



Escuela Superior Politécnica del Litoral
FACULTAD DE INGENIERIA
EN ELECTRICIDAD Y COMPUTACION

**“Diseño de un Sistema de Visualización y Control
para un Sistema de Apertura y Cierre Automático
Utilizando un PLC y la Plataforma de
Visualización Intouch”**

TRABAJO **de** **GRADUACION**

PREVIO A LA OBTENCION DEL TITULO DE
Ingeniero en Electricidad
Especialización Industrial

Presentado por:

Ricardo Noboa Llerena

Tito Farfán Cabrera

Guayaquil 1999

AGRADECIMIENTO

A DIOS,

A NUESTROS PADRES,

A NUESTROS HERMANOS,

Y A TODOS AQUELLOS QUE DE ALGUNA U OTRA MANERA PRESTARON SU COLABORACION EN FORMA DESINTERESADA PARA LA CULMINACION DE ESTE PROYECTO.

DEDICATORIA

A NUESTROS PADRES,

A NUESTROS HERMANOS,

A NUESTRAS FAMILIAS Y AMIGOS.

TRIBUNAL



Ing. Armando Altamirano

Sub-Decano



Ing. Norman Chootong

Director de Tesis



Ing. Dennis Cortés

Miembro Principal



Ing. Carlos Salazar

Miembro Principal

DECLARACION EXPRESA

La responsabilidad por los hechos, ideas y doctrinas expuestas en este proyecto, excepto en aquellas partes donde se anota explícitamente lo contrario, nos corresponden exclusivamente, y el patrimonio intelectual, a la ESCUELA SUPERIOR POLITECNICA DEL LITORAL.

(Reglamento de exámenes y títulos profesionales de la ESPOL).



.....
Esteban Ricardo Noboa Llerena



.....
Tito Germán Farfán Cabrera

RESUMEN

En el mundo moderno toda empresa debe ser eficiente y flexible si desea sobrevivir en el mercado, esto ha provocado un incremento en la demanda de sistemas de control industriales que permitan agilizar las operaciones de una industria y maximizar los recursos que ésta posea.

Bajo este contexto el presente trabajo está orientado a brindar un soporte en el diseño y desarrollo de un sistema de control y monitoreo para un sistema de apertura y cierre automático. En él se puede encontrar una aplicación sencilla que sirve como guía en la utilización del software de visualización "InTouch" y en la aplicación de la familia de PLC's serie S7-200 de Siemens.

Dentro de la variedad de funciones que posee el InTouch se han utilizado las más comunes como son: Animación Dinámica, Gráfico de Tendencias Históricas, Visualización de Alarmas, Impresión de Alarmas y los denominados "Scripts" que son pequeños programas que permiten agilizar el funcionamiento del Software de visualización.

INDICE GENERAL

AGRADECIMIENTO.....	ii
DEDICATORIA.....	iii
TRIBUNAL.....	iv
DECLARACION EXPRESA.....	v
RESUMEN.....	vi
INDICE GENERAL.....	vii
INDICE DE FIGURAS.....	x
INDICE DE TABLAS.....	xii
TABLA DE ABREVIATURAS.....	xiii
INTRODUCCION.....	1
CAPITULO 1	
SISTEMA SCADA.....	5
1.1 DEFINICION.....	5
1.2 COMPONENTES DE UN SISTEMA SCADA.....	6
1.2.1 EL CENTRO DEL CONTROL.....	6
1.2.2 SISTEMAS DE COMUNICACION.....	7
1.2.3 INTERFASE DE MONITOREO Y CONTROL.....	8
1.2.4 EQUIPAMIENTO CONTROLADO.....	10
CAPITULO 2	
DESCRIPCION DE LA PUERTA AUTOMATICA.....	12
2.1 GENERALIDADES DE LOS SISTEMAS ELÉCTRICOS Y ELECTRONICOS UTILIZADOS.....	12
2.1.1 MOTORES DC DE IMAN PERMANENTE.....	12
2.2 CONTROL DE VELOCIDAD DE MOTORES DC.....	13
2.2.1 CONTROL MEDIANTE TROCEADORES (PWM).....	14
2.3 COMPONENTES MECANICOS DEL SISTEMA.....	18
2.4 COMPONENTES ELECTRICOS Y ELECTRONICOS DEL SISTEMA.....	20
2.4.1 COMPONENTES ELECTRICOS.....	20
2.4.1.1 MOTOR ELECTRICO.....	20
2.4.1.2 CIRCUITO CONMUTADOR DE GIRO.....	21
2.4.2 COMPONENTES ELECTRONICOS.....	22
2.4.2.1 SENSORES DE ABIERTO Y CERRADO DE PUERTA.....	23
2.4.2.2 INDICADOR DE POSICION DE LA PUERTA.....	24
2.4.2.3 SISTEMA DETECTOR DE MOVIMIENTO.....	26
2.4.2.4 CONTROL DE VELOCIDAD DEL MOTOR DC DE IMAN PERMANENTE.....	28
2.4.2.4.1 MODULACION DE ANCHO DE PULSO CON C.I TEMPORIZADORES.....	30
CAPITULO 3	

SELECCIÓN DEL CONTROLADOR LOGICO PROGRAMABLE	34
3.1 CRITERIOS DE SELECCIÓN.....	34
3.2 ESTUDIO DE POSIBILIDADES.....	35
3.3 ELECCION DEL CONTROLADORPROGRAMABLE.....	36
3.4 TIPO Y TAMAÑO DE PLC.....	38
3.5 REQUERIMIENTOS DE ENTRADAS/SALIDAS.....	39
3.6 REQUERIMIENTOS DE MEMORIA Y PROGRAMACION.....	39
3.7 JUEGO DE INSTRUCCIONES.....	41
3.8 DESCRIPCION DE LA SERIE S7-200 DE SIEMENS.....	42
3.9 PRINCIPALES COMPONENTES DE UN MICRO-PLC S7-200.....	45
3.9.1 MODULOS DE AMPLIACION.....	46
3.10 ESTABLECER LA COMUNICACIÓN CON UNA CPU S7-200.....	47
3.11 SOFTWARE DE PROGRAMACION STEP7-MICRO/WIN.....	49
3.11.1 INTRODUCIR PROGRAMAS EN KOP.....	50
3.12 CARGAR EL PROYECTO EN LA CPU.....	54
3.13 CONMUTAR LA CPU A MODO RUN.....	55
3.14 OBSERVAR EL ESTADO KOP.....	55
3.15 INTRODUCIR PROGRAMAS EN AWL.....	56
3.16 AJUSTAR EL MODO DE OPERACIÓN DE LA CPU.....	58
3.17 CAMBIAR EL MODO DE OPERACIÓN CON STEP7-MICRO/WIN.....	59
3.18 EL CICLO DE LA CPU.....	60
3.19 LEER LAS ENTRADAS DIGITALES.....	61
3.20 EJECUTAR EL PROGRAMA.....	61
3.21 PROCESAR LAS PETICIONES DE COMUNICACIÓN.....	62
3.22 EJECUTAR EL AUTODIAGNOSTICO DE LA CPU.....	62
3.23 ESCRIBIR LAS ENTRADAS DIGITALES.....	62
3.24 INTERRUPIR EL CICLO.....	63
3.25 IMAGEN DEL PROCESO DE LAS ENTRADAS Y SALIDAS.....	64
3.26 CONTROL DIRECTO DE LAS ENTRADAS Y SALIDAS.....	65
3.27 DESCRIPCION GENERAL DEL PROCESO.....	65
3.28 TABLA DE SIMBOLOS.....	66
3.29 PROGRAMA GENERAL.....	67
CAPITULO 4	
IMPLEMENTACION DE LA INTERFASE MMI PARA EL CONTROL Y	
MONITOREO DEL SISTEMA DE APERTURA Y CIERRE AUTOMATICO	
UTILIZANDO IN-TOUCH	
4.1 INTRODUCCION.....	72
4.2 CARACTERISTICAS DE IN TOUCH.....	73
4.2.1 GRAFICOS ORIENTADOS A OBJETOS.....	73
4.2.2 ENLACES DE ANIMACION.....	74
4.2.3 ASISTENTES.....	74

4.2.4 SCRIPTS.....	75
4.2.5 REFERENCIA DINAMICA.....	75
4.2.6 ALARMAS DISTRIBUIDAS.....	75
4.2.7 TENDENCIA HISTORICA DISTRIBUIDA.....	76
4.3 EL WINDOW MAKER.....	76
4.4 DISEÑO DE LA INTERFASE MMI PARA EL CONTROL DEL SISTEMA DE APERTURA Y CIERRE AUTOMATICO.....	77
4.4.1 OBJETO DE ALARMAS HISTORICAS.....	82
4.4.2 IMPRESIÓN DE ALARMAS.....	91
4.4.3 GRAFICO DE TENDENCIA HISTORICA.....	95
4.4.4 ESCRITURA DE TAGNAMES A LOS ARCHIVOS HISTORICOS.....	98
4.4.5 HABILITACION DEL ALMACENAMIENTO DE ARCHIVOS HISTORICOS.....	99
4.4.6 ACTUALIZACION DE LOS GRAFICOS DE TENDENCIA HISTORICA.....	101
4.4.7 PANEL DE CONTROL.....	102
4.4.8 PANEL DE VISUALIZACION.....	105
4.4.9 SISTEMA DE APERTURA Y CIERRE.....	107
4.4.10 SCRIPT DE APLICACIÓN.....	107
BIBLIOGRAFIA.....	114

INDICE DE FIGURAS

2-1	Cortes transversales de dos motores convencionales de imán Permanente con polos magnetizados radialmente. (a) Dos polos.(b) Cuatro polos.....	13
2-2	Operación básica de un sistema de control de potencia mediante modulación por ancho de pulso.....	15
2-3	Formas de onda en proceso PWM.....	16
2-4	Fotografía del sistema prototipo de puerta automática.....	19
2-5	Esquema de puerta automática con todas sus partes remarcadas.....	20
2-6	Motor eléctrico de 24 Vdc de imán permanente utilizado en el proyecto.....	21
2-7	Circuito conmutador para activación y giro de motor de corriente directa.....	22
2-8	Esquema de los sensores de puerta abierta o puerta cerrada.....	24
2-9	Esquema de sistema detector de posición de la puerta.....	25
2-10	(a) Circuito transmisor. (b)Circuito receptor.....	26
2-11	Diagrama esquemático del circuito de control de velocidad del motor de corriente directa mediante modulación por ancho de pulso (PWM).....	30
2-12	Esquema general de circuito de control de velocidad de motor de corriente directa mediante técnica PWM aplicando dos circuitos temporizadores.....	31
2-13	Formas de onda del sistema de control de motor mediante un modulador por ancho de pulso mostrado en la figura 2-12.....	32
3-1	Micro PLC S7-200.....	43
3-2	Conexión básica.....	43
3.3	El PLC y su módulo de ampliación.....	46
3.4	Comunicación PPI con un CPU.....	47
3.5	Red maestro-esclavo.....	48
3.6	Ajuste de parámetros de comunicación entre PC y un CPU.....	49
3.7	Ventana del editor KOP.....	52
3.8	Elementos básicos de KOP.....	53
3.9	Visualización del estado de un programa en estado KOP.....	56
3.10	Ventana del editor AWL.....	57
3.11	Cambio del modo de operación del CPU con STEP-7 Micro/Win.....	59
3.12	El ciclo del CPU.....	60

4.1	Administrador de aplicaciones.....	77
4.2	Caja de diálogo "creación de nueva aplicación".....	78
4.3	Propiedades de la ventana "principal".....	79
4.4	Propiedades de la ventana "aviso".....	79
4.5	Ventana "principal".....	81
4.6	Selección de Wizard.....	83
4.7	Objeto de alarmas.....	84
4.8	Configuración de alarma.....	84
4.9	Formato del mensaje de alarma.....	86
4.10	Selección de grupo de alarmas.....	89
4.11	Caja de diálogo "Agregar Grupo de Alarmas".....	89
4.12	Diccionario de Tagnames.....	90
4.13	Definición de alarma para "ALARMA".....	91
4.14	Propiedades de Alarmas.....	93
4.15	Formato para la impresión de alarma.....	94
4.16	Caja de diálogo "Configuración de Tendencia Histórica".....	96
4.17	Definición de Tagname "CONT".....	99
4.18	Caja de diálogo "Propiedades de almacenamiento Histórico".....	100
4.19	Panel de Control.....	103
4.20	Panel de visualización.....	106
4.21	Sistema de apertura y cierre.....	107

INDICE DE TABLAS

3.1	Característica de la serie S7-200	44
3.2	Lista de Símbolos utilizados en el Programa	67
4.1	Lista de Tagnames	113

TABLA DE ABREVIATURAS

SCADA.....	Supervisión, control y adquisición de datos.
PLC.....	Control lógico programable.
RTU.....	Unidad terminal remota.
Air Gap.....	Entrehierro.
Vs.....	Voltaje de alimentación.
Rn.....	Resistor.
Dn.....	Diodo rectificador.
t.....	Tiempo (variable).
V _{LD}	Voltaje de carga.
ma.....	Miliamperios.
V _{dc}	Voltaje de corriente directa.
V _{ac}	Voltaje de corriente alterna.
DC.....	Corriente directa.
AC.....	Corriente alterna.
Qm.n.....	Salida del PLC Siemens.
Im.n.....	Entrada del PLC Siemens.
L+.....	Salida positiva de fuente interna del PLC Siemens.
M.....	Salida negativa de fuente interna del PLC Siemens.
A+.....	Entrada positiva analógica de módulo EM235.
A-.....	Entrada negativa analógica de módulo EM235.
S7-CPU212.....	Serie y modelo de PLC Siemens.
EM235.....	Módulo de expansión, entradas y salidas analógicas.
Cn.....	Capacitor.
Pn.....	Potenciómetro.
AOn.....	Amplificador operacional.
E/S.....	Entrada/salida.
INTOUCH.....	Software para supervisión y control de procesos industriales de la Corporación Wonderware.
PWM.....	Modulación de ancho de Pulso.
CPU.....	Unidad de proceso central.
MICROWIN.....	Software de programación para el PLC Siemens S7-200 con CPU 212.
HARDWARE.....	Dispositivos físicos de un sistema de control.
SOFTWARE.....	Elementos intangibles que conforman un sistema de control (Programas).
MMI.....	Interface Hombre-Máquina.
DDE.....	Intercambio dinámico de datos.

Windowmaker.....	Paquete desarrollador de INTOUCH, donde se crean gráficos orientados a objetos.
Windowviewer.....	Paquete de visualización de INTOUCH, donde la animación y comunicación toma lugar.
Runtime.....	Tiempo de ejecución.
Tagname.....	Variables, (campo de una base de datos).
Windows.....	Sistema operativo.
Popup.....	Superposición de ventana.
Toolbox.....	Caja de herramientas de INTOUCH.
Wizard.....	Iconos de instrumentación y objetos prediseñados del INTOUCH.
Trend.....	Gráfico de una variable.
Scripts.....	Ventana donde se programa la lógica en INTOUCH para crear procesos de simulación.
PC.....	Computador personal.
PPL.....	Interface punto a punto.
MPL.....	Interface multi punto.
KOP.....	Lenguaje lógico de contactos.
AWL.....	Lenguaje lógico de instrucciones.

INTRODUCCION

Debido al gran desarrollo y avance tecnológico de los sistemas de control y monitoreo industrial es que en la actualidad tenemos en el mercado muchos productos competitivos y de diferentes procedencias. Estos productos han sido desarrollados tanto a nivel de "Hardware" así como en "software"; por lo tanto es muy importante que el profesional del área industrial se mantenga informado y actualizado referente a las ventajas y aplicaciones de los mismos, todo esto con la finalidad de mantener de forma óptima y adecuada la planta o sistema industrial a su cargo.

Este escrito de tesis está orientada a cumplir con lo remarcado en el párrafo anterior, de manera que lo hacemos explicando el desarrollo de un ejemplo práctico el cual está representado en un sistema de "Apertura y cierre automático" de una compuerta que podríamos decir que está mas orientada a ser aplicada para el ingreso de un centro comercial.

Ha sido de gran provecho para nosotros el escoger este t3pico del 3rea industrial y m3s a3n enfocarnos a la "Automatizaci3n Industrial", para lo cual nos centramos al estudio de los sistemas de control supervisorio y adquisici3n de datos (SCADA) y los controladores l3gicos programables (PLC) que constituyen el par Software y Hardware respectivamente. Esta tecnolog3a combinada con dispositivos sensores y de control electr3nico como lo son los detectores de movimiento mediante barrera infrarroja pasiva y el control de velocidad de motor de corriente directa mediante t3cnica de modulaci3n por ancho de pulso (PWM), lo que hace de este proyecto un sistema completo y pr3ctico para el entendimiento del ingeniero o t3cnico industrial ya que este escrito detalla paso a paso lo concerniente tanto a los sistemas de automatizaci3n industrial asi como los electr3nicos cl3sicos e industriales.

El desarrollo de este documento est3 dado en cuatro capitulos, de los cuales procederemos a hacer una peque3a introducci3n a continuaci3n:

El Capítulo I hace referencia a los sistemas SCADA's dando a conocer la interpretación de sus siglas, las mismas que son de origen inglés además de dar su respectiva definición; también se indican sus componentes, características, aplicaciones y en especial se menciona sus sistemas anexos de interconexión con los fabricantes de PLC's más reconocidos dando así a conocer el sistema de automatización planteado en este proyecto.

La descripción física del sistema de apertura y cierre automático planteado como proyecto en esta tesis se detalla en el Capítulo II. Aquí se desarrollan tópicos importantes tales como la descripción y explicación de los componentes mecánicos, eléctricos y electrónicos del proyecto, lo cual se lo realiza de forma textual con la ayuda de gráficos, esquemas de circuitos y diagramas de tiempo que muestran las formas de onda y señales eléctricas.

El capítulo III detalla las razones y criterios por las cuales se estableció la utilización del PLC S7-200 con CPU212 de Siemens, luego se pasa a dar una explicación del software Micro/win 2.1 que es el que se utiliza con el PLC ya mencionado

y en la última parte de este capítulo se hace una descripción del problema planteado con todas sus especificaciones y de manera inmediata se desarrolla su solución y se presenta el programa de control del sistema además de mostrar la respectiva tabla de símbolos y dirección de las variables utilizadas en este proyecto.

Finalmente en el Capítulo IV se realiza una introducción al INTOUCH que es el SCADA con el que se realiza el monitoreo y control de este proyecto, en el mismo se entrega conceptos generales así como se da una explicación de las diversas utilidades del mismo. En la parte final de este capítulo se lista el programa (Script) con el cual se lleva a cabo la simulación y control del sistema de apertura y cierre automático.

Luego de la lectura de los cuatro capítulos de este escrito se estará en condiciones de poder elaborar o desarrollar soluciones de automatización a cualquier planteamiento requerido en el área industrial o de la producción en general.

CAPITULO I

SISTEMAS SCADA

1.1 DEFINICION.

SCADA ,de sus siglas en inglés, significa Control Supervisorio y Adquisición de Datos. La principal función de este tipo de sistemas es el monitoreo, control y diagnóstico en forma local y/o remota del equipamiento que lleva a cabo un proceso.

El espectro de aplicaciones de estos sistemas en la industria es muy amplio dentro de las cuales tenemos:

- Tratamiento de Aguas.
- Sistemas de Distribución, Transmisión y Generación Eléctrica.

- Aceite y Gas.
- Sistemas de Riego.
- Sistemas de Comunicación.
- Industrias Químicas.

1.2 COMPONENTES DE UN SISTEMA SCADA.

Podemos decir que un sistema SCADA encierra 3 grandes áreas, las cuales son:

- Redes de Computadoras.
- Comunicaciones.
- Electrónica Industrial.

Cada una de estas áreas esta respectivamente relacionada con los componentes de un sistema SCADA. Dichos componentes son:

- Centro de Control.
- Sistemas de Comunicación.
- Interface de Monitoreo y Control (PLC, RTU).
- Equipamiento controlado.

1.2.1 EL CENTRO DE CONTROL.

El Centro de Control es el lugar donde se efectúa el monitoreo, control y diagnóstico del proceso por parte de los operadores del proceso. Es el lugar donde son enviados todos los datos que provee el sistema Scada y que reflejan el estado del proceso controlado, en este centro generalmente se implementa una red de computadoras con la finalidad de procesar los datos proporcionados por el SCADA.

