

ESCUELA POLITÉCNICA NACIONAL

ESCUELA DE FORMACIÓN TECNOLÓGICA

**CARRERA DE TECNOLOGÍA EN
MANTENIMIENTO INDUSTRIAL**

**DISEÑO Y CONSTRUCCIÓN DE UN TABLERO MODULAR DE
FORMACIÓN PROFESIONAL EN LA TECNOLOGÍA DE
AUTOMATIZACIÓN PARA EL LABORATORIO DE
AUTOMATIZACIÓN INDUSTRIAL DE PROCESOS MECÁNICOS
DE LA FACULTAD DE MECÁNICA DE LA EPN**

**PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE TECNÓLOGO EN
MANTENIMIENTO INDUSTRIAL**

**GUAYASAMÍN PAZ IRENE PAULINA
LÓPEZ CARRILLO NANCY FERNANDA**

DIRECTOR: DR. ÁLVARO AGUINAGA

QUITO, 22 DE OCTUBRE 2007

DECLARACIÓN

Yo Irene Paulina Guayasamín Paz y la Srta. Nancy Fernanda López Carrillo declaramos bajo juramento, que el trabajo aquí escrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración, cedemos nuestros derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su reglamento y por la normatividad institucional vigente.

GUAYASAMÍN PAZ IRENE PAUINA

LÓPEZ CARRILLO NANCY FERNANDA

CERTIFICACIÓN

Certificó que el presente trabajo fue desarrollado por las Señoritas:
GUAYASAMÍN PAZ IRENE PAULINA y LÓPEZ CARRILLO NANCY
FERNANDA, bajo mi dirección.

Dr. Álvaro Aguinaga
DIRECTOR DE TESIS

RESUMEN

El tablero de control para el laboratorio de mecánica se construye en vista de la necesidad de involucrar a Ingenieros y Tecnólogos de la Escuela Politécnica Nacional con los procesos de automatización y control industrial, el mismo que esta elaborado de manera didáctica, de tal forma que se puede realizar la simulación de programas con el control de un PLC, así como también la práctica de conexiones para entradas y salidas de operación.

El presente trabajo permitirá reforzar los conocimientos teóricos adquiridos a lo largo de nuestros estudios y ponerlos en práctica. Este proyecto de titulación esta estructurado de la siguiente manera:

En el primer capítulo trata sobre fundamentos generales del PLC: especificaciones, arquitectura, clasificación, módulos de comunicación y en el reconocimiento en prácticas de laboratorio.

El segundo capítulo consiste en la programación del PLC: componentes de un controlador lógico programable, unidad central de procesamiento CPU, memoria, terminal de programación, sistemas de entradas y salidas y fuente de alimentación.

El tercer capítulo contiene el conexionado o instalación del PLC: a la fuente de alimentación, entradas, salidas, límites de corriente y conexión al PC.

El cuarto capítulo consta del dimensionamiento y construcción del módulo didáctico, área del tablero, elementos de construcción y herramientas.

El quinto capítulo consiste en elaborar guías básicas de prácticas, con funciones básicas y aplicables.

El sexto capítulo trata de pruebas y calibración, equipos de pruebas y pruebas realizadas.

AGRADECIMIENTOS

Yo Fernanda López agradezco a todas las personas que de una u otra forma me ayudaron a surgir como persona y llegar a ser una profesional.

A todos mis Profesores que aparte de enseñarme y compartir sus conocimientos me demostraron su don de gente y su calidad humana, en especial al Doc. Álvaro Aguinaga que siempre estuvo ahí guiándome por el sendero del conocimiento, y aparte dándome palabras de Aliento.....y por supuesto al Ing. Mario Granja que siempre ve por el beneficio de sus alumnos y nunca nos a puesto trabas para cualquier emergencia que se a tenido y siempre nos apoyado.

A Dios por haberme guiado siempre y nunca dejarme sola en los momentos más difíciles.

A mis padres que estuvieron junto a mí y por ellos he culminado mis estudios. También a la Escuela de Formación Tecnológica donde hice muchos amigos y compartí momentos tan bellos e inolvidables.

AGRADECIMIENTOS

Yo Paulina Guayasamín Paz agradezco a Dios quien con su grandeza ha iluminado mi mente en todo momento, para poder realizar este proyecto de la mejor manera.

A mis padres, Rodrigo Guayasamín y Jacinta Paz cuyo ejemplo me ha inspirado, a lo largo de mi vida para seguir superándome todos los días.

A mi esposo Alex y mi hija Alejandra que me apoyaron incondicionalmente con su paciencia y amor en todo momento.

A mis profesores les extiendo un agradecimiento especial ya que gracias a todas sus enseñanzas y consejos, que me dieron, contribuyeron en el desenvolvimiento de mi carrera.

A mis amigos de la Escuela Politécnica Nacional con quienes aprendí el valor de la amistad y confianza.

OBJETIVO DEL PROYECTO

OBJETIVO GENERAL

- Desarrollar un tablero modular con un sistema innovador de automatización en el laboratorio.
- Contribuir al proceso de enseñanza, aprendizaje de la tecnología de automatización en el laboratorio.

OBJETIVOS ESPECÍFICOS

- Mostrar de forma práctica como funciona los sistemas de automatización a través de este tablero modular.
- Analizar las características del laboratorio como espacio, temperatura, facilidad de obtención de energía.
- Construir el tablero con material sólido que soporte el corte y perforaciones para la ubicación de los elementos que lo constituyen.
- Utilizar elementos de control automático, control manual y Actuadores.
- Realizar pruebas de automatización para la verificación del funcionamiento

CONTENIDO

CAPITULO I

1. Fundamento

Teórico.....	1
1.1Laboratorio de Automatizacion de Procesesmecánicos.....	1
1.2 PLC.....	1
1.2.1 Funciones Básicas de Plc.....	3
1.2.1.1Nuevas Funcione.....	4
1.2.2 Especificaciones PLC.....	5
1.2.3 Arquitectura del Plc.....	5
1.2.3.1 Estructura Externa.....	6
1.2.3.1.1 Introducción.....	6
1.2.3.1.2 Estructura Compac.....	6
1.2.3.1.3 Estructura Semimodular.....	6
1.2.3.1.4 Estructura Modular.....	7
1.2.3.2 Estructura Interna.....	7
1.2.3.2.1.1 Introducción.....	6
1.2.3.2.1.2 Fuente de Alimentación.....	9
1.2.3.4Interfaces.....	9
1.2.4 Clasificación Del PLC.....	11
1.2.5 Entradas	12
1.2.5.1 Introducción.....	12
1.2.5.2 Entradas Digitales	12
1.2.6 Salidas	13
1.2.6.1 Salidas Digitales	13
1.2.6.2 Salidas Analógicas	14
1.2.7 Módulos e Interfaces De Comunicación.....	15
1.2.7.1 Mecanismos de Comunicación.....	15
1.2.7.2 Funciones de Comunicación.....	16

1.2.7.2.1 Interface Multipu(M.P.I.).....	17
1.2.7.2.2 Interface punto por pun(P.P.I.).....	17
1.2.7.2.3 ProfiDP.....	18

CAPITULO II

2. Programación PLC.....	21
2.1 Introducción a Programación.....	21
2.1.1 Tipos de Señal.....	22
2.1.1.1 SeñDiscreta.....	22
2.1.1.1.1 Entra.....	23
2.1.1.1.Salida.....	23
2.1.1.2 SeAnálogo.....	24
2.1.2 Representación de las CantidadBinarias.....	24
2.1.2.Bit.....	25
2.1.2.2Byte.....	25
2.1.2.3 Palabra.....	25
2.1.3.1DireccionamientoFijo.....	26
2.1.3.1.1 Direccionamiento Fijo del Tipo Oct(BYTE).....	27
2.1.3.1.2 Direccionamiento Fijo del TiHexadecimal.....	27
2.1.4 Programación PLC.....	28
2.1.5 Lenguajes....de programación más utilizados para los PLC son de dos tipo.....	29
2.2 Controlador lógiprogramable.....	32
2.2.1 Unidad centraproceso.....	33
2.2.1.1 Cp312.....	35
2.2.1.2 Cp213.....	35
2.2.1.3 Cpu314.....	36
Cpu315.....	36
2.2.1.4 Cpu 312D.....	36
2.2.1.5 Modos de Operación deCPU.....	36
2.2.1.2.1 Modo RUN-P.....	36

2.2.1.2.2 Modo RUN.....	37
2.2.1.2.3 Modo STOP.....	37
2.2.1.3 Indicadores de la CPU.....	37
2.2.2 Memoria.....	38
2.2.2.1 Existen varios tipos memorias.....	38
2.2.2.2 Memoria interna.....	39
2.2.2.3 Área de imágenes de entradas/salidas y Área interna (IR).....	40
2.2.2.3.1 Área especial (SR).....	40
2.2.2.3.2 Área auxiliar (AR).....	41
2.2.2.3.3 Área de enlace (L.....	41
2.2.2.3.4 Área de temporizadores y contadores (TIM/CNT.....	41
2.2.2.3.5 Área de datos (DM).....	42
2.2.2.4 Memoria Programa.....	42
2.2.3 Terminal de programación.....	42
2.2.2.3.1 Periférico.....	43
2.2.4 Sistemas de entradas ysalidas.....	44
2.2.4.1 Módulos de entradas	44
2.2.4.2 Módulos de salidas.....	45
2.2.4.2.1 Módulos de salidas a relés.....	46
2.2.4.2.2 Módulos de salidas a triac.....	46
2.2.4.2.3Módulos de salidas a Transistores a colector abierto.....	46
2.2.5 Fuente de alimentación.....	47
2.6.1 Fuentes de Alimentación del S7-300.....	47

CAPITULO III

3 Conexionado o instalación plc.....	49
3.1 Cableado de la fuente de alimentación y lacpu.....	49
3.2 Conexión y cableado de losmódulos.....	51
3.2.1 Cableado del módulo analógico.....	54
3.2.2 Encender la alimentación.....	56
3.2.2.1.1 Principio de cableado de transductores de intensidad.....	57

3.2.2.1.2 Principio de cableado de un transductor de intensidad a 2 hilos	57
3.2.2.1.2 Principio de cableado de un transductor de intensidad a 4 hilos	57
3.2.3 Montaje del sistema	58
3.2.4 Montaje del módulo analógico.....	60
3.2.4.1 Componentes del SM331.....	60
3.2.5 Propiedades del módulo analógico.....	61
3.2.6 Adaptadores de margen de medida.....	61
3.2.7 Montaje del módulo sm331.....	63
3.2.8 Montaje e Interconexión de los Módulos.....	64
3.3 Límites de corriente.....	67
3.3.1 Relés internos.....	68
3.3.2 Timers.....	68
3.3.3 Contadores.....	68
3.3.4 Secuenciador.....	69
3.4 Conexión a una pc.....	69

CAPITULO IV

4. Dimensionamiento y Construcción del Módulo Didáctico.....	70
4.1 Área de tablero.....	70
4.1.1 Área de la mesa-tablero.....	70
4.1.2 Longitudes para toma corrientes y cajón.....	72
4.2 Elementos para la construcción.....	72
4.2.1 Herramientas.....	76

CAPITULO V

5. Elaboración de Guías Básicas de Prácticas.....	78
5.1 Introducción	

5.2 Funciones Básicas de un Plc.....	78
5.2.1 Detección.	78
5.2.2 Mando.	78
5.2.3 Dialogo Hombre-Máquina.....	78
5.2.4 Programación.	79
5.3 Nuevas Funciones.....	79
5.3.1 Redes de Comunicación.	79
5.3.2 Sistemas de Supervisión.....	79
5.3.3 Control de Procesos Continuos.	79
5.3.4 Entradas-Salidas Distribuidas:.....	80
5.3.5 Buses de Campo.	80
5.4 Crear un Nuevo Proyecto Step 7	81
5.1.1 Selección de La Cpu	82
5.1.2 Definir Programa de Usuario Base	83
5.1.3 Asignación de un Nombre De Proyecto	84
5.1.4 Proyecto S7 Resultante Terminado	85
5.2 Configuración del Hardware	85
	12

5.2.1 Crear Configuración Del Hardware	85
5.2.2 Añadir Componentes Simatic.....	86
5.2.2.1 Añadir Módulo Analógico	87
5.2.3 Parámetros del Módulo Analógico.....	88
5.2.3.1 Funcionalidades del Sistema Basados En El Ejemplo.....	88
5.2.3.2 Recibir Datos de Parámetros.....	88
5.2.3.3 Explicación de los Diferentes Ajustes	90
5.2.4 Prueba de Conexión	92
5.3 Programa de Usuario Step 7	95
5.3.1 Tareas del Programa De Usuario	95
5.3.2 Crear Programa de Usuario	97
5.3.3 Descarga del Archivo Fuente	97
5.3.4 Importar Archivo Fuente	97
5.3.5 Compilar Código Fuente	99
5.4 Ejemplos En S7-300	101
5.4.1 Semáforo	104
5.4.2 Banda Transportadora	108
5.4.3 Variación de Voltaje	113
5.5 Aplicaciones del S7-300	115
5.5.1 Sistemas de Transporte	115
5.5.2 Sistemas de Elevación.....	115
5.5.3 Otras Aplicaciones	115
5.5.4 Campo De Aplicación	116

CAPITULO VI

6.1 Equipos e Prueba	117
6.2 Pruebas Realizadas.....	117
6.2.1 Probar El Programa De Usuario.....	117
6.2.1.1 Descargar Datos de Sistema y Programa de Usuario.....	117
6.2.1.2 Visualización de los Valores de los Sensores.....	118
6.2.1.2.1 Observación de valores.....	119
6.2.1.2.2 Forzado de valores.....	120
6.2.1.2.3 Particularidad al observar los valore.....	121
6.2.1.3 Representación de Valores Analógicos.....	122
6.2.1.4 Alarma de Diagnóstico.....	123
6.2.1.4.1 Leer desde PG información de diagnóstico.....	123
6.2.1.4.2 Mensajes Generales de Diagnóstico...	124
6.2.1.4.3 Mensajes de Diagnóstico por Canal....	125
6.2.1.4.3.1 Error dConfiguración/parametrización	125
6.2.1.4.3.2 Error de Modo Común.....	125
6.2.1.4.3.3 Rotura de hilo.....	125
6.2.1.4.3.4 Rebase por defecto.....	126
6.2.1.4.3.5 Rebase por exceso.....	126
6.2.1.5 Alarma de Proceso.....	127
6.2.2 Advertencia Tareas de Simulación.....	129
6.2.2.1 Iniciar la Simulación.....	129
6.2.2.2 Abrir un PLC simulado.....	130
6.2.2.3 Elegir el modo de simulación.....	131
6.2.2.3.1 Ciclo individual:	131
6.2.2.3.2. Ciclo continuo:.....	131
6.2.2.4 Cambiar el modo de operación de la CPU.....	132
6.2.2.5 Observar el programa simulado.....	132
6.2.2.5.1 Entrada:.....	132
6.2.2.5.2 Salida:.....	133
6.2.2.5.3 Marca:.....	133
6.2.2.5 .4 Temporizador:.....	133

6.2.2.5.5 Contador:.....	133
6.2.2.5.6 General:.....	133
6.2.2.5.7 Bits verticales:.....	133
6.2.2.5.7.1 Acumuladores:.....	134
6.2.2.5.7.2 Registros de bloques:.....	134
6.2.2.5.7.3 Pilas:	134
6.2.2.3 Utilizar el regulador.....	135
6.2.2.7 Consideraciones relativas a las entradas y salidas.....	136
6.2.2.3 Conectar y desconectar la alimentación de laCPU simulad.	136
6.2.2.8.1 Utilizar direcciones simbólicas.....	137

CAPITULO I

2. FUNDAMENTO TEORICO

2.1 LABORATORIO DE AUTOMATIZACIÓN DE PROCESOS MECÁNICOS

Implementar un laboratorio de automatización que permita generar nuevos conocimientos que ayude a los estudiantes a entender los sistemas de automatización, que encontramos en la industria de este nuevo milenio, para enfrentar retos difíciles debido a la intensificación de la competencia que resulta del proceso de globalización mundial.

Dicho laboratorio contará con una infraestructura adecuada para la ubicación del tablero de control, en el cual los estudiantes de la Escuela Politécnica Nacional tendrán a su alcance la solución integral para los problemas específicos de automatización.

1.2 EL PLC

Un autómata programable industrial (API) o Programmable logic controller (PLC), es un equipo electrónico, programable en lenguaje no informático, diseñado para controlar en tiempo real y en un ambiente de tipo industrial, además procesos secuenciales.

Un PLC trabaja en base a la información recibida por los captadores y el programa lógico interno, actuando sobre los accionadores de la instalación.

