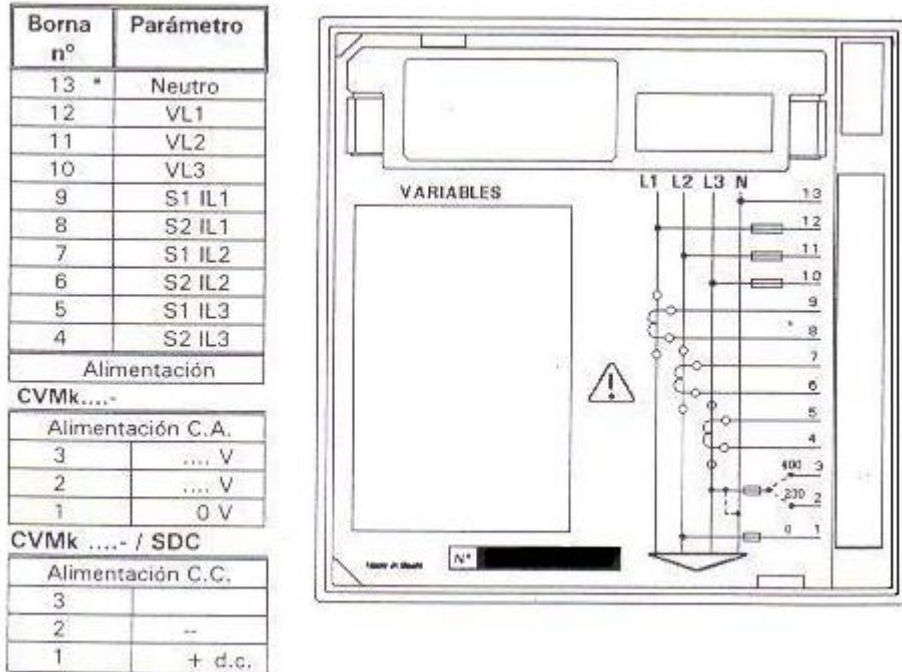


Figura 4.15 Regleta de conexiones del analizador de redes Circutor.

#### 4.16 Esquema de Conexión del CVMK.

A continuación se muestra el esquema de conexión del CVMK en una Red Trifásica de baja tensión:

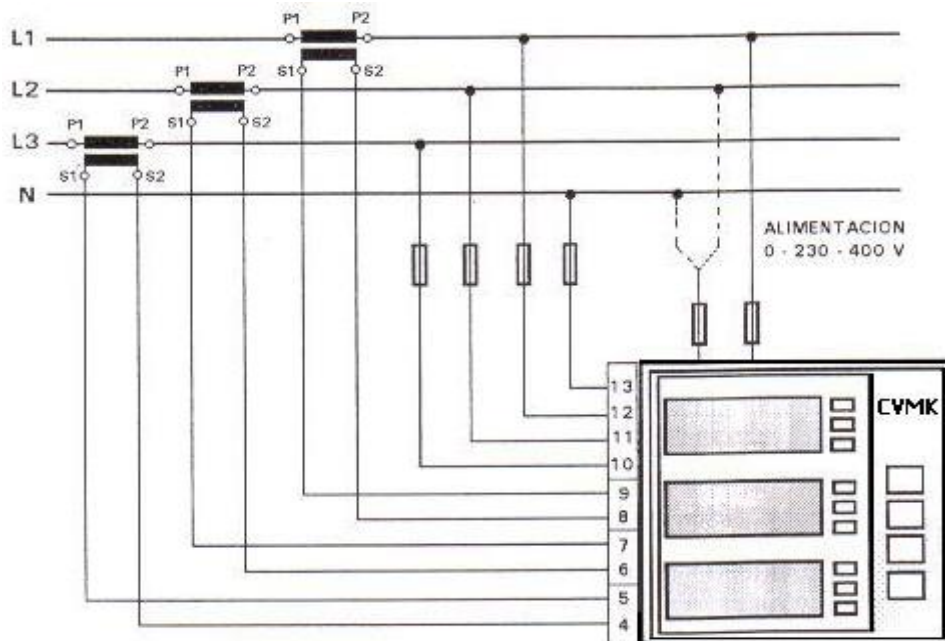


Figura 4.16 Esquema de conexión a TC's del analizador de redes Circutor.



#### 4.17 Conexión RED RS-485 a un Ordenador PC (Mediante la interfaz serial RS-232).

La conexión RS-485 se realiza con cable de comunicación de par trenzado con malla de apantallamiento, de tres hilos como mínimo, con una distancia máxima entre el CVMK y el último periférico de 1200 metros. El CVMK con el módulo CVM/RED (Tarjeta de expansión) utiliza una línea de comunicación RS-485 en la que pueden conectarse hasta un máximo de 32 equipos en paralelo (Mediante el Bus Multipunto) por cada COM del ordenador utilizado.

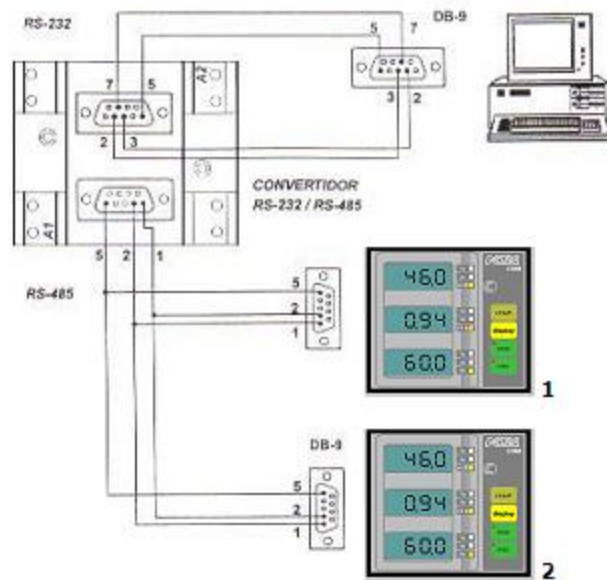


Figura 4.17 Esquema de conexiones de una red de analizadores de redes Circutor.

#### 4.18 Especificaciones Técnicas de los analizadores de redes eléctricos. CIRCUTOR.



Figura 4.18 Analizador de redes CVMk.

#### 4.18.1 Descripción de analizador de redes CVMk.

- Es un instrumento de panel de dimensiones (144 x 144 mm).
- Medición en verdadero valor eficaz.
- Memorización de los valores máximos y mínimos.
- Visualización de los parámetros con escala automática de unidades.
- Display de LCD ó LED (tipos CVMk ó CVMk- L).
- CVMk y CVMk-ITF: Display cristal líquido, 4 dígitos (LCD)
- CVMk L y CVMk L - ITF: Display de LED (4 ½ dígitos)
- CVMk-H: Mide THD y el resto de parámetros de un CVMk estándar
- Teclado de membrana, con 4 teclas, para el control y programación.
- Dispone de 3 x 3 LED luminosos (rojo, verde y amarillo), para indicar el parámetro visualizado en display.
- Posibilidad de comunicación RS-232 ó RS-485 (protocolo CIRCUTOR ó MODBUS ©).

El CVMk permite la visualización de hasta 30 parámetros eléctricos (43 parámetros mediante módulos expansión), mediante 3 display numéricos de grandes dimensiones. En los display se visualiza: (Display 1) La tensión simple o compuesta de las tres fases. (Display 2) 3 parámetros a elegir de los 30 posibles. (Display 3) 3 parámetros a elegir de los 30 posibles.

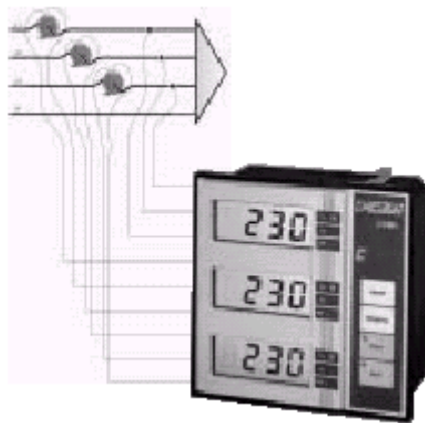


Figura 4.19 Modo de conexión del dispositivo CVMk.

### 2.18.2 Mediciones Eléctricas

La medida se realiza en verdadero valor eficaz, mediante tres entradas de tensión CA y tres entradas de corriente CA La medida se realiza con transformadores de corriente ... / 5 A. (entradas aisladas en el tipo ITF).

### 2.19 Costos del Proyecto.

Los costos del proyecto se estiman en *costos indirectos*, los cuales involucran a los equipos e infraestructura utilizada, pero que pertenece a la entidad que permitió el desarrollo de la aplicación (Universidad Don Bosco); y *costos directos*, los cuales están relacionados con la inversión económica hecha por los interesados en desarrollar la aplicación (incluyendo materiales, equipo y las horas-hombre invertidas en diseño, desarrollo e implementación de la aplicación).

MATERIALES, EQUIPO Y MANO DE OBRA				
Cotización: CSH S.A. de C.V.				
No.	Descripción	Unidades	Precio Unitario	Precio Total
<b>Costos Indirectos</b>				
1	Software CIRNET para Windows (Versión 2.2)	1	\$1,287.54	\$1,287.54
2	Analizadores de Redes CVMk	2	\$493.17	\$986.34
3	Convertidor Inteligente RS 232/485	1	\$300.34	\$300.34
4	PC con características mínimas	1	\$150.00	\$150.00
			<b>Subtotal 1</b>	<b>\$2,724.22</b>
<b>Costos Directos</b>				
5	Módulos de Expansión CVM/RED	1	\$193.67	\$193.67
6	Cable para RS-485 p/metro	150	\$0.20	\$30.00
7	Cable para RS-232 p/metro	20	\$0.50	\$10.00
8	Horas-hombre invertidas en el diseño, desarrollo e implementación de la Aplicación	500	\$5.00	\$2,500.00
			<b>Subtotal 2</b>	<b>\$2,733.67</b>
9	Suministro e Instalación (15% de Subtotal 2)	1		\$410.05
10	Impuesto al valor agregado (IVA 13%)	1		\$355.38
<i>Costos Directos+Suministro+Instalación+Impuestos</i>			<b>TOTAL</b>	<b>\$3,660.26</b>
<b>GRAN TOTAL</b>			<b>\$6,384.48</b>	

Tabla 4.6. Costos de la Aplicación.

Obviamente que para una compañía en particular el costo de inversión en equipos y software puede variar, debido a que estas pueden tener parte del equipo instalado, o ningún equipo que verifique el estado del sistema que quiere supervisarse. Por esta razón el costo de inversión dependerá de la cantidad de equipos que conformen la red, distancias entre los mismos, etc. Por lo tanto estos costos variaran con respecto a los costos presentados con anterioridad.

## CONCLUSIONES.

- A. Se ha podido comprobar que la aplicación realizada posee las características planteadas en el anteproyecto, es decir, que se ha logrado explotar al máximo el software *Cirnet*. La aplicación también posee las herramientas suficientes para dictaminar el estado del sistema, completamente, con respecto al total de variables que supervisa, pero existen algunas limitantes en cuanto a variables de energía y demanda, ya que el equipo con que se cuenta solo permite verificar estas variables solo en una fase del sistema trifásico.
- B. Se tuvo que solventar diversas circunstancias relacionadas con el desarrollo de la aplicación, por ejemplo toda la aplicación incluyendo el sistema supervisorio y la herramienta en Access ha tenido que desarrollarse trabajando en línea siempre con el sistema de potencia. Esto ha incluido otro tipo de limitaciones relacionadas con el tiempo y espacio.
- C. El sistema ha sido desarrollado de tal manera que desde sus inicios puede trabajar en red Intranet, para lo cual se han aprendido técnicas relacionadas con el desarrollo de redes de computadoras, configuración de dispositivos de red, periféricos, etc. Pero básicamente esto ha permitido solventar problemas con sistemas operativos que proporcionaron conflicto con el software *Cirnet*, debido a su plataforma de soporte. Todos los problemas técnicos relacionado con esto se ha solventado para que la aplicación trabaje satisfactoriamente.
- D. Se han podido observar otros sistemas SCADA que poseen características similares al sistema que se ha desarrollado. Este desarrollo también ha estado referido a las herramientas que el software ha permitido llevar a cabo permite y en ese sentido poner a disposición del futuro usuario las máximas posibilidades de monitoreo que este sistema puede dar.

## RECOMENDACIONES.

A continuación se presentan algunas recomendaciones que de tomarse en cuenta añadirían mayor versatilidad al sistema y mejorarían algunos aspectos.