1.2.2 SISTEMAS DE COMUNICACION.

Los sistemas de comunicación constituyen el medio por el que todos los elementos del Scada son enlazados o comunicados de tal manera que la comunicación entre estos elementos sea en todas las direcciones. En estos sistemas de comunicación es muy común la utilización de los denominados "Protocolos de Comunicación" que no son otra cosa que el "Lenguaje" en el que los diferentes elementos del Scada se comunican entre sí. En un mismo sistema Scada es posible que se utilicen más de un Protocolo, de hecho así sucede puesto que la red de computadoras del Centro de Control puede tener su propio Protocolo mientras que los PLCs y el equipo controlado pueden comunicarse en un Protocolo diferente, así mismo el centro de control y el PLC o RTU pueden comunicarse utilizando otro sistema de Protocolo

diferente. De igual manera existen variantes en cuanto a los medios de comunicación utilizados, los más comunes son:

- Comunicación Vía Radio.
- Comunicación Vía Satélite.
- Comunicación serial RS-232.
- Comunicación serial RS-485.

1.2.3 INTERFASE DE MONITOREO Y CONTROL.

Los elementos más utilizados en este componente de un sistema SCADA son los PLC's (Controladores Lógicos Programables) y las RTU (Unidad Terminal Remota). Esta interface es la encargada de recibir las señales que provienen del proceso controlado, procesar esas señales de acuerdo a la programación efectuada en los mismos y reportar los resultados del procesamiento al centro de control.

Un PLC puede ser definido como un sistema electrónico digital diseñado para ser utilizado en industrias, los mismos que utilizan una memoria programable para almacenar internamente instrucciones que llevan a cabo funciones específicas tales como funciones lógicas, secuenciales, de temporización,

conteo y aritméticas para controlar a través de módulos de entrada/salida analógicos o digitales diversos tipos de máquinas o procesos.

El PLC examina las señales de entrada provenientes de un proceso y lleva a cabo instrucciones lógicas (las cuales han sido programadas en su memoria) sobre éstas señales, produciendo señales de salida que comandan equipos o máquinas. Las interfaces estandarizadas integradas en un PLC le permiten conectarse directamente a actuadores y transductores de un proceso como bombas y válvulas, sin la necesidad de circuitería intermedia o relés.

Una RTU realiza exactamente lo mismo que un PLC, la diferencia radica en que una RTU es un equipo con mayor capacidad de comunicación y es utilizado generalmente en aplicaciones que involucran grandes distancias.

Entre los fabricantes más conocidos de PLC's y RTU's tenemos:

- Allan Bradley CO.
- General Electric Fanuc.
- Mitsubishi Electric Ltd.
- Gould Ltd.
- Siemens Ltd.
- Texas Instruments.

- Telemecanique.
- Toshiba.
- Westinghouse.
- Motorola.

1.2.4 EQUIPAMIENTO CONTROLADO.

El equipamiento controlado son todos aquellos equipos que están involucrados en el proceso que se controla como por ejemplo: Bombas, motores, generadores, mezcladores, calderas, bandas, etc. El PLC o RTU es conectado a estos equipos ya sea directamente o a través de los denominados Sensores o Transductores, los cuales tienen la tarea de acondicionar la señal proveniente del proceso para que pueda ser interpretada por el PLC o RTU. Como podemos entender es en este campo en el que se ve involucrada el área de Electrónica Industrial al que hacíamos mención anteriormente.

Dentro del amplio campo de transductores o Sensores utilizados en las diferentes aplicaciones de los sistemas SCADA los más comúnmente utilizados son los siguientes:

- Sensores de Presión.

- Sensores de Temperatura.
- Sensores de Nivel.
- Sensores de Radiación.
- Sensores de Humedad.
- Tacogeneradores.

Como podemos observar estos dispositivos generalmente sensan una variable mecánica y su misión es entregar una señal eléctrica o electrónica para que sea procesada por el sistema. Esta señal electrónica que ya han sido estandarizadas puede ser de corriente o voltaje. Las señales de salida de los sensores más utilizadas son:

De corriente: 0 - 20mA

4 - 20mA

De voltaje: 0 - 5V

-5 - +5V

0 - 10V

-10 - +10V

Puesto que estos son los rangos de las señales que entran al PLC o RTU es necesario definir escalas basadas en los rangos de variación que tienen los parámetros que se están midiendo dentro del proceso.

CAPITULO II

DESCRIPCION DE LA PUERTA AUTOMATICA

2.1 GENERALIDADES DE LOS SISTEMAS ELECTRICOS Y ELECTRONICOS UTILIZADOS

2.1.1 MOTORES DC DE IMAN PERMANENTE.

En un motor de corriente directa convencional de imán permanente simplemente se reemplazan los electroimanes del devanado de campo por imanes permanentes montados en el estator.

En su forma más sencilla, el motor contiene imanes permanentes que están magnetizados radialmente, como se muestra en la figura 2.1(a). El flujo

magnético emerge de la cara del polo norte, pasa a través del entrehierro (air gap), a través del núcleo convencional del rotor, y vuelve a entrar por la cara del polo sur. La trayectoria del flujo magnético se completa a través de la armazón de acero del motor, igual que lo haría en una máquina de campo devanado.

También hay disponibles diseños de cuatro y seis polos, para los cuales sólo se tiene en cuenta que al aumentar el número de polos de campo el devanado de armadura debe rediseñarse para que las bobinas individuales de la armadura tengan el mismo espaciado mecánico que el de los polos adyacentes, como se muestra en la figura 2.1(b).

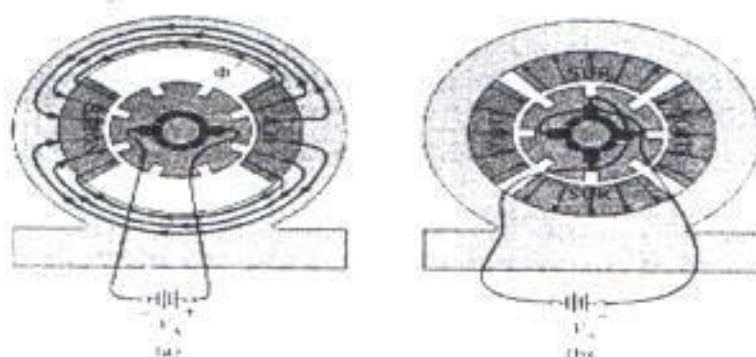


Figura 2.1

Cortes transversales de dos motores convencionales de imán permanente con polos magnetizados radialmente. (a) Dos polos. (b) Cuatro polos.

2.2 CONTROL DE VELOCIDAD DE MOTORES DC

En la mayoría de las situaciones industriales de control, los motores se operan directamente de las líneas de suministro de corriente alterna o de corriente directa o continua. En esta situación, el comportamiento operativo del motor es determinado por la naturaleza de la carga mecánica conectada al eje del motor. En términos sencillos, si la carga es fácil de manejar, el motor tenderá a entregar un par relativamente pequeño, y girará a una velocidad alta. Si la carga es difícil de manejar, el motor tenderá a entregar mucho par, y girará a una velocidad menor.

En resumen, el comportamiento del motor es establecido por su carga (para un voltaje de alimentación fijo), y el operador no tiene control sobre el comportamiento del motor.

En las instalaciones industriales modernas, se requiere que el operador sea capaz de controlar la velocidad del motor. Tal control generalmente se logra mediante la conmutación adecuada de dispositivos electrónicos, tales como: tiristores, transistores de potencia o triacs que conforman un sistema de control de velocidad o sistemas de manejo.

2.2.1 CONTROL MEDIANTE TROCEADORES (PWM).

Ha sido comprobado que los motores grandes de corriente directa son controlados mejor por tiristores de alta potencia, pero los motores de corriente directa de imán

permanente pequeños y medianos, y algunos de estos motores sin escobillas son controlados más exitosamente por transistores de conmutación conectados en serie, manejados mediante técnicas para troceo o control del voltaje de alimentación tales como el control por modulación de ancho de pulso (PWM).

En un sistema de control por modulación de ancho de pulso existen tres partes esenciales:

1. Un generador de rampa, operando generalmente a una frecuencia constante.
2. Un comparador, para detectar el momento en que el voltaje de la rampa ha excedido el voltaje de la señal de control.
3. Un dispositivo electrónico que conmute la potencia de carga en el momento en que el comparador detecte el punto crítico de la forma de onda de la rampa.

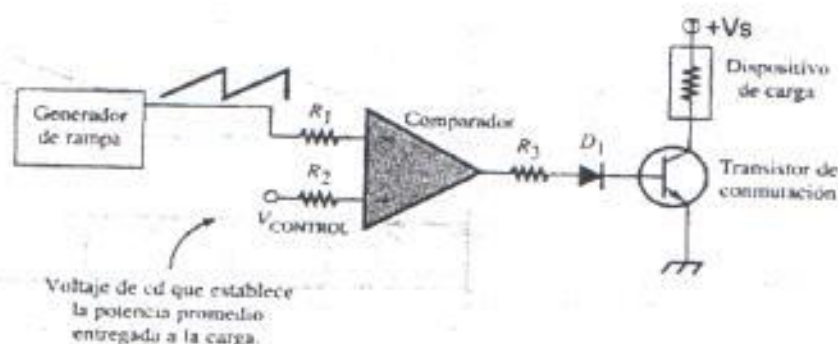


Figura 2.2

Operación básica de un sistema de control de potencia mediante modulación por ancho de pulso

En la figura 2.2, se muestra un esquema básico de control de velocidad mediante modulación por ancho de pulso, se puede observar que el comparador consiste en un amplificador operacional, y el interruptor electrónico es un transistor bipolar que opera en configuración emisor común. A continuación se describe su funcionamiento haciendo referencia en las formas de onda de la figura 2.3.

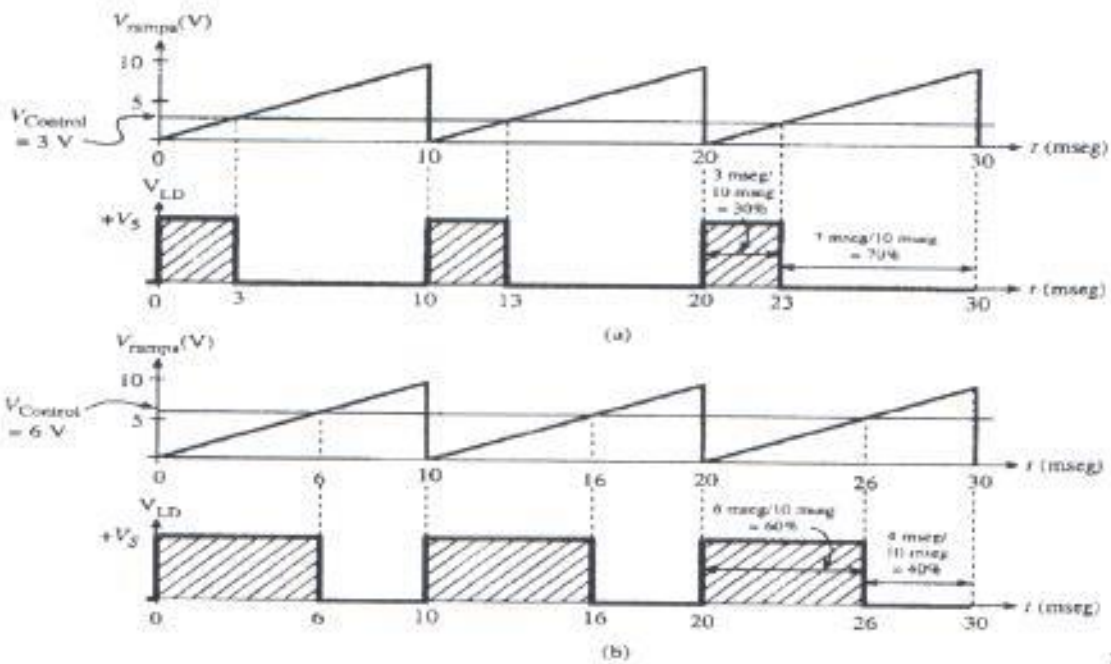


Figura 2.3

Formas de onda en proceso PWM.

En la figura 2.3 (b) el voltaje de control ha aumentado a 6 V. Por lo tanto su ciclo de trabajo varía al 60% aumentando su voltaje promedio a:

$$V_{LD(prom)} = 0.60 * V_s \quad (2-3)$$

Variando el voltaje de control, hemos variado o modulado el ancho de pulso de la carga. Esto varía el voltaje y la potencia promedio de la carga, lo cual es el concepto u objetivo principal de la modulación por ancho de pulso.

La modulación de ancho de pulso conmutada por transistores tiene la ventaja fundamental sobre la operación de la carga mediante transistores lineales. Debido a que el consumo interno de potencia y corriente es casi cero.

2.3 COMPONENTES MECANICOS DEL SISTEMA

La puerta está apoyada sobre una base de madera prensada, la cual ha sido diseñada a escala adecuada para el prototipo como se muestra en la figura 2.4 (a), (b) , y sobre la cual se colocarán los implementos y dispositivos del sistema.

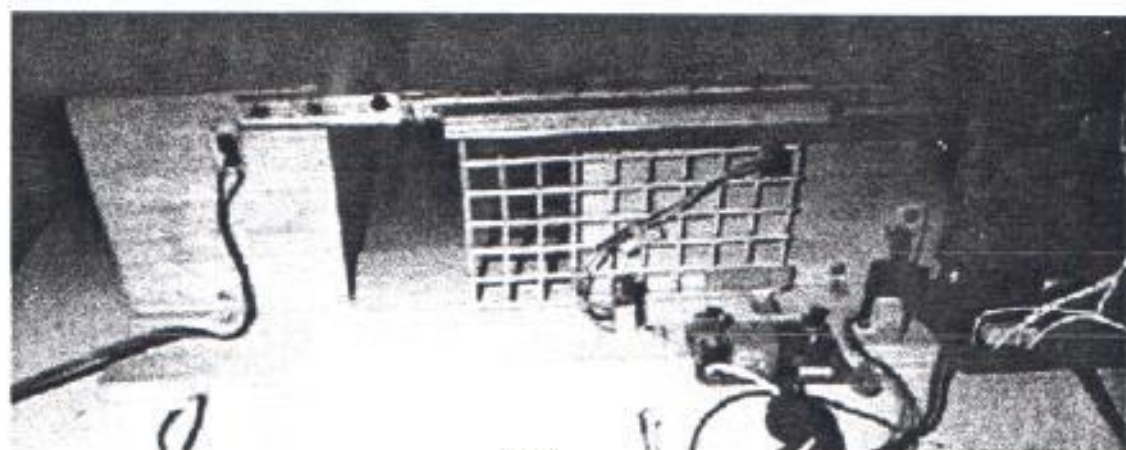
En el momento en que la rampa comienza a ascender, el voltaje de control positivo excede a V_{ramp} . Por tanto el comparador (opamp) produce una saturación positiva. Que satura el transistor, encendiendo el interruptor de potencia. A través de la carga aparece el voltaje de suministro completo V_s .

V_{ramp} va aumentando hacia su valor pico de 10 V. En el momento en que V_{ramp} excede a V_{contr} (3V en figura 2.3 (a)), el comparador conmuta a saturación negativa. Cuando la salida del comparador se vuelve negativa, polariza inversamente el diodo y la unión B-E del transistor. Por tanto, el transistor inmediatamente conmuta a apagado, y la carga se desenergiza, y permanece así el resto del ciclo de la rampa.

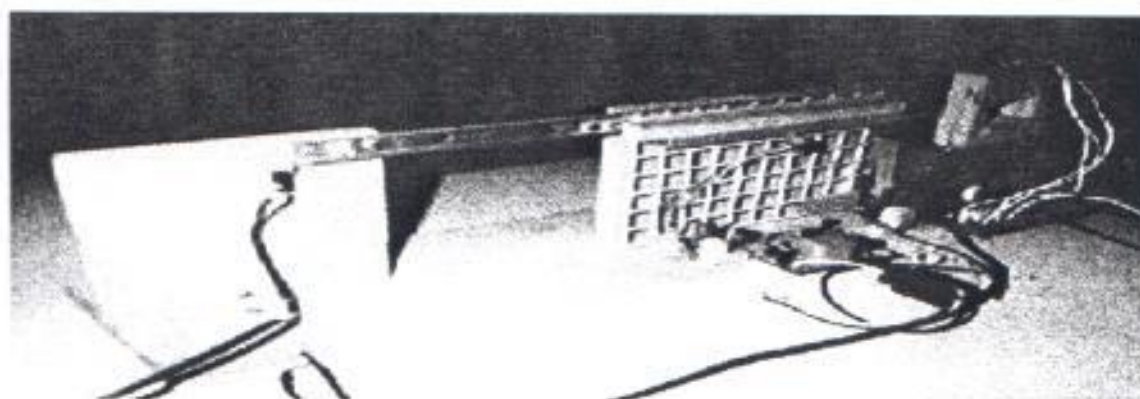
$$\text{duty cycle} = tp/T \quad (2-1)$$

Este proceso se repite para cada ciclo del oscilador de rampa, produciendo un ciclo de trabajo (2-1) del 30% de la onda de carga como se expresa en la siguiente relación:

$$V_{LD(prom)} = 0.30 * V_s \quad (2-2)$$



(a)



(b)

Figura 2.4

Fotografía del sistema prototipo de puerta automática.

La forma para el deslizamiento de la puerta está basado en un riel metálico móvil, el mismo que tiene su parte fija atornillada a la pared de madera y en su parte móvil está sujeta la rejilla que representa la puerta a operar junto con una

cremallera longitudinal de acero, la cual es el dispositivo de enlace o transmisión con el piñón giratorio que se encuentra montado sobre el eje del motor de corriente directa que opera el sistema, tal como se muestra en la figura 2.5.

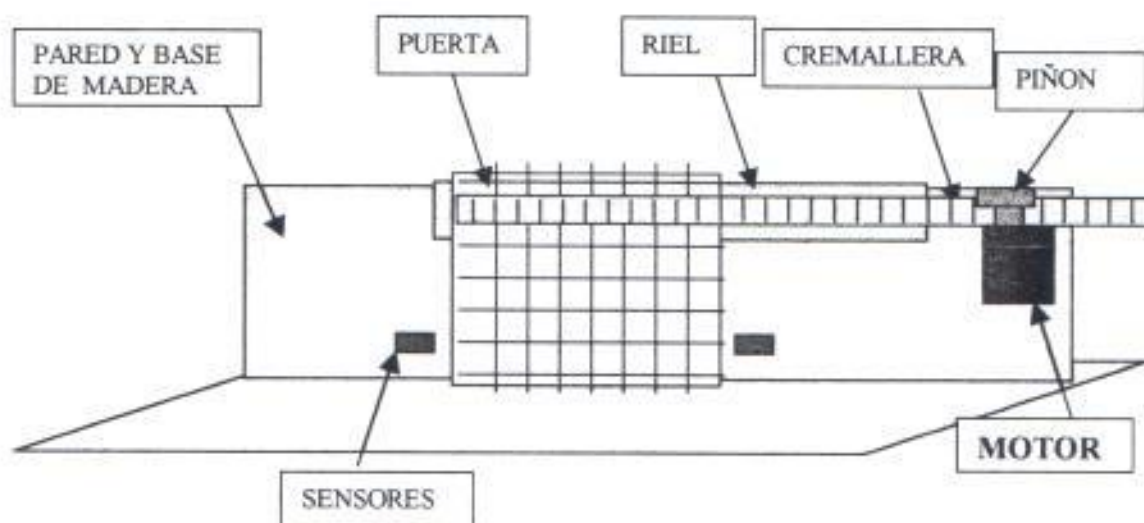


Figura 2.5

Esquema de puerta automática con todas sus partes remarcadas.

2.4 COMPONENTES ELÉCTRICOS Y ELECTRÓNICOS DEL SISTEMA

2.4.1 COMPONENTES ELÉCTRICOS

2.4.1.1 MOTOR ELECTRICO

El sistema eléctrico está conformado básicamente por un pequeño motor de 12 voltios de corriente directa (Vcd) de imán permanente con un consumo de

corriente de 120 miliamperios (ma). Este motor tiene incluido un sistema reductor mecánico que se encarga de establecer la velocidad de giro del eje de transmisión en 30 revoluciones por minuto (RPM) a voltaje nominal.

Al eje del motor está sujeto el piñón que se encuentra acoplado a la cremallera de la puerta, el cual al girar hará desplazar a la misma, ya sea para abrir o cerrar la puerta conforme este gire en uno u otro sentido. En la figura 2.6 se muestra un gráfico del motor utilizado en este proyecto.