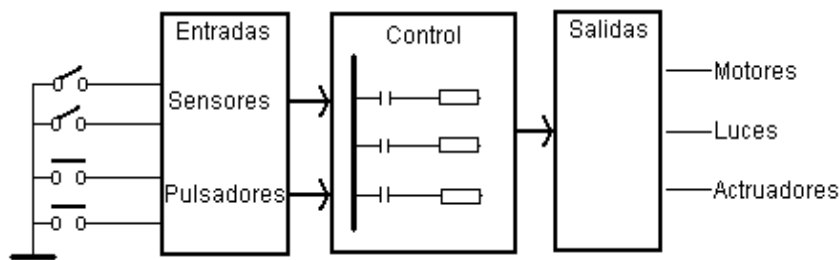


Figura 1-1 Acciones de Entradas y Salidas

Fuente: www.grupo-maser.com/PAG_Curso/Auto/

En este gráfico se puede observar mediante las entradas como son sensores, pulsadores, activan el control de mando para enviar las señales de salida hacia los diferentes elementos.

Que es un PLC? Los PLC son microcomputadoras especialmente preparadas para cumplir la función de un control robusto, para máquinas automáticas y sistemas de alta estabilidad durante su funcionamiento. Todas sus entradas y salidas están preparadas para operar directamente con switches y sensores de campo. Su conexionado se realiza a través de borneras que permiten conectar los cableados provenientes de los sensores y actuadores del sistema a la unidad de procesamiento.

El PLC posee un sistema operativo preinstalado que permite ejecutar un programa desarrollado por medio de un editor de circuitos eléctricos. Este entorno de desarrollo es llamado LADDER. El Ladder permite editar circuitos con contactos, bobinas, relés, contadores, timers y secuenciadores en su PC.

El circuito editado será traducido a un lenguaje de bajo nivel que pueda ser interpretado por el PLC. El programa compilado deberá ser cargado al PLC por medio de una interconexión de los puertos serie entre el PLC y el PC. Una vez que el programa haya sido bajado al PLC, este lo ejecutará indefinidamente, comportándose análogamente al circuito diseñado en la PC. Si el PLC no tuviera el sistema operativo e interprete Ladder, sería muy complicada la programación del sistema con un lenguaje de alto nivel de programación lineal. Gracias al sistema operativo todos los eventos y

ecuaciones lógicas son evaluados simultáneamente sin requerir de ningún esfuerzo por parte del programador, comportándose como un circuito eléctrico real.

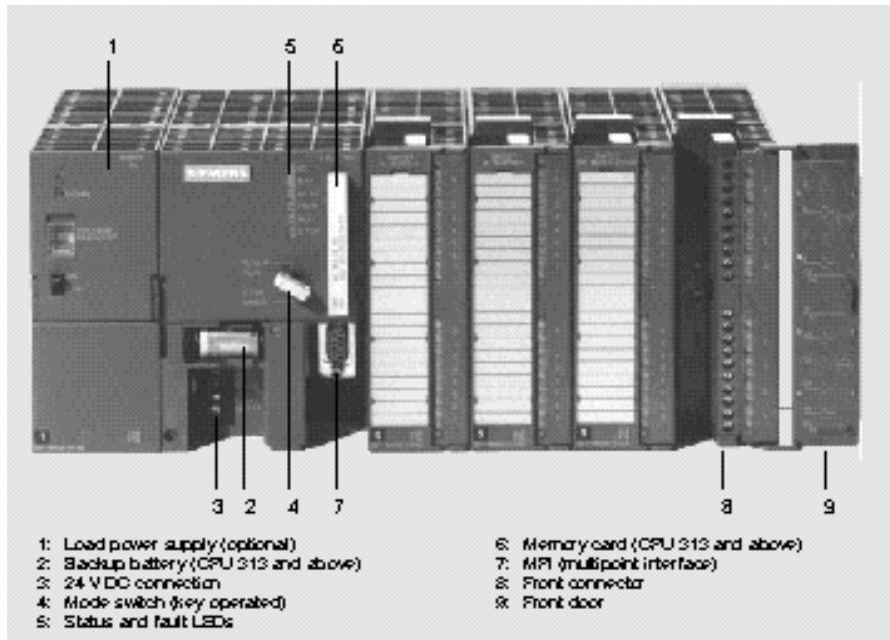


Figura 1-2 Principales Componentes del P.L.C.

Fuente: www.grupo-maser.com/PAG_Curso/Auto/

1.2.1 FUNCIONES BÁSICAS DE UN PLC

Detección: Lectura de la señal de los captadores distribuidos por el sistema de fabricación.

Mando: Elaborar y enviar las acciones al sistema, mediante los accionadores y preaccionadores.

Dialogo hombre máquina: Mantener un diálogo con los operarios de producción, obedeciendo sus consignas e informando del estado del proceso.

Programación: Introduce, elaborar y cambia el programa de aplicación del autómeta. El diálogo de programación debe permitir modificar el programa incluso con el autómeta, controlando la máquina.

1.2.1.1 Nuevas Funciones

Redes de comunicación:

Permiten establecer comunicación con otras partes de control. Las redes industriales permiten la comunicación y el intercambio de datos entre autómatas a tiempo real. En unos cuantos milisegundos pueden enviarse telegramas e intercambiar tablas de memoria compartida.

Sistemas de supervisión:

También los autómatas permiten comunicarse con ordenadores provistos de programas de supervisión industrial. Esta comunicación se realiza por una red industrial o por medio de una simple conexión por el puerto serie del ordenador.

Control de procesos continuos:

Además de dedicarse al control de sistemas de eventos discretos los autómatas llevan incorporadas funciones que permiten el control de procesos continuos. Disponen de módulos de entrada y salida analógicas y la posibilidad de ejecutar reguladores PID que están programados en el autómata.

Entradas- Salidas distribuidas:

Los módulos de entrada salida no tienen porqué estar en el armario del autómata. Pueden estar distribuidos por la instalación; se comunican con la unidad central del autómata mediante un cable de red.

Buses de campo:

Mediante un solo cable de comunicación se pueden conectar al bus, captadores y accionadores reemplazando al cableado tradicional. El autómata

consulta cíclicamente el estado de los captadores y actualiza el estado de los accionadores.

1.2.2 ESPECIFICACIONES DEL PLC

Las especificaciones técnicas que generalmente se encuentran en un PLC, depende de sus características y estas son:

Alimentación:	220/110Vca $\pm 10\%$ 50/60hz
Tolerancia de fuente:	$\pm 10\%$
Modelos de corriente continua	12 Vcc-10+50%, 110/48/24Vcc $\pm 30\%$
Micro corte de alimentación:	10ms
Tiempo de reposición apagado-encendido:	3 a 5 seg
Temperatura de trabajo:	0° a 60°C
Humedad relativa máxima:	90% sin condensar
Fuente incluida:	24Vcc no regulada
Carga máxima:	protección con corte a 300mA

1.2.3 ARQUITECTURA DEL PLC

La arquitectura de un autómata programable se puede describir también en el siguiente esquema:

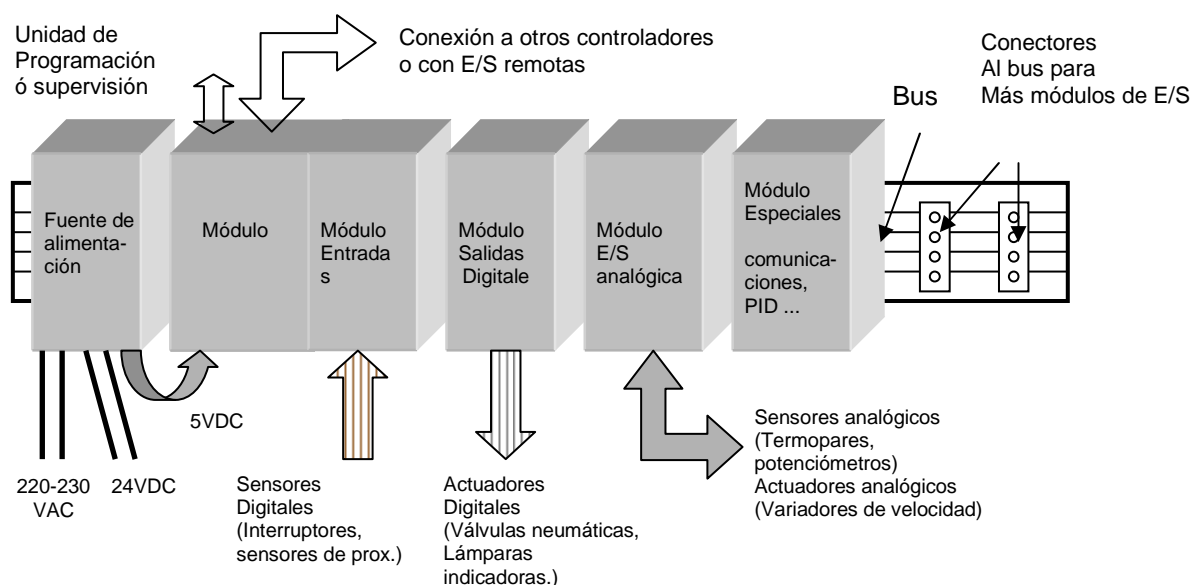


Figura 1-3 Arquitectura Típica de un PLC
Fuente: Manual/EPN/Automatización/Dr. Álvaro Aguinaga

1.2.3.1 Estructura Externa

1.2.3.1.1 Introducción

El término estructura externa o configuración externa de un autómata programable industrial, se refiere al aspecto físico exterior del mismo, bloques o elementos en que está dividido.

Actualmente son tres las estructuras más significativas que existen en el mercado:

Estructura compacta

Estructura semimodular. (Estructura Americana)

Estructura modular. (Estructura Europea)

1.2.3.1.2 Estructura Compacta

Este tipo de autómatas se distingue por presentar en un solo bloque todos sus elementos; esto es: Fuente de alimentación, CPU, Memorias, entradas/salidas, etc.

Son los autómatas de gama baja o nano autómatas los que suelen tener una estructura compacta. Su potencia de proceso suele ser muy limitada dedicándose a controlar máquinas muy pequeñas o cuadros de mando.

1.2.3.1.3 Estructura Semimodular

Se caracteriza por separar las E/S del resto del autómata, de tal forma que en un bloque compacto están reunidas las CPU, memoria de usuario o de programa, fuente de alimentación y separadamente las unidades de E/S.

Son los autómatas de gama media los que suelen tener una estructura semimodular (Americana).

1.2.3.1.4 Estructura Modular

Su característica principal es la de que existe un módulo para cada uno de los diferentes elementos que componen el autómata como puede ser una fuente de alimentación, CPU, E/S, etc. La sujeción de los mismos se hace por carril DIN, placa perforada o sobre RACK, en donde va alojado el BUS externo de unión de los distintos módulos que lo componen.

Son los autómatas de gama alta los que suelen tener una estructura modular, que permiten una gran flexibilidad en su constitución.

1.2.3.2 Estructura Interna

1.2.3.2.1 Introducción

En este apartado vamos a estudiar la estructura interna de cada uno de los diferentes elementos que componen el autómata, las funciones y funcionamiento de cada una de ellas.

El autómata está constituido por diferentes elementos, pero tres son los básicos:

CPU

Entradas

Salidas

Con las partes mencionadas podemos decir que tenemos un autómata pero para que sea operativo son necesarios otros elementos tales como:

Fuente de alimentación

Interfaces

La unidad o consola de programación

Los dispositivos periféricos

La CPU (Central Processing Unit) es la parte inteligente del sistema que interpreta las instrucciones del programa de usuario y consulta el estado de las entradas, dependiendo de dichos estados y del programa, ordena la activación de las salidas deseadas.

La CPU está constituida por los siguientes elementos:

- Procesador
- Memoria monitor del sistema
- Circuitos auxiliares

El procesador está constituido por el microprocesador, el reloj (generador de onda cuadrada) y algún chip auxiliar.

El microprocesador es un circuito integrado (chip), que realiza una gran cantidad de operaciones, que podemos agrupar en:

- Operaciones de tipo lógico
- Operaciones de tipo aritmético.
- Operaciones de control de la transferencia de la información dentro del autómata.

Para que el microprocesador pueda realizar todas estas operaciones está dotado de unos circuitos internos que son los siguientes:

Circuitos de la unidad aritmética y lógica o ALU: Es la parte del microprocesador donde se realizan los cálculos y las decisiones lógicas para controlar el autómata.

Circuitos de la unidad de control (UC) o Decodificador de instrucciones: Decodifica las instrucciones leídas en memoria y se generan las señales de control.

Acumulador: Es la encargada de almacenar el resultado de la última operación realizada por el ALU.

Flags: Flags o indicadores de resultado, que pueden ser consultados por el programa Contador de programa: Encargada de la lectura de las instrucciones de usuario.

Bus (interno): No son circuitos en si, sino zonas conductoras en paralelo que transmiten datos, direcciones, instrucciones y señales de control entre las diferentes partes del microprocesador.

1.2.3.3 Fuente de Alimentación

La fuente de alimentación proporciona las tensiones necesarias para el funcionamiento de los distintos circuitos del sistema.

La alimentación a la CPU puede ser de 24 Vcc, tensión muy frecuente en cuadros de distribución, o en alterna a 110/220 Vca. En cualquier caso es la propia CPU la que alimenta las interfaces conectadas a través del bus interno.

La alimentación a los circuitos E/S puede realizarse, según tipos, en alterna a 48/110/220 Vca o en continua a 12/24/48 Vcc.

La fuente de alimentación del autómatas puede incorporar una batería tampón, que se utiliza para el mantenimiento de algunas posiciones internas y del programa usuario en memoria RAM, cuando falla la alimentación o se apaga el autómatas.

1.2.3.4 Interfaces

En el control de un proceso automatizado, es imprescindible un dialogo entre operador-máquina junto con una comunicación entre la máquina y el autómatas, estas comunicaciones se establecerán por medio del conjunto de entradas y salidas del citado elemento.

Los autómatas son capaces de manejar tensiones y corrientes de nivel industrial, gracias a que disponen un bloque de circuitos de interfaz de E/S muy potente, que les permite conectarse directamente con los sensores y accionamientos del proceso.

De entre todos los tipos de interfaces que existen, las interfaces específicas permiten la conexión con elementos muy concretos del proceso de automatización. Se pueden distinguir entre ellas tres grupos bien diferenciados:

Entradas / salidas especiales.

Entradas / salidas inteligentes.

Procesadores periféricos inteligentes.

Las interfaces especiales del primer grupo se caracterizan por no influir en las variables de estado del proceso de automatización. Únicamente se encargan de adecuar las E/S, para que puedan ser inteligibles por la CPU, si son entradas, o para que puedan ser interpretadas correctamente por actuadores (motores, cilindros, etc.), en el caso de las salidas.

Las del segundo grupo admiten múltiples modos de configuración, por medio de unas combinaciones binarias situadas en la misma tarjeta. De esta forma se descarga la información a la unidad central, con las ventajas que conlleva.

Los procesadores periféricos inteligentes, son módulos que incluyen su propio procesador, memorias y puntos auxiliares de entrada / salida. Estos procesadores contienen en origen un programa especializado en la ejecución de una tarea concreta, a la que le basta conocer los puntos de consigna y los parámetros de aplicación para ejecutar, de forma autónoma e independiente de la CPU principal, el programa de control.

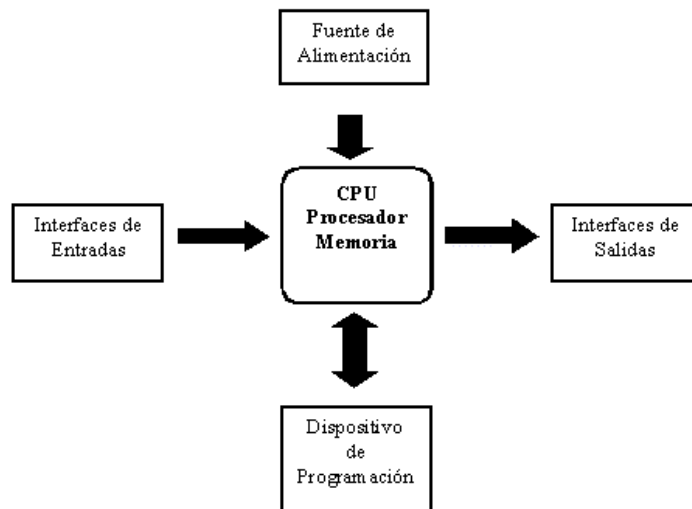


Figura 1-4 Estructura de un Controlador Lógico Programable

Fuente: www.es.wikipedia.org/Automatización_industrial

1.2.4 CLASIFICACIÓN DEL PLC

También se pueden clasificar los controladores lógicos programables de acuerdo con la complejidad de sus aplicaciones.