- A. El uso de tarjetas apropiadas para la conexión de varios nodos SCADA es indispensable. Se recomienda como mínimo una tarjeta de expansión o modulo CVM/ER-485, aunque para mejores resultados se ha especificado el uso de tarjetas de expansión CVM/RED ó REDMAX, las cuales no solo permiten la comunicación a larga distancia (1500 m) sino que también añaden un numero mayor de variables.
- B. El personal a cargo del sistema supervisorio debe estar capacitado en el uso de la aplicación en *Cirnet* y las demás herramientas que esta aplicación necesita. Específicamente, conceptos básicos sobre redes de computadoras, MS Office, y medición en sistemas de potencia.
- C. Se recomienda el uso de un sistema operativo que no cree conflictos con *Cirnet*. Estos pueden sistemas operativos hasta Windows 98, Millenium o incluso algunas versiones de Windows XP, ya que plataformas como Windows NT Profesional, poseen un sistema de archivos diferentes al que maneja *Cirnet*, que son plataformas con soporte de archivos Win 32, (debido a que algunos sistemas operativos que manejan sistemas de archivos de tipo NTSF crean conflicto y evitan que la aplicación SCADA trabaje adecuadamente).
- D. La forma de conexión y configuración de la red SCADA actual esta basada en el aprovechamiento de los puertos de comunicación de la computadora (Terminal). Esto evitó la inversión en más tarjetas que pudieran aumentar las distancias de comunicación entre los equipos Circutor y la Terminal. Cada puerto de comunicación serial (Comm 1, 2, ..., n) puede tener conectados 32 equipos en red, de la misma forma en que se ha conectado el CVM 96 SP, por lo tanto en la Terminal actual se pueden conectar hasta 64 equipos CVM. Con la única desventaja que cada uno de ellos debe

poseer un modulo de red o comunicación que permita la conexión por medio de la interfase RS-485 (módulos CVM: ER-485, RED, REDMAX, etc.).

- E. Se debe tener el cuidado de conectarse a los equipos analizadores de redes Circutor sin cambiar la ultima configuración con que se les interrogó, ya que estos guardan la configuración en su memoria, provocando que si se quiere acceder a dicho equipo con una configuración diferente el equipo se bloquee y no permita la adquisición de datos. Este problema puede darse en redes nómadas, es decir redes que están siendo montadas y desmontadas constante, agregando o quitando equipo, etc. Por lo tanto la configuración de ultimo acceso se convierte en una contraseña para poder acceder desde otro equipo o Terminal (PC).
- F. Como toda herramienta software, la herramienta creada necesita soporte técnico, es decir, que cada cierto tiempo se debe verificar que el sistema opere adecuadamente. Verificar las mediciones del equipo con respecto a las monitoreadas a través de la interfaz, actualización de equipos, software, etc.
- G. Debe procurarse que los equipos (terminales) que sean montados inicialmente para el soportar el sistema de monitoreo, posean las características mínimas propuestas, y debe preverse que serán equipos (computadoras) que permanecerán operando varias horas al día, por lo cual debe proveérseles un ambiente adecuado para evitar que se dañen prematuramente. Así mismo, a medida se vaya extendiendo la red los equipos deberán cumplir con las especificaciones mínimas, las cuales se refieren a la correcta operación de los diferentes programas con que cuenta la aplicación.

GLOSARIO.

**Alarma:** Mensaje que notifica a un operador o administrador que existe un problema en los elementos de red administrados.

**Carga:** El término carga, en el lenguaje habitual de la electrotecnia, puede tener varias acepciones:

- Conjunto de valores eléctricos que caracterizan la solicitud a que está sometido un equipamiento eléctrico (transformador, máquina, etc.).
- Equipamiento eléctrico que absorbe potencia.
- Potencia (o corriente) transferida por un equipamiento eléctrico.
- Potencia instalada.

Por otra parte, para un circuito o equipamiento eléctrico hablamos de funcionamiento en carga cuando está consumiendo potencia y funcionamiento en vacío en caso contrario.

**DDE (Dynamic Data Exchange):** protocolo que permite interactuar entre aplicaciones dentro del sistema operativo Windows; actúa concretamente en modo cliente (solicita datos a otras aplicaciones), o en modo servidor (suministrar datos a otras aplicaciones).

**Demanda:** Es la carga en las terminales receptoras, tomada como valor medio en un intervalo de tiempo determinado; por lo tanto, hablar de demanda carece de sentido si no se explicita dicho intervalo. Se expresa en kW, kVA o Amperes.

**DCS (Distributed Control Systems):** Son sistemas de control que geográficamente se encuentran alejados pero que realizan una función común.

**Factor de potencia:** Es la relación entre la potencia activa o efectiva y la potencia aparente de una instalación, es decir, que:

$$\text{Factor de potencia} = P/S = \cos \Phi$$



**Interfaz:** cable de un número determinado de hilos que permite la comunicación entre los diferentes equipos que conforman una red, ya sea conectándolos directamente (RS 232) o por medio de periféricos llamados convertidores de señal (RS 485 a RS 232), y limitadas por ciertas distancias normadas internacionalmente.

**Periféricos:** permiten el enlace entre las diferentes interfaces para poder establecer la comunicación entre los distintos equipos. Funcionan como amplificadores de señales digitales.

**Medir:** Se refiere a verificar el estado presente o continuo de las variables en el funcionamiento de una planta.

**MDI (Multiple Documents Interface):** Interfase de múltiples documentos, es la que permite una mejor visualización de los diferentes procesos supervisados por el SCADA, la cual es característica de los programas basados en plataforma Windows.

**Potencia Aparente (S):** Se expresa en voltamperios (VA) y se determina a partir de las indicaciones de un amperímetro y un voltímetro. Está expresada por:  $S = V I$ . Cuando se trata de una instalación de corriente alterna trifásica, la potencia aparente se calcula por la fórmula:  $S = \sqrt{3} V I$ .

**Potencia Activa (P):** Llamada también potencia efectiva o potencia real, que se expresa en vatios (W) y se mide directamente con vatímetro. Solamente esta potencia se puede transformar en potencia mecánica o en potencia calorífica y, por consiguiente, solamente esta potencia es la que interesa al usuario. Esta expresada por  $P = V I \cos \phi$ . Si se trata de una instalación de corriente alterna trifásica, la potencia activa está expresada por  $P = \sqrt{3} V I \cos \phi$ .

**Formulario:** ventana de visualización que se genera en aquellos programas que permiten la manipulación de bases de datos.

**Potencia Reactiva (Q):** Llamada también potencia magnetizante, que se expresa en voltamperios reactivos (VAR) y que se calcula a partir de las otras potencias. Resulta necesaria para el funcionamiento de ciertas máquinas y dispositivos eléctricos (motores, transformadores, bobinas, relés, etc.) pero no puede transformarse en potencia mecánica o calorífica útil, esta expresada por  $Q = V I \text{ Sen } \phi$ . En el caso de una instalación de corriente alterna trifásica, la potencia reactiva esta expresada por  $P = \sqrt{3} V I \text{ Sen } \phi$ .

**SCADA (Supervisory Control And Acquisition Data):** sistema de control de lazo cerrado, el cual posee las características de controlar de forma supervisoria y por medio de la adquisición de datos que luego pueden ser almacenados como registros del estado de un sistema determinado. En estos sistemas el operario desempeña una función fundamental: cierra el lazo de retroalimentación del sistema de control.

**Servidor:** Computadora que ofrece sus prestaciones a varios ordenadores clientes conectados a una red.

**SSL (Security Socket Layer):** protocolo de seguridad que permite tener un acceso controlado y supervisado de un sitio de Internet en particular, el cual posee información que no puede ser divulgada o verificada por cualquier usuario de Internet, sino solo por aquellos que posean una llave de acceso.

**Telemetría:** Es una manera avanzada y precisa de verificar el estado de una planta a distancia.

**Script:** párrafos de programación que se adicionan a los comandos de las distintas funciones SCADA en el sistema, para que ejecuten ciertas acciones, por ejemplo las alarmas se programan dentro de un rango de valores por medio de un script.

**Bus:** infraestructura cableada en el cual están conectados diferentes dispositivos que pueden conformar una red.

**Comm:** puerto de comunicación serial que poseen los computadores, por medio de los cuales se puede establecer comunicación con diversos equipos, por medio de interfaces y periféricos adecuados.

**LAN:** red de área local (Local Area Network), son redes creadas para trabajar en cierta región o espacio limitado.

**WAN:** red de área amplia (Wide Area Network), redes que poseen una mayor cobertura que las de área local, dentro de las cuales se encuentra la red Internet.

**Master:** del ingles maestro, es el dispositivo que funciona como servidor en una red. Interroga y estimula la comunicación, es el receptor.

**Slave:** del ingles esclavo, dispositivo que es interrogado y no actúa sino existe un estímulo por parte del maestro.

## BIBLIOGRAFÍA.

- **MANUAL DE USUARIO DE MS ACCESS.**  
Microsoft Press 2000.
- **DEFINICIONES Y HERRAMIENTAS DE CIRNET PARA WINDOWS (ELECTRICAL CONTROL AND MESURING SOFTWARE).**  
Circutor 1995-1998. Barcelona, España.
- **DISEÑO E IMPLEMENTACIÓN DE SISTEMA DE MONITOREO REMOTO A TRAVÉS DE INTERNET, DE UNIDADES DE POTENCIA ININTERRUMPIDA INSTALADAS POR LA EMPRESA FASOR S.A. DE C.V.**  
Trabajo de graduación para optar al grado de Ingeniero Electrónico presentado por: José E. Alvarado Ayala y José R. Soriano Velásquez  
Octubre de 2002. Universidad Don Bosco.
- **SISTEMAS OPERATIVOS** Stallings, William.  
Prentice-Hall, 2da edición.
- **IEEE POWER & ENERGY MAGAZINE.**
- **INTERNET.**

# ANEXOS

PHP CON ALGUNAS APLICACIONES COMUNES, CÓDIGOS DE PROGRAMACIÓN EN  
ACCESS Y PHP, Y ACERCA DE HTML.

## **PHP CON ALGUNAS APLICACIONES COMUNES**

## Instalación de PHP

PHP Es un lenguaje de programación orientado a servidor, es decir, que nuestra máquina no realiza ninguna operación de compilado, la máquina encargada de esto es el servidor web en el que esté puesta nuestra página; es más si miramos el código de la página en nuestro explorador no vamos a ver nada de php, porque lo que hace php es escribir html (entre otras cosas)

PHP es un lenguaje ídneo para el manejo de base de datos MySQL, y se pueden llegar a hacer verdaderas maravillas con la combinación de estas dos herramientas, las principales características de PHP son:

- Manejo ultra sencillo de las bases de datos
- Creación de imágenes dinámicas
- Hace que la creación de una página web sea mucho más rápida y mucho más sencilla
- La creación de carritos de la compra, entre otras cosas es super sencilla.
- Y otras muchas cosas más que se me olvidan

1º - Bajarse el PHP: La página principal de PHP es <http://www.php.net> ; pero como siempre a lo interesante; podrás bajar el PHP de [aquí](#)

2º - La instalación de PHP es muy sencilla, una vez bajado hay que descomprimirlo; una vez descomprimido en c:\ tendremos una carpeta que se llame c:\php-4.2.3-Win32, lo mejor es cambiarle el nombre, la llamaremos c:\php a secas.

3º - Bien, ahora tenemos una carpeta que se llama c:\php con el contenido de php; lo siguiente que tenemos que hacer es entrar en esa carpeta y copiar el archivo php.ini-dist a nuestro directorio de Windows (normalmente c:\windows) con el nombre php.ini; el archivo debe quedar así: c:\windows\php.ini

4º - La instalación básica de PHP ya ha finalizado, ahora hay que darle soporte a Apache para PHP, y hacer algunas modificaciones en el archivo php.ini para que todo funcione correctamente

## Modificaciones en Apache para que soporte PHP

5º - Para que Apache soporte PHP tenemos que abrir el archivo httpd.conf (se encuentra en c:\apache\apache2\conf\ ) y añadir al final del mismo este texto:

```
ScriptAlias /php/ "c:/php/"  
AddType application/x-httpd-php .php  
Action application/x-httpd-php "/php/php.exe"
```

Si tenemos iniciado nuestro servidor Apache, lo tenemos que reiniciar para que soporte PHP, ahora mismo nos funcionaría PHP en nuestro servidor web (que caña eh?), pero es mejor hacer algunas modificaciones en el php.ini para evitar futuros dolores de cabeza.

## Modificaciones en el php.ini

1º - Lo primero, abrir el archivo php.ini (está en c:\windows\php.ini) y buscar el texto que pone:

```
; You should do your best to write your scripts so that they do not require  
; register_ to be on; Using form variables as globals can easily lead  
; to possible security problems, if the code is not very well thought of.  
register_globals = Off
```

Para que cuando vayamos a usar los métodos POST y GET funcionen correctamente es necesario poner:

```
register_globals = On
```

2º - La segunda y última modificación que vamos a hacer sobre el php.ini es esta; buscar este texto:

```
[mail function]
```



; For Win32 only.

SMTP =

Bien vamos a modificar el valor de smtp poniendo el servidor de correo saliente que tengamos, por ejemplo:

SMTP = mailhost.terra.es

## 1. Introducción

Aunque PHP y ASP tienen básicamente la misma funcionalidad, su sintaxis es completamente distinta y su semántica también presenta diferencias, por lo que aunque lo que podamos hacer con ASP también lo podamos hacer en PHP (y viceversa), la adaptación de uno a otro lenguaje puede resultar algo complicada al principio.