Figura 2.6

Motor eléctrico de 24 Vdc de imán permanente utilizado en el proyecto.

2.4.1.2 CIRCUITO CONMUTADOR DE GIRO

El control de giro del motor se lleva a cabo a través de un circuito eléctrico de conmutación conformado por dos relés de 24 voltios de corriente directa de dos vías de contactos. En los terminales de estos contactos se encuentran

conectados los puntos de polarización del motor por un extremo y en el otro los terminales de activación del control electrónico de velocidad del motor (M+ y M-). Las bobinas de los relés están controladas por las señales de salida Q0.0 y Q0.1 del PLC para hacer girar al motor en uno u otro sentido. El esquema del circuito de control de giro se muestra en la figura 2.7

CONTROL DE ACTIVACION DE MOTOR
DC CON INVERSION DE GIRO

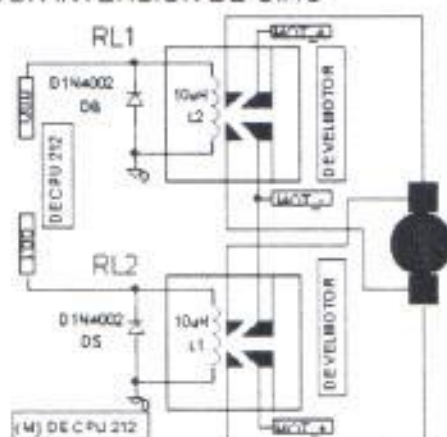


Figura 2.7

Circuito conmutador para activación y giro de motor de corriente directa.

2.4.2 COMPONENTES ELECTRÓNICOS

La estructura electrónica de este proyecto está distribuida básicamente para la implementación de sensores, tales como: sensores de fin de carrera (puerta abierta

y puerta cerrada), sensores infrarrojos de proximidad o movimiento; además, se estableció un sistema detector de posicionamiento, así como, un control de velocidad de motor. Todos estos sistemas se detallan a continuación:

2.4.2.1 SENSORES DE ABIERTO Y CERRADO DE PUERTA

Esta parte está implementada a base de dos módulos interruptores opto-acopladores externos con su respectivo circuito de polarización acondicionado a los 24Vdc de la fuente de alimentación, así como se puede ver en el esquema de la figura 2.8.

Los módulos interruptores opto-acopladores de haz de luz infrarroja permanentemente detectarán la interrupción del mismo debido a el cruce de una platina de hierro que se encuentra sujeta al borde de la puerta. Esta acción hará que force el transistor de salida del opto-acoplador cambie a un estado de corte, y envíe una señal de alto (24 Vcd) a las entradas del CPU (I0.1 – I0.2), ya que este estaba en un estado de saturación, con lo cual envía una señal de bajo (cero voltios) a los terminales de entrada del PLC (I0.1 – I0.0) que en estado normal está en estado alto (veinticuatro voltios), simulando así la acción de dos fines de carrera que indicarán si la puerta se encuentra en estado abierto o cerrado.



Figura 2.8

Esquema de los sensores de puerta abierta o puerta cerrada.

2.4.2.2 INDICADOR DE POSICIÓN DE LA PUERTA

El sistema de detección de posición de la puerta se fundamenta en un divisor de voltaje, determinado por una resistencia fija de 2K y un potenciómetro ajustable de 14 vueltas tipo trimmer de 0.5K como se muestra en el esquema de la figura 2.9. El divisor de voltaje que está conformado por estas dos resistencias, está conectado a la fuente de voltaje de 24 Vdc del CPU (L+ - M) el cual servirá de referencia para alimentar una porción controlada de este voltaje a través del terminal medio del potenciómetro en referencia al terminal negativo de la fuente hacia la entrada de señal analógica de voltaje (A+ - A-) del módulo de

entradas analógicas EM235. Esta señal de voltaje se ajustará automáticamente con el movimiento del eje del potenciómetro que se encuentra acoplado mecánicamente al piñón del eje del motor eléctrico, es decir, conforme gire el motor, se ajustará el potenciómetro con diferentes valores de resistencia y esto dará como resultado diferentes niveles de voltaje (0 – 10Vdc) en la entrada (A+) del EM235. Esta señal es receptada por el CPU del plc y de este será transmitida al computador donde se encuentra activo el programa SCADA que está enlazado al PLC, y que se encarga de procesar la misma para determinar la respectiva ubicación de la puerta y adecuarla a la ventana abierta en la pantalla del computador para establecer el monitoreo en tiempo real del sistema físico implementado.

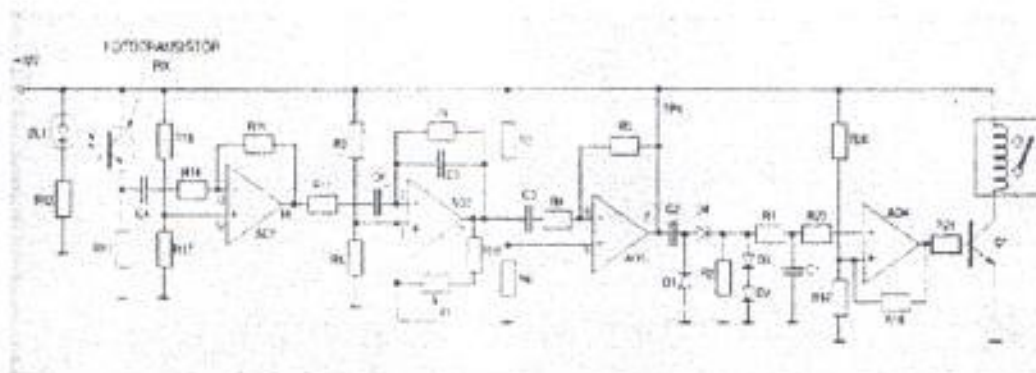
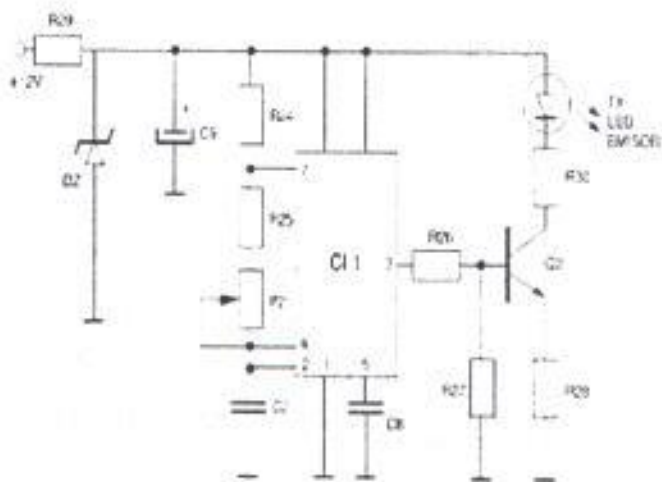


Figura 2.9

Esquema de sistema detector de posición de la puerta.

2.4.2.3 SISTEMA DETECTOR DE MOVIMIENTO.

(a)



(b)

Figura 2.10

(a) Circuito transmisor. (b) circuito receptor.

Este circuito consiste en un sistema de detección de interrupción de una barrera que opera en base a un haz de infrarrojo invisible con un alcance promedio de 4 metros.

Se trata de un circuito transmisor y un circuito receptor, estableciéndose entre ambos un enlace infrarrojo que al ser interrumpido activa un relé que operará el dispositivo deseado, en este caso las entradas del I0.1 y I0.2 del PLC para exterior e interior de la puerta respectivamente.

En la figura 2.10. (a) se muestra el esquema del circuito transmisor que está conformado básicamente por un oscilador biestable de onda cuadrada cuya frecuencia se fija por P2. Este circuito tiene como elemento principal el circuito integrado NE 555, el mismo que es un Temporizador – oscilador configurable, esta vez como oscilador biestable cuya salida es aplicada a Q2 que se encarga de entregar la corriente necesaria al LED emisor infrarrojo para que este emita.

El circuito receptor de este sistema infrarrojo se muestra en la figura 2.10. (b), este consiste de un amplificador sintonizado de tres etapas, el mismo que está constituido principal mente por amplificadores operacionales que permitirán la acción de un relé cuando no se reciba la señal de transmisión.

En este circuito la función del LED DL1 es la de mantener iluminado el fototransistor para que el equipo pueda trabajar en la oscuridad absoluta, dándole una pequeña polarización para que pueda operar correctamente.

El amplificador A01 trabaja con corriente alterna con una ganancia que depende de la relación R14 y R15 teniendo que superar un valor fijado por el divisor R16 y R17. A02 es un amplificador filtro activo selectivo cuya frecuencia central está dada por $R8 - C6 - R11$ pudiendo variar la ganancia del mismo y la frecuencia de recepción a través de P1. El A03 proporciona una ganancia adicional para luego de convertir la señal de corriente alterna en otra de corriente continua por medio de D4 y sus componentes asociados, esta señal dispara un Schmitt Trigger formado por A04, lo que hará que desactive el relé al pasar Q3 al estado de corte.

2.4.2.4 CONTROL DE VELOCIDAD DEL MOTOR DC DE IMAN PERMANENTE

El proyecto tiene implementado un sistema de control de velocidad para el motor dc de imán permanente que se está utilizando. Este control de velocidad va a ser regulado por medio de un nivel de voltaje (2 - 8Vdc) que será suministrado por el PLC a través de la salida de voltaje del módulo E/S

analógicas EM235, el mismo que va a ser comandado por la variable de control que se establece en un potenciómetro deslizante (slider) del grupo de elementos de control graficados en las pantalla de INTOUCH.

El diagrama esquemático del circuito se muestra en la figura 2.11. En este se puede notar que el control está basado en la interconexión de dos configuraciones del circuito integrado temporizador 555.

La primera es una etapa generadora de una especie de onda rampa para activar el disparo del segundo 555 que está en configuración de un circuito de un solo disparo variable en función del nivel de voltaje de control aplicado a este. Este esquema completo constituye un circuito de control de velocidad mediante técnica de modulación de ancho de pulso (PWM). El funcionamiento específico de este circuito se detalla a continuación.

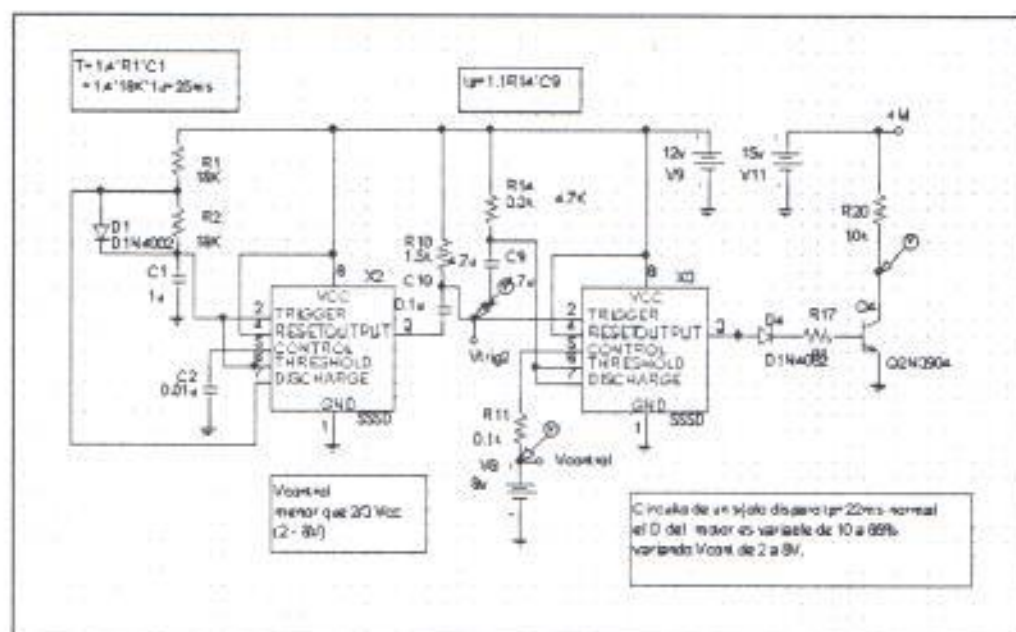


Figura 2.11

Diagrama esquemático de circuito de control de velocidad de motor de corriente directa mediante modulación por ancho de pulso (PWM).

2.4.2.4.1 MODULACIÓN DE ANCHO DE PULSO CON C.I. TEMPORIZADORES.

El temporizador-oscilador 555 es un circuito integrado útil para lograr modulación por ancho de pulso. En una configuración práctica y completa de un sistema de modulación por ancho de pulso se utiliza dos integrados 555.

El primer 555 sirve como oscilador de frecuencia constante, y un segundo temporizador sirve como circuito de un solo disparo variable para suministrar el pulso de encendido al transistor de conmutación de la carga. En la figura 2.12 se presenta el esquema completo de este circuito mencionado, donde el primer 555 opera como oscilador, usando el flanco negativo de su salida principal para disparar un segundo 555 que opera como circuito de un solo disparo variable. Entonces amplificamos la forma de onda de la salida principal del circuito de un solo disparo para operar el devanado de armadura de un motor de dc de imán permanente.

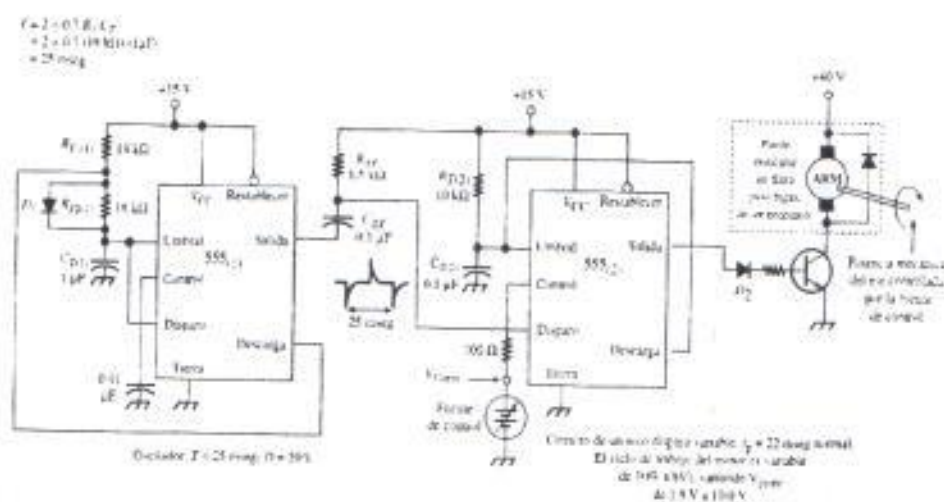


Figura 2.12

Esquema general de circuito de control de velocidad de motor de corriente directa mediante técnica PWM aplicando dos circuitos integrados temporizadores.

En la figura 2.13 se muestran las formas de onda para el circuito anterior y en estas gráficas se puede notar el resultado para diferentes valores de V_{contr} que está en valores de 4, 6, y 8 V generando en la salida frecuencias (t_p) de 6, 15 y 22 mseg respectivamente, que con relación a los 25 mseg de T de la señal del circuito oscilador representa niveles de ciclo de trabajo del 24, 60, y 88 %.

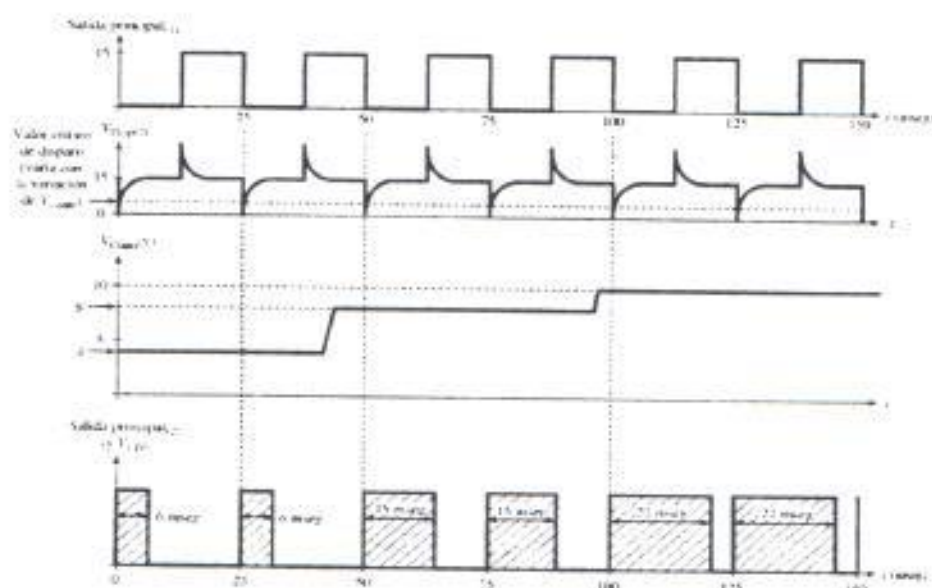


Figura 2.13

Formas de onda del sistema de control de motor mediante un modulador por ancho de pulso mostrado en la figura 2-12.

$$\text{ciclo de trabajo} = (t_{\text{ENC}} / T) * 100\% \quad (2-4)$$

Conforme se va aumentando el ciclo de trabajo del devanado de armadura del motor, se va creando más par sobre el eje que gira a una velocidad mayor por el efecto del aumento de la potencia del mismo.

A continuación se detallan todas las expresiones y relaciones válidas para un oscilador con 555:

Para un oscilador con dos resistores de temporización:

$$t_{ENC} = 0.7 * \tau_{carga} = 0.7 * (R_{T1} + R_{T2}) * C_T \quad (2-5)$$

$$t_{APAG} = 0.7 * \tau_{descarga} = 0.7 * R_{T2} * C_T \quad (2-6)$$

$$T = t_{ENC} + t_{APAG} \quad (2-7)$$

$$f = 1/T = 1.4 / (R_{T1} + 2R_{T2}) * C_T \quad (2-8)$$

Para un 555 de un solo disparo sin entrada de control:

$$T_p = 1.1 * \tau_{carga} = 1.1 * R_T * C_T \quad (2-9)$$

CAPITULO III

SELECCIÓN DEL CONTROLADOR LOGICO PROGRAMABLE

3.1 CRITERIOS DE SELECCIÓN.

Cuando nos encontramos en un proyecto de automatización, debe recordarse que el controlador es una herramienta o vehículo utilizado para llevar a cabo las tareas necesarias. Estas tareas son por lo tanto muy importantes, el controlador solo es un medio para realizarlas. Por lo tanto debe desarrollarse una definición exacta de las funciones que el sistema debe realizar. La experiencia nos dice que los errores que existen en ésta etapa son extremadamente costosos al corregirlos en etapas posteriores donde el diseño del software y la codificación ya se han realizado.

Resulta más económico en términos de tiempo y dinero tener una especificación correcta, y por lo tanto un diseño del sistema más confiable.

3.2 ESTUDIO DE POSIBILIDADES.

Este tipo de estudio es a menudo conveniente realizarlo antes de tomar cualquier decisión sobre qué solución se adoptará para una tarea específica. El alcance de éste estudio puede variar enormemente, puede ir desde un sencillo planteamiento de las posibilidades del proyecto hasta un análisis del caso con recomendaciones completas en equipamiento. Normalmente éste tipo de estudio encierra varias áreas específicas de investigación:

- A) **Posibilidades Económicas**, consiste de la evaluación de los costos de instalación y desarrollo considerados en relación con la renta o beneficio resultantes de un sistema desarrollado.
- B) **Posibilidades Técnicas**, donde el proceso y los equipos son estudiados en términos funcionales y de rendimiento los cuales podrían relacionarse para conseguir un sistema aceptable.
- C) **Alternativas**, con una investigación y evaluación de alternativas dirigidas al desarrollo del sistema aceptable.

La realización completa de una proposición técnica requiere que conozcamos las necesidades presentes y futuras de la compañía en términos de la automatización de la planta y de los sistemas de información.

Una vez que las funciones de control han sido exactamente definidas, se debe elegir un adecuado sistema de control programable. Después de la identificación de un adecuado PLC, el trabajo sobre aspectos del diseño eléctrico del hardware y el diseño del software puede empezar.

3.3 ELECCIÓN DEL CONTROLADOR PROGRAMABLE.

Existe una amplia gama de PLCs disponibles hoy en día, con adiciones o reemplazos continuamente fabricados con características extendidas de un tipo u otro. Los avances de la tecnología son rápidamente adoptados por los fabricantes para mejorar el funcionamiento y status de sus productos. Sin embargo, la mayoría de los PLCs en cada rango de tamaño son muy similares en cuanto a sus medios de control. Las diferencias significantes radican en los métodos de programación y lenguajes, así como también diferentes estándares de respaldo y soporte del fabricante. Este último punto a menudo no es tomado en cuenta cuando se elige una adecuada marca de controlador.