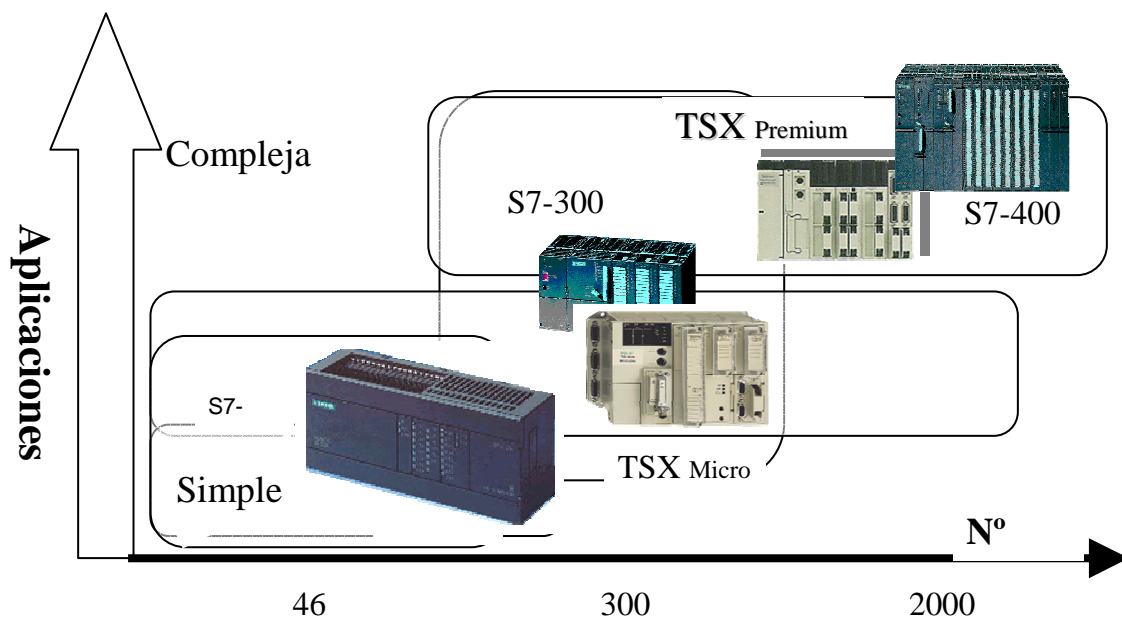


Figura 1-5 Gama de PLC's industriales

Fuente: Manual/EPN/Automatización/Dr. Alvaro Aguinaga

1.2.5 ENTRADAS

1.2.5.1 Introducción

La sección de entradas mediante el interfaz, adapta y codifica de forma comprensible para la CPU las señales procedentes de los dispositivos de entrada o captadores.

Hay dos tipos de entradas:

Entradas digitales

Entradas analógicas

La sección de salida también mediante interfaz trabaja de forma inversa a las entradas, es decir, decodifica las señales procedentes de la CPU, las amplifica y manda con ellas los dispositivos de salida o actuadores como lámparas, relés, etc. Aquí también existen unos interfaces de adaptación a las salidas de protección de circuitos internos.

Hay dos tipos de salidas:

Salidas digitales

Salidas analógicas

1.2.5.2 Entradas Digitales

Los módulos de entrada digitales permiten conectar a los autómatas captadores de todo tipo.

Los módulos de entrada digitales trabajan con señales de tensión, por ejemplo cuando por una vía llegan 24 voltios se interpreta como un "1" y cuando llegan cero voltios se interpreta como un "0"

El proceso de adquisición de la señal digital consta de varias etapas:

Protección contra sobretensiones

Filtrado

Puesta en forma de la onda

Aislamiento galvánico

1.2.5.3 Entradas Analógicas

Los módulos de entrada analógicas permiten que los autómatas programables trabajen con accionadores de mando analógico y lean señales de tipo analógico como pueden ser la temperatura, la presión o el caudal.

Los módulos de entradas analógicas convierten una magnitud analógica en un número que se deposita, en una variable interna del autómata. Lo que realiza es una conversión A/D, puesto que el autómata solo trabaja con señales digitales. Esta conversión se realiza con una precisión o resolución determinada (número de bits) y cada cierto intervalo de tiempo (periodo muestreo).

Los módulos de entrada analógica pueden leer tensión o intensidad.

El proceso de adquisición de la señal analógica consta de varias etapas:

Filtrado

Conversión A/D

Memoria interna

1.2.6 SALIDAS

1.2.6.1 Salidas Digitales

Un módulo de salida digital permite al autómata programable actuar sobre los preaccionadores y accionadores que admitan ordenes de tipo todo o nada.

El valor binario de las salidas digitales se convierte en la apertura o cierre de un relé interno del autómatas en el caso de módulos de salidas a relé.

En los módulos estáticos (bornero), los elementos que conmutan son los componentes electrónicos como transistores o triacs, y en los módulos electromecánicos son contactos de relés internos al módulo.

Los módulos de salidas estáticos al suministrar tensión, solo pueden actuar sobre los elementos que trabajan a la misma tensión, en cambio los módulos de salida electromecánicos, al ser libres de tensión, pueden actuar sobre elementos que trabajen a tensiones distintas.

El proceso de envío de la señal digital consta de varias etapas:

- Puesta en forma
- Aislamiento
- Circuito de mando (relé interno)
- Protección electrónica
- Tratamiento cortocircuitos

1.2.6.2 Salidas Analógicas

Los módulos de salida analógica permiten que el valor de una variable numérica interna del autómatas se convierta en tensión o intensidad.

Esta tensión o intensidad puede servir de referencia de mando para actuadores que admitan mando analógico como pueden ser los variadores de velocidad, las etapas de los tiristores de los hornos, reguladores de temperatura, permitiendo al autómatas realizar, funciones de regulación y control de procesos continuos.

El proceso de envío de la señal analógica consta de varias etapas:

- Aislamiento galvánico
- Conversión D/A
- Circuitos de amplificación y adaptación
- Protección electrónica de la salida

Como hemos visto las señales analógicas sufren un gran proceso de adaptación tanto en los módulos de entrada como en los módulos de salida.

Las funciones de conversión A/D y D/A que realiza son esenciales. Por ello los módulos de E/S analógicos se les consideran módulos de E/S especiales.

1.2.7 MODULOS E INTERFACES DE COMUNICACIÓN

El simatic S7-300 tiene diferentes interfaces de comunicación:

Procesadores de comunicación CP 343-5, CP 343-1 y CP 343 TCP para conexión al PROFIBUS y sistemas bus Industrial.

Procesador de comunicaciones CP 340 para conexión a sistemas punto a punto.

La interface multipunto (MPI) está integrada al CPU; para conexión simultánea de los mandos de programación, PC, sistemas MMI y sistemas de automatización SIMATIC S7, M7 o C7.

1.2.7.1 Mecanismos de Comunicación

El simatic S7-300 tiene varios mecanismos de comunicación:

Intercambio cíclico del conjunto de datos entre redes de CPU mediante la comunicación global de datos.

Comunicación de resultado transmitido por las redes utilizando bloques de comunicación.

Mediante el servicio de comunicación global de datos, las redes de CPU pueden intercambiar datos cíclicamente con cada una de las otras unidades centrales de procesamiento. Esto permite a una CPU acceder a la memoria de datos de otra CPU. La comunicación global de datos solo puede ser enviada vía interfaces multipunto (MPI).

1.2.7.2 Funciones de Comunicación

El PLC, al ser un elemento destinado a la Automatización y Control de los objetivos principales en el aumento de la Productividad o Cadencia y la disminución de los Tiempos Ciclos, no puede o mejor dicho no es un simple ejecutador de datos almacenados en su memoria para transmitir directivas a sus dispositivos que controla. Es decir, debe ser un elemento que en cualquier momento sea capaz de cambiar la tarea que realiza con simples cambios en su programación, ésta tarea sería imposible sin la ayuda de otros dispositivos tales como PCI, programadoras o paneles de control, dispositivos de campo, PLC, etc. Por lo tanto necesitamos comunicar al PLC. Estos conceptos no solo se basan en la Fabricación Flexible y una comunicación eficiente, depende esencialmente de la red en la que se encuentra trabajando el PLC. No solamente el PLC sino también los computadores industriales, unidades de programación, etc., que una vez conectados todos a la red, desde cualquier punto es posible acceder a cada uno de los componentes.

En particular el S7-300 de Siemens viene dotado con 3 interfaces para trabajar en equipo o red, ellos son:

El M.P.I. (Interface Multi Punto)

El P.P.I. (Interface Punto por Punto)

El Profibus-DP

Existen además a nivel industrial otras redes tales como la Profibus-FMS,

Industrial Ethernet, etc., pero no intervendrán en nuestro trabajo a pesar de que también puede ser conectado a cualquiera de ellas.

1.2.7.2.1 Interface Multipunto (M.P.I.)

Todas las CPU (312, 313, 314, 315 y 315 -2DP), lo incorporan desde fábrica. Con éste puerto se puede comunicar fácilmente a distancias reducidas sin requerir módulos adicionales; por ejemplo hacia equipos de M+V (manejo + visualización), unidades de programación y otros autómatas S7-300 o S7- 400 para probar programas o consultar valores de estado.

Se pueden enviar datos a 4 distintos aparatos al mismo tiempo y utilizando siempre el mismo puerto a una velocidad de 187,5 Kbits / seg o 187,5 Kbaudios. Para pequeñas redes de comunicación o pequeños volúmenes de datos la CPU ofrece el servicio de Datos Globales, que permite intercambiar cíclicamente cantidades de datos en paquetes de hasta 22 bytes como máximo.

Distancia máxima entre dos estaciones o nudos de red de MPI adyacentes: 50 metros (sin repetidores); 1100 metros (con dos repetidores); 9100 metros (con más de 10 repetidores en serie); por encima de los 500 Klm. (cable de fibra óptica, con módulos de conexión ópticas).

Capacidad de expansión: los componentes comprobadores de campo son usados para configurar la comunicación de interface multipunto: cables LAN, conectores LAN y repetidores RS485, desde el PROFIBUS y la línea de productos de entradas/salidas distribuidas.

Estos componentes permiten una óptima utilización de la configuración.

1.2.7.2.2 Interface punto por punto (P.P.I)

Esta interface permite la comunicación de nuestro dispositivo con otros tales

como módems, scanner, impresoras, etc., situados a una cierta distancia del PLC. En la parte frontal del módulo de la CPU posee fichas DB 9 o DB 25 para la comunicación serial vía RS 232 y RS 485. La conexión Punto a Punto puede ser establecida económicamente y convenientemente por medio del procesador de comunicaciones CP 340. Hay varios protocolos disponibles por debajo de las tres interfaces de conexión:

20 mA (TTY)

RS 232 C/V.24

RS 422 / RS485

Los siguientes dispositivos pueden ser conectados:

Controladores programables SIMATIC S7 y SIMATIC S5

Impresoras

Robots controladores

Modems

Scanners, lectores de códigos de barras, etc.

1.2.7.2.3 Profibus DP

Esta interface de comunicación es usada para gran capacidad de transmisión de datos, llamada Simatic Net o Sinec L2 de Siemens. El S7 300 mantiene una relación muy estrecha con él. Un módulo de comunicación permite conectarlo al Sinec L2 para comunicarse con otros autómatas Simatic y dispositivos de campo. La CPU 315 – 2DP ya la trae integrada. De éste modo, el autómata se adapta armoniosamente en arquitecturas descentralizadas que integran componentes de automatización y dispositivos de campo. El PLC puede desenvolverse aquí como maestro – esclavo, además también se dispone de los prácticos servicios de comunicación llamados Datos Globales. Para entablar comunicación se utilizan cables LAN, conectores LAN, repetidores, etc.

Digamos entonces que es una red suplementaria que ofrece un gran rendimiento, arquitectura abierta o descentralizada y gran robustez o confiabilidad.

Existe además la gran ventaja del Manejo + Visualización (paneles de operador, llamados Coros) que permite tanto en ésta interface como en las otras, de la búsqueda de errores a partir de cualquier dispositivo y así por ejemplo: generar una base de datos con los errores (hora y tipo) que puedan existir.

Los siguientes dispositivos pueden ser conectados como maestros:

SIMATIC S7-300 (vía CPU 315-2DP o CP 342-5DP)

SIMATIC S7-400 (vía CPU 41-2)

SIMATIC S5-115U-H, S5-135U, S5-155UH, S5 95U con interface de PROFIBUS, SIMATIC TI505

Dispositivos programables y Pc's con STEP7 (solo con CPU 41-2 y CPU 315-2)

Paneles del operador (OP).

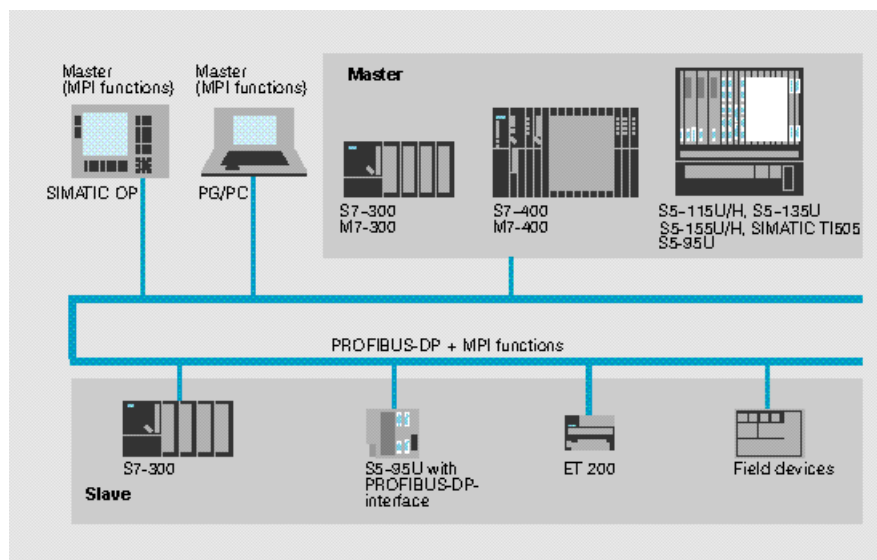


Figura 1-6 Profibus DP

Fuente: www.deremate.com.ar/accdb/viewitem.asp?idi=17454992

Los siguientes dispositivos pueden conectarse como esclavo:

ET 200U/B/C/L/M con dispositivos de entrada y salida distribuida.

S7-300 vía CP342-5

CPI 315-2 DP

Por encima de 10 repetidores, pueden ser conectados en series, por ejemplo para interconectar largas distancias entre alguno de los nodos del MPI dados.

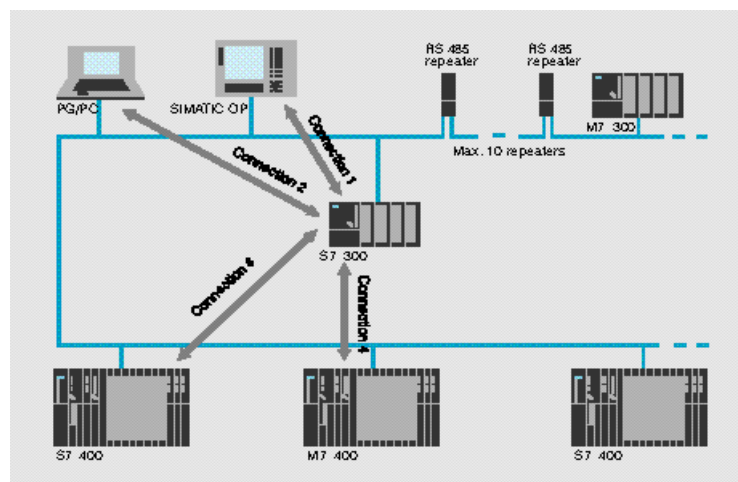


Figura 1-7 Profibus DP

Fuente: www.deremate.com.ar/accdb/viewitem.asp?idi=17454992

CAPITULO II

2. PROGRAMACIÓN DEL PLC

2.1 INTRODUCCION A LA PROGRAMACIÓN

El PLC, en la primera mitad de los 80, eran programado usando sistemas de programación propietarios o terminales de programación especializados, que ha menudo tenían teclas de función dedicadas a las representación de los elementos lógicos de los programas de PLC. Los programas eran guardados en cintas. Más recientemente, los programas PLC son escritos en aplicaciones especiales en un ordenador, y luego son descargados directamente mediante un cable o una red al PLC. Los PLC viejos usan una memoria no volátil (magnetic core memory) pero ahora los programas son guardados en una RAM con batería propia o en otros sistemas de memoria no volátil como las memoria flash.

El PLC fue diseñado para ser usado por electricistas que podían aprender a programar los PLC en el trabajo. Este PLC era programado con “lógica de escalera” ("ladder logic"). Los PLC's modernos pueden ser programados de muchas formas, desde la lógica de escalera hasta lenguajes de programación tradicionales como el BASIC o C. Otro método es usar la Lógica de Estados (State Logic), un lenguaje de programación de alto nivel diseñado para programas PLC basándose en los diagramas de transición de estados.