ASP ("Active Server Pages") es, más que un lenguaje de programación en sí, una tecnología que permite insertar en una página HTML código que se ejecutará en el servidor. Este código puede ser Java, JavaScript o, más comúnmente, VBScript. Por lo tanto cuando en este curso hagamos referencia al "lenguaje ASP", y no a la tecnología, nos referiremos a VBScript.

Por su parte, PHP ("PHP: Hypertext Preprocessor") provee de una tecnología similar a ASP para insertar código en las páginas HTML, pero PHP además de la tecnología es el lenguaje. En PHP sólo podemos incluir código en un lenguaje, PHP, que será el lenguaje que vamos a estudiar en el presente curso.

PHP es un lenguaje interpretado basado principalmente en C, C++ y Java, con los que comparte prácticamente toda su sintaxis y semántica, y aporta también algunas características de lenguajes interpretados como Perl y Bash. Debido a esto, una de sus principales características y una gran baza a su favor es que la curva de aprendizaje para programadores que ya conozcan estos lenguajes es muy suave, prácticamente pueden sentarse delante del ordenador y comenzar a escribir código.

En cuanto a la tecnología detrás de PHP, ya en la versión 3.0 el intérprete de PHP era bastante más rápido que los intérpretes existentes de ASP, lo que junto con su buena integración con el servidor HTTP Apache y su capacidad de acceder a unos 20 sistemas de Bases de Datos distintos, lo ha convertido en un fuerte competidor frente a las “soluciones” de Microsoft. Con la versión 4.0 de PHP que ha visto la luz hace pocos meses la situación ha mejorado todavía más: el intérprete es más rápido (hasta 12 veces más rápido que el de la versión 3.0); se ha perfeccionado la integración de PHP con otros servidores además de Apache, entre otros IIS; y se ha modularizado todo el diseño interno, entre otras cosas independizado el intérprete del lenguaje PHP (Zend) del módulo de comunicación con el servidor, con lo que a partir de ahora es posible utilizar PHP como lenguaje interpretado en cualquier otro proyecto (ya se está trabajando para utilizar PHP como lenguaje para procedimientos en MySQL.)

El presente curso está estructurado en dos secciones: En la primera se introduce la sintaxis y semántica básica de PHP, haciendo especial hincapié en sus diferencias con ASP; En la segunda parte veremos más a fondo algunas de las características de PHP con las que tendremos que trabajar. Por último, enumeraremos algunas características más de PHP que nos hemos dejado en el tintero.

## 2. El lenguaje PHP

PHP es bastante diferente en su sintaxis a ASP. ASP está basado en VBScript, mientras que PHP toma bastantes características de lenguajes como C, C++ y Java, y algunas también de Perl.

A continuación analizaremos en detalle estas diferencias al tiempo que vamos introduciendo las bases de la programación en PHP.

### 2.1. Extensión de los ficheros

La primera diferencia entre ASP y PHP viene a la hora de dar nombre a los ficheros. La extensión es importante ya que el servidor HTTP (en nuestro caso, generalmente Apache) decide si debe pasárselo al procesador de PHP o no en función de esta extensión.

Las extensiones que indican al servidor HTTP que el fichero contiene código PHP que debe ser procesado son:

- .php3 Indica código PHP 3.x.
- .php4 Indica código PHP 4.x.
- .php Indica código PHP. Preferiremos esta extensión por ser más genérica.
- .phtml Actualmente en desuso.

Aunque haya extensiones diferentes para PHP 3 y 4, no tienen efecto en el intérprete que las procesará IGUAL. En general, PHP 4 es compatible con 3, salvo unas pocas excepciones. Este curso se centra en PHP 4, así que no analizaremos estos casos de incompatibilidad con las versiones anteriores.

### 2.2. Delimitadores

El código ASP se delimita dentro del HTML entre los signos '<% ' y %>'. Por ejemplo:

```
<%@ Language=VBScript %>
<html>
<body>
<%IF Hour(time)>20 OR Hour(time)<4 THEN%>
Buenas noches.
<%ELSE%>
Buenos días.
<%END IF%>
</body>
</html>
```

En PHP disponemos de cuatro opciones para delimitar el código:

```
<? echo 'Primer método de delimitar código PHP'; ?>
<?php echo 'Segundo método, el más usado'; ?>
<script language="php">
echo 'Algunos editores (como el FrontPage) sólo entienden este
método';
</script>
<% echo 'Método de compatibilidad con ASP'; %>
```

Los métodos primero y cuarto no siempre estarán disponibles, ya que se pueden activar o desactivar al compilar el intérprete de PHP. En general, preferiremos usar el segundo método.

El ejemplo anteriormente expuesto en ASP podría traducirse así a PHP:

```
<html>
<body>
<?php if ( Hour(time)>20 || Hour(time)<4) { ?>
Buenas noches.
<?php } else { ?>
Buenos días.
<?php }?>
</body>
</html>
```

O también así, utilizando la función *echo* para imprimir cadenas de texto:

```
<html>
<body>
<?php
if ( Hour(time)>20 || Hour(time)<4) {
    echo "Buenas noches.";
} else {
    echo "Buenos días.";
}
?>
</body>
</html>
```

Además, se puede imprimir de forma rápida con `<%= $var %>`, al igual que en ASP.

## 2.3. Fin de línea

En ASP el carácter de fin de línea representa el final de un comando, llamada a una subrutina, etc. En cambio PHP ignora cualquier carácter de espaciado presente en el código,

incluyendo espacios, tabuladores y saltos de línea (salvo si se encuentran dentro de una cadena de texto). El fin de sentencia se marca con un carácter de punto y coma.

Por ejemplo, los dos trozos de código siguientes son equivalentes:

```
print( date("M d, Y H:i:s", time()) );

print (
    date( "M d, Y H:i:s",
          time()
        )
    )
;
```

## 2.4. Comentarios

En PHP hay tres formas de introducir comentarios en el código, frente al ' de ASP:

```
/* Comentarios estilo C.
 * Pueden extenderse durante varias líneas.
 */

// Comentarios estilo C++. Cubren hasta el final de la línea.

# Comentarios estilo Bash/Perl. Cubren hasta el fin de línea.
```

Hay que hacer notar que los comentarios de una línea cubren hasta final de línea O HASTA EL FIN DEL BLOQUE PHP ('>' o el que corresponda.)

## 2.5. Variables

Las variables son una parte fundamental de todo lenguaje de programación. En ellas se almacenan valores con los que se puede operar, se pueden comparar entre sí y se puede hacer variar el flujo del programa en función de su valor. Vamos a ver cómo trabajar con variables en PHP.

### 2.5.1. Declaración y uso

Al contrario que en la mayoría de lenguajes de programación, y al igual que en ASP, en PHP NO hace falta declarar las variables antes de usarlas: tras la primera aparición en el código quedan declaradas. Como diferencia frente a ASP, en PHP todas las variables llevan delante el signo del dólar '\$'. Ej. :

```
$var_1 = 123;
```

```
$var_2 = 'hola';  
$var_3 = $var_1 * 2;
```

## 2.5.2. Tipado de variables

Las variables en PHP están débilmente tipadas, de forma similar a ASP (el tipo mixto *Variant* de ASP es similar al *mixed* de PHP.) Una variable se asocia a un contenido, del tipo que sea, estando el tipo indicado en el contenido, no en la variable en sí. Visto de otra forma, una misma variable se puede reutilizar asignándole a lo largo del tiempo datos de distinto tipo. Por ejemplo:

```
$mi_variable = 'Inicializamos como una cadena de texto';  
$mi_variable = 3; // Ahora es un entero.  
$mi_variable = 3.14 * $mi_variable; // Ahora un float.  
$mi_variable = new MiClase(); // Ahora un objeto.
```

A pesar de esto, el tipo es importante en algunos casos (para efectuar operaciones o llamadas a funciones.) En estos casos, PHP se encarga de realizar las transformaciones necesarias de forma automática.

```
$mivar = 123;  
echo $mivar;
```

En el ejemplo anterior, PHP convierte el valor entero 123 a la cadena de texto “123” antes de pasárselo a la función echo.

```
$mivar = '3';  
$mivar = 2 + $mivar;
```

En este caso *\$mivar* comienza almacenando la cadena “3”. En la segunda línea, y para realizar la suma, se convierte al entero 3, se realiza la suma y se asigna a *\$mivar* el entero 5.

También se puede forzar la conversión a un tipo específico, por ejemplo:

```
$mivar = (string)123;
```

Y se puede cambiar el tipo de una variable con:

```
$mivar = 12;  
settype($mivar, "double");
```

## 2.5.3. Ámbito

El ámbito de una variable hace referencia a dónde está disponible esa variable y dónde no, y depende del contexto en el que haya sido definida la variable:

- En el cuerpo de un fichero, las variables son GLOBALES al fichero y a cualquier código que se haya incluido con los comandos “include” o “require” (explicados más adelante.)
- En una función, son LOCALES a esa función y no pueden ser accedidas desde fuera.
- Dentro de una clase, sólo pueden ser accedidas a través del operador “->” sobre el nombre del objeto (las clases se explicarán con detalle más tarde.)

Aunque todos estos conceptos (*includes*, *funciones*, *clases*) los presentaremos más tarde, introduciremos ya una posible fuente de errores: Los programadores acostumbrados a C deben llevar cuidado con las variables globales, ya que no se comportan como en C. Por ejemplo, considérese el código siguiente:

```
$mivar = 3;
function mifuncion() {
    echo $mivar;
}
```

En C, la función *mifuncion()* imprimiría '3'. En cambio en PHP no imprimiría nada (si no da un error) ya que la variable *\$mivar* de *mifuncion* es una variable DISTINTA a la global y local a esta función.

Para acceder a las variables globales desde una función, hay que utilizar la palabra reservada *global*:

```
$mivar = 3;
function mifuncion() {
    global $mivar;
    echo $mivar;
}
```

## 3.8. Acceso a Bases de Datos

PHP dispone de varias funciones para acceder a Bases de Datos, pero son propias de cada una de estas Bases de Datos, es decir, no hay una función “conectar” genérica, si no una para conectar a MySQL, otra para conectar a Informix, etc. Para evitar confusiones entre las funciones de unas BD y otras, el nombre de todas estas funciones sigue este patrón:

<nombreBD>\_<nombreFUNCION>

de forma que tenemos una función *mysql\_connect*, una función *ifx\_connect*... Todas las funciones con el mismo nombre de función pero para distintas BD comparten la misma funcionalidad y parámetros. Lo único que tendremos que comprobar es que una

determinada función esté disponible para la BD que vamos a atacar, ya que dependiendo de la BD y de su grado de integración con PHP tendremos más o menos funciones.

Vamos a explicar las funciones más importantes, con ejemplos en MySQL ya que es la BD que mejor integrada está en PHP. Se puede obtener una lista completa de todas las opciones en el manual de PHP:

<http://www.php.net/manual/ref.mysql.php>

o en las páginas de funciones de las otras BD.

### 3.8.1. Abrir y cerrar una conexión

Para conectar a una BD tenemos la función *mysql\_connect* con la siguiente definición:

```
int mysql_connect ([string servidor [:puerto] [:/camino/al/socket]
                  [, string usuario [, string contraseña]])
```

Todos los parámetros son opcionales. Si no se indican, se utiliza "localhost" como servidor por defecto, el nombre del usuario que esté ejecutando PHP como usuario, y la cadena vacía como contraseña. La función devuelve un identificador de la conexión que necesitaremos más tarde para hacer referencia a este enlace en concreto con la BD, o un error si algo ha ido mal.

La conexión con la BD se cierra automáticamente al finalizar la ejecución del fichero actual. También se puede cerrar explícitamente con *mysql\_close*:

```
int mysql_close ([int identificador])
```

Esta función cierra la conexión indicada por el *identificador* o la última conexión que se haya abierto si se omite, y devuelve TRUE o FALSE según si la operación ha finalizado con éxito o no. Por ejemplo, así nos conectaríamos a la BD MySQL en el servidor linux:

```
$link = mysql_connect ("www.mmlabx.ua.es", "nobody", "");
if (!$link) {
    die ("No se pudo conectar");
}
print ("Conexión realizada");
mysql_close ($link);
```

Con algunas BD también se pueden usar enlaces persistentes que no se cerrarán automáticamente al acabar la ejecución del fichero, si no que permanecerán abiertos y se podrán utilizar desde los siguientes ficheros .php que se ejecuten. Estos enlaces se abren y cierran con *mysql\_pconnect* y *mysql\_pclose*. Su definición es idéntica a la de *connect* y *close*:

```
int mysql_pconnect ([string servidor [:puerto] [:/camino/al/socket]
                   [, string usuario [, string contraseña]])
```



```
int mysql_pclose ([int identificador])
```

### 3.8.2. Elegir una BD

Para elegir la base de datos con la que queremos trabajar de entre todas las disponibles se utiliza la función *mysql\_select\_db*:

```
int mysql_select_db (string nombre_bd [, int identificador])
```

Aquí, *nombre\_bd* especifica el nombre de la BD a la que queremos acceder, y el campo opcional *identificador* es el número de identificación de enlace devuelto por *mysql\_connect*. Si se omite este último campo, se utiliza el último enlace abierto, y si no existiera, se trata de crear uno con *mysql\_connect* (sin parámetros). La función devuelve TRUE o FALSE, según si todo ha ido bien o no.