Para seleccionar un fabricante específico, dependiendo de la tarea involucrada, el cliente puede acudir directamente a un fabricante o una compañía especializada en sistemas, quienes normalmente toman el proyecto en su totalidad y subcontratan varios aspectos como por ejemplo el cableado. Estas compañías tienden a trabajar con uno o dos fabricantes de PLCs, por lo tanto si elige una compañía especializada ésta elegirá el fabricante de PLCs que más le convenga.

El cliente que posee experiencia en PLCs y sistemas de control por lo general explora la excelencia técnica y el funcionamiento de cierto fabricante más que la asistencia en el diseño e instalación. Cuando se trata de alguien que no posee experiencia en éste campo, es aconsejable que tenga presente, antes de una selección, los siguientes puntos:

- Es posible obtener asistencia en el diseño del sistema.
- Evaluar la experiencia del fabricante en el área de aplicación necesaria.
- Ofrece el fabricante cursos de entrenamiento sobre el probable tipo de PLC para conocer las necesidades del cliente.
- Existe la documentación necesaria disponible en el lenguaje requerido.
- ¿Qué compatibilidad existe entre cualquier posible sistema y otros tipos de PLC del mismo o diferente fabricante ?

Para el usuario inexperto es una gran ventaja que el fabricante pueda ofrecerle asistencia en el diseño del sistema junto con entrenamiento y ayuda posterior en el diseño y codificación del programa .

3.4 TIPO Y TAMAÑO DE PLC

El tamaño y tipo de PLC puede ser seleccionado junto con el fabricante puesto que existen más de una marca de PLC que pueden satisfacer una aplicación particular el cliente puede obtener sistemas similares de varios fabricantes de equipos originales. La elección de un sistema específico resulta de las especificaciones funcionales y de entrada/salida requeridas.

Una vez que se ha decidido el tamaño del sistema PLC hay varios puntos que tienen que ser considerados :

- Capacidad necesaria de entradas/salidas.
- Tipos de entradas/salidas requeridas.
- Tamaño de memoria requerida.
- Velocidad y potencia requerida del CPU y juego de instrucciones.

Todos estos puntos están relacionados entre sí, el tamaño de memoria está directamente relacionado con la cantidad de entradas/salidas así como con el tamaño

del programa. Al crecer el número de entradas/salidas y el tamaño de memoria, el tiempo de procesamiento se incrementa requiriendo un procesador más rápido y potente.

3.5 REQUERIMIENTOS DE ENTRADAS/SALIDAS.

Las secciones de entrada/salida de un PLC deben ser capaces de contener suficientes módulos para conectar todas las señales y líneas de control para el proceso. Estos módulos deben estar conformes con las especificaciones básicas del sistema como por ejemplo mirar niveles de voltaje, cargas, etc., además:

- El tipo y número de puntos de entrada/salida requeridos por módulo.
- Aislamiento requerido entre el controlador y el proceso.
- La necesidad de entradas/salidas rápidas, o entradas/salidas remotas, o cualquier otra característica especial.
- Las necesidades futuras de la planta en términos de expansión potencial y puntos instalados de entrada/salida desocupados.
- Los requerimientos de alimentación de potencia de los puntos de entrada/salida.

3.6 REQUERIMIENTOS DE MEMORIA Y PROGRAMACIÓN.

El sistema de memoria de un PLC puede implementarse integrado a la tarjeta del CPU o en tarjetas adicionales permitiendo un tamaño de memoria que puede ser incrementado de acuerdo a las necesidades lo que permite una mayor adaptación, sin la necesidad de cambiar la tarjeta del CPU.

Como se mencionó anteriormente el tamaño de la memoria está relacionado tanto con la cantidad de entradas/salidas requeridas por el sistema como también con el programa de control que se almacene. El tamaño exacto de éste programa no se puede definir hasta que todo el software haya sido diseñado, codificado, instalado y probado. El tamaño del programa también está relacionado con el número de entradas/salidas, puesto que debe incluir instrucciones que lean desde o escriban hacia cada punto de entrada/salida. Adicionalmente se debe proveer un espacio de memoria adicional para permitir cambios en el programa o para futuras expansiones del sistema.

Existen dos alternativas en cuanto al tipo de memoria, RAM o EPROM. La RAM es la más común y permite una alteración rápida y sencilla del programa ya sea antes o después de la instalación del sistema. El contenido de la RAM se hace semipermanente mediante la provisión de baterías de respaldo en su suministro de potencia. La memoria RAM debe ser utilizada siempre para las entradas/salidas y para las funciones de datos, cuando éstas impliquen datos dinámicos. La memoria

EPROM puede ser utilizada solamente para almacenamiento del programa y requiere la utilización de un programador/borrador especial de memoria EPROM para alterar el código almacenado. El uso de EPROMS es ideal cuando varias máquinas son controladas por PLCs idénticos que tienen el mismo programa. Sin embargo el almacenamiento en la memoria RAM puede utilizarse hasta que el programa haya sido totalmente desarrollado y probado.

Los microcomputadores son utilizados comúnmente como estaciones desarrolladoras de programas. La gran cantidad de memoria RAM y espacio de almacenamiento del disco que poseen éstas máquinas permiten el desarrollo y almacenamiento muchos programas de PLC, incluyendo la redacción de texto y documentación. Los programas pueden ser transferidos entre el microcomputador y el PLC para su alteración y pruebas. La programación de la EPROM también puede llevarse a cabo mediante el microcomputador.

3.7 JUEGO DE INSTRUCCIONES.

Cualquier sistema considerado debe proveer un conjunto de instrucciones que sea adecuado para realizar las tareas del proceso. A pesar del tamaño, todos los PLCs pueden manejar controles lógicos, secuencias, etc.. Las diferencias radican en el manejo de datos, funciones especiales y comunicaciones. Los PLCs más grandes

tienen instrucciones más poderosas que los pequeños en éstos aspectos, pero también es posible encontrar pequeñas unidades y de tamaño medio capaces de llevar a cabo funciones específicas con excelentes niveles de funcionamiento.

En los controladores programables modulares existen varias alternativas para la elección del CPU, las cuales ofrecen diferentes niveles de funcionamiento en términos de velocidad y funcionalidad. A medida que el número de funciones y de entradas/salidas se incrementa, las exigencias sobre el CPU también se incrementan debido a que existe un mayor número de señales que tienen que procesarse cada ciclo. Esto requeriría la utilización de un CPU más rápido para que los tiempos de exploración no sean afectados.

3.8 DESCRIPCIÓN DE LA SERIE S7-200 DE SIEMENS.

La serie S7-200 comprende diversos sistemas de automatización pequeños (Micro-PLCs) que se pueden utilizar para numerosas tareas. Gracias a su diseño compacto, su capacidad de ampliación, su bajo costo y su amplio juego de operaciones, Los Micro-PLCs S7-200 se adecúan para numerosas aplicaciones pequeñas de control. Además los diversos tamaños y fuentes de alimentación de las CPUs ofrecen la flexibilidad necesaria para la solución de tareas de automatización. En la figura 3-1 se muestra un Micro-PLC S7-200.



Figura 3.1
Micro PLC S7-200

En la figura 3-2 se muestra la estructura básica de un Micro-PLC S7-200 que incluye una CPU S7-200, un PC, el software de programación STEP7-Micro/WIN y un cable de comunicación. Para poder utilizar un PC se requiere un cable PC/PPI o una tarjeta MPI.

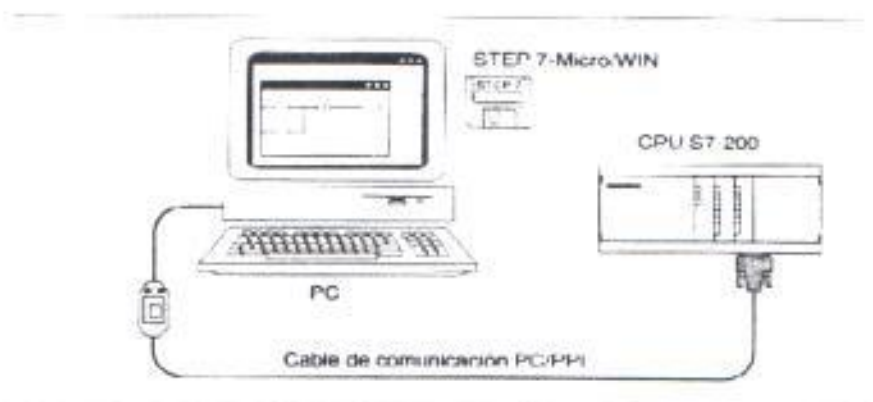


Figura 3.2
Conexión Básica

Función	CPU 212	CPU 214	CPU 215	CPU 216
Tamaño físico	160 x 80 x 62mm	197 x 80 x 62mm	218 x 80 x 62mm	218 x 80 x 62mm
Memoria				
Programa (EEPROM)	512 palabras	2K palabras	4K palabras	4K palabras
Datos de usuario	512 palabras	2K palabras	2.5K palabras	2.5K palabras
Marcas internas	128	256	256	256
Cartucho de memoria	No	Si EEPROM	Si EEPROM	Si EEPROM
Cartucho de pila opcional	No	200 días (opcional)	200 días (opcional)	200 días (opcional)
Respaldo (condensador de alto rendimiento)	50 horas (tip.)	190 horas (tip.)	190 horas (tip.)	190 horas (tip.)
Entradas/salidas (E/S)				
E/S integradas	8DI/6DO	14DI/10DO	14DI/10DO	24DI/16DO
Módulos de ampliación (máx.)	2 módulos	7 módulos	7 módulos	7 módulos
Imagen del proceso E/S	64DI/64DO	64DI/64DO	64DI/64DO	64DI/64DO
E/S analógicas (ampliación)	16AI/16AQ	16AI/16AQ	16AI/16AQ	16AI/16AQ
Filtros de entrada	No	Si	Si	Si
Operaciones				
Velocidad de ejecución booleana	1.2µs/operación	0.8µs/operación	0.8µs/operación	0.8µs/operación
Contadores / temporizadores	64/64	128/128	256/256	256/256
Bucles FOR/NEXT	No	Si	Si	Si
Aritmética en coma fija	Si	Si	Si	Si
Aritmética en coma flotante	No	Si	Si	Si
PID	No	No	Si	Si
Funciones adicionales				
Contadores rápidos	1 S/W	1 S/W, 2 H/W	1 S/W, 2 H/W	1 S/W, 2 H/W
Potenciómetros analógicos	1	2	2	2
Salidas de impulsos	No	2	2	2
Interrupciones de comunicación	1 emisor / 1 receptor	1 emisor / 1 receptor	1 emisor / 2 receptores	2 emisores / 4 receptores
Interrupciones temporizadas	1	2	2	2
Entradas de interrupción de hardware	1	4	4	4
Reloj de tiempo real	No	Si	Si	Si
Comunicación				
Nº de interfaces	1 (RS-485)	1 (RS-485)	2 (RS-485)	2 (RS-485)
Protocolos asistidos Interface 0: Interface 1:	PPI, Freeport N/A	PPI, Freeport N/A	PPI, Freeport, DP/T DP, DPV2	PPI, Freeport, DP/T PPI, Freeport, DP/T
Punto a punto	Sólo esclavo	Si	Si	Si

Tabla 3.1
Características de la Serie S7-200.

La serie S7-200 comprende diversas CPUs. Por lo tanto, se dispone de una amplia gama de funciones que permiten diseñar soluciones de automatización a un precio razonable. En la tabla 3-1 se resumen las principales funciones de cada CPU.

3.9 PRINCIPALES COMPONENTES DE UN MICRO-PLC S7-200.

La CPU S7-200 es un aparato autónomo compacto que comprende una unidad central de procesamiento (CPU), la fuente de alimentación, así como entradas y salidas digitales.

- La CPU ejecuta el programa y almacena los datos para la tarea de automatización o el proceso.
- La fuente de alimentación le proporciona corriente a la unidad central y a los módulos de ampliación conectados.
- Las entradas y salidas controlan el sistema de automatización. Las entradas vigilan las señales de los dispositivos de campo (por ejemplo sensores e interruptores) y las salidas comandan las bombas, motores u otros dispositivos del proceso (actuadores).
- La interface de comunicación permite conectar la CPU a una unidad de programación o a otros dispositivos. Algunas CPUs S7-200 disponen de dos interfaces de comunicación.

- Los diodos luminosos indican el modo de operación de la CPU (RUN o STOP), el estado de las entradas y salidas integradas, así como los posibles fallos del sistema que se hayan detectado.

3.9.1 Módulos de Ampliación.

Los módulos de ampliación para las CPU S7-200 ofrecen un número determinado de entradas y salidas integradas. Si se conecta un módulo de ampliación se dispondrá de más entradas y salidas. Como se observa en la figura 3-3, éstos módulos disponen de un conector de bus para su conexión al aparato central.

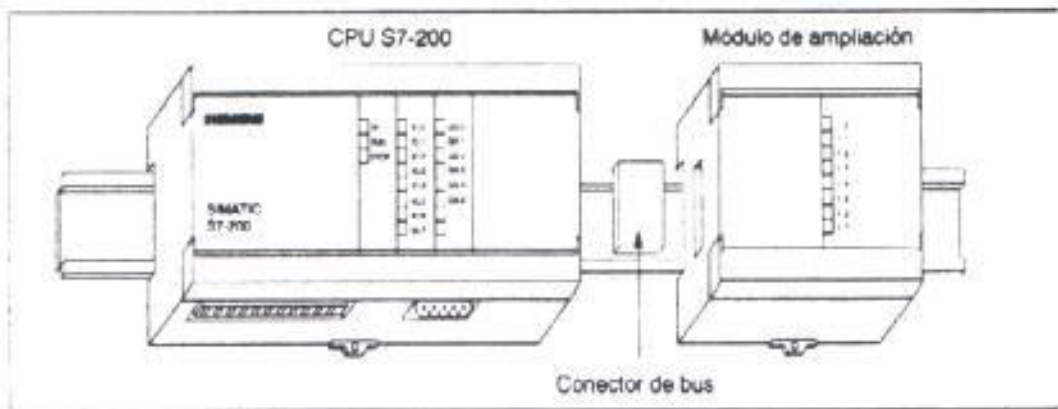


Figura 3.3

El PLC y su Módulo de Ampliación

3.10 ESTABLECER LA COMUNICACIÓN CON UNA CPU S7-200

Conectar el computador a la CPU S7-200 para establecer la comunicación PPI

La figura 3-4 muestra una configuración típica para conectar el computador personal a la CPU mediante el cable PC/PPI. Para establecer un enlace correcto entre los componentes:

- 1.- Ajuste los interruptores del cable PC/PPI para una velocidad de transferencia.
- 2.- Conecte el extremo RS-232 (PC) del cable PC/PPI al interfase de comunicación de su ordenador (COM1 o COM2) y apriete los tornillos de conexión.
- 3.- Conecte el otro extremo (RS-485) del cable PC/PPI al interfase de comunicación de la CPU y apriete los tornillos de conexión.

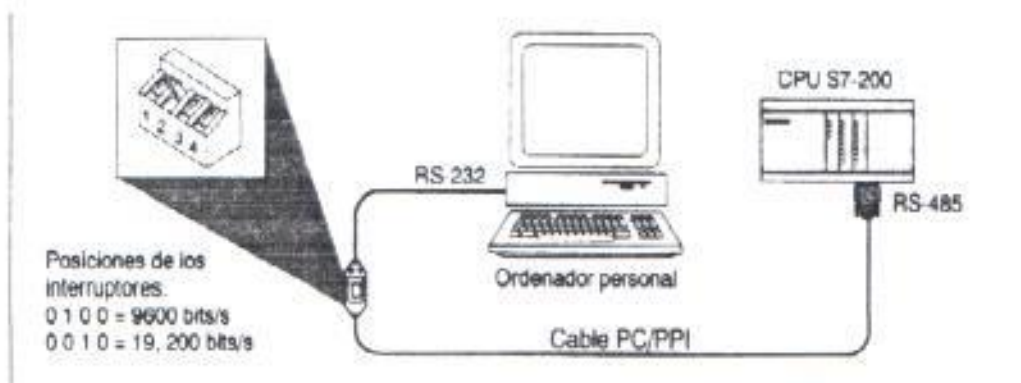


Figura 3.4
Comunicación PPI con un CPU

Conectar el computador a la CPU S7-200 para establecer la comunicación MPI STEP7-Micro/WIN también se puede utilizar con una tarjeta de interfase multipunto (MPI). La tarjeta MPI incorpora un solo interfase RS-485 para su conexión a la red mediante un cable directo.

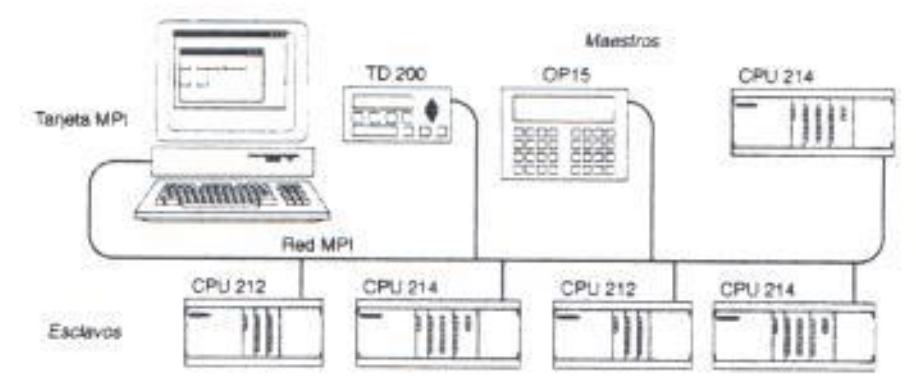


Figura 3.5
Red Maestro-Eslavo

Una vez establecido un enlace MPI es posible conectar STEP7-Micro/WIN a una red que contenga otros maestros, tal como se aprecia en la figura 3-5. (Cada maestro debe tener una dirección unívoca).

Se debe tener en cuenta que con STEP7-Micro/WIN no es posible ejecutar simultáneamente dos aplicaciones diferentes que necesiten la misma tarjeta MPI. Cierre las demás aplicaciones antes de conectar STEP7-Micro/WIN a la red mediante la tarjeta MPI.

Ajustar los parámetros de comunicación

La figura 3-6 muestra el cuadro de diálogo "Comunicación". Las dos primeras opciones de interfases están previstas para los enlaces PPI. La tercera opción es para una tarjeta MPI. Para ajustar los parámetros de comunicación:

- 1.- Elija el comando **Instalar > Comunicación...**
- 2.- Compruebe si la información contenida en el cuadro de diálogo concuerda con su configuración. Si así lo desea, puede utilizar el botón "Buscar" para comprobar si se han establecido los enlaces correctamente. Al hacer click en "Buscar" se visualiza la primera dirección de la red, cambiando luego a "Siguiente" para permitirle establecer un enlace con otra estación de la red.
- 3.- Confirme sus ajustes haciendo click en "Aceptar" o pulsando la tecla de introducción.



Figura 3.6
Ajuste de Parámetros de Comunicación entre PC y un CPU

3.11 SOFTWARE DE PROGRAMACIÓN STEP7-MICRO/WIN.

STEP7-MICRO/WIN es un software de programación ejecutable bajo Windows. Éste permite programar el sistema de automatización S7-200. El paquete de software incorpora diversas herramientas necesarias para programar el S7-200 utilizando el lenguaje Lista de Instrucciones (AWL) o Esquema de Contactos (KOP).

3.11.1 INTRODUCIR PROGRAMAS EN KOP.

La ventana del editor KOP permite escribir programas utilizando el lenguaje Esquema de Contactos (KOP). La barra de herramientas incluye algunos de los elementos KOP usados para introducir los programas. Para acceder a una de las operaciones agrupadas por funciones en un cuadro de lista despegable (izquierda), haga clic en la misma o pulse la tecla F2. Al seleccionar un grupo de funciones aparecen en el cuadro de lista situado a la derecha todas las operaciones de dicho grupo. Si pulsa la tecla F9 o elige todas las operaciones puede también visualizar una lista de todas las operaciones en orden alfabético.