Recientemente, el estándar internacional IEC 61131-3 se está volviendo muy popular. IEC 61131-3 define cinco lenguajes de programación para los sistemas de control programables:

FBD (Function block diagram),

LD (Ladder diagram),

ST (Structured text, similar al language de programación Pascal),

IL (Instruction list)

SFC (Sequential function chart).

Antes de empezar con la programación propiamente dicha, es necesario definir algunos conceptos que proporcionen al estudiante las bases suficientes para comprender de la manera más clara, el desarrollo de los temas que se tocarán más adelante en lo referente a la programación básica y avanzada, así por ejemplo, el lector deberá estar en condiciones de diferenciar una señal discreta de una análoga, representar las cantidades binarias, estructurar una instrucción de mando, y tener presente las reglas básicas para las diferentes representaciones de los lenguajes de programación, etc. Por consiguiente, el éxito que se tenga en lo sucesivo dependerá de lo aprendido en esta parte introductoria.

2.1.1 TIPOS DE SEÑALES

Existen dos tipos de señales bien definidas que un PLC puede procesar, estos son:

2.1.1.1 Señal Discreta

Este tipo de señal es conocido también con los siguientes nombres:

Señal binaria

Señal digital

Señal lógica

Señal todo o nada (TON)

Se caracteriza porque sólo pueden adoptar uno de dos posibles estados o niveles. A estos dos estados posibles se le asocia para efectos del procesamiento el estado de señal "0" y el estado de señal "1". Así mismo, estos estados cuando se relaciona de acuerdo a su condición eléctrica se dice: no existe tensión y existe tensión. La magnitud de la tensión no interesa ya que

dependerá del diseño del componente electrónico que pueda asumir esta tensión nominal.

Como ejemplo se pueden citar aquellos dispositivos de campo de entrada y salida de donde provienen o se asigna una señal discreta con respecto a un PLC.

2.1.1.1.1 Entrada

Pulsador

Interruptor deposición

Interruptor fotoeléctrico, etc.

2.1.1.1.2 Salida

Contactor

Lámpara indicadora, etc.

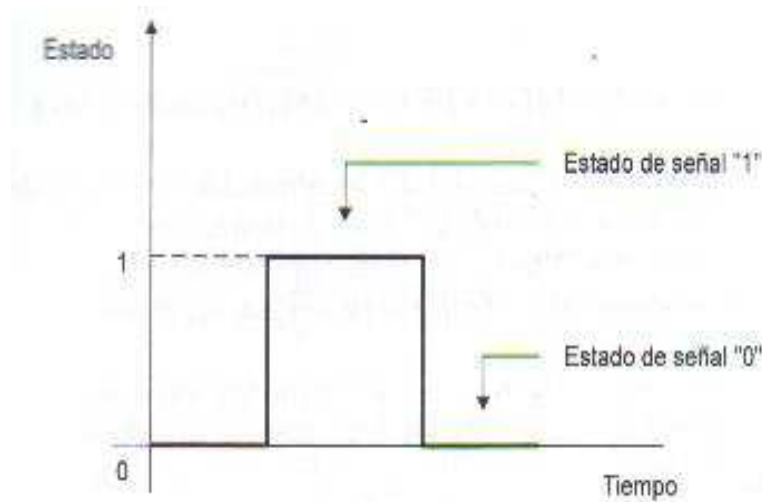


Figura 2-1 Estado vs Tiempo (1-0)

Fuente: www.grupo-maser.com/PAG_Curso/Auto/

2.1.1.2 Señal Análoga

Se conoce como señal análoga, aquella cuyo valor varía con el tiempo y en forma continua, pudiendo asumir un número infinito de valores entre sus límites mínimos y máximos.

A continuación se citan algunos parámetros físicos muy utilizados en los procesos industriales, tal que, en forma de señal análoga pueden ser controlados y medidos.

Temperatura

Velocidad

Presión

Flujo,

Nivel, etc.

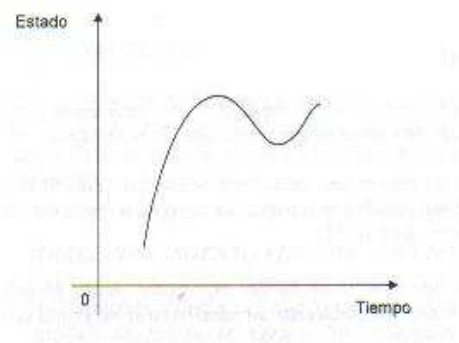


Figura 2-2 Estado vs Tiempo Señal Analógica

Fuente: www.grupo-maser.com/PAG_Curso/Auto/

2.1.2 REPRESENTACIÓN DE LAS CANTIDADES BINARIAS

Dado que el PLC recibe la información proveniente del proceso ya sea en forma discreta o análoga, la información se almacena en forma de una agrupación binaria, es preciso por lo tanto, disponer de un medio de representación que facilite su manejo y mejore la capacidad de procesamiento.

Para ello se emplean con mayor frecuencia tres tipos de representación para la información, éstos son: bit, byte y palabra, en algunos casos se utilizan la doble palabra.

2.1.2.1 Bit

El bit es la unidad elemental de información donde sólo puede tomar dos valores un "1" ó un "0", es decir, un bit es suficiente para representar una señal binaria.

2.1.2.2 Byte

El byte es una unidad compuesta por una agrupación ordenada de 8 bits, es decir, ocho dígitos binarios. Los bits se agrupan de derecha a izquierda tomando como número de bit del 0 al 7. En un byte se puede representar el estado de hasta ocho señales binarias, puede usarse para almacenar un número cuya magnitud como máximo sería:

Número máximo de un byte = $1\ 1\ 1\ 1\ 1\ 1\ 1\ 1 = 2^8 - 1 = 255$

2.1.2.3 Palabra

Para obtener mayor capacidad de procesamiento a veces se agrupan los bytes formando lo que se denomina las palabras. La palabra es una unidad mayor compuesta de 16 bits = 2 bytes. Los bits de una palabra se agrupan de derecha a izquierda tomando como número de bit del 0 al 15. En una palabra se pueden representar hasta 16 señales binarias, puede usarse para almacenar un número cuya magnitud como máximo sería

Número máximo en una Palabra = $2^{16} - 1 = 65535$

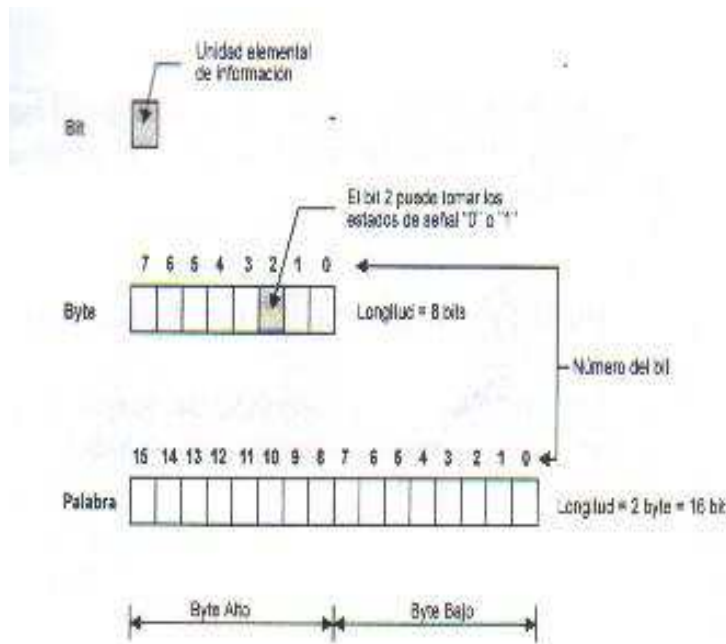


Figura 2-3 Representación del Bit, Byte y Palabra

Fuente: www.grupo-maser.com/PAG_Curso/Auto/

2.1.3 DIRECCIONAMIENTO DE BITS

Cuando se elabora un programa de control, se van indicando las diferentes instrucciones de mando donde cada instrucción indica que operación se debe ejecutar, también figura la dirección exacta del módulo y canal o terminal de conexión de las señales de E/S involucradas en el proceso.

El direccionamiento puede realizarse de dos formas

Direccionamiento Fijo

Direccionamiento Variable

2.1.3.1 Direccionamiento Fijo

Cuando la dirección de las señales de E/S queda determinada por la posición o puesto de enchufe en que están ubicados los módulos de E/S respecto al CPU, se dice que el direccionamiento es fijo. Además, un direccionamiento fijo puede ser del tipo Octal (byte) o hexadecimal

2.1.3.1.1 Direccionamiento Fijo del Tipo Octal (BYTE)

Un direccionamiento del tipo octal queda determinado cuando a cada módulo de E/S se le agrupa los terminales por bytes, es decir, en grupos de 8 bits del (0 al 7).

En este caso, en la dirección se especificará el byte correspondiente al terminal seleccionado y que pertenece al puesto de enchufe según la posición que ocupa.

2.1.3.1.2 Direccionamiento Fijo del Tipo Hexadecimal

Este direccionamiento se diferencia del anterior en el agrupamiento de los terminales, siendo para este caso del tipo hexadecimal, ósea en grupos de 16 bits del (0 al F).

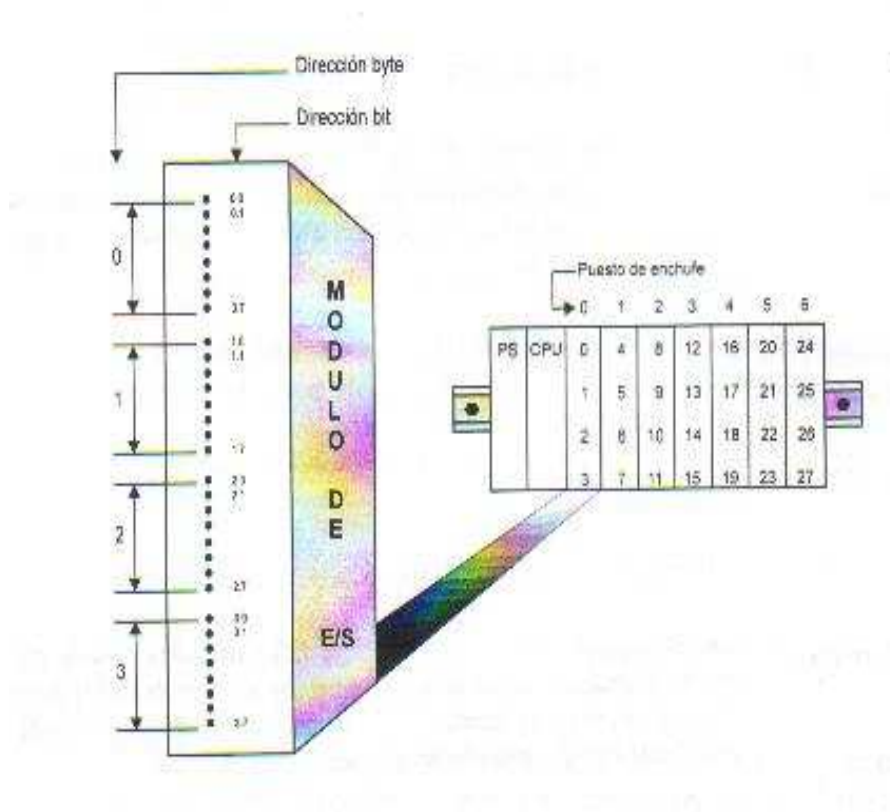


Figura 2-4 Direccionamiento fijo

Fuente: www.grupo-maser.com/PAG_Curso/Auto/

2.1.4 PROGRAMACIÓN DEL PLC:

Los programas para el PLC comprenden tres partes básicas:

Programa principal

Subrutinas (opcional)

Rutinas de interrupción (opcional)

Estos tienen varios lenguajes de programación de entre los cuales los mas importantes son: KOP (esquema de contactos) y AWL (lista de instrucciones).

El programa principal contiene las operaciones que controlan la aplicación, las cuales se ejecutan en forma secuencial en cada ciclo. El programa principal termina con una sentencia de finalización (Bobina absoluta en KOP o sentencia MEND en AWL).

Una subrutina es una secuencia de operaciones que se ejecutan cada vez que son invocadas en el programa principal. Esta o éstas se colocan al final del programa principal y terminan con una operación de retorno absoluto (RET).

Las rutinas de interrupción son secuencias de operaciones que se ejecutan cada vez que se presenta el correspondiente evento de interrupción. Esta o éstas se ubican al final del programa principal y terminan con una operación de retorno absoluto (RETI).

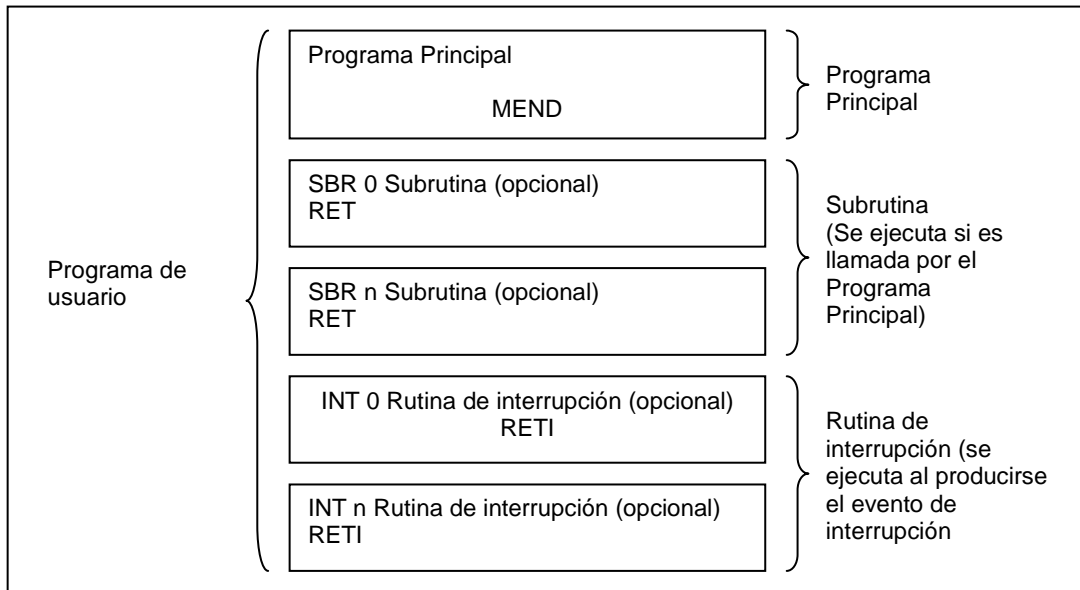


Figura 2-5 Ubicación de las partes del programa.

Fuente: Manual/EPN/Automatización/Dr. Álvaro Aguinaga

2.1.5 LENGUAJES DE PROGRAMACIÓN MÁS UTILIZADOS PARA LOS PLC'S SON DE DOS TIPOS:

AwI: Que comprende un juego de operaciones nemotécnicas que representan las funciones del CPU. Este lenguaje contiene una lista de instrucciones en el que cada línea del programa tiene una operación que utiliza una abreviatura nemotécnica para representar una función de la CPU. Las operaciones se colocan en secuencia lógica de acuerdo a los requerimientos del programa.

La programación con este tipo de lenguaje tiene las siguientes reglas:

Cada sección de programación se divide en segmentos, cuya palabra clave es NETWORK.

Los comentarios se escriben luego de dos barras inclinadas (/). Cada línea adicional de comentario debe comenzar así mismo con dos barras inclinadas. Finalice cada línea pulsando Enter .

La primera columna corresponde a la operación. La operación es una sentencia lógica.

La segunda columna que se separa de la anterior por un espacio en blanco, corresponde al operando. El operando es la dirección del dato sobre el que actúa la operación.

Kop: Es un lenguaje de programación gráfico con componentes similares a los elementos de los esquemas de circuitos, los cuales conforman un segmento de operaciones lógicas. Los diagramas que se establecen son similares a gráficos en escalera (LADDER).

Los elementos básicos de un lenguaje KOP son:

Contactos: Que representan interruptores por los que circula la corriente cuando está cerrado. Esto implica que hay dos tipos de contactos: cerrados y abiertos.

Bobinas: Que representan a relés que se excitan cuando se aplica voltaje.

Cuadros: Que representan una función que se ejecuta cuando la corriente circula por él. Un cuadro puede representar, por ejemplo, un contador, un temporizador, etc.

Segmentos: Que constituyen un circuito completo. La corriente circula desde la barra de alimentación ubicada a la izquierda pasando por los contactos cerrados para excitar las bobinas o cuadros.

Debe señalarse que un mismo programa puede ser escrito sea en lenguaje KOP o en A WL.