Para elegir la BD “prueba” utilizada en el sistema de gestión de asignaturas en nuestro servidor Linux, utilizaríamos:

```
if (!mysql_select_db("prueba", $link)) {  
    die ("No existe la BD");  
}
```

### 3.8.3. Interrogar a la BD

Para hacer queries a una BD se utiliza *mysql\_query*:

```
int mysql_query (string query [, int identificador])
```

En *query* va la sentencia en SQL que queremos ejecutar. Al igual que con *mysql\_select\_db*, si se omite el *identificador* se usa el último o se trata de crear. Esta función devuelve FALSE si ha habido un error, o un valor no negativo (TRUE) que identifica el resultado y que más tarde deberá ser tratado para extraer la información necesaria.

Para obtener la información de las asignaturas en la BD de ejemplo:

```
$query = "SELECT codigo, nombre, descripcion, creditos, tipo  
          FROM asignatura";  
$asignaturas = mysql_query($query, $link);  
if (!$asignaturas) {  
    die ("Error en el query");  
}
```

También podemos utilizar esta otra función, en la que además del query indicamos la BD a la que queremos interrogar:

```
int mysql_db_query (string nombre_bd, string query  
                  [, int identificador])
```

Cuando hayamos acabado con el resultado, podemos liberar memoria con:

```
int mysql_free_result (int result)
```

aunque realmente no es necesario ya que el recolector de basura de PHP se encargará de liberarla al acabar la ejecución del fichero actual.

### 3.8.4. Extraer información del resultado del query

Si la última operación fue un INSERT, UPDATE o DELETE (operaciones que *modifican* algún registro), se puede utilizar esta función para ver el número de filas afectadas:

```
int mysql_affected_rows ([int identificador])
```

Para obtener el número de filas en el resultado de un query, tenemos:

```
int mysql_num_rows (int resultado)
```

Donde *resultado* es el valor devuelto por *mysql\_query*.

Para obtener toda una fila del resultado:

```
array mysql_fetch_row (int resultado)
```

Cada llamada sucesiva a esta función devuelve un array con la siguiente fila del resultado, hasta que no queden filas y devuelva FALSE. Las columnas están indexadas por orden en el array, comenzando la primera en la posición 0.

Para este mismo fin también disponemos de otra función, que es una extensión de la anterior:

```
array mysql_fetch_array(int resultado [, int tipo_resultado])
```

Esta función es capaz de indexar el array de forma numérica o asociativa por el nombre de las columnas. El parámetro *tipo\_resultado* puede valer MYSQL\_ASSOC, MYSQL\_NUM, o MYSQL\_BOTH para indicar el tipo de indexado que queremos (por defecto, ambos). Hay que destacar que, a pesar de la funcionalidad extendida sobre *mysql\_fetch\_row*, esta función NO es significativamente más lenta que la anterior.

Continuando con el ejemplo anterior, podemos obtener los resultados del query así:

```
while ($fila = mysql_fetch_array($signaturas)) {  
    echo $fila['codigo'] . ' ' . $fila['nombre'] . "\n";  
}
```

Otra función que podemos utilizar para obtener los resultados de un query es *mysql\_fetch\_object*:

```
object mysql_fetch_object (int result)
```

que en lugar de almacenar el resultado en un array, lo almacena en un objeto cuyas propiedades (variables) coinciden con los nombres de las columnas:

```
while ($fila = mysql_fetch_object($asignaturas)) {  
    echo $fila->codigo . ' ' . $fila->nombre . "\n";  
}
```

Si queremos saber el tamaño de los campos de la última fila devuelta por *mysql\_fetch\_array*, *object* o *row*, tenemos:

```
array mysql_fetch_lengths (int resultado)
```

Esta función devuelve un array con tantos campos como columnas el último resultado, con los tamaños de estas columnas.

Por último, podemos “saltar” directamente a una fila del resultado, o “rebobinar” una vez hemos llegado al final, con la función *mysql\_data\_seek*. Las filas comienzan en el 0.

```
int mysql_data_seek (int resultado, int numero_fila)
```

## 3.9. Acceso a Bases de Datos unificado

Hemos ideado un método para reducir al mínimo las modificaciones en el código al cambiar de Sistema Gestor de Bases de Datos. El método utiliza la POO para definir una clase “virtual” que define un interfaz genérico de acceso a bases de datos, clase de la que luego heredarán este interfaz las clases especializadas en el acceso a cada BD en concreto. En nuestro programa utilizaremos objetos de estas clases especializadas para acceder a las BD, pero como todos ellos compartirán los mismos métodos, la única línea de código que habrá que cambiar será aquella en la que se cree el objeto de una clase (BD) u otra. Y para no tener que modificar esta línea en todos los ficheros, tendremos un fichero en el que se crea el objeto y se incluirá desde todos los demás. De esta forma, tendremos un objeto accesible en todos los ficheros y sólo tendremos que modificar uno para cambiar de BD.

Este es el código de la clase *ConexionGenerica*. Define el interfaz que será comun a todas las clases específicas de cada BD, y provee ya de mensajes de error para el caso de que una clase no defina un método en concreto:

```
<?php
```

```

/*
 * Clase base para definir una conexión genérica a una BD desde PHP.
 * Se deberán definir clases que hereden de esta redefiniendo sus métodos
 * para cada BD a la que queramos acceder.
 *
 * Vicente Aguilar <vaguilar@linuxfreak.com>
 *
 * 7/9/2000
 */

/**
 * CLASE ConexionGenerica
 *
 * Esta clase es una plantilla para definir la estructura básica (variables
 * y métodos) de todas las clases para conectar a distintas BBDD que más
 * tarde heredarán de esta, redefiniendo los métodos.
 */
class ConexionGenerica {

    /** El enlace con la BD */
    var $link;

    /**
     * Constructor. No hace nada. Creo que se podría quitar ...
     */
    function ConexionGenerica() {}

    /**
     * FUNCIÓN connect( servidor, usuario, clave )
     * Conecta con el servidor, identificándose como usuario y clave.
     * No hace nada, debe ser redefinida por las clases que hereden de esta.
     */
    function connect($servidor, $usuario, $clave) {
        echo "<h1>El m&eacute;todo <i>connect</i> no est&aacute; .
            "implementado en la clase <i>" . get_class($this) . "</i></h1>";
        return FALSE;
    }

    /**
     * FUNCIÓN close()
     * Finaliza la conexión con el servidor.
     * No hace nada, debe ser redefinida por las clases que hereden de esta.
     */
    function close() {
        echo "<h1>El m&eacute;todo <i>close</i> no est&aacute; .
            "implementado en la clase <i>" . get_class($this) . "</i></h1>";
        return FALSE;
    }

    /**
     * FUNCIÓN select db( base datos )
     * Elige la base de datos con la que trabajar.
     * No hace nada, debe ser redefinida por las clases que hereden de esta.
     */
    function select db($base datos) {
        echo "<h1>El m&eacute;todo <i>select_db</i> no est&aacute; .
            "implementado en la clase <i>" . get_class($this) . "</i></h1>";
        return FALSE;
    }

    /**
     * FUNCIÓN query( query )
     * Realiza una consulta o actualización en la BD.
     * No hace nada, debe ser redefinida por las clases que hereden de esta.
     */
    function query($query) {
        echo "<h1>El m&eacute;todo <i>query</i> no est&aacute; .
            "implementado en la clase <i>" . get_class($this) . "</i></h1>";
        return FALSE;
    }
}

```

```

/**
 * FUNCIÓN fetch_array( resultado )
 * Realiza una consulta o actualización en la BD.
 * No hace nada, debe ser redefinida por las clases que hereden de esta.
 */
function fetch_array($resultado) {
    echo "<h1>El m&eacute;todo <i>query</i> no est&aacute; implementado en la clase <i>" . get_class($this) . "</i></h1>";
    return FALSE;
}

/**
 * FUNCIÓN free_result( resultado )
 * Libera la memoria ocupada por un resultado.
 * No hace nada, debe ser redefinida por las clases que hereden de esta.
 */
function free_result($resultado) {
    echo "<h1>El m&eacute;todo <i>free_result</i> no est&aacute; implementado en la clase <i>" . get_class($this) . "</i></h1>";
    return FALSE;
}
}
?>

```

Un ejemplo de clase especializada en una BD sería la siguiente para MySQL:

```

<?php
/**
 * Clase para conectar a una base de datos MySQL.
 * Hereda de ConexionGenerica y redefine sus métodos de acceso a la BD.
 *
 * Vicente Aguilar <vaguilar@linuxfreak.com>
 *
 * 7/9/2000
 */

// La clase "plantilla" de la que heredamos
require ("ConexionGenerica.inc");

/**
 * CLASE ConexionMySQL
 * Métodos para conectar e interactuar con una BD MySQL.
 */
class ConexionMySQL extends ConexionGenerica {

    /**
     * Constructor. No hace nada. Creo que se podría quitar ...
     */
    function ConexionMySQL() {}

    /**
     * FUNCIÓN connect( servidor, usuario, clave )
     * Conecta con el servidor, identificándose como usuario y clave.
     */
    function connect($servidor="localhost", $usuario="nobody", $clave="") {
        return $this->link = mysql_connect($servidor, $usuario, $clave);
    }

    /**
     * FUNCIÓN close()
     * Finaliza la conexión con el servidor.
     */
    function close() {
        return mysql_close($this->link);
    }
}

```

```

/**
 * FUNCIÓN select_db( base_datos )
 * Elige la base de datos con la que trabajar.
 */
function select_db($base_datos) {
    return mysql_select_db($base_datos, $this->link);
}

/**
 * FUNCIÓN query( query )
 * Realiza una consulta o actualización en la BD.
 */
function query($query) {
    return mysql_query($query, $this->link);
}

/**
 * FUNCIÓN fetch_array( resultado )
 * Realiza una consulta o actualización en la BD.
 */
function fetch_array($resultado) {
    return mysql_fetch_array($resultado);
}

/**
 * FUNCIÓN free_result( resultado )
 * Libera la memoria ocupada por un resultado.
 */
function free_result($resultado) {
    return mysql_free_result($resultado);
}
}
?>

```

Además de estas dos clases, tenemos otro fichero CreaConexion.inc que es el encargado de crear el objeto de la clase que vayamos a usar:

```

<?php
/**
 * CreaConexion
 * Crea una conexion (variable $conexion) con una base de datos.
 *
 * Vicente Aguilar <vaguilar@linuxfreak.com>
 *
 * 7/9/2000
 */
/*
 * Habra que cambiar estas dos lmeas segzn la BD a la que se quiera
 * acceder y la clase que se quiera utilizar para ello.
 */
require ("ConexionMySQL.inc");
$conexion = new ConexionMySQL();
?>

```

Para acceder a una base de datos, tan sólo tendremos que incluir este último fichero y acceder a las funciones de las BD a través de él, de esta forma:

```

<?php
require ("CreaConexion.inc");

$conexion->connect("localhost", "nobody");
$conexion->select db("prueba", $link);

```

```
$asignaturas = $conexion->query("SELECT codigo, nombre, descripcion,  
                                creditos, tipo  
                                FROM asignatura");  
?>
```

## **4. Características y funciones adicionales**

Muchas características avanzadas de PHP, que en muchos casos en ASP vienen en paquetes a parte por los que hay que pagar más, se nos han quedado en el tintero. Algunos de estas características podrían no venir en la instalación por defecto de PHP, ser opcionales a la hora de compilar o incluso necesitar de paquetes a parte, por lo que podrían no estar disponibles en un sistema concreto.



## CÓDIGO DE PROGRAMACIÓN EN ACCESS

A continuación se presenta el lenguaje de programación que posee la aplicación en MS Access, para el menú principal, submenús y formularios realizados en la aplicación .