A cada segmento le corresponden dos comentarios descritos a continuación:

- Los comentarios del segmento en una sola línea siempre son visibles en el editor KOP. Para acceder a ellos haga click en cualquier parte del comentario.
- Para acceder a los comentarios del segmento que utilicen más de una línea haga doble click en el número del segmento. Dichos comentarios sólo pueden ser indicados en un cuadro de diálogo, pero su impresión es completa.

Para comenzar a introducir un programa:

- Para introducir el título del programa, elija el comando **Edición > Título...**
- Para introducir elementos KOP, elija el tipo de elemento deseado haciendo click en el botón correspondiente o seleccionándolo de la lista de operaciones. A continuación, pulse la tecla de introducción o haga doble click en el cuadro del cursor.
- Introduzca los operandos o los parámetros en cada cuadro de texto y pulse la tecla de introducción.

En la figura 3-7 se muestra la ventana del editor KOP.

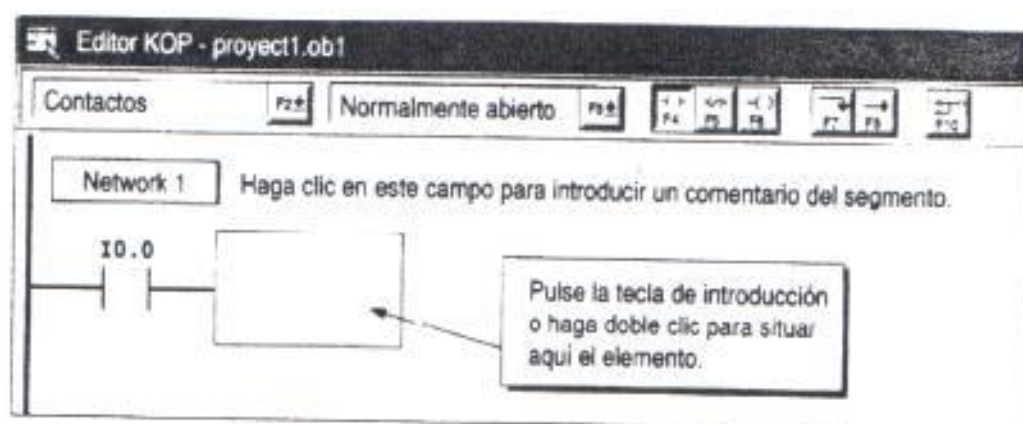


Figura 3.7
Ventana del Editor KOP

Al programar con KOP, se crean y se disponen componentes gráficos que conforman un segmento de operaciones lógicas. Como se muestra en la figura 3-8, se ofrecen los siguientes elementos básicos para crear programas:

- **Contactos:** un contacto representa un interruptor por el que fluye la corriente cuando está cerrado.
- **Bobinas:** una bobina representa un relé que se excita cuando se aplica tensión.
- **Cuadros:** un cuadro representa una función que se ejecuta cuando la corriente fluye por él.
- **Segmentos:** Un segmento constituye un circuito completo. La corriente fluye desde la barra de alimentación izquierda pasando por los contactos cerrados para excitar las bobinas o cuadros.

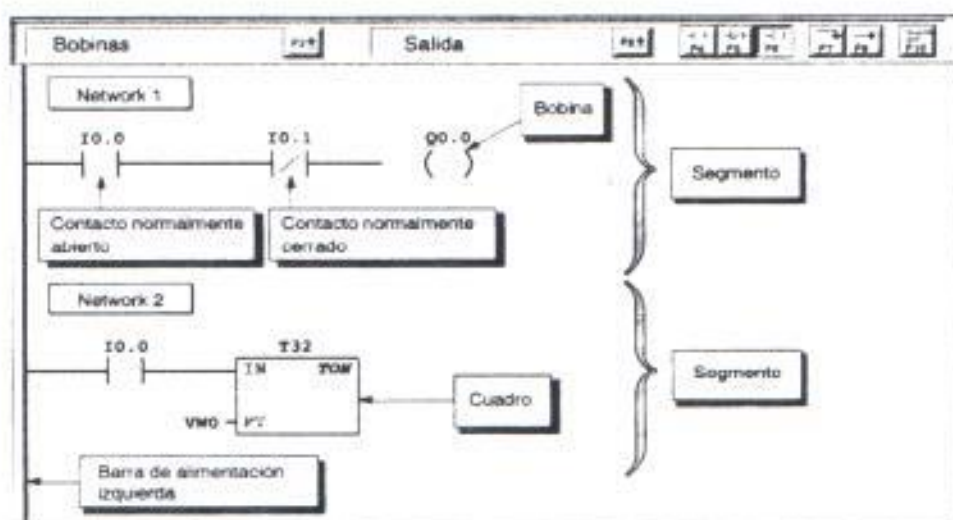


Figura 3.8

Elementos Básicos de KOP

Una vez completado el programa , compruebe la sintaxis eligiendo el comando **CPU > Compilar** o haciendo click en el botón "Compilar". Si ha introducido correctamente todos los segmentos obtendrá el mensaje "Compilación Finalizada" que incluye también información sobre el número de segmentos y sobre la cantidad de memoria utilizada por el programa. En caso contrario, el mensaje de compilación indicará qué segmentos contienen errores.

Una vez que la compilación se ha efectuado con éxito puede guardar su proyecto eligiendo el comando **Proyecto > Guardar todo** o hacer click en el botón correspondiente.

Ahora debe cargar el programa en la CPU y poner la CPU a modo RUN.

Entonces puede visualizar y comprobar el programa y su procesamiento.

3.12 CARGAR EL PROYECTO EN LA CPU.

Antes de cargar el programa en la CPU, asegúrese de que ésta se encuentra en modo STOP. Para cargar el programa en la CPU:

- 1.- Ponga el selector de modos de operación de la CPU en la posición TERM o STOP.
- 2.- Elija el comando **CPU > STOP** o haga click en el botón "STOP" de la ventana principal.
- 3.- Responda con "Sí" para confirmar éste paso.
- 4.- Elija el comando **CPU > Cargar en CPU** o haga click en el botón "Cargar en CPU" de la ventana principal.
- 5.- El cuadro de diálogo "Cargar en CPU" permite especificar los componentes del proyecto que desea cargar. Pulse la tecla de introducción o haga click en "Aceptar".

Es importante notar que STEP 7-Micro/WIN comprueba si su programa utiliza operandos o direcciones de entradas y salidas válidas para la CPU en cuestión. Si

intenta cargar en la CPU un programa que utilice comandos o direcciones no comprendidas en el rango de la CPU u operaciones no asistidas por la misma, La CPU rechazará el intento de cargar el programa y visualizará un mensaje de error. Asegúrese de que todas las direcciones de la memoria, así como las direcciones de entradas/salidas y las operaciones utilizadas en su programa, sean válidas para la CPU en cuestión.

3.13 CONMUTAR LA CPU A MODO RUN

Si la operación de carga se efectuó con éxito, se podrá conmutar entonces la CPU a modo RUN:

- 1.- Elija el comando **CPU>RUN** o haga click en el botón RUN de la ventana principal.
- 2.- Responda con "Sí" para confirmar este paso.

3.14 OBSERVAR EL ESTADO KOP

Estando la CPU en modo RUN, al activar el estado KOP, se visualizará el estado actual de los eventos en su programa. Abra la ventana del editor KOP y elija el comando **TEST>Activar estado KOP**.

Si tiene un simulador de entradas conectado a los terminales de su CPU puede colocar los interruptores en posición "ON" para observar el flujo de corriente y la ejecución de la lógica. En la figura 3-9 se muestra un ejemplo en el cual las entradas de "Marcha_1" y "Paro_1" se encuentran activas y estando la entrada "Nivel_Superior" desactivada, se completará el flujo de corriente del segmento 1. El aspecto del segmento será entonces como se observa en la figura 3-9.

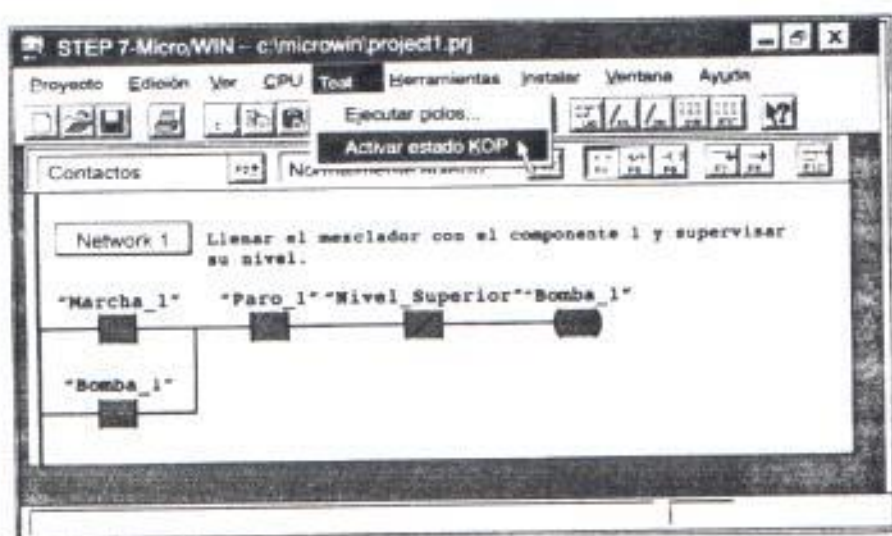


Figura 3.9
Visualización del estado de un Programa en KOP

3.15 INTRODUCIR PROGRAMAS EN AWL

El editor AWL es un editor de texto que brinda cierta flexibilidad en cuanto a la forma de introducir las operaciones del programa. En la figura 3-10 se muestra un ejemplo

de un programa AWL.

```

STL Editor - proyect1.ob1
//Programa para tren transportador

NETWORK //Marcha motor:
LD "Marchal" //Si I0.0 está activada
UN "Para_Em1" //y I0.1 no está activada,
= Q0.0 //poner en marcha el motor del traspos

Network //Paro emergencia transportador:
LD I0.1 //Si "Paro_Em1" esta activada
O I0.3 //o si "Paro_Em2" está activada,
R Q0.0, 1 //parar el motor del transportador.

NETWORK //fin del programa
MEND
  
```

Para poder visualizar el programa tanto en AWL como en KOP es preciso dividir los segmentos lógicos con la palabra clave NETWORK.

Figura 3.10
Ventana del Editor AWL.

Tenga en cuenta lo siguiente para introducir programas en AWL:

- Para poder visualizar un programa AWL en KOP debe dividir los segmentos lógicos en segmentos independientes, introduciendo para tal efecto la palabra "NETWORK (Segmento)". (Los números de los segmentos se generan automáticamente al compilar o cargar el programa).
- Cada comentario debe ir precedido de dos barras inclinadas (/). Cada línea adicional de comentario debe comenzar asimismo con dos barras inclinadas.
- Finalice cada línea pulsando la tecla de introducción.
- Separe cada operación de su dirección o parámetro con un espacio en blanco o con un tabulador.

- No introduzca espacios entre el área de memoria y la dirección (p. ej., introduzca I0.0 y no I 0.0).
- Dentro de una operación, separe cada operando mediante una coma, un espacio en blanco o un tabulador.
- Utilice comillas al introducir los nombres simbólicos. Por ejemplo, si su tabla de símbolos contiene el nombre simbólico Marchal para la dirección I0.0, introduzca la operación de la siguiente forma:
LD "Marchal"

Una vez completada una serie de segmentos puede comprobar la sintaxis lógica eligiendo el comando **CPU > Compilar** o haciendo click en el botón correspondiente.

Luego de que la compilación se ha llevado a cabo con éxito se puede proceder a grabar el programa y cargarlo en la CPU mediante el procedimiento antes mencionado.

3.16 AJUSTAR EL MODO DE OPERACION DE LA CPU

La CPU S7-200 tiene dos modos de operación:

- **STOP:** La CPU no ejecuta el programa. Cuando está en el modo STOP, es posible cargar programas o configurar la CPU.
- **RUN:** La CPU ejecuta el programa. Cuando está en modo RUN no es posible cargar programas ni configurar la CPU.

El diodo luminoso (LED) en la cara frontal de la CPU indica el modo de operación actual. Para poder cargar un programa en la memoria de la CPU es preciso cambiar a modo STOP.

La CPU tiene un selector ubicado debajo de la tapa de acceso a la CPU, éste selector puede ser utilizado para cambiar el modo de operación de la CPU en forma manual como se mencionó anteriormente.

3.17 CAMBIAR EL MODO DE OPERACIÓN CON STEP7-MICRO/WIN

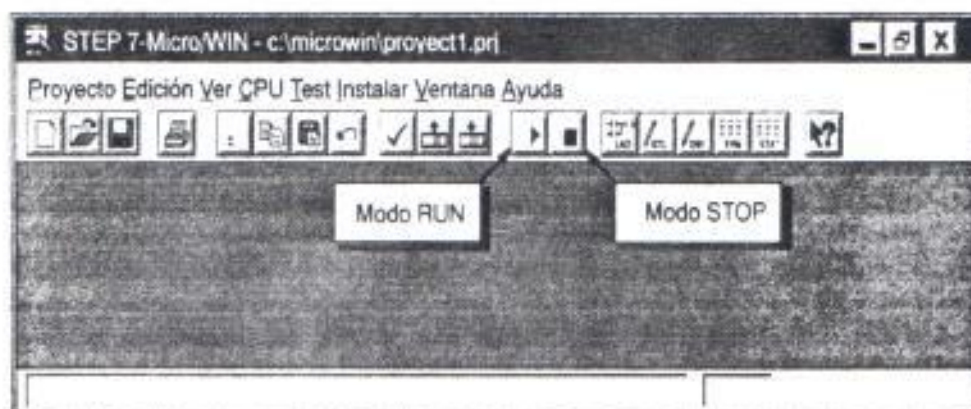


Figura 3.11

Cambio de Modo de Operación del CPU con STEP-7 Micro/Win

Como se observa en la figura 3-11, el modo de operación de la CPU se puede cambiar también con STEP7-Micro/WIN. Para que ésto sea posible, el selector de la CPU deberá estar en la posición TERM o RUN.

3.18 EL CICLO DE LA CPU

La CPU ejecuta el programa en un ciclo continuo. Como se muestra en la figura 3-12, el ciclo de la CPU abarca las siguientes tareas:

- Leer las entradas.
- Ejecutar el programa de usuario.
- Procesar las peticiones de comunicación.
- Ejecutar el autodiagnóstico de la CPU.
- Escribir las salidas.



Figura 3.12

El Ciclo del CPU

3.19 LEER LAS ENTRADAS DIGITALES

Al principio de cada ciclo se leen los valores actuales de las entradas digitales, escribiéndose luego en la imagen del proceso de las entradas.

La CPU reserva un espacio de la imagen del proceso de las entradas en incrementos de ocho bits (un byte). Si la CPU o el módulo de ampliación no proporcionan una entrada física para cada bit del byte reservado, no será posible asignar dichos bits a los módulos subsiguientes en la cadena de entradas/salidas o utilizarlos en el programa de usuario. Al comienzo de cada ciclo, la CPU pone a 0 dichos bits no utilizados en la imagen del proceso. No obstante, si la CPU asiste varios módulos de ampliación y no se está utilizando su capacidad de Entradas/Salidas (porque no se han instalado los módulos de ampliación), los bits de entradas de ampliación no utilizados se pueden usar como marcas internas adicionales.

La CPU no actualiza automáticamente las entradas analógicas como parte del ciclo y no prevé imagen del proceso para las mismas. A las entradas analógicas se debe acceder directamente desde el programa de usuario.

3.20 EJECUTAR EL PROGRAMA

Durante esta fase del ciclo, la CPU ejecuta el programa desde la primera operación hasta la última. El control directo de las entradas y salidas permite acceder directamente a las mismas mientras se ejecuta el programa o una rutina de interrupción.

Si se utilizan interrupciones, las rutinas asociadas a los eventos de interrupción se almacenan como parte del programa. Las rutinas de interrupción no se ejecutan como parte del ciclo, sino solo cuando ocurre el evento (en cualquier momento del ciclo).

3.21 PROCESAR LAS PETICIONES DE COMUNICACIÓN

Durante ésta fase del ciclo, la CPU procesa los mensajes que haya recibido por el interfase de comunicación.

3.22 EJECUTAR EL AUTODIAGNÓSTICO DE LA CPU

Durante el autodiagnóstico se comprueba el firmware de la CPU y la memoria del programa, así como el estado de los módulos de ampliación.

3.23 ESCRIBIR LAS ENTRADAS DIGITALES

Al final de cada ciclo, la CPU escribe los valores de la imagen del proceso de las salidas en las salidas digitales.

La CPU reserva un espacio de la imagen del proceso de las salidas en incrementos de ocho bits (un byte). Si la CPU o el módulo de ampliación no proveen una salida física para cada bit del byte reservado, no será posible asignar dichos bits a los módulos subsiguientes en la cadena de entradas/salidas. No obstante, los bits no utilizados de la imagen del proceso de las salidas se pueden usar como marcas internas adicionales.

La CPU no actualiza automáticamente las salidas analógicas como parte del ciclo y no provee imagen del proceso para las mismas. A las salidas analógicas se debe acceder directamente desde el programa del usuario.

3.24 INTERRUMPIR EL CICLO

Si se utilizan interrupciones, las rutinas asociadas a los eventos de interrupción se almacenan como parte del programa. Las rutinas de interrupción no se ejecutan como parte del ciclo, sino solo cuando ocurre el evento (en cualquier momento del ciclo). La CPU procesa las interrupciones habilitadas de forma asíncrona al ciclo y

ejecuta las rutinas de interrupción cuando se presente el correspondiente evento, conforme a su orden de aparición y prioridad.

3.25 IMAGEN DEL PROCESO DE LAS ENTRADAS Y SALIDAS

Al ejecutarse el programa, los accesos a las entradas y salidas se efectúan generalmente a través de la respectiva imagen del proceso, y no de forma directa.

Las imágenes del proceso existen por tres razones principales:

- El sistema comprueba todas las entradas al comenzar el ciclo. De este modo se sincronizan y congelan los valores de éstas entradas durante la ejecución del programa. La imagen del proceso actualiza las salidas cuando termina de ejecutarse el programa. Ello tiene un efecto estabilizador en el sistema.
- El programa de usuario puede acceder a la imagen del proceso mucho más rápido de lo que podría acceder directamente a las entradas y salidas físicas, con lo cual se acelera su tiempo de ejecución.
- Las entradas y salidas son unidas de bit a las que se debe acceder en formato de bit. No obstante, la imagen del proceso permite acceder a ellas en formato de bits, bytes, palabras y doble palabras, lo que ofrece flexibilidad adicional.

Otra ventaja es que las imágenes del proceso son lo suficientemente grandes para procesar el número máximo de entradas y salidas. Puesto que un sistema real

comprende tanto entradas como salidas, en la imagen del proceso existe siempre un número de direcciones que no se utilizan. Estas direcciones libres pueden utilizarse como marcas internas adicionales.

3.26 CONTROL DIRECTO DE LAS ENTRADAS Y SALIDAS

El acceso a las entradas y salidas se efectúa generalmente a través de las imágenes del proceso. Sin embargo el control directo de las mismas permite acceder inmediatamente a una entrada o salida físicas. El acceso directo a una entrada no modifica la dirección correspondiente en la imagen del proceso de las entradas. En cambio, el acceso directo a una salida actualiza simultáneamente la dirección correspondiente en la imagen del proceso de las salidas.

3.27 DESCRIPCIÓN GENERAL DEL PROCESO

Considerando que el estado inicial del proceso es aquel en el que la puerta se encuentra totalmente cerrada, el proceso de apertura y cierre automático puede describirse de la siguiente manera:

- La puerta deberá abrirse en el momento que los sensores de movimiento externo o interno detecten la presencia de una persona.

- La orden de apertura permanecerá activa hasta que el switch de fin de carrera denominado "Final_A" se accione, lo que indica que la puerta esta totalmente abierta.
- En este momento, el sistema esperará un tiempo de 5 segundos antes de enviar a cerrar la puerta, el inicio de este tiempo lo determina la activación del sensor de movimiento ubicado en el lado opuesto del sensor que mandó a abrir la puerta.
- Si durante el cierre de la puerta cualquiera de los sensores de movimiento se activan la puerta deberá ser abierta inmediatamente.
- La orden de cierre permanecerá activa hasta que el switch de fin de carrera denominado "Final_C" se accione, lo que indica que la puerta esta totalmente cerrada.