Una vez escrito el programa, el software permite compilar, y así revisar la sintaxis lógica, de tal manera que si existen errores éstos, el mismo tienen los mensajes respectivos que permiten corregidos.

El software instalado en el ordenador permite cambiar automáticamente el programa escrito en KOP o A WL, al elegir el comando respectivo y realizar los cambios de presentación requeridos, como por ejemplo: operaciones y direcciones escritos completamente en mayúsculas, tabuladores entre las operaciones y las direcciones, entre otros.

Otra operación importante es la creación de un bloque de datos con el objeto de predefinir e inicializar las variables utilizadas en el programa. Al efectuar esta labor se puede especificar la columna de dirección, el valor de los datos y los comentarios respectivos.

En los circuitos de control deben diferenciarse claramente el circuito de mando o de control que tiene baja potencia e intensidad de corriente y el circuito de fuerza o potencia que puede manejar alta potencia eléctrica y altos valores de intensidad de corriente o amperaje.

Para sistemas de control electromecánicos más complejos, por ejemplo el arranque de motores trifásicos en que se debe temporizadamente, pasar de una configuración a otra (triángulo/estrella), se debe utilizar una serie de otros componentes electromecánicos como son: contactores, temporizadores, contadores, pulsadores, etc. Con estos elementos se puede dar la secuencia o el “programa” de control, para esto se utiliza también un esquema de programación Ladder. Sin embargo como se podrá concluir, el uso de estos sistemas de control implica un costo muy alto y una gran complejidad mientras el sistema sea más grande. Esta realidad ha determinado que en la actualidad se defina el control electrónico como el que se impone en la automatización industrial y los elementos de control electromecánico que se utilizan como actuadores para relacionar los circuitos de alta potencia y amperaje con el circuito de control en que siempre existirá baja potencia y bajo amperaje y en que se encontrarán los PLC's.

La forma básica de programación más común en los PLC es la programación de escalera. Está especificada por cada una de las tareas de un programa como si fueran los peldaños de una escalera. En cada peldaño se

especifica, por ejemplo, la revisión de los interruptores A y B (las entradas); si ambos están cerrados, se proporciona energía a un solenoide (la salida).

La secuencia que sigue un PLC para realizar un programa se resume de la siguiente manera:

1. Explora las entradas asociadas a un peldaño del programa de escalera.
2. Solución de la operación lógica de cada una de las entradas.
3. Encendido/apagado de las salidas del peldaño.
4. Continúa con el siguiente peldaño y repite los pasos 1, 2, 3.
5. Continúa con el siguiente peldaño y repite los pasos 1,2 Y 3.
6. Continúa con el siguiente peldaño y repite los pasos 1,2 Y 3.

Y así sucesivamente, hasta finalizar el programa.

2.2 CONTROLADOR LÓGICO PROGRAMABLE

Los PLC (*Programmable Logic Controller* en sus siglas en inglés) son dispositivos electrónicos muy usados en Automatización Industrial. Su historia se remonta a finales de la década de 1960, cuando la industria buscó en las nuevas tecnologías electrónicas una solución más eficiente para reemplazar los sistemas de control basados en circuitos eléctricos con relés, interruptores y otros componentes comúnmente utilizados para el control de los sistemas de lógica combinatorial.

Hoy en día, los PLC no sólo controlan la lógica de funcionamiento de máquinas, plantas y procesos industriales, sino que también pueden realizar operaciones aritméticas y manejar señales analógicas para realizar estrategias de control, tales como controladores proporcional integral derivativo (PID). Los PLC actuales pueden comunicarse con otros controladores y computadoras en redes de área local, y son una parte fundamental de los modernos sistemas de control distribuido.

Existen varios lenguajes de programación, tradicionalmente los más utilizados son el diagrama de escalera LADDER, preferido por los electricistas, lista de instrucciones y programación por estados, aunque se han incorporado lenguajes más intuitivos que permiten implementar algoritmos complejos mediante simples diagramas de flujo más fáciles de interpretar y mantener. Un lenguaje mas reciente, preferido por los informáticos y electrónicos, es el FBD (en ingles Function Block Diagram) que emplea compuertas lógicas y bloques con distintas funciones conectados entre si.

En la programación se pueden incluir diferentes tipos de operandos, desde los más simples como lógica booleana, contadores, temporizadores, contactos, bobinas y operadores matemáticos, hasta operaciones más complejas como manejo de tablas (recetas), apuntadores, algoritmos PID y funciones de comunicación multiprotocolo que le permitira interconectarse con otros dispositivos

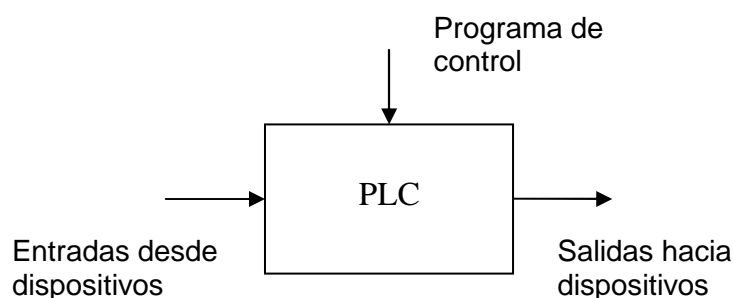


Figura 2-6 Controlador Lógico Programable

Fuente: Manual/EPN/Automatización/Dr.
Álvaro Aguinaga

2.2.3 UNIDAD CENTRAL DE PROCESO (CPU)

EL **CPU**, que es el "cerebro" del PLC, procesa la información que recibe del exterior a través de la interfaz de entrada y de acuerdo con el programa, activa una salida a través de la correspondiente interfaz de salida.

Evidentemente, las interfaces de entrada y salida se encargan de adaptar las señales internas a niveles del la CPU. Por ejemplo, cuando la CPU

ordena la activación de una salida, la interfaz adapta la señal y acciona un componente (transistor, relé, etc.)

Pero, Cómo funciona el CPU? Al comenzar el ciclo, el CPU lee el estado de las entradas. A continuación ejecuta la aplicación empleando el último estado leído. Una vez completado el programa, el CPU ejecuta tareas internas de diagnóstico y comunicación. Al final del ciclo se actualizan las salidas. El tiempo de ciclo depende del tamaño del programa, del número de E/S y de la cantidad de comunicación requerida.

El sistema modular comprende de cinco CPU para distintas exigencias, módulos de entradas y salidas analógicas y digitales, módulos de función de conteo rápido, posicionamiento de lazo abierto y lazo cerrado, así como también módulos de comunicación para el acoplamiento a redes en bus.

La CPU más potente puede tratar 1024 instrucciones binarias en menos de 0,3 ms. Pero como las instrucciones puramente binarias constituyen más bien la excepción, tenemos que mencionar los tiempos de ejecución de las instrucciones mixtas: 65% de instrucciones con bits y un 35% con palabras, el más rápido de los autómatas puede con 1K en sólo 0,8 ms.

Otro detalle es la simplicidad de diagnóstico. Los datos de diagnóstico de todo el autómata están fijamente almacenados en la CPU (hasta 100 avisos). Estos datos pueden consultarse centralizadamente en la CPU, ya que todos los módulos relevantes son accesibles vía interfaces MPI de ésta, lo que permite ahorrarse gastos suplementarios y evita molestas manipulaciones de conectores.

En una configuración de PLC en red, el puesto central de mando puede acceder directamente a cualquier CPU y a cualquier módulo de función, a cualquier panel de operador y a cualquier procesador de comunicaciones de la red, todo ello sin hardware ni software adicional.

El sistema de diagnóstico inteligente del CPU se activa al reemplazar un módulo: se encarga de verificar si la configuración del autómata es aún

compatible y evita así funcionamientos anómalos en la instalación, incluso la destrucción de módulos. Además realiza automáticamente el registro de la hora y la memorización de los fallos, contribuyendo así a un diagnóstico rápido y puntual a posteriori cuando ya no se manifieste más el defecto o cuando éste sea de naturaleza esporádica.

Si se nombraran sus características generales tendríamos:

Los cinco ofrecen hasta 2048 marcas, 128 temporizadores y 64 contadores

Según el tipo de CPU, una parte de ellos o su totalidad pueden hacerse remanente, es decir, no volátil.

La salvaguarda y gestión de datos está asegurada por una memoria especial exenta de mantenimiento y que funciona sin pila (depende del tipo de CPU)

2.2.3.1 Cpu 312 IFM

Este es capaz de procesar 1024 instrucciones binarias en 0,6 ms. Es la solución óptima para aplicaciones que requieren funciones simples como conteo y medición de frecuencias.

Para tareas sencillas no hay más que usar la función Contador con dos canales para contar atrás y adelante (el contador puede contar señales de hasta 10 KHz y tiene un ancho de banda de 32 bits). Puesto que ésta CPU lleva incorporada una memoria para el programa de usuario, (E) EPROM y dispositivos de respaldo sin pilas, no necesita mantenimiento alguno.

2.2.1.2 Cpu 313

Es similar al CPU 312 IMF con la diferencia de que tiene el doble de memoria. Además permite guardar el programa en una Memory Card, con lo cual éste autómata tampoco requiere mantenimiento.

2.2.1.3 Cpu 314

Ejecuta el programa al doble de velocidad, es decir, en 0,3 ms. Por 1K de instrucciones binarias. Tampoco hay peligro de perder datos pues también permite guardar el programa en una Memory Card tipo Flash-EEPROM.

2.2.1.4 Cpu 315

Tiene la misma rapidez que la CPU 314 (1K de instrucciones al bit en 0,3ms.), pero dos veces más de memoria (48 Kbytes), es decir, para más de 16.000 instrucciones. También contiene una memoria Flash del tipo EPROM que le permite salvaguardar los datos. Además, el reloj está asociado a un acumulador de energía dotado de una reserva de marcha de 4 semanas en caso de falla de la red.

2.2.1.5 Cpu 315-2DP

Si configuramos el S7-300 con ésta CPU, es posible extender el autómatas a 64 estaciones DP (periferia descentralizada), totalizando más de 1000 entradas/salidas a varios kilómetros de distancia y con puertos abiertos y normalizados.

Esta posibilidad que brinda el CPU 315-DP, confiere una flexibilidad total, ya que permite la libertad de direccionamiento de entradas/salidas centralizadas y descentralizadas.

2.2.3.2 Modos de Operación de la CPU

2.2.1.2.1 Modo RUN-P

La CPU ejecuta el programa y permite modificar tanto el programa como sus parámetros. Si desea utilizar las aplicaciones de STEP 7 para cambiar alguno de los parámetros del programa durante su ejecución, deberá cambiar

la CPU a modo RUN-P. Las subventanas creadas en S7-PLCSIM se pueden utilizar para modificar los datos del programa.

2.2.1.2.2 Modo RUN

La CPU lee las entradas, ejecuta el programa y actualiza luego las salidas. Por defecto, no es posible cargar programas en la CPU ni utilizar aplicaciones de STEP 7 para cambiar parámetros (como por ejemplo los valores de las entradas) cuando la CPU esté en modo RUN. No obstante, si ha configurado elementos CeR (configuración en RUN) en el proyecto de STEP 7, podrá cargarlos en modo RUN. Si la CPU simulada está en modo RUN, podrá utilizar las subventanas creadas en S7-PLCSIM para modificar los datos utilizados por el programa.

2.2.1.2.3 Modo STOP

La CPU no ejecuta el programa. A diferencia del modo STOP de una CPU real, las salidas no se ajustan a valores estándar ("seguros"), sino que conservan el estado en el que se encontraban cuando la CPU cambió a modo STOP. Los programas se pueden cargar en la CPU cuando ésta está en modo STOP. Al cambiar de STOP a RUN se inicia la ejecución del programa a partir de la primera operación. Los modos de operación de la CPU, los indicadores de la CPU y el botón de borrado total se visualizan en la subventana CPU. El modo de operación de la CPU se puede ajustar con el comando de menú Posición del selector de modo. La ejecución del programa en el PLC simulado se puede detener cuando la CPU se encuentre en modo RUN o RUN-P.

2.2.1.3 Indicadores de la CPU

La subventana "CPU" dispone de una serie de indicadores que corresponden a los diodos luminosos de una CPU real:}

SF (error de sistema) indica que la CPU ha encontrado un error de sistema que ha causado un cambio del modo de operación.

DP (periferia descentralizada o distribuida) indica el estado de la comunicación con unidades de periferia descentralizadas (distribuidas).

DC (alimentación) indica si la alimentación de la CPU está conectada o desconectada.

RUN indica que la CPU está en modo RUN.

STOP indica que la CPU está en modo STOP.

2.2.4 MEMORIA

La memoria es el almacén donde el autómata guarda todo cuanto necesita para ejecutar la tarea de control.

Datos del proceso:

Señales de planta, entradas y salidas

Variables internas, de bit y de palabra

Datos alfanuméricos y constantes

Datos de control:

Instrucciones de usuario (programa)

Configuración del autómata (modo de funcionamiento, número de e/s conectadas).

2.2.2.1 Existen varios tipos de memorias:

RAM. Memoria de lectura y escritura

ROM. Memoria de solo lectura, no reprogramable.

EPROM. Memoria de solo lectura, reprogramables con borrado por

ultravioletas

EEPROM. Memoria de solo lectura, alterables por medios eléctricos

La memoria RAM se utiliza principalmente como memoria interna, y únicamente como memoria de programa en el caso de que pueda asegurarse el mantenimiento de los datos con una batería exterior.

La memoria ROM se utiliza para almacenar el programa monitor del sistema como hemos visto en el apartado dedicado a la CPU.

Las memorias EPROM se utilizan para almacenar el programa de usuario, una vez que ha sido convenientemente depurada.

Las memorias EEPROM se emplean principalmente para almacenar programas, aunque en la actualidad es cada vez más frecuente el uso de combinaciones RAM + EEPROM, utilizando estas últimas como memorias de seguridad que salvan el contenido de las RAM. Una vez reanudada la alimentación, el contenido de la EEPROM se vuelca sobre la RAM. Las soluciones de este tipo están sustituyendo a las clásicas RAM + batería; puesto que presentan muchos menos problemas.

2.2.2.2 Memoria interna

En un autómata programable, la memoria interna es aquella que almacena el estado de las variables que maneja el autómata: entradas, salidas, contadores, relés internos, señales de estado, etc. Esta memoria interna se encuentra dividida en varias áreas, cada una de ellas con un cometido y características distintas.

La clasificación de la memoria interna no se realiza atendiendo a sus características de lectura y escritura, sino por el tipo de variables que almacena y el número de bits que ocupa la variable. Así, la memoria interna del autómata queda clasificada en las siguientes áreas.

2.2.2.3 Área de imágenes de entradas/salidas y Área interna (IR).

En esta área de memoria se encuentran:

Los canales (registros) asociados a los terminales externos (entradas y salidas).

Los relés (bit) internos (no correspondidos con el terminal externo), gestionados como relés de E/S.

Los relés E/S no usados pueden usarse como IR.

No retienen estado frente a la falta de alimentación o cambio de modo de Operación.

2.2.2.3.1 Área especial (SR).

Son relés de señalización de funciones particulares como:

Servicio (siempre ON, OFF)

Diagnos (señalización o anomalías)

Temporizaciones (relojes a varias frecuencias)

Cálculo

Comunicaciones

No conservan su estado en caso de fallo de alimentación o cambio de modo

2.2.2.3.2 Área auxiliar (AR).

Contienen bits de control e información de recursos de PLC como: Puerto RS232C, puertos periféricos, casetes de memoria.

Se dividen en dos bloques:

Señalización: Errores de configuración

Datos del sistema

Memorización y gestión de datos:

Es un área de retención.

Accesible en forma de bit o de canal.

No conservan su estado en caso de fallo de alimentación o cambio de modo.

2.2.2.3.3 Área de enlace (LR).

Se utilizan para el intercambio de datos entre dos PLC's unidos en forma:

PC Link(1:1).

Dedicados al intercambio de información entre PLC's.

Si no se utilizan como LR pueden usarse como IR

Accesible en forma de bit o canal

No conservan su estado en caso de fallo de alimentación o cambio de modo.

Mantienen su estado ante fallos de alimentación o cambio de modo de PLC

Son gestionados como los IR y direccionables como bit o como canal.

2.2.2.3.4 Área de temporizadores y contadores (TIM/CNT).

Es el área de memoria que simula el funcionamiento de estos dispositivos

Son usados por el PLC para programar retardos y contajes.

2.2.2.3.5 Área de datos (DM).

Esta área suele contener los parámetros de configuración del PLC (setup).

Se trata de memoria de 16 bits (palabra).

Utilizable para gestión de valores numéricos.

Mantiene su estado ante cambios de modos de trabajo o fallo de alimentación.

2.2.2.4 Memoria de Programa

La memoria de programa, normalmente externa y enchufable a la CPU mediante casete de memoria, almacena el programa escrito por el usuario para su aplicación.