## Menu Principal

```
Option Compare Database
Private Sub cmdExcel_Click()
DoCmd.OutputTo acOutputQuery, "qryExpPROMIPAC1", acFormatXLS
DoCmd.OutputTo acOutputQuery, "qryExpPROMIPAC2", acFormatXLS
End Sub
Private Sub Command1_Click()
On Error GoTo Err_Command1_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "frmEncPROMIPAC"
    DoCmd.OpenForm stDocName, , stLinkCriteria
Exit_Command1_Click:
    Exit Sub
Err_Command1_Click:
    MsgBox Err.Description
    Resume Exit_Command1_Click
End Sub
Private Sub Command3_Click()
On Error GoTo Err_Command3_Click
    DoCmd.Quit
Exit_Command3_Click:
    Exit Sub
Err_Command3_Click:
    MsgBox Err.Description
    Resume Exit_Command3_Click
End Sub
Private Sub Command4_Click()
DoCmd.OpenReport "zmrptPrintTables", acViewPreview
End Sub
Private Sub Command3_Enter()
End Sub
Private Sub Detail_Click()
End Sub
Private Sub Form_Open(Cancel As Integer)
DoCmd.Restore
End Sub
Private Sub Comando6_Click()
On Error GoTo Err_Comando6_Click
    Dim stDocName As String
```

```

    Dim stLinkCriteria As String
    stDocName = "MENU SUB1"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Comando6_Click:
    Exit Sub
Err_Comando6_Click:
    MsgBox Err.Description
    Resume Exit_Comando6_Click
End Sub
Private Sub Comando7_Click()
On Error GoTo Err_Comando7_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "MENU SUB2"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Comando7_Click:
    Exit Sub
Err_Comando7_Click:
    MsgBox Err.Description
    Resume Exit_Comando7_Click
End Sub
Private Sub Comando9_Click()
On Error GoTo Err_Comando9_Click
    Dim stAppName As String
    stAppName = "C:\Archivos de programa\Circutor\Cirnet para
Windows\WCIRNET.EXE C:\ARCHIV~1\CIRCUTOR\PROY.APL"
    Call Shell(stAppName, 1)
Exit_Comando9_Click:
    Exit Sub
Err_Comando9_Click:
    MsgBox Err.Description
    Resume Exit_Comando9_Click
End Sub

```

## **MENU- SUB1**

```

Option Compare Database
Private Sub cmdExcel_Click()
DoCmd.OutputTo acOutputQuery, "qryExpPROMIPAC1", acFormatXLS
DoCmd.OutputTo acOutputQuery, "qryExpPROMIPAC2", acFormatXLS
End Sub
Private Sub Command1_Click()
On Error GoTo Err_Command1_Click

```

```

    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "frmEncPROMIPAC"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Command1_Click:
    Exit Sub
Err_Command1_Click:
    MsgBox Err.Description
    Resume Exit_Command1_Click

End Sub
Private Sub Command3_Click()
On Error GoTo Err_Command3_Click
    DoCmd.Quit
Exit_Command3_Click:
    Exit Sub
Err_Command3_Click:
    MsgBox Err.Description
    Resume Exit_Command3_Click
End Sub
Private Sub Command4_Click()
DoCmd.OpenReport "zmrptPrintTables", acViewPreview
End Sub
Private Sub Form_Open(Cancel As Integer)
DoCmd.Restore
End Sub
Private Sub Comando6_Click()
On Error GoTo Err_Comando6_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "DEMANDA Y ENERGIA SUB1"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Comando6_Click:
    Exit Sub
Err_Comando6_Click:
    MsgBox Err.Description
    Resume Exit_Comando6_Click
End Sub
Private Sub Comando7_Click()
On Error GoTo Err_Comando7_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "INTENSIDADES DE CORRIENTE POR FASE SUB1"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Comando7_Click:
    Exit Sub
Err_Comando7_Click:

```

```

    MsgBox Err.Description
    Resume Exit_Comando7_Click
End Sub
Private Sub Comando8_Click()
On Error GoTo Err_Comando8_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "TENSIONES DE FASE SUB1"
    DoCmd.OpenForm stDocName, , , stLinkCriteria

Exit_Comando8_Click:
    Exit Sub
Err_Comando8_Click:
    MsgBox Err.Description
    Resume Exit_Comando8_Click
End Sub
Private Sub Comando9_Click()
On Error GoTo Err_Comando9_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "TENSIONES DE LINEA SUB1"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Comando9_Click:
    Exit Sub
Err_Comando9_Click:
    MsgBox Err.Description
    Resume Exit_Comando9_Click
End Sub
Private Sub Comando10_Click()
On Error GoTo Err_Comando10_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "TENSIONES DE LINEA Y FASE SUB1"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Comando10_Click:
    Exit Sub
Err_Comando10_Click:
    MsgBox Err.Description
    Resume Exit_Comando10_Click
End Sub
Private Sub Comando11_Click()
On Error GoTo Err_Comando11_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "POTENCIAS FASE 1 SUB1"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Comando11_Click:

```

```

Exit Sub
Err_Comando11_Click:
    MsgBox Err.Description
    Resume Exit_Comando11_Click
End Sub
Private Sub Comando12_Click()
On Error GoTo Err_Comando12_Click
    Dim stDocName As String
    Dim stLinkCriteria As String

    stDocName = "POTENCIAS FASE 2 SUB1"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Comando12_Click:
    Exit Sub
Err_Comando12_Click:
    MsgBox Err.Description
    Resume Exit_Comando12_Click
End Sub
Private Sub Comando13_Click()
On Error GoTo Err_Comando13_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "POTENCIAS FASE 3 SUB1"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Comando13_Click:
    Exit Sub
Err_Comando13_Click:
    MsgBox Err.Description
    Resume Exit_Comando13_Click
End Sub
Private Sub Comando14_Click()
On Error GoTo Err_Comando14_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "POTENCIA PROMEDIO TRIFASICA SUB1"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Comando14_Click:
    Exit Sub
Err_Comando14_Click:
    MsgBox Err.Description
    Resume Exit_Comando14_Click
End Sub
Private Sub Comando15_Click()
On Error GoTo Err_Comando15_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "PROMEDIO VOLTAJES Y CORRIENTES TRIFASICOS SUB1"

```

```

    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Comando15_Click:
    Exit Sub
Err_Comando15_Click:
    MsgBox Err.Description
    Resume Exit_Comando15_Click
End Sub
Private Sub Comando16_Click()
On Error GoTo Err_Comando16_Click

    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "TODOS LOS PARAMETROS ELECTRICOS SUB1"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Comando16_Click:
    Exit Sub
Err_Comando16_Click:
    MsgBox Err.Description
    Resume Exit_Comando16_Click
End Sub
Private Sub Comando18_Click()
On Error GoTo Err_Comando18_Click

    DoCmd.Close
Exit_Comando18_Click:
    Exit Sub
Err_Comando18_Click:
    MsgBox Err.Description
    Resume Exit_Comando18_Click
End Sub
Private Sub Comando19_Click()
On Error GoTo Err_Comando19_Click

    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "REGISTROS SUB1 2004"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Comando19_Click:
    Exit Sub
Err_Comando19_Click:
    MsgBox Err.Description
    Resume Exit_Comando19_Click
End Sub

```

## **MENU- SUB2**

```

Option Compare Database
Private Sub cmdExcel_Click()

```

```

DoCmd.OutputTo acOutputQuery, "qryExpPROMIPAC1", acFormatXLS
DoCmd.OutputTo acOutputQuery, "qryExpPROMIPAC2", acFormatXLS
End Sub
Private Sub Command1_Click()
On Error GoTo Err_Command1_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "TENSIONES DE FASE SUB2"
    DoCmd.OpenForm stDocName, , , stLinkCriteria

Exit_Command1_Click:
    Exit Sub
Err_Command1_Click:
    MsgBox Err.Description
    Resume Exit_Command1_Click
End Sub
Private Sub Command3_Click()
On Error GoTo Err_Command3_Click
    DoCmd.Quit
Exit_Command3_Click:
    Exit Sub
Err_Command3_Click:
    MsgBox Err.Description
    Resume Exit_Command3_Click
End Sub
Private Sub Command4_Click()
DoCmd.OpenReport "zmrptPrintTables", acViewPreview
End Sub
Private Sub Comando10_Click()
On Error GoTo Err_Command1_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "TENSIONES DE LINEA Y FASE SUB2"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Command1_Click:
    Exit Sub
Err_Command1_Click:
    MsgBox Err.Description
    Resume Exit_Command1_Click
End Sub
Private Sub Comando11_Click()
On Error GoTo Err_Command1_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "POTENCIAS FASE 1 SUB2"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Command1_Click:

```



```

Exit Sub
Err_Command1_Click:
    MsgBox Err.Description
    Resume Exit_Command1_Click
End Sub
Private Sub Comando12_Click()
On Error GoTo Err_Command1_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "POTENCIAS FASE 2 SUB2"
    DoCmd.OpenForm stDocName, , , stLinkCriteria

Exit_Command1_Click:
    Exit Sub
Err_Command1_Click:
    MsgBox Err.Description
    Resume Exit_Command1_Click
End Sub
Private Sub Comando13_Click()
On Error GoTo Err_Command1_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "POTENCIAS FASE 3 SUB2"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Command1_Click:
    Exit Sub
Err_Command1_Click:
    MsgBox Err.Description
    Resume Exit_Command1_Click
End Sub
Private Sub Comando14_Click()
On Error GoTo Err_Command1_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "POTENCIA PROMEDIO TRIFASICA SUB2"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Command1_Click:
    Exit Sub
Err_Command1_Click:
    MsgBox Err.Description
    Resume Exit_Command1_Click
End Sub
Private Sub Comando15_Click()
On Error GoTo Err_Command1_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "PROMEDIO VOLTAJES Y CORRIENTES TRIFASICOS SUB2"

```

```

    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Command1_Click:
    Exit Sub
Err_Command1_Click:
    MsgBox Err.Description
    Resume Exit_Command1_Click
End Sub
Private Sub Comando16_Click()
On Error GoTo Err_Command1_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "TODOS LOS PARAMETROS ELECTRICOS SUB2"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Command1_Click:
    Exit Sub
Err_Command1_Click:
    MsgBox Err.Description
    Resume Exit_Command1_Click
End Sub
Private Sub Comando19_Click()
On Error GoTo Err_Command1_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "REGISTROS SUB2 2004"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Command1_Click:
    Exit Sub
Err_Command1_Click:
    MsgBox Err.Description
    Resume Exit_Command1_Click
End Sub
Private Sub Comando6_Click()
On Error GoTo Err_Command1_Click
    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "DEMANDA Y ENERGIA SUB2"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Command1_Click:
    Exit Sub
Err_Command1_Click:
    MsgBox Err.Description
    Resume Exit_Command1_Click
End Sub
Private Sub Comando7_Click()
On Error GoTo Err_Command1_Click
    Dim stDocName As String
    Dim stLinkCriteria As String

```

```

    stDocName = "INTENSIDADES DE CORRIENTE POR FASE SUB2"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Command1_Click:
    Exit Sub
Err_Command1_Click:
    MsgBox Err.Description
    Resume Exit_Command1_Click
End Sub
Private Sub Comando8_Click()
On Error GoTo Err_Command8_Click

    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "TENSIONES DE FASE SUB2"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Comando8_Click:
    Exit Sub
Err_Comando8_Click:
    MsgBox Err.Description
    Resume Exit_Comando8_Click
End Sub
Private Sub Comando9_Click()
On Error GoTo Err_Comando8_Click

    Dim stDocName As String
    Dim stLinkCriteria As String
    stDocName = "TENSIONES DE LINEA SUB2"
    DoCmd.OpenForm stDocName, , , stLinkCriteria
Exit_Comando8_Click:
    Exit Sub
Err_Comando8_Click:
    MsgBox Err.Description
    Resume Exit_Comando8_Click
End Sub
Private Sub Form_Open(Cancel As Integer)
DoCmd.Restore
End Sub
Private Sub Comando17_Click()
On Error GoTo Err_Comando17_Click

    DoCmd.Close
Exit_Comando17_Click:
    Exit Sub
Err_Comando17_Click:
    MsgBox Err.Description
    Resume Exit_Comando17_Click
End Sub

```

## **MENU DEMANDA Y ENERGIA SUB1**

```

Option Compare Database
Private Sub Comando14_Click()
On Error GoTo Err_Comando14_Click
    DoCmd.Close
Exit_Comando14_Click:
    Exit Sub
Err_Comando14_Click:
    MsgBox Err.Description
    Resume Exit_Comando14_Click

End Sub

```

## **INTENSIDADES POR FASE SUB1**

```

Option Compare Database
Private Sub Comando14_Click()
On Error GoTo Err_Comando14_Click
    DoCmd.Close
Exit_Comando14_Click:
    Exit Sub
Err_Comando14_Click:
    MsgBox Err.Description
    Resume Exit_Comando14_Click

End Sub

```

## **TODOS LOS PARÁMETROS ELÉCTRICOS SUB1**

```

Option Compare Database
Private Sub Comando77_Click()
On Error GoTo Err_Comando77_Click
    DoCmd.Close
Exit_Comando77_Click:
    Exit Sub
Err_Comando77_Click:
    MsgBox Err.Description
    Resume Exit_Comando77_Click

End Sub
Private Sub Command78_Click()
On Error GoTo Err_Command78_Click
    Dim stDocName As String