Además se ha implementado un control de la posición de la puerta mediante una señal Analógica de Voltaje DC de entrada, la misma que es monitoreada durante el proceso de cierre como el de apertura, con la finalidad de accionar una alarma en el momento que la puerta se detiene debido a algún desperfecto mecánico o eléctrico durante su recorrido

3.28 TABLA DE SIMBOLOS

Para un mejor entendimiento del programa a continuación en la tabla 3-2 se detallarán todos los elementos utilizados en la programación con sus respectivos nombres simbólicos y direcciones en el CPU con un breve comentario.

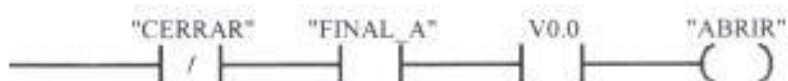
NOMBRE SIMBOLICO	DIRECCION	COMENTARIO
DECT.EXT.	I0.1	DETECTOR EXTERNO DE MOVIMIENTO
DECT.INT	I0.2	DETECTOR INTERNO DE MOVIMIENTO
FINAL_C	I0.3	INTERRUPTOR FINAL DE PUERTA CERRADA
FINAL_A	I0.4	INTERRUPTOR FINAL DE PUERTA ABIERTA
RESET	I0.5	REPOSICION DE ALARMA
ABRIR	Q0.0	SE ABRE LA PUERTA
CERRAR	Q0.1	SE CIERRA LA PUERTA
ALARMA	Q0.3	SALIDA DE ALARMA EN EL MOTOR
POSICION	AIW0	INDICA LA POSICION DE LA PUERTA
POS.ANT.A	VW0	ALMACENA POSICION ANTERIOR DURANTE LA APERTURA DE LA PUERTA
POS.ANT.C	VW2	ALMACENA POSICION ANTERIOR DURANTE EL CIERRE DE LA PUERTA

TABLA 3-2
Lista de Símbolos utilizados en el Programa.

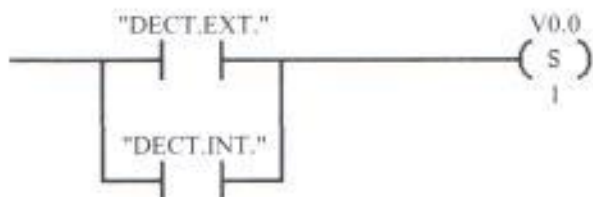
3.29 PROGRAMA GENERAL

El programa completo diseñado en STEP-7 Micro/Win es el siguiente:

nto 1 INICIO DEL PROGRAMA



nto 2



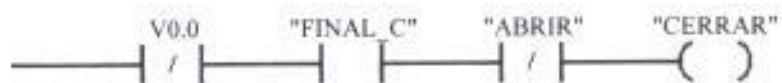
nto 3



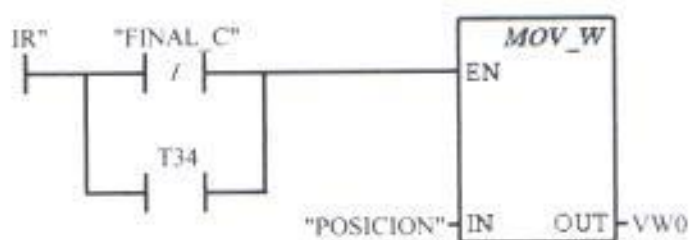
p. 4



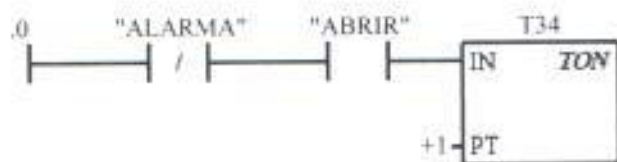
p. 5

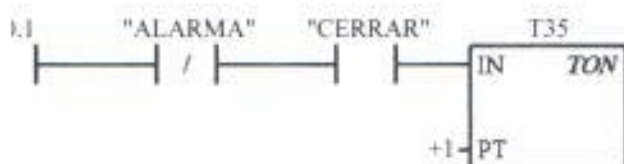
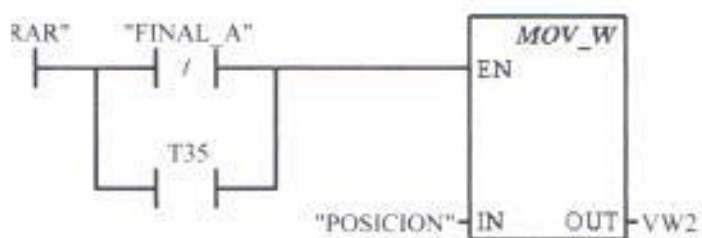
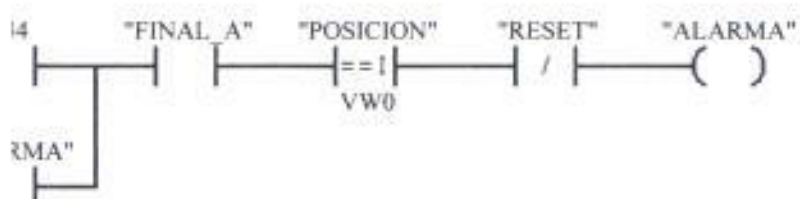
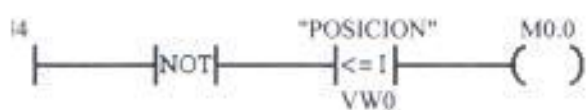


p. 6 VERIFICACION DE POSICION EN LA APERTURA



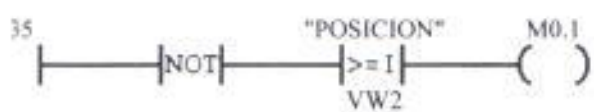
p. 7



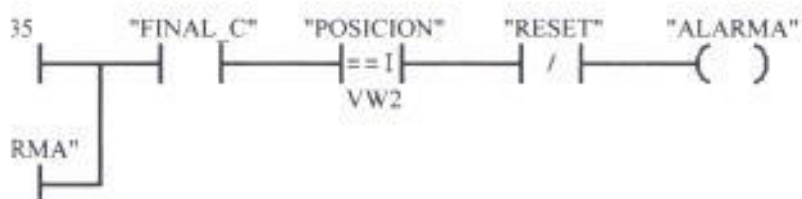


ito 12

71



ito 13



ito 14

4D)

CAPITULO IV

IMPLEMENTACION DE LA INTERFASE MMI PARA EL CONTROL Y MONITOREO DEL SISTEMA DE APERTURA Y CIERRE AUTOMATICO UTILIZANDO IN-TOUCH

4.1 INTRODUCCIÓN.

IN-TOUCH es una potente y flexible herramienta de desarrollo de interfaces de operador para la creación de sistemas personalizados en entornos de fabricación discretos, de proceso, DCS, SCADA y otros.

Permite a los ingenieros, supervisores, administradores y operadores, ver en pantalla mediante representaciones gráficas de procesos en tiempo real, los trabajos de una operación completa. Algunas versiones (para Windows NT) incluyen varias funciones de arquitectura distribuida entre las que se encuentran la gestión de alarmas distribuidas, conversión de resolución dinámica y desarrollo y mantenimiento remoto de aplicaciones para su uso en grandes redes basadas en PC. IN-TOUCH incluye además un potente nodo de sólo visualización (Factory Focus) que permite a los supervisores y administradores ver los datos de planta en tiempo real desde un PC de escritorio en cualquier punto de la red.

4.2 CARACTERÍSTICAS DE INTOUCH.

En IN-TOUCH también se incorporan funciones que aumentan el rendimiento las cuales son:

4.2.1 GRÁFICOS ORIENTADOS A OBJETOS.

Los objetos y grupos de objetos pueden moverse, redimensionarse y animarse de forma más rápida y sencilla que los gráficos de mapa de bits. Las potentes herramientas de desarrollo orientadas a objetos, facilitan dibujar, organizar,

alinear, disponer en capas, espaciar, rotar, invertir, duplicar, cortar, copiar, pegar y mucho más.

4.2.2 ENLACES DE ANIMACIÓN.

Los enlaces de animación pueden combinarse para ofrecer tamaños, colores, movimientos y/o cambios de posición complejos. Incluyen entradas de contacto discretas, analógicas y de cadena; deslizadores horizontales y verticales; pulsadores discretos y de acción; pulsadores para mostrar y ocultar ventanas; enlaces de color de línea, relleno y texto para valores y alarmas discretos y analógicos; enlaces de altura y anchura de objetos; enlaces de posición horizontal y vertical y mucho más.

4.2.3 ASISTENTES.

IN-TOUCH incluye una biblioteca completa de asistentes complejos preconfigurados como interruptores, deslizadores y medidores, que el usuario puede modificar y duplicar libremente. Con frecuencia los asistentes utilizados pueden añadirse a la barra de herramientas de IN-TOUCH para facilitar el acceso durante el desarrollo de aplicaciones. El Extensibility Toolkit permite a

los usuarios y a desarrolladores, crear asistentes muy complejos, como un asistente de conversión de dibujos de AutoCAD.

4.2.4 SCRIPTS.

Este lenguaje de IN-TOUCH es tan potente, flexible y fácil de usar que se pueden crear scripts simplemente apuntando y haciendo click, sin necesidad de tocar el teclado. También es posible escribir sus propias funciones de script y agregarlas al menú mediante el Extensibility Toolkit.

4.2.5 REFERENCIA DINÁMICA.

Esta función permite cambiar las referencias de base de datos a etiquetas de entrada/salida durante la ejecución, lo que significa que los usuarios pueden cambiar en todo momento las referencias de datos para direcciones PLC, celdas de hojas de cálculo de Excel y referencias de intercambio dinámico de datos (DDE).

4.2.6 ALARMAS DISTRIBUIDAS.

Esta función admite varios servidores o suministradores de alarmas simultáneamente, lo que da a los operadores la capacidad de ver información de alarmas desde varias ubicaciones remotas al mismo tiempo. Las nuevas funciones permiten a los usuarios implementar un reconocimiento de alarmas del tipo “apuntar y hacer click”, barras de desplazamiento de alarmas y muchas otras funciones para su uso en red.

4.2.7 TENDENCIA HISTÓRICA DISTRIBUIDA.

Este sistema permite especificar de forma dinámica diferentes fuentes de datos de archivos históricos para cada una de las plumas de un gráfico de tendencia. Como IN-TOUCH permite usar hasta ocho plumas por gráfico, los usuarios pueden disponer de una cantidad sin precedentes de datos históricos en un instante dado.

4.3 EL WINDOW MAKER

El Window Maker es uno de los dos elementos principales del software Intouch. Es el ambiente de desarrollo de aplicaciones en el que constan todas las funciones y herramientas requeridas para la animación de objetos, ventanas sensitivas,

implementación lógica, implementación de alarmas, graficos de tendencia, y la creación de una base de datos (“Diccionario de Tagnames”). Las ventanas a través de la base de datos (“Tagnames”), son enlazadas a controladores industriales, sistemas de E/S y a otras aplicaciones de ambiente Windows. El objetivo de este capítulo es explicar en detalle el diseño de nuestra interface MMI para nuestra aplicación.

4.4 DISEÑO DE LA INTERFASE MMI PARA EL CONTROL DEL SISTEMA DE APERTURA Y CIERRE AUTOMATICO

Para el desarrollo de nuestra aplicación es necesario crear un directorio en el que se almacenarán todos los archivos generados en dicha aplicación. Para tal efecto es necesario abrir la caja de diálogo “Intouch-Administrador de Aplicaciones” siguiendo la trayectoria INICIO / PROGRAMAS / WONDERWARE FACTORY SUITE / INTOUCH .



Figura 4.1
Administrador de Aplicaciones

En la figura 4.1 se muestra la caja de diálogo "Intouch-Administrador de Aplicaciones".

Para crear un nuevo directorio presionamos FILE/New en el menú Principal y a continuación aparece la caja de diálogo que se muestra en la figura 4.2.

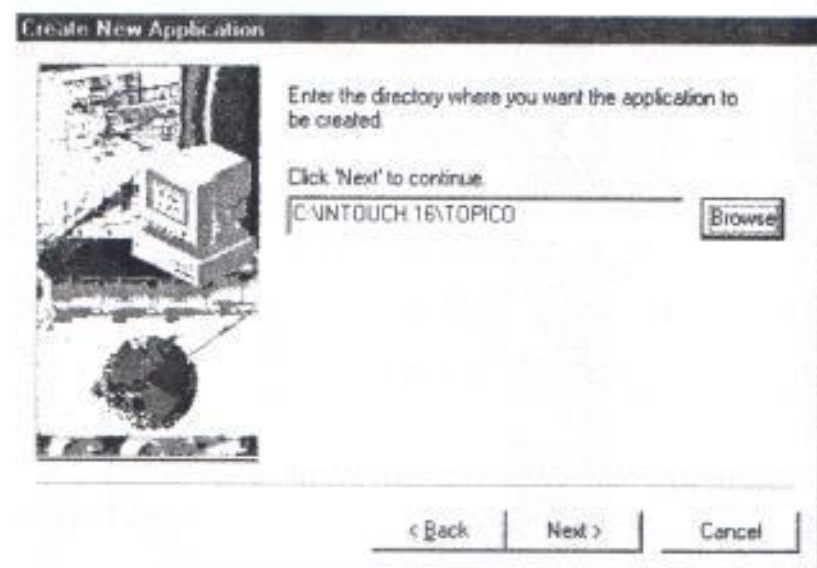


Figura 4.2

Caja de Diálogo "Creación de Nueva Aplicación"

Ingrese el nombre del directorio en el campo correspondiente, en nuestra aplicación dicho directorio se denomina "TOPICO".

Para iniciar la sesión con el Window Maker resalte el nombre del directorio de la aplicación y presione dos veces con el botón derecho del ratón. A continuación se abrirá la ventana del desarrollador Window Maker.

Una vez que se abre la ventana del desarrollador procedemos a crear las ventanas que nuestra aplicación requiera, que en nuestro caso son dos denominadas “Principal” y “Aviso”. La creación de estas ventanas se efectúa en la caja de dialogo “Propiedades de Ventana” que se muestran en las figuras 4.3 y 4.4.

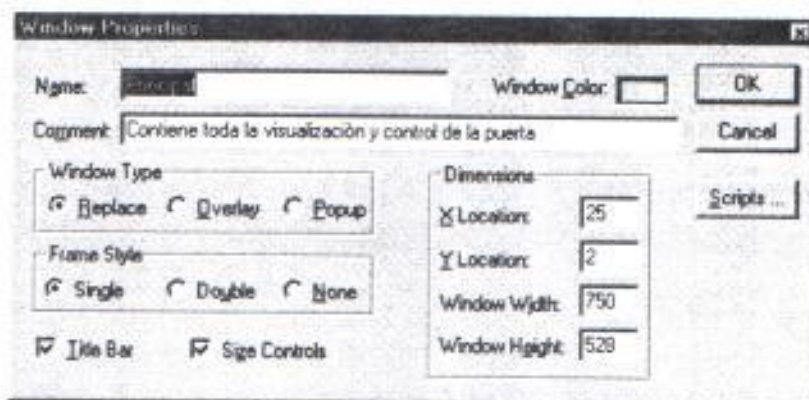


Figura 4.3

Propiedades de la Ventana “Principal”

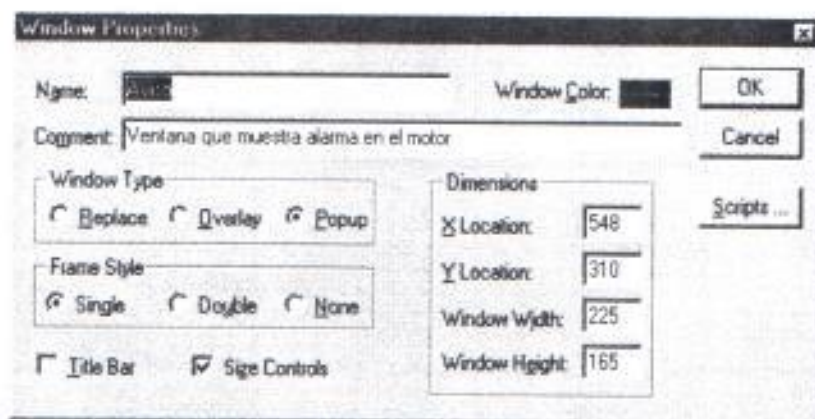


Figura 4.4

Propiedades de la Ventana “Aviso”

La ventana "Principal" es una ventana del tipo "Reemplazo", lo cual le da la propiedad de cerrar cualquier otro tipo de ventana cuando ésta es invocada y aparece sobre la pantalla.

Como se observa en la figura 4.3 en esta caja de diálogo también se especifican otras características como las coordenadas de la posición de la ventana (X,Y), ancho y alto, estilo del marco, barra de título, control de tamaño y color de la ventana.

La ventana "Aviso" es una ventana del tipo "Popup" lo que significa que permanecerá sobre otras ventanas, sin cerrar las otras ventanas, hasta que no sea removida por el usuario. Asimismo en la caja de diálogo mostrada en la figura 4.4 se especifican las otras características de ventana anteriormente mencionadas.

Luego de definir las características de las ventanas utilizadas en nuestra aplicación estamos en capacidad de comenzar el diseño de cada una de ellas.

A continuación en la figura 4.5 se muestra el esquema general de la ventana denominada "Principal".

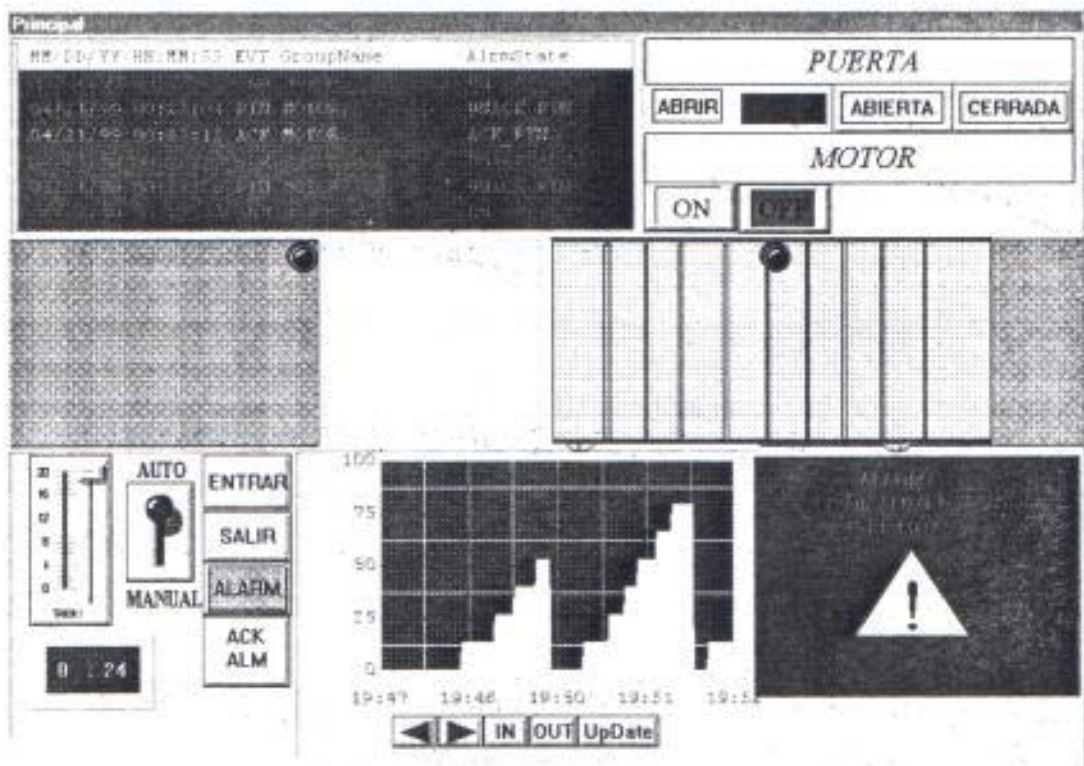


Figura 4.5

Ventana "Principal"


La ventana "Principal" consta de las siguientes partes:

- Objeto de Alarmas Históricas.
- Gráfico de Tendencia Histórica.
- Panel de Control.
- Panel de Señalización.
- Ventana "Aviso".
- Sistema de Apertura y Cierre.

4.4.1 OBJETO DE ALARMAS HISTORICAS.

InTouch provee un sistema de notificación para informar a los operadores de las condiciones del proceso y del sistema que están siendo monitoreados y controlados. Este sistema de notificación permite mostrar, grabar e imprimir las alarmas del proceso y los eventos del sistema. Las alarmas representan los estados de alertas presentes en las condiciones del proceso, los eventos representan mensajes del estado normal del sistema.