Cada instrucción del usuario ocupa un paso o dirección del programa. Las memorias de programa o memorias de usuario son siempre de tipo permanente RAM + batería o EPROM / EEPROM. Por lo general la mayoría de los fabricantes de autómatas ofrecen la posibilidad de utilizar memorias RAM con batería para la fase de desarrollo y depuración de los programas, y de pasar estos a memorias no volátiles EPROM o EEPROM una vez finalizada esta fase.

La ejecución del programa en el módulo es siempre prioritaria, de forma que si se da tensión al autómata con un módulo conectado, la CPU ejecuta su programa y no el contenido en memoria RAM interna.

2.2.3. TERMINAL DE PROGRAMACIÓN

El terminal o consola de programación es el que permite comunicar al operario con el sistema.

Las funciones básicas de éste son las siguientes:

Transferencia y modificación de programas.

Verificación de la programación.

Información del funcionamiento de los procesos.

Como consolas de programación pueden ser utilizadas las construidas específicamente para el autómatas, tipo calculadora o bien un ordenador personal, PC, que soporte un software especialmente diseñado para resolver los problemas de programación y control



Figura 2-7 Terminal de programación portátil

Fuente: www.grupo-maser.com/PAG_Curso/Auto/

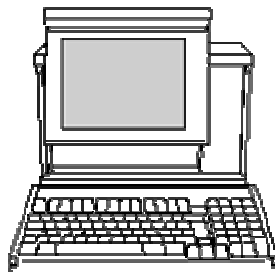


Figura 2-8 Terminal de programación compatible PC

Fuente: www.grupo-maser.com/PAG_Curso/Auto/

2.2.3.1 Periféricos

Los periféricos no intervienen directamente en el funcionamiento del autómatas, pero sin embargo facilitan la labor del operario.

Los más utilizados son:

Grabadoras a cassettes.

Impresoras.

Cartuchos de memoria EEPROM.

Visualizadores y paneles de operación OP

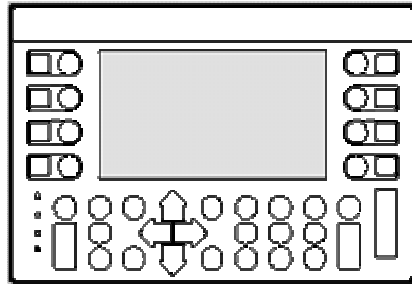


Figura 2-9 Panel de Operación

Fuente: www.grupo-maser.com/PAG_Curso/Auto/

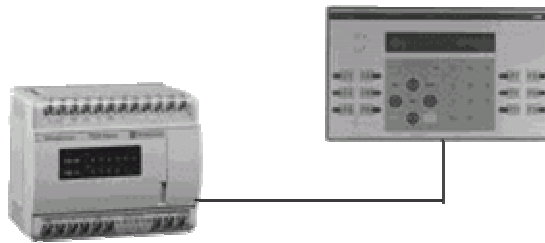


Figura 2-10 Conexión de un visualizador a un autómata

Fuente: www.grupo-maser.com/PAG_Curso/Auto/

2.2.4 SISTEMAS DE ENTRADAS Y SALIDAS

2.2.4.1 Módulo de entradas

A este módulo se unen eléctricamente los captadores (interruptores, finales de carrera, pulsadores).

La información recibida en él, es enviada a la CPU para ser procesada de acuerdo la programación residente.

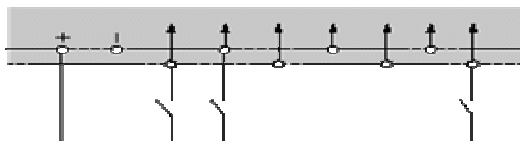


Figura 2-11 Captadores pasivos

www.grupo-maser.com/PAG_Curso/Auto/

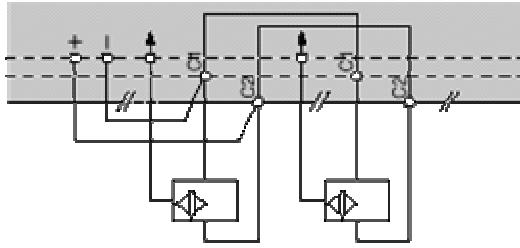


Figura 2-12 Captadores activos

www.grupo-maser.com/PAG_Curso/Auto/

Se pueden diferenciar dos tipos de captadores conectables al módulo de entradas: los Pasivos y los Activos.

Los Captadores Pasivos son aquellos que cambian su estado lógico, activado - no activado, por medio de una acción mecánica. Estos son los Interruptores, pulsadores, finales de carrera, etc.

Los Captadores Activos son dispositivos electrónicos que necesitan ser alimentados por una tensión para que varíen su estado lógico. Este es el caso de los diferentes tipos de detectores (Inductivos, Capacitivos, Fotoeléctricos). Muchos de estos aparatos pueden ser alimentados por la propia fuente de alimentación del autómatas.

El que conoce circuitos de automatismos industriales realizados por contactores, sabrá que puede utilizar, como captadores, contactos eléctricamente abiertos o eléctricamente cerrados dependiendo de su función en el circuito.

2.2.4.2 Módulo de salidas

El modulo de salidas del autómatas es el encargado de activar y desactivar los actuadores (bobinas de contactores, lámparas, motores pequeños, etc). La información enviada por las entradas a la CPU, una vez procesada, se envía al módulo de salidas para que estas sean activadas y a la vez los actuadores que en ellas están conectados.

Según el tipo de proceso a controlar por el autómata, podemos utilizar diferentes módulos de salidas .

Existen tres tipos bien diferenciados:

A relés.

A triac.

A transistores.

2.2.4.2.1 Módulos de salidas a relés:

Son usados en circuitos de corriente continua y alterna. Están basados en la conmutación mecánica, por la bobina del relé, de un contacto eléctrico normalmente abierto.

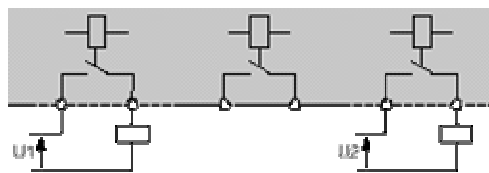


Figura 2-13 Salidas a relés

www.grupo-maser.com/PAG_Curso/Auto/

2.2.4.2.2 Módulos de salidas a triacs:

Se utilizan en circuitos de corriente continua y corriente alterna que necesiten maniobras de conmutación muy rápidas

2.2.4.2.3 Módulos de salidas a Transistores a colector abierto

El uso del este tipo de módulos es exclusivo de los circuitos de c.c. Igualmente que en los de Triacs, es utilizado en circuitos que necesiten maniobras de conexión/desconexión muy rápidas

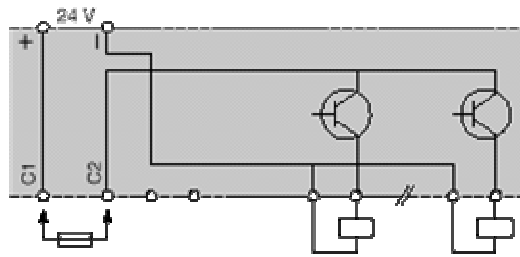


Figura 2-14 salidas a transistores
www.grupo-maser.com/PAG_Curso/Auto/

2.2.5 FUENTE DE ALIMENTACION

Para cada sistema de automatización están disponibles distintas fuentes de alimentación.

2. 2.5.1 Fuentes de Alimentación del S7-300

Para la alimentación eléctrica de la CPU se puede utilizar cualquier red eléctrica de 24V (industrial).



Figura 2-15 Fuente de voltaje 24 V DC - Modular, 24 V DC, de 5 a 40 Amperios
Fuente: www.siemens.com.co/SiemensDotNetClient_Andina/

La gama de módulos del sistema S7 incluye las siguientes fuentes de alimentación, las cuales han sido construidas especialmente para el S7-300:

Denominación	Intensidad de salida	Tensión de salida	Tensión de entrada
PS 307	2A	DC 24V	AC 120V / 230V
PS 307	5A	DC 24V	AC 120V / 230V
PS 307	10A	DC 24V	AC 120V / 230V

Tabla 2-1 de variación de intensidad de salida
www.grupo-maser.com/PAG_Curso/Auto/

CAPITULO III

3. CONEXIONADO O INSTALACIÓN DEL PLC

Introducción

En este capítulo se demuestra la manera de cablear eléctricamente las diferentes partes del sistema.

3.1 CABLEADO DE LA FUENTE DE ALIMENTACIÓN Y LA CPU



Figura 3-1 Cableado de la fuente de alimentación y la CPU

Fuente: SIEMENS-getting-started-teild.pdf.

El cableado se realiza de la forma siguiente.


Nº	Gráfico	Descripción
1		Abrir las tapas frontales de la fuente de alimentación y la CPU.
2		Soltar la abrazadera de protección contra los tirones en el cable de la fuente de alimentación.
3		Pele el cable de red, dado el caso, enganchar las punteras (en caso de cable multifilar) y conecte en la fuente de alimentación.
4		Apretar las abrazaderas anti tirones para el cable.
5		Insertar dentro de la fuente de alimentación y la CPU dos cables y atornillar.
6		<p>Controlar si el selector de la tensión de red esta en la posición correcta para el caso de aplicación.</p> <p>La fuente de alimentación viene ajustada desde la fábrica con una tensión de red de 230 V AC para cambiar se procede de la siguiente forma: levantar la tapa de protección usando un destornillador ajustar el selector a la tensión de la red presente y vuelva a insertar la tapa de protección.</p>

Tabla 3 -1 Cableado de la fuente de alimentación y la CPU

Fuente: SIEMENS-getting-started-teild.pdf.

La fuente de alimentación proporciona las tensiones necesarias para el funcionamiento de los distintos circuitos del sistema.

La alimentación a la CPU puede ser de continua a 24 Vcc, tensión muy frecuente en cuadros de distribución, o en alterna a 110/220 Vca. En cualquier caso es la propia CPU la que alimenta las interfaces conectadas a través del bus interno.

La alimentación a los circuitos E/S puede realizarse, según tipos, en alterna a 48/110/220 Vca o en continua a 12/24/48 Vcc.

La fuente de alimentación del autómata puede incorporar una batería tampón, que se utiliza para el mantenimiento de algunas posiciones internas y del programa usuario en memoria RAM, cuando falla la alimentación o se apaga el autómata.

3.2 CONEXIÓN Y CABLEADO DE LOS MÓDULOS

Tipos de módulos disponibles:

Tanto si son analógicas o digitales como si son entradas o salidas, éste autómata trata las señales a medida que se van presentando.

Módulos de entradas digitales

Los módulos de entradas digitales convierten las señales digitales externas del proceso al nivel interno del autómata.

Por ejemplo, si se va a utilizar detectores de proximidad o finales de carreras con una tensión de 24 VDC, se debe elegir el módulo de entrada de 24 V., que le ofrece 16/32 entradas y conecta los sensores con separación galvánica y en grupos de 8 entradas con contacto común.

Para señales de corriente alterna de 120 ó 230 V., existe un módulo de 8

canales que se encarga de traducir las señales para que las pueda leer el autómeta.

Módulos de salidas digitales

Los módulos de salidas digitales convierten las señales internas del S7-300 en señales externas adaptadas al proceso.

Por ejemplo, si desea conectar electroválvulas, contactores, pequeños motores, lámparas, etc., entonces necesitará un módulo de éste tipo. En lo que respecta a los actuadores de 24 VDC, como por ejemplo contactores y válvulas, el autómeta ofrece varias alternativas como ser: desde módulos de 16/32 canales y 0,5 A. Con separación galvánica hasta módulos de relé de 8 a 16 canales

Módulos de entradas analógicas

Este convierte las señales analógicas en señales digitales en donde el autómeta procesa internamente. Se puede conectar sensores y emisores de señal de tipo tensión o intensidad, resistencia, así como termopares y termoresistencias y se puede elegir entre módulos que van de los 2 a 8 canales.

Módulos de salidas analógicas

Este módulo convierte las señales digitales del S7-300 en señales analógicas para el proceso. Es una herramienta indispensable para convertidores de frecuencias, regulaciones, etc. Además dispone de 2 ó 4 canales y tiene una resolución de 4 bits, con posibilidad de configuración para señales tipo tensión o corriente.

Módulos económicos

Este módulo es especial cuando el factor económico es fundamental. Tiene una resolución de 8 bits, convierte señales analógicas en digitales

y viceversa, y está dotado de 4 entradas y 2 salidas.

Módulos de función para tareas especiales

Son módulos de conteo rápido que superan el ámbito de los 100 kHz y son idóneos para medir frecuencias, procesar los valores medidos, medir revoluciones o longitudes, así como para realizar tareas de posicionamiento.

Se ofrecen diversos módulos de posicionamiento para controlar tareas de posicionamiento, motor paso a paso, así como para simular controladores de levas y accionamiento de 2 marchas (lenta/rápida).

Módulo de simulación

Este módulo se utiliza para comprobar el programa de aplicación antes de poner el sistema en marcha, o durante su funcionamiento. Este módulo permite simular señales de sensores mediante interruptores y averiguar los estados de señal de las salidas por medio de indicadores LED. Se coloca en lugar de un módulo de E/S digitales.

Módulo de suministro de energía

Este módulo es la fuente de alimentación del autómatas que transforma la tensión externa de suministro en la tensión operativa interna. Las tensiones de alimentación posibles para el S7-300 son: 24 VCC, 115 VCA o 230 VCA.

Módulos de interconexión o interface

Estos módulos permiten la comunicación entre los distintos racks. Se encuentran IM360, IM361, IM 365.

3.2.1 CABLEADO DEL MÓDULO ANALÓGICO

El cableado de un transductor de valor analógico depende de su tipo y no del módulo SM331.

El cableado del módulo analógico comprende las tareas siguientes:

- Conectar la fuente de alimentación (cable rojo)
- Conectar el transductor de intensidad a 2 hilos (cables verdes)
- Conectar resistencias a los canales no utilizados
- Conectar el primer transductor de intensidad a 4 hilos (cables verdes)
- Conectar el segundo transductor de intensidad a 4 hilos (cables verdes)
- Cablear la masa y cortocircuitar los canales no utilizados (cable azul)

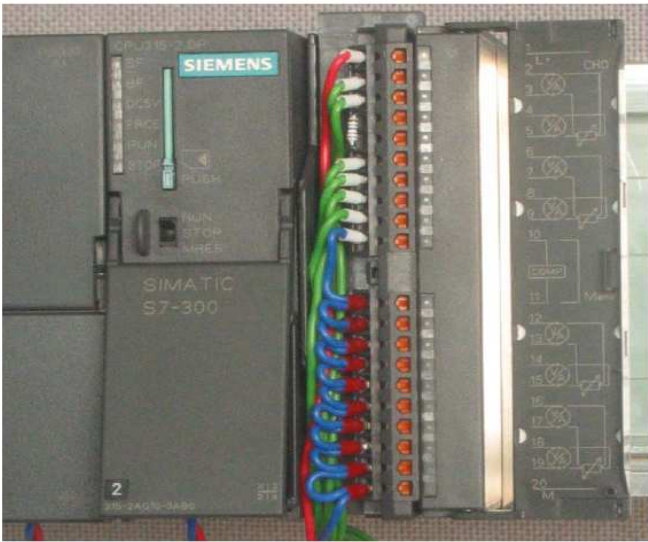


Figura 3-2 Cableado del conector frontal del SM331

Fuente: SIEMENS-getting-started-teild.pdf.

Operaciones pasó a paso para el cableado:

Gráfico	Cableado	Comentario
	Abrir la puerta frontral del SM331	La puerta frontal incluye la serigrafía de los bornes.







	Pelar a 6mm de los extremos de los cables que desee enchufar en el conector frontal y enganche en ellas las punteras adecuadas.	
	Cablear el conector frontal como sigue Borne 1: L+	Alimentación de módulo
	Borne 2: M+ sensor 1 Borne 3: M- sensor 1	Cableado estándar para transductores a dos hilos
	Conectar los bornes 4 y 5 con una resistencia de 1.5 y 3.3 ohm	Para mantener la capacidad de diagnostico del grupo de canales es necesario dotar con una resistencia a una entrada no utilizada.
	Borne 6: M+ sensor 2 Borne 7: M- sensor 2	Cableado estándar de un transductor de 4 hilos
	Borne 8: M+ sensor 3 Borne 9: M- sensor 3	
	Borne 10 (comp) y Borne 11 (M _{ana}) conexión a M Cortocircuitar los bornes 12 a 19 y conectarlos con M _{ana} Borne 20: M	Para medir la intensidad no se utiliza com. Especificado para transductores de dos hilos Los grupos de canales no utilizados deberán cortocircuitarse con M _{ana}

Tabla 3-2 Cableado del conector frontal del SM331

Fuente: SIEMENS-getting-started-teild.pdf.