```

```
Dim MyForm As Form
stDocName = "POTENCIA PROMEDIO TRIFASICA SUB1"
Set MyForm = Screen.ActiveForm
DoCmd.SelectObject acForm, stDocName, True
DoCmd.PrintOut
DoCmd.SelectObject acForm, MyForm.Name, False
Exit_Command78_Click:
Exit Sub
Err_Command78_Click:
MsgBox Err.Description
Resume Exit_Command78_Click
End Sub
```

## CÓDIGO DE PROGRAMACIÓN EN ACCESS

### **Pantalla Acceso**

```
<html>  
<head>  
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">  
<meta name="juliux" content="UDB">  
<meta name="keywords" content="Scada">
```

```

<meta name="description" content="Scada Software">
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<meta http-equiv="refresh" content="5; URL=access.php">
<title>...:Scada V1.0:...:</title>
</head>
<BODY>
<?php

//subestacion1
include('adodb/adodb.inc.php');
$conn = &ADONewConnection('access');
$conn->PConnect('sub1');
$recordSet = &$conn->Execute('select DATE from 04095000');
$recordSet->Close();
$conn->Close();

$user="root";
$pass="vacaloca";
$db="scada";
$link=mysql_connect("10.0.10.96",$user,$pass);
if (!$link)
die("No me puedo conectar a MySQL");
mysql_select_db($db,$link)
or die("No pude abrir $db: " .mysql_error());
$envio = "INSERT INTO parametros1
VALUES(NULL,'$CKW_III','$CKVARL_III','$CKVARC_III','$CPF_III','$CV_1','$CA_
1','$CPF_1','$CHZ','$CV_2','$CA_2','$CPF_2','$CV_3','$CA_3','$CPF_3','$CKWH_1','$C
KVARLH_1','$CKVARCH_1)";
mysql_query($envio,$link) or die ("Error" .mysql_error());
mysql_close($link);
?>
</BODY>
</HTML>

```

## Pantalla Cerrar

```

<?php
    session_start();
    session_destroy();
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

```

```

<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
  <meta name="GENERATOR" content="Quanta Plus">
  <meta name="juliux" content="UDB">
  <meta name="copyright" content="UDB">
  <meta name="keywords" content="Tutorias">
  <meta name="description" content="Software para Tutorias">
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <meta http-equiv="refresh" content="5; URL=index.php">
<title>...:TUTOR V1.0:::</title>
<link rel="stylesheet" type="text/css" href="estilista.css" title="estilos en cascada">
</head>
<body bgcolor="#9090a5" alink="#9090a5" vlink="#9090a5" leftmargin="0"
topmargin="0" marginheight="0" marginwidth="0">
<br>
<table align="center" border="0" cellspacing="0" cellpadding="0">

      <table align="center" border="0" cellspacing=0 cellpadding=0>
      <tr>
      <td bgcolor="#ffffff"></td>
      <td bgcolor="#ffffff">&nbsp;</td>
      <td bgcolor="#ffffff"></td>
      </tr>

      <tr>
      <td bgcolor="#ffffff"></td>
      <td bgcolor="#ffffff"><div align="center"><b><font size='+1'>...:
Sistema SCADA de Monitoreo en L&#237;nea :::</font></b></div><hr>
      <p class=negro>
          <div align="center"></div><br>
          <div align="center"><font color="#47475e" size=+2>Su sesion se cerro
correctamente!!!</font></div><hr>
      </p>
      </td>
      <td bgcolor="#ffffff"></td>
      </tr>

      <tr>
      <td bgcolor="#ffffff"></td>
      <td bgcolor="#ffffff"><div align="center"></div></td>

```



```

                <td bgcolor="#ffffff"></td>
            </tr>
        </table><br>
</body>
</html>

```

## Pantalla Index

```

<?php
    if($ok){
        $user="x";
        $pass="123456";
        $db="scada";
        $link=mysql_connect("localhost",$user,$pass);
        if (!$link)
            die("Couldn't connect to MYSQL");
        mysql_select_db($db,$link)
        or die("Couldn't open $db: " .mysql_error());
        $result1 = mysql_query("SELECT COUNT(*) FROM usuarios WHERE
login='$usuario' AND contrasenia='$x'");
        list($y)=mysql_fetch_row($result1);
        mysql_free_result($result1);
        mysql_close($link);

        if($y>0){
            $user="x";
            $pass="123456";
            $db="scada";
            $link=mysql_connect("localhost",$user,$pass);
            if (!$link)
                die("Couldn't connect to MYSQL");
            mysql_select_db($db,$link)
            or die("Couldn't open $db: " .mysql_error());
            $result1 = mysql_query("SELECT nivel_acceso FROM
usuarios WHERE login='$usuario'");
            while($col=mysql_fetch_row($result1)){
                foreach($col as $campo){
                    switch($campo){
                        case "2":
                            header('location: sub1.php');
                            break;
                        case "1":

```

```

construcciontemporal.php');
                                header('location:
                                break;
                                case "0":
                                header('location:
construcciontemporal.php');
                                break;
                                }
                                }
                                mysql_free_result($result1);
                                mysql_close($link);
                                }
                                }
                                else{
                                $error_login=1;
                                }
                                }
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>...:SISTEMA SCADA:...</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<meta name="GENERATOR" content="Quanta Plus">
<link rel="stylesheet" type="text/css" href="estilista.css" title="estilos en cascada">
<script language="javascript">
function validar(){
var mg="";

if(document.formulario1.usuario.value.length == 0){
mg = mg+"Ingrese su ID de usuario!\n";
}

if(document.formulario1.x.value.length == 0){
mg=mg+"Ingrese el Password correspondiente al ID de usuario!\n";
}

if(document.formulario1.x.value.length < 6){
mg=mg+"El Password debe contener por lo menos 6 caracteres!\n";
}

if (mg.length == 0){
return true;
}

else{

```

```

        alert(mg);
        return false;
    }
}
</script>
</head>
<body bgcolor="#9090a5" alink="#9090a5" vlink="#9090a5" leftmargin="0"
topmargin="0" marginheight="0" marginwidth="0">
<table align="center" border=0 cellpadding=0 cellspacing=0 width=100%>
<tr>
<td bgcolor="#FFFFFF" align="right"></td>
</tr>
</table><br>

<table align="center" border="0" cellspacing="0" cellpadding="0">

        <table align="center" border="0" cellspacing=0 cellpadding=0>

                <tr>
                <td bgcolor="#ffffff"></td>
                <td bgcolor="#ffffff">&nbsp;</td>
                <td bgcolor="#ffffff"></td>
                </tr>

                <tr>
                <td bgcolor="#ffffff"></td>
                <td bgcolor="#ffffff"><div align="center"><b>::: Sistema SCADA de
Monitoreo en L&#237;nea :::</b></div><hr>
                <p class=negro>
                <div align="center"></div><br>
                <div align="center"><font
color="#47475e" size=+2>Identificaci&#243;n</font></div><br>
                <hr>
                <form name="formulario1" onsubmit="return validar();"
action="index.php" method="POST">
                <table align="center" border=0 cellpadding=5 cellspacing=5>
                <tr align=center>
                <th class=negrolleno bgcolor="#C0C0FF">Usuario:</th>
                <th class=negrolleno bgcolor="#C0C0FF"><input type="text"
name="usuario"></th>
                </tr>

```

```

        <tr align=center>
        <th class=negrolleno bgcolor="#FFDCA8">Password:</th>
        <th class=negrolleno bgcolor="#FFDCA8"><input type="password"
name="x"></th>
        </tr>
        <tr align=center>
        <th class=negrolleno colspan="2"><input type="submit" name="ok"
value="Enviar">&nbsp;<input type="reset" name="res" value="Limpiar"></th>
        </tr>
        </table><hr>

                                <?php

        if($error_login==1){
                                                                print
        "<div align=center class=l1leno11><img src=pics/dialog-warning.png width=48 height=48
align=middle border=0><br>Usuario/Password Incorrectos!</div>";

        $error_login=0;
                                                                }
                                                                ?>

        </form>
        </p>
        </td>
        <td bgcolor="#ffffff"></td>
        </tr>
        <tr>
        <td bgcolor="#ffffff"></td>
        <td bgcolor="#ffffff"><div align="center"><font color="red">
</font></div></td>
        <td bgcolor="#ffffff"></td>
        </tr>

        </table><br>

</body>
</html>

```

## Pantalla Agregar Usuario

```

<?php
    session_start();
    session_register('tutorname1');

```

```

session_register('tutorape1');
session_register('telcasa1');
session_register('celular1');
session_register('email1');
session_register('flagtutor');
session_register('pass3');
session_register('login1');
session_register('loginx');
session_register('$perfil1');
if($send){
    $tutorname1=$tutorname;
    $tutorape1=$tutorape;
    $pass3=$pass1;
    $telcasa1=$telcasa;
    $celular1=$celular;
    $email1=$email;
    $login1=$logina;
    $perfil1=$perfil;
    $user="x";
    $pass="123456";
    $db="scada";
    $link=mysql_connect("localhost",$user,$pass);
    if (!$link)
        die("No me puedo conectar a MySQL");
    mysql_select_db($db,$link)
        or die("No pude abrir $db: " .mysql_error());

    $result1 = mysql_query("SELECT login FROM usuarios WHERE
login='$login1'");

    if(mysql_num_rows($result1)==0){
        $envio = "INSERT INTO usuarios
VALUES('NULL','$login1','$pass1','$tutorname1','$tutorape1','$telcasa1','$celular1','$email
1','$perfil')";

        mysql_query($envio,$link) or die ("Error" .mysql_error());
        mysql_close($link);
        $flagtutor=1;
        header('location:sub1.php');
    }
    else{
        $loginx=1;
        mysql_free_result($result1);
        mysql_close($link);
        header('location: sub1.php');
    }
}
?>

```

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>...:TUTOR V1.0:...</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<meta name="GENERATOR" content="Quanta Plus">
<link rel=stylesheet type="text/css" href="estilista.css" title="estilos en cascada">
<script language="javascript">
    function mensaje(){
        alert("Sus datos no seran guardados!");
    }

    function validar(){
        var mg = "";

        if(document.formulario1.logina.value.length==0){
            mg=mg+"Ingrese el ID de usuario!\n";
        }
        if(document.formulario1.pass1.value.length==0){
            mg=mg+"Ingrese el password del tutor!\n";
        }
        if(document.formulario1.pass2.value.length==0){
            mg=mg+"Reescriba el password!\n";
        }

        if(document.formulario1.pass1.value!=document.formulario1.pass2.value){
            mg=mg+"El password no coincide!\n";
        }

        if((document.formulario1.pass1.value.length<6)||
        (document.formulario1.pass2.value.length<6)){
            mg=mg+"El password debe contener al menos
6 caracteres!\n";
        }
        if(document.formulario1.tutorname.value.length==0){
            mg=mg+"Ingrese los nombres del tutor!\n";
        }
        if(document.formulario1.tutorape.value.length==0){
            mg=mg+"Ingrese los apellidos del tutor!\n";
        }
        if(document.formulario1.celular.value.length==0){
            mg=mg+"Ingrese el numero de celular, o N/A
si no posee uno!\n";
        }
        if(document.formulario1.email.value.length==0){

```

```

                                mg=mg+"Ingrese el email, o N/A si no posee
uno!\n";
                                }
                                if (mg.length == 0){
                                    return true;
                                }
                                else{
                                    alert(mg);
                                    return false;
                                }
                            }
</script>
</script>
</head>
<body bgcolor="#9090a5" leftmargin="0" topmargin="0" marginheight="0"
marginwidth="0">
<br>

<table align="center" border=0 width="100%">
    <tr>
        <td width="10%" align="center" valign="top">
            <div align="center" class=lleno10><img src='pics/udb_logo.gif'
width='68' height='79' align='middle' border='0'></div>
            <div align="center" class=lleno10><a
href="newtutor.php">Eliminar usuarios</a></div>
            <div align="center" class=lleno10><a href="newtutor.php">Agregar
usuarios</a></div>
            <div align="center" class=lleno10><a href="sub1.php">Subestacion
1</a></div>
            <div align="center" class=lleno10><a href="sub2.php">Subestacion
2</a></div>
            <div align="center" class=lleno10><a href="">Graficos</a></div>
            <div align="center" class=lleno10><a href="">Reportes</a></div>
            <div align="center" class=lleno10><a
href="cerrar.php">Salir</a></div>
        </td>
        <td width="90%" align="center" valign="top">
            <table align="center" border="0" cellspacing="0"
cellpadding="0">
                <table align="center" border="0" cellspacing=0 cellpadding=0>
                    <tr>
                        <td bgcolor="#ffffff"></td>
                        <td bgcolor="#ffffff">&nbsp;</td>
                        <td bgcolor="#ffffff"></td>
                    </tr>
                </table>
            </table>
        </td>
    </tr>
</table>

```