En un Objeto de Alarmas Históricas, se muestran mensajes correspondientes a la ocurrencia de una alarma o evento, la hora en la que fue reconocida dicha alarma o evento y la hora en la que terminó la condición de alarma y se retornó a la normalidad. En este tipo de objeto todos los mensajes anteriormente generados se muestran en una lista y a medida que se generan nuevos mensajes éstos se van agregando al final de la lista.

Para crear un Objeto de Alarmas Históricas presione el botón correspondiente a la herramienta "Wizard"  para habrir la caja de diálogo "Selección de Wizard" que se muestra en la figura 4.6 .

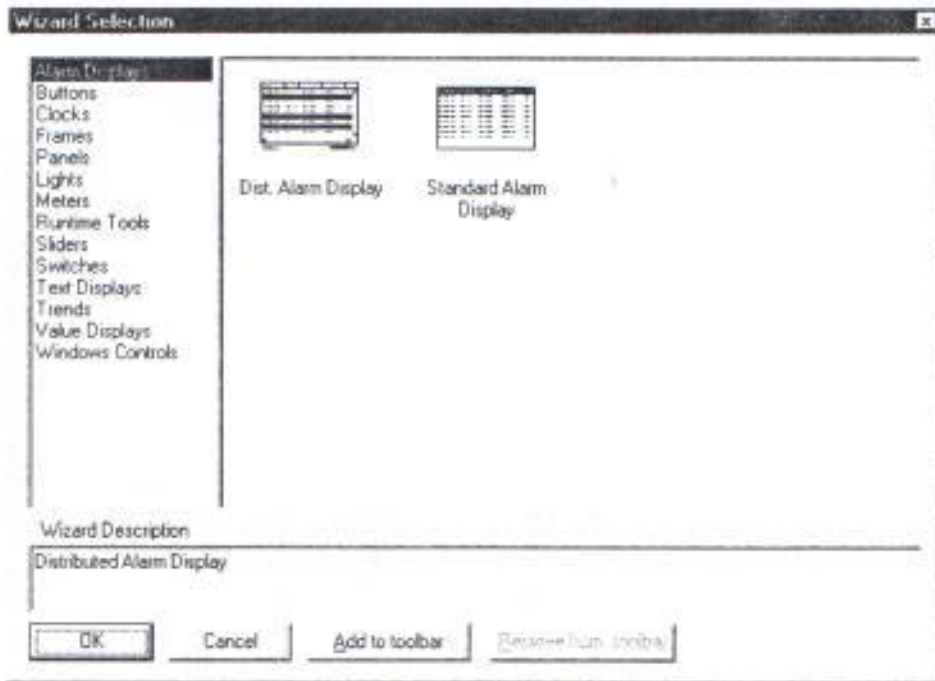


Figura 4.6

Selección de Wizard

En esta caja de diálogo seleccione la opción "Alarm Displays" y presionando dos veces elija el objeto "Standart Alarm Display" en la parte derecha de la caja. La caja de diálogo se cerrará y la ventana "Principal" reaparecerá con el cursor en el modo "Pegar".

Posicione el cursor en el lugar deseado de la ventana y presione una vez el botón izquierdo del ratón para pegar el objeto de alarmas. El objeto de alarmas aparecerá como se muestra en la figura 4.7.

MM/DD/YY HH:MM:SS	EVT	GroupName	AlarmState
MM/DD/YY HH:MM:SS	EVT	GroupName	AlarmState
MM/DD/YY HH:MM:SS	EVT	GroupName	AlarmState
MM/DD/YY HH:MM:SS	EVT	GroupName	AlarmState
MM/DD/YY HH:MM:SS	EVT	GroupName	AlarmState
MM/DD/YY HH:MM:SS	EVT	GroupName	AlarmState

Figura 4.7

Objeto de Alarmas.

Inicialmente el Objeto de Alarmas aparecerá con una configuración por defecto o una configuración que corresponde a la de algún otro Objeto de Alarmas previamente utilizado. Para cambiar la configuración de nuestro Objeto de Alarmas señálelo con el cursor y presione dos veces el botón izquierdo del ratón, esto provocará que la caja de diálogo “Configuración de Alarma” que se muestra en la figura 4.8 aparezca.

Figura 4.8
Configuración de Alarma

Para definir nuestro Objeto de Alarmas como de tipo "Histórico" seleccione la opción "Alarm History" en la sección denominada "Window Type". Como se mencionó anteriormente un Objeto de Alarmas Histórico muestra una lista de todos los eventos relacionados con las alarmas del proceso. Si se quiere que en nuestro Objeto se muestre una barra de título con las etiquetas de cada columna del display seleccione la opción "Titles". Habilitando esta opción se mostrarán las cajas de selección "Title Bar Color" y "Title Text Color", por medio de las cuales puede seleccionar el color de la barra de título y el color del texto de la barra.

Seleccione el color del fondo del display y el color del borde del Objeto de Alarmas en las opciones "Window Color" y "Border Color" respectivamente. Las opciones "UnAck Alm Color", "Ack Alm Color", "RTN Color" y "EVT Color" nos permiten seleccionar el color de texto para una alarma no reconocida, para una alarma reconocida, para una alarma que ha retornado a la normalidad y para los eventos respectivamente.

Un evento ocurre en cualquier momento en que un dato ha sido cambiado por el operador o como respuesta a un cambio en el estado del proceso.

La opción “Local” de la sección “Display Alarms” nos permite observar los eventos del nodo local.

Para definir el formato en el que aparecerán los mensajes de alarmas en nuestro Objeto de Alarmas presione el botón “Format Alarm Message...”, esto producirá que la caja de diálogo “Formato del Mensaje de Alarma” aparezca. En la figura 4.9 se muestra la mencionada caja de diálogo.



Figura 4.9

Formato del Mensaje de Alarma.

En esta caja de diálogo podemos definir el formato de los mensajes de alarmas que se muestran en el Objeto de Alarmas. El orden vertical en el que se

muestran las diferentes opciones corresponde al orden horizontal en el que se mostrarán los mensajes de alarmas en el Objeto de Alarmas, de esta forma la fecha aparecería en el extremo izquierdo del mensaje y el estado de la alarma aparecería en la última columna derecha del mensaje de alarma.

Las opciones que hemos seleccionado para nuestro Objeto de Alarmas son las siguientes:

- Fecha.
- Hora.
- Evento.
- Nombre del Grupo de Alarma.
- Estado de la Alarma.

La fecha y hora puede seleccionarse en los formatos que se muestran a la derecha de cada una de estas opciones. Las demás opciones se seleccionan marcando con una "X" el pequeño cuadro a la izquierda de cada una de ellas. Un ejemplo del formato seleccionado para los mensajes de alarmas es mostrado en una línea ubicada en la parte inferior de la caja de diálogo. Luego de escoger las opciones deseadas presione el botón "OK", esto cerrará la caja de diálogo "Formato del Mensaje de Alarma" y la definición de dicho formato

habrá concluido.

Las Alarms en INTouch son ordenadas en grupos y gerarquías con la finalidad de facilitar la administración de las mismas, si el usuario no crea un nuevo grupo de alarmas INTouch automáticamente asigna las alarmas creadas al denominado grupo Raíz “\$System”. Una de las opciones seleccionadas para nuestro formato de alarmas es “Nombre del Grupo de Alarma”, esto permitirá observar el nombre del grupo de alarma al cual pertenece la alarma mostrada en el Objeto de Alarmas en determinado momento.

La creación de un nuevo grupo de alarmas puede efectuarse de dos formas: Seleccionando el comando Special / Alarm Groups desde la barra de menú principal del Window Maker o creándolos al mismo tiempo en el que se esta definiendo un Tagname en el Diccionario de Tagnames. En cualquier caso la caja de diálogo “Selección de Grupo de Alarmas” que se muestra en figura 4.10 aparecerá.

Presione el botón “Add...” para acceder a la caja de diálogo “Agregar Grupo de Alarmas” que se muestra en la figura 4.11

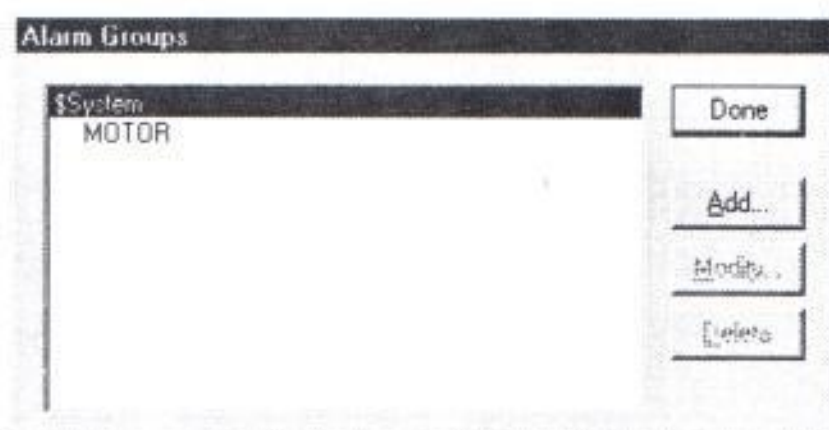


Figura 4.10

Selección de Grupo de Alarmas

Los botones “Modify” y “Delete” estarán deshabilitados hasta que un nuevo grupo de alarmas sea creado. Cabe indicar que el Grupo Raíz “\$System” no puede ser modificado o borrado.

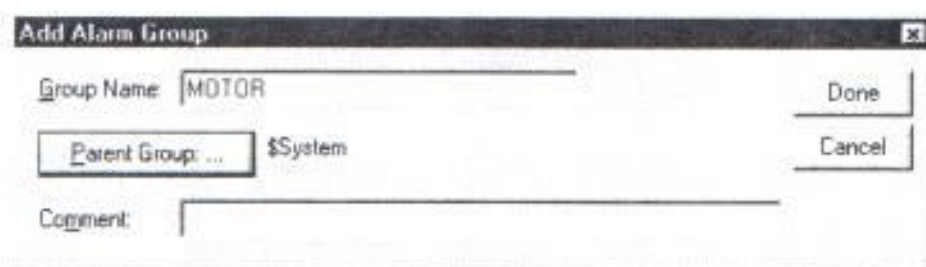


Figura 4.11

Caja de diálogo “Agregar Grupo de Alarmas”

Ingrese el nombre del nuevo grupo de alarmas que esta creando en el campo “Group Name”. Los grupos de alarmas pueden tener a su vez otros subgrupos de alarmas de menor jerarquía, esta organización de grupos no puede tener más

de 8 niveles de jerarquía donde el nivel más alto de gerarquía le corresponde al grupo raíz "\$System". Para definir el grupo de mayor jerarquía al cual pertenece el grupo que estamos creando presione el botón "Parent Group" y escoja un grupo de la lista que se muestra. Una vez hecho esto se mostrará el nombre del grupo escogido a la derecha del botón "Parent Group" de la caja de diálogo mostrada en la figura 4.11.

Para asignar un Tagname a un grupo de alarma, abra el Diccionario de Tagnames que se muestra en la figura 4.12 invocando el comando Special / Tagname Dictionary (Ctrl+T), seleccione el Tagname deseado de la lista que se muestra al presionar el botón "Select" del Diccionario y luego presionando el botón "Group" escoja el nombre del grupo de alarmas al cual se quiere asignar el Tagname especificado.

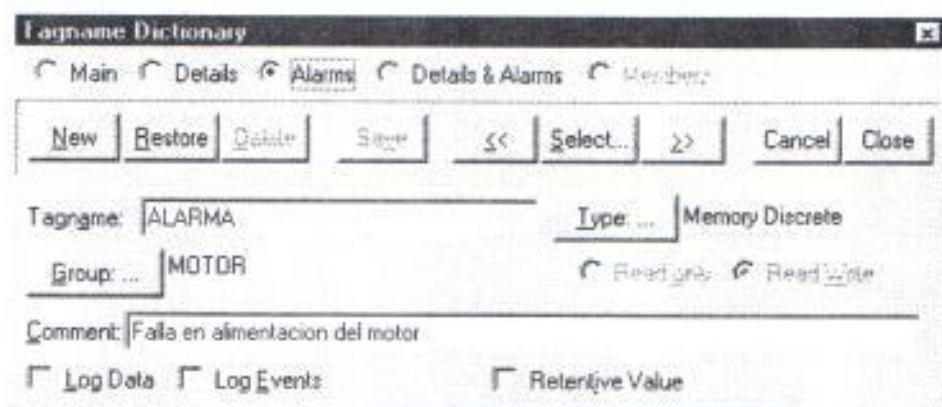


Figura 4.12

Diccionario de Tagnames.

En nuestra aplicación el Tagname “Alarma” representa un estado de alarma provocado por falla de alimentación en el motor. Como se observa en la figura 4.12 este Tagname es de tipo “Memoria Discreto” y su estado de alarma se activa cuando su valor es “1”, además ha sido asignado al grupo de alarmas denominado “MOTOR”. Para definir la alarma asignada a este tagname marque la opción “Alarms” en el Diccionario de Tagnames, esto producirá que se despliegue la ventana mostrada en la figura 4.13.



Figura 4.13

Definición de Alarma para “ALARMA”.

En la sección “Alarm State” seleccione el estado en el cual el Tagname “Alarma” representará una alarma. Si selecciona “On” el Tagname Discreto representará una alarma cuando su valor sea “1”, si selecciona “Off” el Tagname representará una alarma cuando sea “0”. En el campo “Priority” ingrese un valor entero entre 1 y 999, este valor representa la prioridad de la alarma. La prioridad es otro concepto que nos permite administrar eficazmente la impresión, preestación en pantalla y almacenamiento en archivos históricos de todas las alarmas definidas en nuestra aplicación.

4.4.2 IMPRESION DE ALARMAS.

La impresión de alarmas es una herramienta que nos permite obtener reportes escritos de la sucesión de alarmas y eventos generados en nuestro proceso o sistema controlado. En nuestra aplicación se ha considerado la impresión de alarmas pertenecientes al grupo "MOTOR", lo que nos permite llevar un registro escrito de todas las fallas producidas en el motor que acciona nuestro sistema de apertura y cierre.

Para definir la impresión de alarmas seleccione el comando Special / Configure / Alarms.. en la barra de menú del Window Maker. Este comando provocará que la caja de diálogo "Propiedades de Alarmas" que se muestra en la figura 4.14 aparezca sobre la pantalla.

En la caja de diálogo "Propiedades de Alarmas" se podrán definir además de la impresión de las mismas los atributos para el almacenamiento de los eventos de alarmas en los archivos históricos de InTouch. En nuestra aplicación esta función no ha sido utilizada.

En la figura 4.14 podemos observar la configuración realizada en las propiedades de las alarmas para la impresión de las mismas.

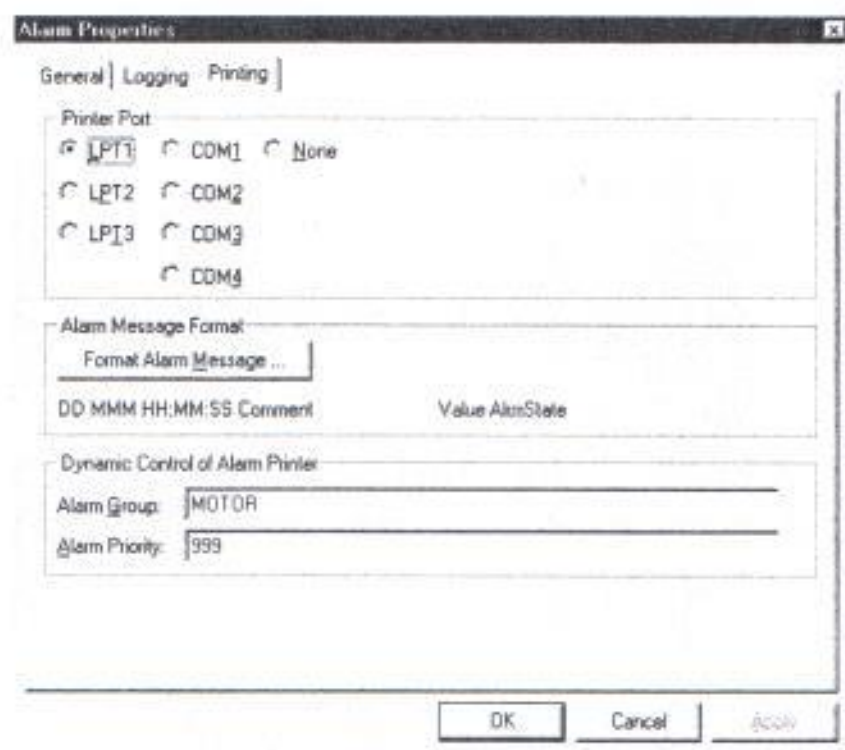


Figura 4.14
Propiedades de Alarmas.

Seleccione el puerto utilizado por la impresora para la impresión de las alarmas marcando el círculo que se encuentra a la izquierda de cada opción, en nuestro caso el puerto seleccionado es el puerto paralelo “LPT1”.

Al igual que en la configuración del Objeto de Alarmas, podemos elegir el formato para la impresión de las alarmas presionando el botón “Format Alarm Message...”, esto hará que la caja de diálogo “Formato del Mensaje de Alarma” que se muestra en la figura 4.9 aparezca.

Format Alarm Message

Date MM/DD MMM DD MM/DD/YY MMM DD YYYY
 DD/MM DD MMM DD/MM/YY DD MMM YYYY

Time 24 Hour AM/PM HH MM SS MSec

Event (ACK, RTN, ALM, ...)

Alarm Type (HIHI, SDEV, OPF, ...)

Operator Length:

Priority

Comment Length:

Tagname Length:

Group Name Length:

Value Length:

Limit Length:

Alarm State (UNACK_ALM, ACK_ALM, etc.)

DD MMM HH:MM:SS Comment Value AlarmState

figura 4.15

Formato para la Impresión de Alarmas.


De la misma manera en que lo hicimos antes podemos elegir el formato para la impresión de las alarmas, en la figura 4.15 se muestra el formato seleccionado para la impresión.

Como podemos observar en la figura 4.15 el formato para la impresión de las alarmas consta de Fecha, Hora, Comentario, Valor y Estado de la Alarma. El comentario que será impreso es el mismo que se introdujo en el campo "Comment" en el Diccionario de Tagnames al definir la Alarma para el Tagname "ALARMA".

4.4.3 GRAFICO DE TENDENCIA HISTORICA.

Los Gráficos de Tendencia Histórica son gráficos que muestran una imagen instantánea de datos correspondientes a una hora y fecha en el pasado. Este tipo de Gráficos necesitan ser actualizados por medio de un script, una expresión o mediante una acción efectuada por el operador, de otra forma el Gráfico de Tendencia Histórica no mostrará la gráfica correspondiente a un intervalo de tiempo actualizado en el que se incorpore la hora actual.

En nuestra aplicación se ha implementado un Gráfico de Tendencia Histórica que nos permita observar en forma gráfica el número de aperturas de nuestro sistema con período de 2 minutos.

Para crear un Gráfico de Tendencia Histórica, seleccione la herramienta “Historical Trend” presionando el botón  en la caja de herramientas del Window Maker. Posicione el cursor en el lugar deseado de la ventana “Principal” y coloque el Gráfico de Tendencias en el lugar seleccionado. Para configurar el Gráfico de Tendencias posicione el puntero sobre nuestro Gráfico y presione dos veces el botón izquierdo del ratón. La caja de diálogo

“Configuración de Tendencia Histórica” que se muestra en la figura 4.16 aparecerá en la pantalla.

En el campo “Historical Tag” ingrese el nombre del Tagname asignado para el Gráfico de Tendencias, el tipo de este Tag será definido automáticamente por el sistema como “Hist Trend”. En el campo “Initial Time Span” de la sección “Chart Time” ingrese la longitud de tiempo que el Gráfico de Tendencias mostrará en el eje X y seleccione las unidades para la longitud de tiempo ingresada. Para que el gráfico cubra el rango de valores mínimo

Historical Trend Configuration

Historical Tag:

Chart Time
Initial Time Span:
 Secs Mins Hrs Days

Initial Display Mode
 Min/Max Average

Color
 Chart Color:
 Border Color:

Time Divisions
 Number of Major Div:
 Minor Div/Major Div:
 Top Labels Bottom Labels
 Major Div/Time Label:
 Time: MM DD YY
 HH MM SS

Value Divisions
 Number of Major Div:
 Minor Div/Major Div:
 Left Labels Right Labels
 Major Div/Value Label:
 Min Value: Max:

Pen	Tagname	Color	Width	Pen	Tagname	Color	Width
1	CONT	<input type="checkbox"/>	1	5		<input checked="" type="checkbox"/>	1
2		<input checked="" type="checkbox"/>	1	6		<input checked="" type="checkbox"/>	1
3		<input type="checkbox"/>	1	7		<input type="checkbox"/>	1
4		<input checked="" type="checkbox"/>	1	8		<input checked="" type="checkbox"/>	1

Allow runtime changes

Figura 4.16
Caja de Diálogo “Configuración de Tendencia Histórica”.

a máximo de la escala de los tagnames graficados escoja la opción "Min/Max" en la sección "Initial Display Mode". A continuación escoja los colores para el fondo del gráfico y para el borde del mismo en las opciones "Chart Color" y "Border Color" respectivamente.