3.2.2 ENCENDER LA FUENTE DE ALIMENTACIÓN

Para comprobar el cableado, entonces encienda la fuente de alimentación.

No olvidar poner la CPU en STOP (ver círculo rojo).



Figura 3-3 Cableado terminado con éxito, CPU en posición STOP

Fuente: SIEMENS-getting-started-teild.pdf.

Si luce un LED rojo existe un error en el cableado. Comprobar en éste caso el cableado

3.2.2.1.1 Principio de cableado de transductores de intensidad

Dependiendo del transductor de intensidad que se utilice es necesario adaptar el cableado de la fuente de alimentación. Se distingue entre el cableado de un transductor a 2 hilos y el de un transductor a 4 hilos.

3.2.2.1.2 Principio de cableado de un transductor de intensidad a 2 hilos

Este tipo de transductor es alimentado por el propio módulo de entrada analógica.

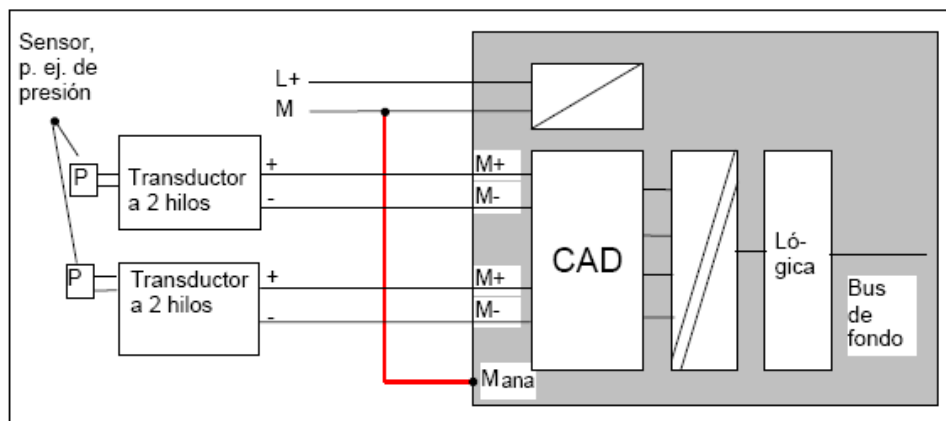


Figura 3-4 Cableado: transductor de intensidad a 2 hilos

Fuente : SIEMENS-getting-started-teild.pdf.

3.2.2.1.2 Principio de cableado de un transductor de intensidad a 4 hilos

A diferencia de la variante a 2 hilos este tipo de transductor dispone de alimentación propia.

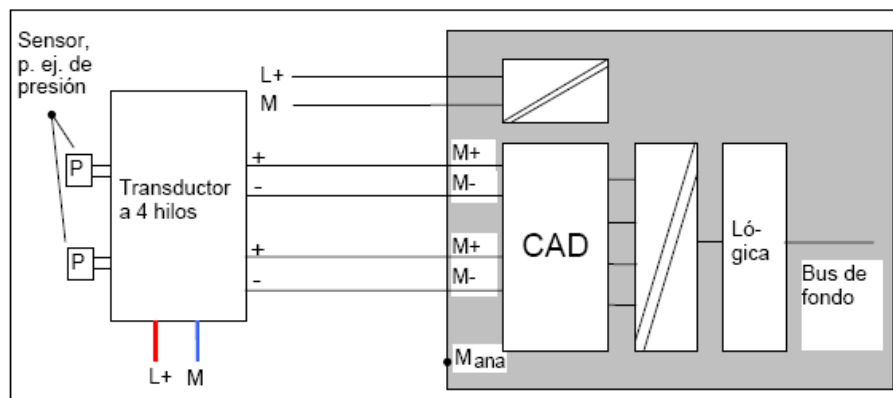


Figura 3-5 Cableado de transductor de intensidad a 4 hilos

Fuente: SIEMENS-getting-started-teild.pdf.

3.2.3 MONTAJE DEL SISTEMA



Antes de poder aplicar el módulo de entrada analógica SM331 se requiere una configuración base utilizando componentes SIMATIC S7-300.

Secuencia de montaje, siempre de izquierda a derecha:

Fuente de alimentación PS307

CPU 314

SM331

Gráfico	Descripción
	<p>Atornillar el perfil soporte a la base (tornillo: M6) de forma que quede con un mínimo de 40mm de espacio por encima y debajo del perfil.</p> <p>Si la base es una placa metálica puesta a tierra o una chapa porta equipos puesta a tierra, atienda la conexión entre el perfil soporte y la base que ofrezca baja resistencia.</p> <p>Conectar el perfil soporte al conductor de protección con el fin de que el perfil soporte tenga un tornillo M 6 reservado al efecto.</p>
	<p>Montaje de la fuente de alimentación:</p> <ul style="list-style-type: none">• Colgar la fuente en la parte superior del perfil soporte.• Abatirla y atornillarle en la parte inferior del perfil.



	<p>Conectar el conector de bus (incluido en el suministro SM331) en el conector izquierdo posterior de la CPU.</p>
	<p>Montaje de la CPU:</p> <ul style="list-style-type: none"> • Ubicar la CPU arriba en el perfil soporte • Correrle hacia la izquierda hacia la fuente • Abatirla hacia abajo • Atornillar abajo el perfil soporte

Tabla 3-3 Montaje Mecánico

Fuente: SIEMENS-getting-started-teild.pdf.

3.2.4 MONTAJE DEL MÓDULO ANALÓGICO

Antes de montar el SM331 se enchufa en él el conector frontal adecuado y se ajusta el tipo de medida deseado de las entradas.

3.2.4.1 Componentes del SM331

Un módulo analógico operativo consta de los componentes:

Módulo SM331 (en nuestro ejemplo 6ES7331-7KF02-0AB0)

Conector frontal de 20 polos. El conector frontal está disponible en 2 versiones:

Con bornes de resorte (referencia 6ES7392-1BJ00-0AA0)

Con bornes de tornillo (referencia 6ES7392-1AJ00-0AA0)



Figura 3-6 Componentes del SM331

Fuente: SIEMENS-getting-started-teild.pdf.

Componentes
Módulo
Tiras rotulables
Conectores de bus
2 bridas de cable (no en figura) para fijar los cables externos

Tabla 3-4 Alcance de suministro del módulo SM3

Fuente: SIEMENS-getting-started-teild.pdf.

3.2.5 PROPIEDADES DEL MÓDULO ANALÓGICO

El módulo analógico, tiene 8 entradas en 4 grupos de canales (cada grupo tiene dos entradas del mismo tipo). La resolución de medida es ajustable para cada grupo de canales

Margen de medida ajustable para cada grupo de canales:

Tensión

Intensidad

Resistencia

Temperatura

Alarma de diagnóstico parametrizable

Dos canales con alarmas de límite (parametrizable sólo en canal 0 y canal 2)

Aislamiento galvánico respecto del interface al bus de fondo

Aislamiento galvánico respecto a la tensión de carga (excepción: como mínimo un adaptador de margen de medida esté enchufado en la posición D)

El módulo tiene carácter universal y ha sido proyectado para los casos de aplicación más habituales. El tipo de medida deseado se ajusta directamente en el módulo utilizando los adaptadores de margen de medida al efecto (ver apartado 4.2.3).

3.2.6 ADAPTADORES DE MARGEN DE MEDIDA

El módulo SM331 tiene 4 adaptadores de margen de medida (uno por cada grupo de canales). Un adaptador de margen de medida puede enchufarse en 4 posiciones diferentes (A, B, C o D). La posición define qué transductor se conecta al grupo de canales respectivo.

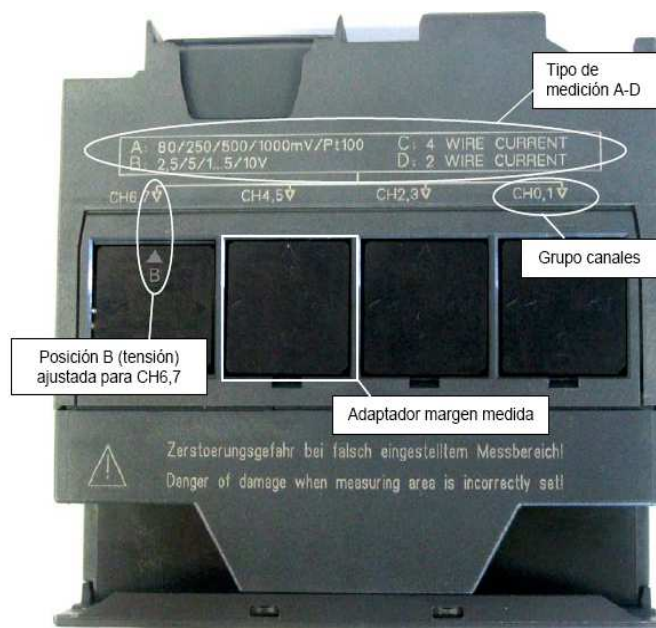


Figura 3-7 Cuatro adaptadores de margen de medida ajustados de fábrica a B (tensión)

Fuente: SIEMENS-getting-started-teild.pdf.

Posición	Tipo de medida
A	Termopar / medida de resistencia
B	Tensión (ajuste de fábrica)
C	Intensidad/Corriente (transductor a 2 hilos)
D	Intensidad/Corriente (transductor a 4 hilos)

Tabla 3-5 Posiciones de los adaptadores de margen de medida

Fuente: SIEMENS-getting-started-teild.pdf.

En éste ejemplo al grupo de canales 1 en la entrada 0 se conecta un sensor con un transductor a 2 hilos 4 a 20mA.

En el grupo de canales 2, en las entradas 2 y 3, se conecta un transductor a 4 hilos. Por ello, el primer adaptador de margen de medida debe tener la posición D y el segundo la posición C.

Gráfico	Descripción
	Usar un destornillador, extraer dos adaptadores de margen de medida
	Gire los adaptadores de margen de medida a la posición deseada.
	Vuelva a enchufa en el módulo los adaptadores de margen de medida

Tabla 3-6 Posicionamiento de los adaptadores de margen de medida

Fuente: SIEMENS-getting-started-teild.pdf.

3.2.7 MONTAJE DEL MÓDULO SM331

Una vez preparado adecuadamente el módulo analógico, éste se monta también en el perfil de soporte.



Gráfico	Descripción
	Montaje del SM331: <ul style="list-style-type: none">• Colgar el SM331 arriba en el perfil soporte,• desplazarlo hacia la izquierda hasta la CPU,• abatirlo hacia abajo• y atornillarlo abajo al perfil soporte
	Montar el conector frontal: <ul style="list-style-type: none">• Pulse el botón superior del conector.• Enchufe el conector en el módulo hasta que el botón del mismo encaje en la posición superior.

Tabla 3-7 Montaje del módulo SM331

Fuente: SIEMENS-getting-started-teild.pdf.

Con ello queda ya montado mecánicamente el sistema

3.2.8 Montaje e Interconexión de los Módulos

El diseño simple permite que el S7-300 sea flexible y fácil de utilizar.

Rieles de montaje DIN: Los módulos son enganchados de la parte superior del riel, ajustándola hasta el tope y luego atornillando arriba y abajo. En cuanto se refiere a la interconexión de módulos, éstos llevan incorporados el bus posterior (de fondo de panel), lo que significa que no hay mas que enchufar los conectores de bus suministrados en la parte posterior de la carcasa y así, todos los módulos quedan correctamente

interconectados.

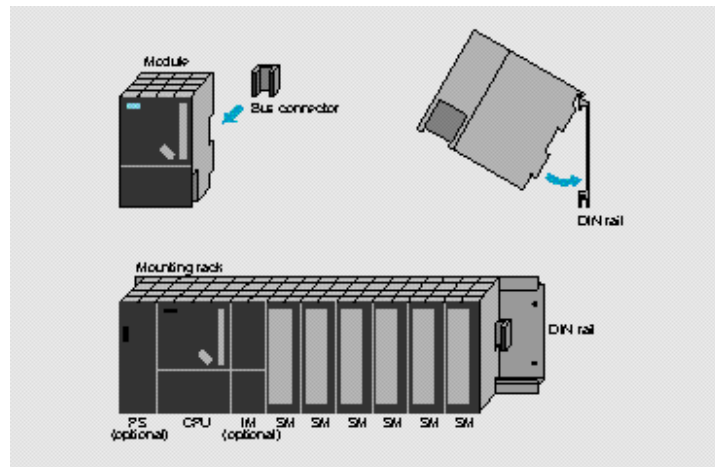


Figura 3-8: Conexión de módulos

Fuente: SIEMENS-getting-started-teild.pdf.

Además, al querer montar una CPU o cambiar solamente un módulo oprimiendo un pulsador, se suelta el conector frontal, quedando a la vista el esquema de conexiones del módulo; por su parte, los conectores frontales están codificados por lo que resulta imposible enchufar accidentalmente en un módulo equivocado (además, el plano de conexiones está situado en la parte interior de la tapa frontal, por lo que siempre estará disponible) y, en posición de montaje, se interrumpe la conexión eléctrica.

Otra ventaja que tiene el S7-300 es el sistema de precableado (llamado SITOP) que se compone solamente de elementos pasivos, tales como conectores frontales, cables planos en vaina redonda, bloques de bornes y además el cableado ya viene preparado. Este sistema permite establecer conexiones a 1, 2 ó 3 hilos con toda facilidad y evitar errores en el cableado.

Es especialmente útil cuando los módulos E/S y los sensores y actuadores conectados se encuentran a una distancia de 30 mts., como máximo.

Prestaciones: Si hablamos de las prestaciones, diremos que la CPU permite montar 256 entradas/salidas digitales en un sólo perfil y, si bien es cierto que en la fila central sólo caben 8 módulos de E/S además de la CPU, pueden emplearse otras cuatro filas de éste tipo: por ejemplo, la CPU 314 permite incorporar hasta 32 módulos, repartidos en cuatro filas. Para enlazar las distintas filas basta usar los módulos de interconexión, también llamados interfaces (IM). Estos se encargan por sí solos de comunicar las demás filas, incluso salvando las distancias de hasta 10 mts.

Los módulos de interconexión son dos: IM360 e IM361. El IM360 se sitúa en la fila central y por cada fila adicional se coloca un IM361, respectivamente. Si solo necesita una fila adicional, la pareja de módulos IM365 es la más económica (el primero de ellos se coloca en la fila central y el segundo, en la fila adicional).

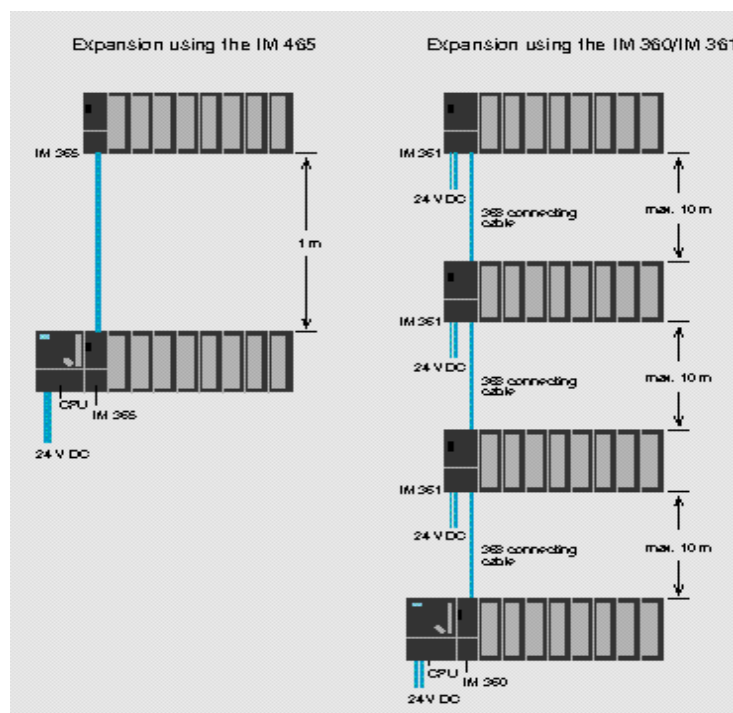


Figura 3-9: Conexión de módulos

Fuente: SIEMENS-getting-started-teild.pdf.

Pueden ser instalados 32 módulos en 4 racks: un total de 3 racks de expansión pueden ser conectados al rack central. Ocho módulos pueden ser conectados en cada rack.

Módulos de conexión vía interfaces: cada rack tiene su propio módulo de interfaces. Este es siempre conectado en la ranura adyacente al CPU.

Instalación separada: los racks individuales pueden ser instalados también en forma separada. La distancia máxima entre racks es de 10 metros

Distribución versátil: los racks pueden ser instalados horizontalmente o verticalmente, de manera de obtener la distribución óptima en el espacio del que se dispone.