```

        </tr>
        <tr>
        <td bgcolor="#ffffff"></td>
        <td bgcolor="#ffffff"><div align="center"><b><font size='+1'>...:
Sistema SCADA de Monitoreo en L&#237;nea :::</font></b></div>
        <p classub1.phps=negro>
        <div align="center"></div><br>
        <div align="center"><font
color="#47475e" size="+2">Agregar usuario:</font></div><br>
        <hr>
        <h3><b>Datos del
usuario:</b></h3><br>
        <form name="formulario1"
onsubmit='return validar();' action="newtutor.php" method="POST">
        <table align="center" border="0"
width="100%" cellspacing="5" cellpadding="5">
        <tr>
        <td align="center"
bgcolor="#C0C0FF"><b>Login (ID):</b></td>
        <td align="center"
bgcolor="#C0C0FF"><input type="text" name="logina" size="20"></td>
        <td align="center"
bgcolor="#C0C0FF"><b>Perfil:</b></td>
        <td align="center"
bgcolor="#C0C0FF">
        <select name="perfil">
        <option
value="2">Administrador</option>
        <option
value="1">Monitor</option>
        </select>
        </td>
        </tr>
        <tr>
        <td align="center"
bgcolor="#FFDCA8"><b>Password:</b></td>
        <td align="center"
bgcolor="#FFDCA8"><input type="password" name="pass1" size="20"></td>
        <td align="center"
bgcolor="#FFDCA8"><b>Reescriba el Password:</b></td>
        <td align="center"
bgcolor="#FFDCA8"><input type="password" name="pass2" size="27"></td>
        </tr>
        <tr>
        <td align="center"
bgcolor="#C0C0FF"><b>Nombres:</b></td>

```



```

                                <td align="center"
bgcolor="#C0C0FF"><input type="text" name="tutorname" size="20"></td>
                                <td align="center"
bgcolor="#C0C0FF"><b>Apellidos:</b></td>
                                <td align="center"
bgcolor="#C0C0FF"><input type="text" name="tutorape" size="27"></td>
                                </tr>
                                <tr>
                                <td align="center"
bgcolor="#FFDCA8"><b>Telefono de casa:</b></td>
                                <td align="center"
bgcolor="#FFDCA8"><input type="text" name="telcasa" size="20"></td>
                                <td align="center"
bgcolor="#FFDCA8"><b>Celular:</b></td>
                                <td align="center"
bgcolor="#FFDCA8"><input type="text" name="celular" size="27"></td>
                                </tr>
                                <tr>
                                <td align="center"
bgcolor="#C0C0FF"><b>Email:</b></td>
                                <td align="left"
bgcolor="#C0C0FF" colspan='3'><input type="text" name="email" size="60"></td>
                                </tr>
                                <tr>
                                <td align="center"
bgcolor="#0000FF" colspan="4">
                                <input type="submit" name=send
value="Validar y enviar">&nbsp;<input type="reset" value="Limpiar">&nbsp;<
a href="sub1.php"><input
type="button" name ="cancel" value="Cancelar" onClick="mensaje();"></a></td>
                                </tr>
                                </table><hr>

                                </p></td>
                                <td bgcolor="#ffffff"></td>
                                </tr>

                                <tr>
                                <td bgcolor="#ffffff"></td>
                                <td bgcolor="#ffffff"></td>
                                <td bgcolor="#ffffff"></td>
                                </tr>
                                </table><br>

                                </td>
</tr>

```

```
</table>
</body>
</html>
```

## Pantalla Subestación 1:

```
<?php
    session_start();
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
    <title>:::SISTEMA SCADA:::</title>
    <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
    <meta name="GENERATOR" content="Quanta Plus">
    <link rel="stylesheet type="text/css" href="estilista.css" title="estilos en cascada">
</head>
<body bgcolor="#9090a5" leftmargin="0" topmargin="0" marginheight="0"
marginwidth="0">
    <br>
    <table align="center" border=0 width="100%">
        <tr>
            <td width="10%" align="center" valign="top">
                <div align="center" class=lleno10><img src='pics/udb_logo.gif'
width='68' height='79' align='middle' border='0'></div>
                <div align="center" class=lleno10><a
href="newtutor.php">Eliminar usuarios</a></div>
                <div align="center" class=lleno10><a href="newtutor.php">Agregar
usuarios</a></div>
                <div align="center" class=lleno10><a href="sub1.php">Subestacion
1</a></div>
                <div align="center" class=lleno10><a href="sub2.php">Subestacion
2</a></div>
                <div align="center" class=lleno10><a href="">Graficos</a></div>
                <div align="center" class=lleno10><a href="">Reportes</a></div>
                <div align="center" class=lleno10><a
href="cerrar.php">Salir</a></div>
            </td>
            <td width="90%" align="center" valign="top">
                <table align="center" border="0" cellspacing="0"
cellpadding="0">
                    <table align="center" border="0" cellspacing=0
cellpadding=0>
                        <tr>
```

```

width="5%"></td>
width="90%">&nbsp;</td>
width="5%"></td>
</tr>
<tr>
<td bgcolor="#ffffff"></td>
<td bgcolor="#ffffff"><div
align="center"><b><font size='+1'>...: Sistema SCADA de Monitoreo en L&#237;nea
...</font></b></div><hr>
<p class=negro>
<div align="left">
<?php
print "<font size='+1'><b>Fecha:
</b></font>";
color='blue'><b>";
print "<font size='+1'
print date("j/m/Y");
print "</b></font><br>";
?></div>
<div align="center"><font color='blue'
size='+2'><b>Subestaci&#243;n 1</b></font></div>
cellpadding='5' cellspacing='5'>
cellpadding='5' cellspacing='5'>
bgcolor='#FFDCA8'><b>Par&#225;metro</b></td>
bgcolor='#FFDCA8' colspan='3'><b>Valor</b></td>
</tr>
<tr>
<td align='left'
bgcolor='#FFFFC0'><b>Potencia Activa Trifasica [KW]</b></td>

```

```

        <td align='center'
bgcolor='#FFFFC0' colspan='3'></td>
    </tr>
    <tr>
        <td align='left'
bgcolor='#DCDCDC'><b>Potencia Inductiva Trifasica [KVARL]</b></td>
        <td align='center'
bgcolor='#DCDCDC' colspan='3'></td>
    </tr>
    <tr>
        <td align='left'
bgcolor='#FFFFC0'><b>Potencia Capacitiva Trifasica [KVARC]</b></td>
        <td align='center'
bgcolor='#FFFFC0' colspan='3'></td>
    </tr>
    <tr>
        <td align='left'
bgcolor='#DCDCDC'><b>Factor de potencia Trifasico</b></td>
        <td align='center'
bgcolor='#DCDCDC' colspan='3'></td>
    </tr>
</table>
</td>
<td align='center'>
<table align='center' border='2'
cellpadding='5' cellspacing='5'>
    <tr>
        <td align='center'
bgcolor='#FFDCA8'><b>Par&#225;metro</b></td>
        <td align='center'
bgcolor='#FFDCA8'><b>F1</b></td>
        <td align='center'
bgcolor='#FFDCA8'><b>F2</b></td>
        <td align='center'
bgcolor='#FFDCA8'><b>F3</b></td>
    </tr>
    <tr>

```

bgcolor='#FFFFC0'><b>Voltaje [V]</b></td>

bgcolor='#FFFFC0'></td>

bgcolor='#FFFFC0'></td>

bgcolor='#FFFFC0'></td>

<td align='left'

<td align='center'

<td align='center'

<td align='center'

</tr>

<tr>

<td align='left'

<td align='center'

<td align='center'

<td align='center'

</tr>

<tr>

<td align='left'

<td align='center'

<td align='center'

<td align='center'

</tr>

<tr>

<td align='left'

<td align='center'

<td align='center'

<td align='center'

</tr>

</table>

bgcolor='#DCDCDC'><b>Corriente [A]</b></td>

bgcolor='#DCDCDC'></td>

bgcolor='#DCDCDC'></td>

bgcolor='#DCDCDC'></td>

bgcolor='#FFFFC0'><b>Factor de potencia</b></td>

bgcolor='#FFFFC0'></td>

bgcolor='#FFFFC0'></td>

bgcolor='#FFFFC0'></td>

bgcolor='#DCDCDC'><b>Frecuencia [Hz]</b></td>

bgcolor='#DCDCDC'></td>

bgcolor='#DCDCDC'></td>

bgcolor='#DCDCDC'></td>

</td>

```

<td align='center'>
    <table align='center' border='2'
cellpadding='5' cellspacing='5'>
    <tr>
    <td align='center'
bgcolor='#FFDCA8'><b>Par&#225;metro</b></td>
    <td align='center'
bgcolor='#FFDCA8' colspan='3'><b>Valor</b></td>
    </tr>
    <tr>
    <td align='left'
bgcolor='#FFFFC0'><b>Energia Activa [KWH]</b></td>
    <td align='center'
bgcolor='#FFFFC0' colspan='3'></td>
    </tr>
    <tr>
    <td align='left'
bgcolor='#DCDCDC'><b>Energia Inductiva [KVALH]</b></td>
    <td align='center'
bgcolor='#DCDCDC' colspan='3'></td>
    </tr>
    <tr>
    <td align='left'
bgcolor='#FFFFC0'><b>Energia Capacitiva [KVARCH]</b></td>
    <td align='center'
bgcolor='#FFFFC0' colspan='3'></td>
    </tr>
    </table>
</td>
</tr>
</table>
<?php
    if($flagtutor==1){
        print ("<div
align=center class=l1leno11><img src=pics/dialog-warning.png width=48 height=48
align=middle border=0><br>El usuario ha sido agregado correctamente!</div>");

```

```

$flagtutor=0;
}
if($loginx==1){
    print ("<div
align=center class=lleño11><img src=pics/dialog-error.png width=48 height=48
align=middle border=0><br>El usuario ya existe en el sistema!</div>");
    $loginx=0;
}
?>
</p></td>
<td bgcolor="#ffffff"></td>
</tr>
<tr>
<td bgcolor="#ffffff"></td>
<td bgcolor="#ffffff"></td>
<td bgcolor="#ffffff"></td>
</tr>
</table><br>
</table>
</td>
</tr>
</table>
</body>
</html>

```

**Nota:** Para la Subestación 2 será el mismo código fuente de Subestación 1, solo que referenciado al acceso de la base datos de dicha Subestación (Base de Datos generada por el software CIRNET = 04095031.DBF, que ha sido vinculada por medio del ODBC).

## ACERCA DE HTML



El **HTML** no es más que una aplicación del **SGML** (*Standard Generalized Markup Language*), un sistema para definir tipos de documentos estructurados y lenguajes de marcas para representar esos mismos documentos. El término **HTML** se suele referir a ambas cosas, tanto al tipo de documento como al lenguaje de marcas.

En realidad aún no existe un estándar del **HTML** en Internet, ya que existen tres revisiones o niveles de estandarización que aún no han sido aceptadas: la versión 1.0, que en realidad no existe como estándar ya que nunca se hizo una especificación formal, la 2.0, que es un estándar de hecho, aunque todavía está en la etapa de propuesta de estándar documentada en el RFC-1866 de noviembre de 1995. El siguiente nivel, el HTML 3.0, está en una etapa experimental, aunque muchos de los visores aceptan algunas de las extensiones propuestas.

En este apartado comentaremos la especificación del HTML 2.0 tal y como se describe en el RFC-1866. En el punto siguiente hablaremos de algunas de las propuestas de ampliación para el HTML 3.0 y algunas extensiones soportadas por algunos de los visores.

## Atributos

Cuando una marca inicial admite atributos, éstos se escriben a continuación del nombre del elemento. Generalmente los atributos tienen la forma nombre, signo

igual, valor del atributo aunque en algunos casos basta con el nombre del atributo. Se pueden poner espacios en blanco antes y después del signo igual.

El valor de un atributo puede ser:

- Una cadena de caracteres entre comillas (simples o dobles) que no contenga el símbolo de fin de marca ">".
- Un nombre como los definidos en el apartado anterior.

La longitud del valor de un atributo no puede superar los 1024 caracteres.

---

## Comentarios

Para incluir comentarios en HTML se emplea la declaración de comentarios. Una declaración de comentarios comienza con <!, le siguen uno o varios comentarios y termina con >. Cada comentario comienza con -- e incluye todo el texto hasta la siguiente aparición de --. Dentro de una declaración de comentarios, se pueden poner espacios en blanco después de cada uno de ellos, pero no antes del primero. Toda la declaración del comentario se ignora.

---

## Identificación del Nivel HTML de un Documento

Para identificar un documento como HTML que sigue el estándar 2.0, cada documento debe comenzar con la siguiente declaración:

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
```

Existen más identificadores que especifican otros detalles, por ejemplo si el documento contiene formularios. Este mecanismo también puede ser empleado por los visores para reconocer otros tipos de documentos.

Los documentos en formato **HTML** son un conjunto de elementos anidados. En el nivel más alto nos encontramos el elemento **HTML** (marca inicial <HTML> y final </HTML>) que consta de dos partes: cabecera y cuerpo.

La cabecera se emplea para proporcionar información acerca del documento, mientras que el cuerpo contiene el texto de la página, es decir, la información que se va a presentar al usuario. En el cuerpo se pueden incluir todo tipo de elementos y marcas.