El número de divisiones mayores para la escala de tiempo (eje X) y el número de divisiones menores por cada división mayor para la misma escala, así como sus respectivos colores, es determinado en los campos "Number of Major Div." y "Minor Div / Major Div" de la sección "Time Divisions" respectivamente. los campos "Number of Major Div." y "Minor Div. / Major Div" de la sección "Value Divisions" determinan respectivamente el número de divisiones mayores y el número de divisiones menores por cada división mayor de la escala vertical del gráfico.

Luego de definir las divisiones tanto verticales como horizontales, la siguiente sección nos permite configurar las etiquetas para las escalas horizontales y verticales. Para la escala de tiempo, seleccione la opción "Bottom Labels" para que las etiquetas aparezcan en la parte inferior de la escala horizontal. El campo "Major Div / Time Label" permite definir el número de divisiones mayores por

cada etiqueta de la escala de tiempo. También puede escoger el formato para las etiquetas de la escala de tiempo.

En los campos "Min Value" y "Max." ingrese el rango para la escala vertical. Las unidades para este rango están en porcentajes de la escala de ingeniería definida para los tags que se van a graficar, por esta razón se recomienda que estos valores sean 0 - 100%.

Los siguientes ocho campos se utilizan para introducir las expresiones o los tags que van a ser dibujados en el Gráfico de Tendencias con sus respectivos colores. La capacidad de poder graficar expresiones en las que se involucren tagnames, nos permite mostrar gráficos de tendencias de tagnames con rangos ampliamente diferentes.

La selección de la opción "Allow Runtime Changes" le permite al operador realizar cambios en la configuración del Gráfico de Tendencias durante la ejecución del Runtime de InTouch. Para efectuar dichos cambios coloque el puntero sobre el Gráfico y presione una vez el botón izquierdo del ratón.

4.4.4 ESCRITURA DE TAGNAMES A LOS ARCHIVOS HISTORICOS.

Debido a que los Gráficos de Tendencia Histórica toman los datos que son graficados de los archivos históricos, es necesario que todos los tagnames que van a ser graficados sean escritos en dichos archivos históricos. Para que los valores de un tagname sean grabados en los archivos históricos de InTouch es necesario seleccionar la opción "Log Data" al definir el tagname en el Diccionario de Tagnames como se muestra en la figura 4.17 para el tagname "CONT".

4.4.5 HABILITACION DEL ALMACENAMIENTO DE ARCHIVOS HISTORICOS.

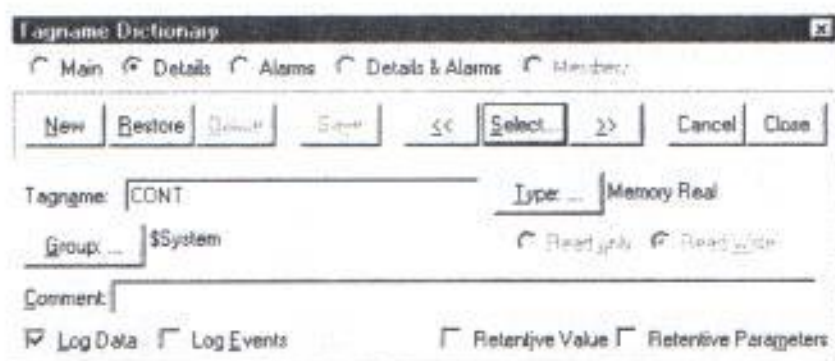


Figura 4.17

Definición del Tagname "CONT".

Para permitir que los tagnames sean escritos a los archivos históricos, la función de "Almacenamiento" debe estar habilitada. Para tal efecto ejecute el comando

Special / Configure / Historical Logging, esto producirá que la caja de diálogo “Propiedades de Almacenamiento Histórico” que se muestra en la figura 4.18 aparezca sobre la pantalla.

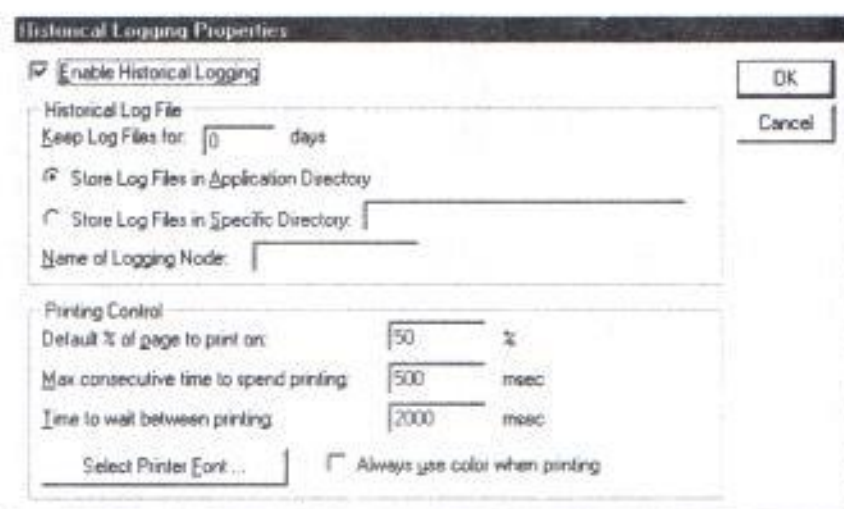


Figura 4.18

Caja de diálogo “Propiedades de Almacenamiento Histórico”. Para habilitar la función de Almacenamiento Histórico seleccione la opción “Enable Historical Logging..”. Al habilitar esta función se crearán diariamente dos archivos, los cuales serán colocados en el directorio seleccionado (que en nuestro caso es el mismo directorio que contiene nuestra aplicación, ya que se ha seleccionado la opción “Store Log Files in Application Directory” . Estos archivos serán automáticamente nombrados de la siguiente manera:

YYMMDD00.LGH

YYMMDD00.IDX

Donde: YY es el año de creación del archivo.

MM es el mes en el que fue creado el archivo (01-12).


DD es el día de creación del archivo (01-31).


00 siempre se muestran.



El campo "Keep Log Files for ... Days" nos permite definir el número de días que van a ser retenidos en el disco duro los archivos creados. Si ingresamos "0" en este campo, los archivos históricos serán mantenidos indefinidamente.

4.4.6 ACTUALIZACION DE LOS GRAFICOS DE TENDENCIA HISTORICA.

Como mencionamos anteriormente los Gráficos de Tendencia Histórica necesitan ser actualizados mediante un script o mediante una acción ejecutada por el operador para mostrar una gráfica correspondiente a la longitud de tiempo especificada que involucre la hora actual. En nuestro caso la actualización del gráfico la puede realizar manualmente el operador presionando el botón "UpDate" ubicado en la parte inferior del Gráfico. Este botón tiene asignado un Script de Acción que contiene la línea "HTUpdateToCurrent Time ("Aperturas");". Este comando permite actualizar a la hora actual el Gráfico de Tendencia Histórica que en nuestra aplicación se denomina "Aperturas".

El botón  nos permite hacer un acercamiento en el Gráfico de Tendencia Histórica reduciendo el ancho del gráfico a la mitad tomando como referencia el tiempo central del gráfico.

El botón  permite que el ancho del gráfico sea multiplicado por dos manteniendo el tiempo central en la nueva escala de tiempo, esta función proporciona un efecto de alejamiento del Gráfico de Tendencia.

Los botones  y  nos permiten desplazarnos a la izquierda y a la derecha en la escala de tiempo, de esta forma podemos observar el Gráfico de Tendencias para un intervalo de tiempo en el pasado.

4.4.7 PANEL DE CONTROL.

La figura 4.19 nos muestra el Panel de Control de nuestra aplicación, en él se encuentran los mandos que le permitirán al operador controlar el sistema.



Figura 4.19

Panel de Control.

El botón “ENTRAR” es utilizado para simular la activación del sensor de movimiento exterior, cuando este botón sea presionado el sensor se activará y permanecerá así hasta volver a presionar el botón. Para indicar la activación de este sensor se ha implementado una luz intermitente ubicada sobre el lado izquierdo de la puerta.

El botón “SALIR” es utilizado para simular la activación del sensor de movimiento interior, cuando este botón sea presionado el sensor se activará y permanecerá así hasta volver a presionar el botón. Para indicar la activación de este sensor se ha implementado como una luz intermitente ubicada sobre el lado derecho de la puerta.

El “Slider” vertical que se muestra en la figura 4.19 es utilizado para controlar

la velocidad de apertura y cierre de la puerta, la velocidad es incrementada al subir el Slider y reducida al bajarlo.

También se ha implementado el botón "ALARM" para simular la activación de una alarma cuando la alimentación en el motor falle y produzca la paralización del mismo, interrumpiendo de esta forma el normal funcionamiento del sistema. Cuando este botón es presionado se activará la alarma y aparecerá la ventana "AVISO" en la esquina inferior derecha de la ventana "Principal". La finalidad de esta ventana "AVISO" de tipo "PopUp" es hacer notar al operador la ocurrencia de una alarma. Asimismo cuando ocurre una alarma se efectuará un reporte impreso de la misma y aparecerá en el Objeto de Alarmas con los formatos correspondientes anteriormente indicados. Para simular el retorno de la alarma a la normalidad presione nuevamente el botón "ALARM", se producirá también un reporte impreso y se observará dicho evento en el Objeto de Alarmas.

El botón "ACK ALM" es utilizado para que el operador efectúe el reconocimiento de las alarmas activas y de esta forma esté enterado del estado

la velocidad de apertura y cierre de la puerta, la velocidad es incrementada al subir el Slider y reducida al bajarlo.

También se ha implementado el botón "ALARM" para simular la activación de una alarma cuando la alimentación en el motor falle y produzca la paralización del mismo, interrumpiendo de esta forma el normal funcionamiento del sistema. Cuando este botón es presionado se activará la alarma y aparecerá la ventana "AVISO" en la esquina inferior derecha de la ventana "Principal". La finalidad de esta ventana "AVISO" de tipo "PopUp" es hacer notar al operador la ocurrencia de una alarma. Asimismo cuando ocurre una alarma se efectuará un reporte impreso de la misma y aparecerá en el Objeto de Alarmas con los formatos correspondientes anteriormente indicados. Para simular el retorno de la alarma a la normalidad presione nuevamente el botón "ALARM", se producirá también un reporte impreso y se observará dicho evento en el Objeto de Alarmas.

El botón "ACK ALM" es utilizado para que el operador efectúe el reconocimiento de las alarmas activas y de esta forma esté enterado del estado actual del sistema o proceso. Para reconocer la alarma el operador deberá presionar este botón, al hacerlo observará que la ventana "AVISO"

desaparecerá. También se producirá un reporte impreso y se podrá observar este evento en el Objeto de Alarmas.

Existe también una “Palanca Sectora” para los modos de operación manual y automático del sistema. Para seleccionar el modo deseado el operador deberá posicionar el puntero sobre este objeto y presionar una vez el botón izquierdo del ratón, el operador observará el cambio de posición de la palanca y verificará que se produce un reporte impreso de este evento.

Finalmente, se ha implementado un reloj digital en la esquina inferior izquierda del panel para tener una visualización rápida de la hora.

4.4.8 PANEL DE VISUALIZACION.

En la figura 4.20 se muestra el “Panel de Visualización” por medio del cual el operador podrá obtener una descripción resumida en forma visual del estado del proceso de apertura y cierre, y del estado del motor.

La parte superior del Panel de Visualización está destinada a la puerta, en ella se

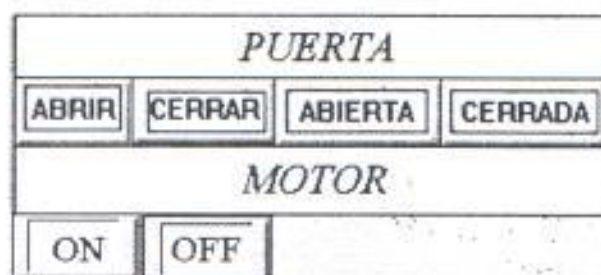


Figura 4.20

Panel de Visualización.

observan los botones “ABRIR” y “CERRAR” que son utilizados para abrir y cerrar la puerta respectivamente cuando el modo de operación seleccionado es “Manual”. Para abrir o cerrar la puerta el operador deberá presionar una vez el botón correspondiente, al hacerlo cada uno de los botones presentará un color característico, verde en el caso de “ABRIR” y rojo en el caso de “CERRAR”. Cabe indicar que cuando el sistema se encuentra operando en modo “AUTO” estos botones desaparecen.

Asimismo se ha implementado una señalización para indicar la activación de los denominados Switch de Fin de Carrera que indican los estados de “ABIERTA” y “CERRADA” de la puerta y a la vez mandan a desenergizar el motor cuando la puerta se encuentra en uno de los dos estados.

La parte inferior del Panel está destinada al motor, aquí se han implementado

los visualizadores “ON” y “OFF” que nos indican respectivamente los estados de “Encendido” y “Apagado” del Motor. Cuando el motor se encuentra en uno de estos dos estados los visualizadores se tomarán de color verde.

4.4.9 SISTEMA DE APERTURA Y CIERRE.

En la figura 4.21 se muestran los denominamos componentes del Sistema de Apertura y Cierre. Consta de la puerta, dos paredes laterales y dos luces intermitentes de señalización que se activan cuando los sensores de movimiento también lo hacen. Esta es la parte de nuestra aplicación que posee animación dinámica, el operador podrá observar el desplazamiento de la puerta hacia la derecha cuando ésta se abre y el desplazamiento de retorno hacia la izquierda cuando la puerta se cierra.

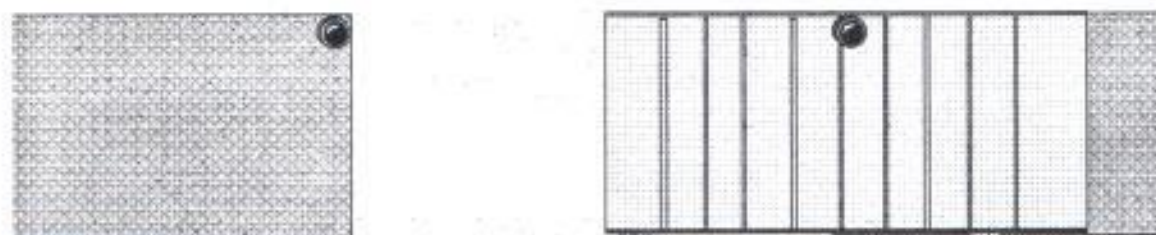


Figura 4.21

Sistema de Apertura y Cierre.

4.4.10 SCRIPT DE APLICACION.

Los Scripts son herramientas que permiten expandir las capacidades de InTouch proveyendo la habilidad para ejecutar comandos y operadores lógicos basados en criterios específicos. Por medio de los Scripts una amplia variedad de funciones de sistema personalizadas y automáticas pueden ser creadas como por ejemplo una tecla presionada, una ventana que se abre, un valor que cambia, etc.

Los Scripts de Aplicación son enlazados a toda la aplicación y pueden ser usados para correr otras aplicaciones, crear simulaciones de procesos, calcular variables, etc. En nuestra aplicación este tipo de Script ha sido utilizado para simular todo el proceso de apertura y cierre automático. A continuación se lista el Script de Aplicación.

```
1) IF ALARMA AND ALARMA.UnAck THEN
```

```
    Show "AVISO";
```

```
ENDIF;
```

```
2) IF ALARMA <> 1 THEN
```

```
    IF AUTO THEN
```

```
        IF (ENTRAR OR SALIR) THEN
```

```
CERRAR=0;
ABRIR=1;
ENDIF;
IF (ABIERTA AND NOT( ENTRAR OR SALIR) )THEN
TIEMPO = TIEMPO +100;
IF TIEMPO >= 5000 THEN
TIEMPO = 0;
ABRIR=0;
CERRAR= 1;
ENDIF;
ENDIF;
ELSE
ENTRAR = 0;
SALIR = 0;
ENDIF;
3) IF (ABRIR AND NOT BAND2) THEN
CERRAR=0;
BAND1=1;
ELSE
BAND1=0;
```

```
    INC=0;
ENDIF;
IF (CERRAR AND NOT BAND1) THEN
    ABRIR = 0;
    BAND2=1;
ELSE
    BAND2=0;
ENDIF;
IF ABRIR == 1 THEN
    AVANCE = AVANCE + INCREMENTO;
    IF INC==0 THEN
        CONT=CONT+1;
        INC=1;
    ENDIF;
    IF AVANCE >= 300 THEN
        AVANCE =300;
        ABRIR=0;
        ABIERTA=1;
    ENDIF;
ENDIF;
```

MINUTO1=\$Minute /2;

IF (Int(MINUTO1)≠MINUTO1) THEN

IF (\$Second >= 1 AND \$Second <=2) THEN

CONT=0;

ENDIF;

ENDIF;

IF AVANCE <> 300 THEN

ABIERTA=0;

ENDIF;

IF (CERRAR =1 AND AVANCE >=0) THEN

AVANCE= AVANCE - INCREMENTO;

IF AVANCE <= 0 THEN

AVANCE = 0;

CERRAR=0;

CERRADA=1;

ENDIF;

ENDIF;

IF AVANCE <> 0 THEN

```
CERRADA=0;
```

```
ENDIF;
```

```
ENDIF;
```

El bloque 1 se ha implementado para mostrar la ventana tipo "Pop Up" denominada "Aviso" siempre que una alarma esté activa y no halla sido reconocida por el operador. En el bloque 2 se comprueba primero que no exista una alarma activa y luego que el sistema se encuentre en modo automático, las líneas siguientes estan orientadas a la simulación del proceso en modo automático.

Las líneas del bloque 3 estan orientadas a la simulación del proceso en modo automático y a la programación de algunos comandos que son utilizados en los dos modos de funcionamiento.

A continuación se muestra una tabla donde se listan todos los Tagnames utilizados en el Script de Aplicación.

En la columna "Nombre" se indican las denominaciones que tiene cada Tagname dentro del programa. En la columna "Tipo" se indica la naturaleza

definida dentro de InTouch para cada uno de los Tagnames. Finalmente la columna "Descripción" nos muestra la utilidad y el significado de cada Tagname en el programa.

NOMBRE	TIPO	DESCRIPCION
ALARMA	Memoria-Discreta	Indica una Alarma en el motor.
AUTO	Memoria-Discreta	Indica el modo automático de operación.
ENTRAR	Memoria-Discreta	Simula un sensor externo de movimiento.
SALIR	Memoria-Discreta	Simula un sensor interno de movimiento.
CERRAR	Memoria-Discreta	Comando utilizado para cerrar la puerta.
ABRIR	Memoria-Discreta	Comando utilizado para abrir la puerta.
ABIERTA	Memoria-Discreta	Indica que la puerta está totalmente abierta.
CERRADA	Memoria-Discreta	Indica que la puerta está totalmente cerrada.
TIEMPO	Memoria-Entero	Tiempo de espera antes de cerrar la puerta.
AVANCE	Memoria-Entero	Indica la posición de la puerta.
INCREMENTO	Memoria-Entero	Utilizada para variar la velocidad del motor.
\$MINUTE	Sistema	Indica los minutos de la hora actual.
\$SECOND	Sistema	Indica los segundos de la hora actual.
CONT.	Memoria-Entero	Cuenta el número de aperturas cada 2 min.

TABLA 4-1
Lista de Tagnames.

BIBLIOGRAFIA

- 1.- Yan G. Warnock, "Programable Controllers Operations Applications",
Prentice Hall International , 1988.

- 2.- Wonderware Corporation, "Wonderware's Intouch Basic Training Course
Manual", Wonderware, 1996.

- 3.- Timothy J. Maloney, "Electrónica Industrial Moderna", Prentice Hall,
1997.

- 4.- Simatic, "Sistema de automatización S7-200 / Manual de sistema", Siemens
1997.