No es mala idea diseñarse una *plantilla* para desarrollar siempre a partir de ella. Además del esqueleto de la página (marcas de texto HTML, cabecera y cuerpo) podemos incluir en ella otras informaciones que siempre queramos que aparezcan (tanto comentarios como elementos del cuerpo o la cabecera). Prácticamente todos los editores específicos permiten el uso de plantillas y macros para introducir información útil, como por ejemplo la fecha de la última modificación. La siguiente página puede ser un buen punto de partida:

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<!-- Comentario sobre el autor -->
<HTML>
<HEAD>
<TITLE></TITLE>
<!-- Datos adicionales -->
</HEAD>
<BODY>
<!-- Datos cabecera pagina -->
<HR>
<HR>
<!-- Datos pie de pagina -->
</BODY>
</HTML>
```

Los datos adicionales dependerán del autor. La cabecera y pie pueden ser útiles para dar a todas las páginas el mismo aspecto, aunque dependerán de que la persona que escribe las páginas lo considere necesario.

Entraremos ahora en la descripción de los elementos que pueden aparecer en la cabecera y el cuerpo.

---

## Cabecera

La cabecera es una colección de información acerca del documento. Las marcas de principio y fin son <HEAD> y </HEAD>.

La cabecera puede contener los siguientes datos, sin importar el orden en que aparezcan:

- *Título* (TITLE). Indica el nombre del documento, los visores lo emplean generalmente como etiqueta de la ventana. Este campo es el único obligatorio en la cabecera.
- *Dirección de base* (BASE). Proporciona una dirección de base para interpretar los enlaces relativos cuando el documento se lee fuera de su contexto (por ejemplo cuando está guardado en un disco).
- *El documento es un índice* (ISINDEX). Si se pone la palabra clave ISINDEX, el cliente interpretará que la página es un índice y permitirá al usuario introducir palabras clave para buscarlas.
- *Enlaces relacionados* (LINK). En la cabecera podemos incluir varios enlaces relacionados con el documento como versiones anteriores, dirección del autor, etc.
- *Metainformación* (META). Este campo sirve para proporcionar información sobre el documento que no pueda ser expresada en los campos anteriores. La información se escribe usando pares nombre/valor que pueden ser empleados para dos propósitos:
  1. Proporcionar datos al servidor de HTTP para que genere campos de cabecera (como por ejemplo la fecha de caducidad de un documento que se actualiza periódicamente) o,
  2. Para que un usuario incluya información adicional sobre el documento, como palabras clave o el nombre del autor.

Para la primera función se emplean atributos del tipo HTTP-EQUIV y para la segunda los del tipo NAME. En ambos casos, el valor se asigna en el campo CONTENT. Un ejemplo del primer caso sería:

```
<META HTTP-EQUIV ="Expires" CONTENT="Dec 1996">
```

Y del segundo:

```
<META NAME="Autor" CONTENT="Plo+Serg">
```

El atributo NAME se refiere a nombres elegidos al azar por el usuario, mientras que HTTP-EQUIV significa que el valor tiene una cabecera equivalente real en el protocolo HTTP.

- *Siguiente Identificador* (NEXTID). En la actualidad este campo ya no se usa, lo empleaban los editores de HTML para asignar nombres a documentos de forma automática.

---

## Cuerpo

Como ya hemos dicho, el cuerpo contiene el texto de la página que se va a presentar al usuario. Las marcas de principio y fin del cuerpo son <BODY> y </BODY> respectivamente.

A continuación describiremos los elementos que pueden aparecer en el cuerpo, clasificados por categorías.

---

## Encabezados

Los encabezados se emplean para dividir los documentos en secciones, o más concretamente para marcar los títulos de esas secciones. Las marcas son del tipo <H#>Título</H#>, donde # puede ser un número cualquiera entre 1 y 6.

Aunque el estándar no lo especifica, es recomendable usar los niveles en orden, es decir, después de un encabezado de nivel uno deberemos usar encabezados de nivel dos para las subsecciones y no saltar directamente al tres o al cuatro, por ejemplo.

La representación de los encabezados depende del visor, generalmente se representan en negrita y van cambiando de tamaño y espacios antes y después, de más a menos, en función del nivel (el nivel uno es el mayor).

Ejemplo:

```
<H1>Don Quijote de la Mancha</H1>
<H2>Capítulo 1</H2>
<P>En un lugar de la Mancha de cuyo nombre no quiero acordarme ...
```

---

## Definición de Bloques

Para definir y separar bloques de texto se emplea una serie de marcas que definen párrafos, texto preformateado o bloques con un significado especial como direcciones o citas. También, y aunque no son propiamente para definir bloques, hablamos en este punto de dos marcas *especiales*, una para representar saltos de línea y otra que inserta una línea horizontal. Ambas permiten dividir el texto, por lo que las hemos incluido aquí.

Las marcas de bloque son:

- `<P>` para separar párrafos. Dado que para el HTML todo el texto es continuo, necesitamos algún mecanismo para indicar el principio y fin de un párrafo. Las marcas inicial y final son `<P>` y `</P>`, aunque generalmente sólo se emplea la inicial, terminando el mismo cuando encontramos cualquier elemento que cause un salto de línea.
- `<PRE>` para texto preformateado. Esta marca se emplea para texto escrito en tipo de letra de caja fija (mono-espaciada) y dentro del bloque marcado los saltos de línea sí son respetados. Dentro de estos elementos se pueden emplear anclajes y marcado tipográfico, pero no elementos que definan formato de párrafo (como marcas de párrafo, encabezados, etc.).  
El elemento acepta el atributo opcional `WIDTH`, que indica el máximo número de caracteres por línea para que el visor ajuste el tamaño y tabulación del texto. Además de la marca `PRE` existen dos elementos muy similares a él: `EXP` (ejemplo) y `LISTING` (listado), que no permiten ningún tipo de elemento anidado. Dado que con `PRE` podemos obtener el mismo resultado, no es recomendable el uso de estas marcas.
- `<ADDRESS>`, empleada para indicar que un texto representa una dirección o una firma. Generalmente se representa en cursiva y puede estar tabulado.
- `<BLOCKQUOTE>`, que indica que un texto es una cita de otra fuente. Se suele representar con tabulaciones a izquierda y derecha y en cursiva. En sistemas que no permiten representar cursivas se puede emplear algún tipo de símbolo al principio de las líneas, de manera similar a lo que se hace en las réplicas o citas (*quote*) del correo electrónico.

- <BR>. Este elemento sólo tiene marca inicial y se usa para que el visor termine la línea en el punto en el que encuentre el salto.
  - <HR>. Al igual que la anterior, sólo consta de una marca inicial. Se emplea para presentar una línea horizontal en el texto, ya sea usando caracteres o un gráfico, dependiendo del visor.
- 

## Listas

En realidad, también son marcas que permiten definir bloques, pero con características especiales. Las listas se emplean para presentar de forma ordenada una serie de líneas.

En función de su carácter lógico se distinguen los siguientes tipos de lista:

- Lista desordenada, <UL> (*Unordered List*).
- Lista ordenada, <OL> (*Ordered List*).
- Directorio, <DIR> (*Directory*).
- Menú, <MENU> (*Menu*).
- Lista de definición, <DL> (*Definition List*).

Exceptuando las listas de definición, el marcado de las líneas es igual en todos los casos: poniendo <LI> para marcar el principio de cada una, la línea termina cuando aparece un nuevo símbolo <LI> o se cierra la lista.

Para las listas de definición se emplean las marcas <DT> (*Definition Term*) y <DD> (*Definition Data*) para cada término y su correspondiente definición. Se pueden poner varios términos antes de una definición (marcas <DT>), pero no dos definiciones para un solo término.

Un ejemplo sería:

Esto es una lista desordenada:

```
<UL>
```

```
<LI>Primer elemento
```

```
<LI>Segundo elemento
```

```
</UL>
```

Esto es una definición:

```
<DL>
```

```
<DT>Perro  
<DD>Animal al que el hombre tiene la fea costumbre de morder  
</DL>
```

---

## Marcado Lógico de Frases

Existen multitud de marcas para indicar que una palabra o frase tiene una connotación especial.

Los elementos son:

- <CITE>, indica que el texto es una cita (por ejemplo de un título). Generalmente se representa en cursiva.
  - <CODE>, lo marcado es un ejemplo de código dentro del texto. Se representa usando un tipo de letra de caja fija.
  - <EM>, denota énfasis. Generalmente se representa con letra cursiva.
  - <KBD>, indica que el texto se introduce desde el teclado. Se representa usando un tipo de letra de caja fija.
  - <SAMP>, ejemplo, es decir, una secuencia de caracteres literales. Se representa usando un tipo de letra de caja fija.
  - <STRONG>, denota énfasis fuerte. Generalmente en negrita.
  - <VAR>, lugar de una variable, por ejemplo la que se le pasa a un programa. Se representa usando un tipo de letra de caja fija.
- 

## Marcado Tipográfico de Frases

Se usan para indicar explícitamente el formato tipográfico de una palabra o frase.

Los formatos y sus marcas correspondientes son:

- Negrita, <B></B>
- Cursiva, <I></I>
- Texto de teletipo (tipo de letra de caja fija) <TT></TT>

Aunque no están en el estándar, algunos visores pueden soportar otras marcas de formato tipográfico como <STRIKE> (texto tachado) o <U> (subrayado).

---



## Marcado de Anclaje de Hiperenlaces

Un caso especial de marcado es el representado por el elemento <A>, que se emplea para indicar que un texto hace referencia a otro, es decir, está anclado mediante un hiperenlace.

Para el marcado se emplean la marca inicial con atributos, el texto a anclar y la marca final. Los atributos pueden ser:

- HREF. El valor es el URI (*Uniform Resource Identifier*) del anclaje principal de un hiperenlace. El URI se puede referir a otro documento HTML, a un anclaje del mismo documento o a cualquier otro tipo de recurso (**FTP**, **Gopher**, etc.).
- NAME. Nombra un anclaje para poder ser usado como anclaje principal de un hiperenlace, es decir, el punto nombrado puede ser referenciado desde otro anclaje que emplee el atributo HREF. Por ejemplo, si en el documento test.html incluimos un anclaje <A NAME="indice">&iacute;ndice</A>, dentro del mismo lo podremos referenciar mediante <A HREF="#indice">Volver al &iacute;ndice</A>. Si queremos referirnos a ese punto dentro del documento desde otro documento que se encuentra en el mismo directorio, la referencia será <A HREF="test.html#indice">&iacute;ndice del test</A>.
- TITLE. Sugiere un título para el recurso destino. Este atributo es sólo informativo y puede ser usado para que se visualice al colocarnos sobre el enlace o para nombrar recursos que no incluyen un título, como gráficos.
- REL. relaciones descritas por el hiperenlace. El valor es una lista de nombres de relaciones separadas por blancos.
- REV. Igual que REL pero en dirección contraria (si A se relaciona con B por X, un enlace de A a B con REL="X" expresa lo mismo que un enlace de B a A con REV="X").
- URN. Especifica un identificador para el primer anclaje del hiperenlace. La sintaxis de los URN (*Uniform Resource Namer*) a\_n no está especificada.
- METHODS. Especifica métodos a usar para acceder al destino. Se escriben como una lista de palabras separadas por espacios y dependen del tipo de URI. Al igual que el elemento TITLE, son sólo orientativos para el visor.

---

## Imágenes

Para incluir imágenes en documentos HTML se emplea la marca <IMG>. Esta marca puede tener los siguientes atributos:

- SRC. Indica la fuente de la imagen, en realidad se trata de un enlace con el documento que contiene la imagen. En la práctica se suelen emplear solo imágenes en formatos de mapa de bits como gif o jpeg.
- ALT. Indica un nombre para referirnos a la imagen si ésta no se representa. Generalmente lo emplean los visores de solo texto o los visores gráficos cuando el usuario selecciona no cargar automáticamente las imágenes. Es el nombre que aparece habitualmente cuando interrumpimos la carga de una página o ésta se corta dejando imágenes sin traer.
- ALIGN. Alineación de la imagen respecto al texto, puede tomar los valores: TOP (arriba), MIDDLE (en medio) o BOTTOM (abajo). En general, si no se especifica, los visores asumen BOTTOM.
- ISMAP. Indica que la imagen es un mapa (lo veremos más adelante)

El único atributo imprescindible es el SRC (como es lógico, sin la imagen poca utilidad tiene la marca).

Una página con varias referencias a imágenes podría ser la siguiente:

```
<HTML>
<HEAD>
<TITLE>Página de prueba de imágenes</TITLE>
</HEAD>
<BODY>
<IMG SRC="foto.gif" ALT="Foto" ALIGN=MIDDLE>
Este soy yo.
<P>Selecciona lo que quieras de este mapa:
<A HREF="/cgi-bin/imagemap/mapa"><IMG SRC="mapa.gif" ISMAP></A>
</BODY>
</HTML>
```