3.3 LIMITES DE CORRIENTE

Cada transistor de salida (NPN) puede drenar una corriente pico de 1.0A. Los transistores están agrupados en drivers de cuatro transistores:

Driver	1:	Salidas	1,2,3,4
Driver	2:	Salidas	5,6,7,8
Driver	3:	Salidas	9,10,11,12
Driver 4: Salidas 13,14,15,16			

Sin la utilización de disipadores, cada driver es capaz de drenar una corriente continua de 1.0A. Para una operación adecuada, asegúrese que la suma de la corriente drenada por los cuatro drivers no exceda 1.0A en total. También se recomienda que cada pin de salida no drene mas de 500mA continuamente.

Se recomienda un sistema de ventilación para aplicaciones que estén cerca de estos límites, incluyendo la posibilidad de utilizar un ventilador.

3.3.1 RELES INTERNOS

El SCMT20H soporta hasta 128 reles internos. Los reles internos son utilizados para almacenar estados lógicos temporarios o pueden servir como flags de estado para que una computadora remota consulte el estado de este flag a traves de un link serie. Estos reles corresponden a los primeros 128 items de la tabla de reles Ladder.

3.3.2 TIMERS

Los 20 timers ocupan las 20 primeras entradas de la tabla de timers Ladder. El valor preseteable del timer es de 0.1 a 999.9 segundos.

Pueden crearse timers adicionales utilizando los relojes internos e ingresando a contadores no utilizados. El SCMT20H soporta 6 fuentes de clock de varios periodos: 0.01s, 0.02s, 0.1s, 0.2s, 1 seg y 1 minuto. El reloj de 0.01s puede ser utilizado para crear timers de alta velocidad con 0.01s de resolución.

3.3.3 CONTADORES

Hay 20 contadores independientes en el SCMT20H. Estos contadores utilizan las 20 entradas a la tabla de contadores de Ladder. Pueden realizar cuentas desde 1 a 9999.

Estos contadores pueden operar como contadores descendentes con un valor representable o como contadores ascendentes / descendentes. Cualquiera o todos los 8 primeros contadores pueden ser utilizados como secuenciadores o maquinas de estado.

3.3.4 SECUENCIADOR

El SCMT20H soporta 8 secuenciadores de 32 pasos cada uno. Los secuenciadores son un excelente recurso para la programación de máquinas o

procesos secuenciales. Cualquiera de todos los 8 primeros pueden ser utilizados como pasos, que corresponden a los secuenciadores "Seq1" a "Seq8.

Cada paso del secuenciador (hasta 31) puede ser utilizado como contacto en el programa Ladder "SeqN:XX

Donde:

N = secuenciador # 1 a 8. XX = Step # 0 - 31.

3.4 CONEXIÓN A UNA PC

La programación del PLC es realizada enteramente desde una PC, y el programa se descarga al controlador a través de un port serie de la PC. Se requiere de un adaptador para interconectar el PLC al port. Y el adaptador debe ser conectado a los pines.

El conector macho puede ser conectado al COM1: o COM2: de la PC. Si su computadora posee un conector puede instalar un adaptador.

CAPITULO IV

4. DIMENCIONAMIENTO Y CONSTRUCCIÓN DEL MÓDULO DIDACTICO

4.1 DIMENSINES DEL TABLERO

Las dimensiones del tablero de automatización esta dado según las medias de los elementos que van instalados.

4.2.1 ÁREA DE LA MESA - TABLERO

Determinar las longitudes del contorno de la estructura metálica para establecer el área de la madera y el tool que se coloca sobre esta estructura.



Figura 4-1 Medición de área (mesa-tablero)

Recortar la lámina metálica con las medidas exactas según el área que se evaluó en la medición anterior. (68*110cm)



Figura 4-2 Área del tablero

Realizar el procedimiento anterior con la madera. (120*56cm)



Figura 4-3 Área de la mesa

4.1.2 LONGITUDES PARA LOS TOMA CORRIENTES Y CAJÓN



Figura 4-4 Medición de las longitudes la ubicación de tomacorriente (11*7cm) y el cajón



Figura 4-4 Medición de área del espacio para tomacorriente y cajón

4.2 ELEMENTOS PARA LA CONSTRUCCIÓN

Los elementos utilizados para la construcción del tablero de automatización son:

<p>PLC(s7-300)</p> <p>CPU 314</p> <p>Fuente</p> <p>Módulos de entradas digitales</p> <p>Módulos de salidas digitales</p> <p>Módulos de entradas analógicas</p> <p>Módulos de salidas analógicas</p> <p>Rieldin</p>	A photograph of a Siemens PLC rack installed in a control cabinet. The rack contains several modules, including a CPU 314, a power supply, and various input and output modules. The rack is labeled 'Siemens' and 'S7-300'. The modules are connected to a terminal block at the bottom of the rack.
--	---

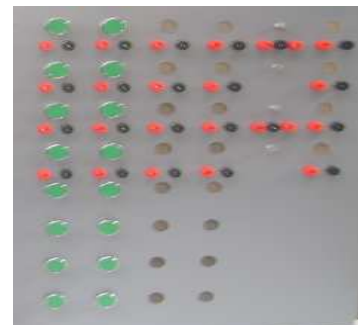
3 Tubos cuadrados para la estructura o esqueleto del tablero.



Una plancha metálica para la colocación de todos los elementos en si del tablero.



16 pulsadores
48 jack



4 Breik de 10 A para proteger los circuitos de cualquier controversia y evitar que se dañe el PLC.
2 Rieldin para ubicar los breik, PLC y borneras.



2 rollos de alambre flexible 18 AWG para realizar todas las conexiones.
Una funda de punteras



2 canaletas para mandar los cables y quede el circuito de una forma estética.



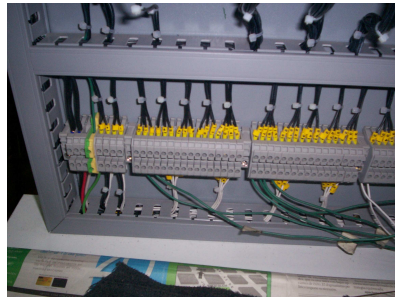
2 tomacorrientes



1 bornera tierra para conectar la tierra.

40 borneras para las conexiones de los cables

Las marcas para realizar las enumeraciones respectivas de cada entrada y salida del PLC



Una plancha de madera



Base de pintura



Pintura



4.2.1 HERRAMIENTAS

Taladro es la máquina que se utiliza para el taladrado de los agujeros para la ubicación de los jack y los potenciómetros.



Soplete es una pistola de pulverización para colocar la base y la pintura sobre el tablero.



Sacabocados es el útil para realizar los agujeros de 22 mm para colocar los pulsadores y los focos.



Destornillador es la herramienta básica que se utilizo para atornilla las conexiones del cableado y ubicación de canaletas sobre la plancha metálica.



<p>Peladora de cables es herramienta utilizada para retirar el recubrimiento (aislante) de los conductores.</p>	
<p>Soldadora es herramienta que nos permite unir los materiales base, para la construcción del esqueleto del tablero.</p>	
<p>Machuelo es útil para realizar roscas en los agujeros.</p>	

CAPITULO V

5. ELABORACION DE GUIAS BASICAS DE PRÁCTICAS

5.1 INTRODUCCIÓN

Cuanto más complejidad tiene un proceso automatizado más importante es lograr una comunicación "hombre-máquina" adecuada al mismo.

La productividad de la producción depende fuertemente de la flexibilidad de los sistemas de control en ella utilizados. Sin embargo, a medida que se recurre a soluciones descentralizadas con sus grandes ventajas de flexibilidad, se incrementan también las necesidades de intercambio de datos entre los autómatas y con el computador central.

5.2 FUNCIONES BÁSICAS DE UN PLC

5.2.1 DETECCIÓN:

Lectura de la señal de los captadores distribuidos por el sistema de fabricación.

5.2.2 MANDO:

Elaborar y enviar las acciones al sistema mediante los accionadores y preaccionadores.

5.2.3 DIALOGO HOMBRE-MAQUINA

Mantener un diálogo con los operarios de producción, obedeciendo sus consignas e informando del estado del proceso.

5.2.4 PROGRAMACION:

Para introducir, elaborar y cambiar el programa de aplicación del autómata. El dialogo de programación debe permitir modificar el programa incluso con el autómata controlando la máquina.

5.3 NUEVAS FUNCIONES

5.3.1 REDES DE COMUNICACIÓN:

Permiten establecer comunicación con otras partes de control. Las redes industriales permiten la comunicación y el intercambio de datos entre autómatas a tiempo real. En unos cuantos milisegundos pueden enviarse telegramas e intercambiar tablas de memoria compartida.

5.3.2 SISTEMAS DE SUPERVISIÓN:

También los autómatas permiten comunicarse con ordenadores provistos de programas de supervisión industrial. Esta comunicación se realiza por una red industrial o por medio de una simple conexión por el puerto serie del ordenador.

5.3.3 CONTROL DE PROCESOS CONTINUOS:

Además de dedicarse al control de sistemas de eventos discretos, los autómatas llevan incorporados funciones que permiten el control de procesos continuos, disponen de módulos de entrada y salida analógicas con la posibilidad de ejecutar reguladores PID que están programados en el autómata.

5.3.4 ENTRADAS- SALIDAS DISTRIBUIDAS:

Los módulos de entrada y salida, no tienen porqué estar en el armario del autómat. Pueden estar distribuidos por la instalación ya que se comunican con la unidad central del autómat mediante un cable de red.

5.3.5 BUSES DE CAMPO:

Mediante un solo cable de comunicación se pueden conectar al bus captadores y accionadores, reemplazando al cableado tradicional. El autómat consulta cíclicamente el estado de los captadores y actualiza el estado de los accionadores.

5.4 CREAR UN NUEVO PROYECTO STEP 7

Para configurar la nueva CPU 315-2DP utilizar el Administrador SIMATIC con STEP7 V5.2 ó superior. Iniciar el Administrador SIMATIC haciendo clic en el icono “Administrador SIMATIC” en el escritorio de Windows y crear un nuevo proyecto utilizando el asistente Nuevo proyecto.

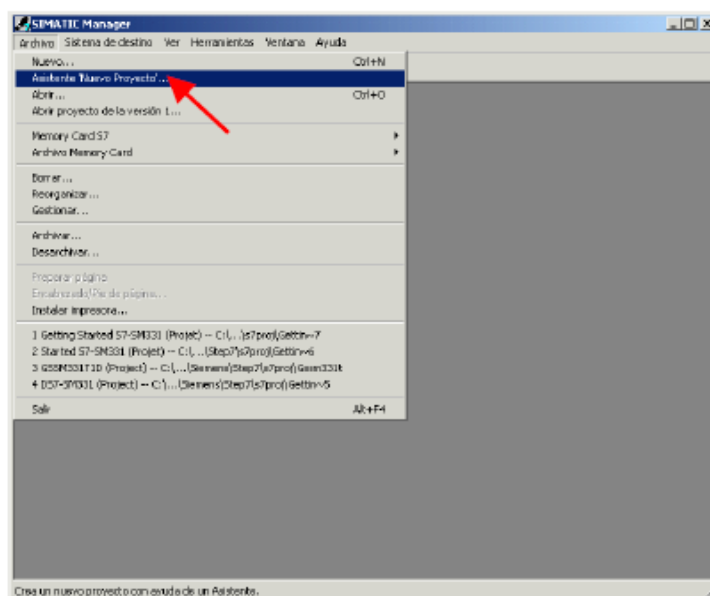


Figura 5-1 Abrir asistente .Nuevo proyecto.

Fuente: SIEMENS-getting-started-teild.pdf.

Se presenta la pantalla de bienvenida del asistente de proyecto. El asistente va guiando por toda la rutina de creación de un nuevo proyecto.

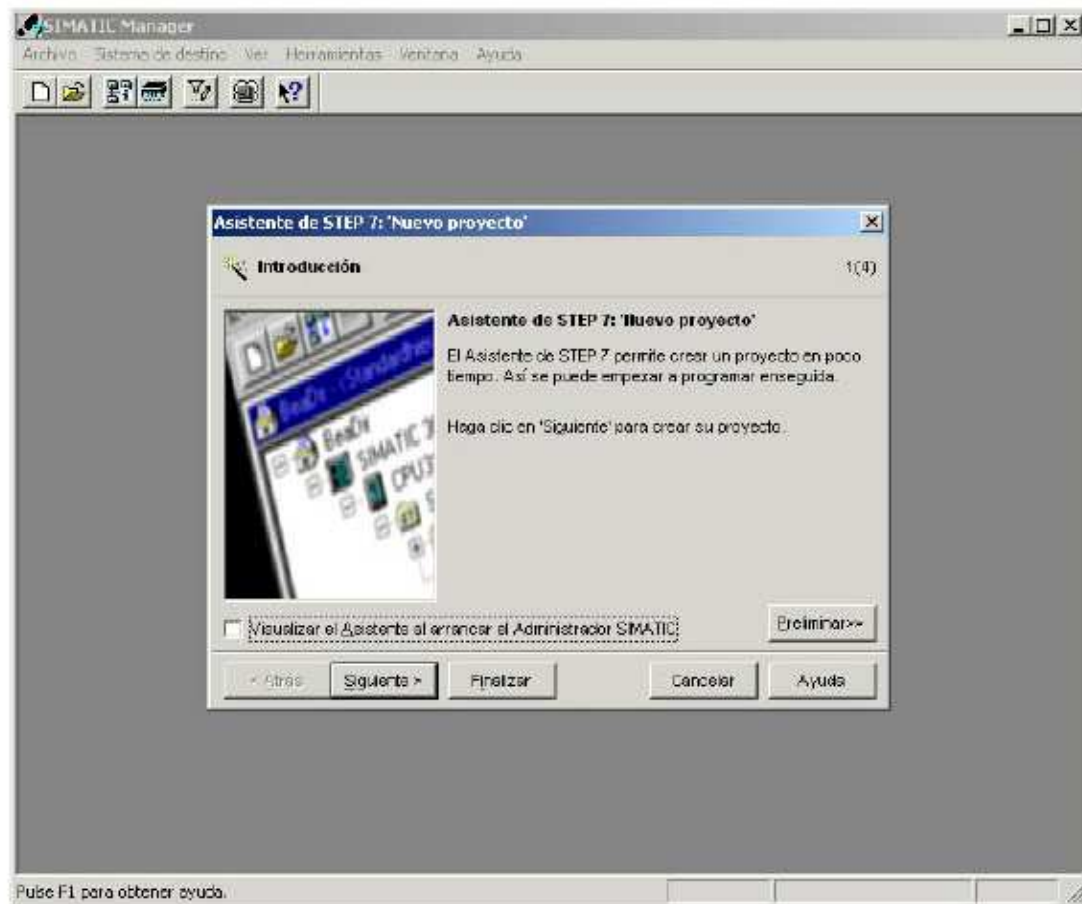


Figura 5-2 Inicio del asistente .Nuevo proyecto.

Fuente: SIEMENS-getting-started-teild.pdf.

Durante la creación se requiere introducir los datos siguientes:

- Tipo de CPU
- Definir un programa de usuario base
- Bloques de organización
- Nombre del proyecto

Hacer clic en .Siguiente

5.1.1 SELECCIÓN DE LA CPU

Para el ejemplo seleccionar la CPU 315-2DP. (Este ejemplo puede usarse también para otras CPU).

Seleccionar entonces su CPU.

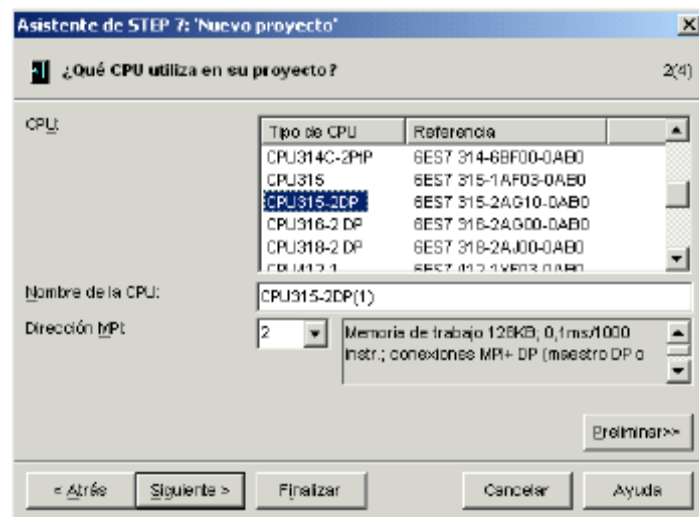


Figura 5-3 Asistente .Nuevo proyecto., selección de CPU

Fuente: SIEMENS-getting-started-teild.pdf.

Hacer clic en Siguiente

5.1.2 DEFINIR PROGRAMA DE USUARIO BASE

Elegir el lenguaje AWL y seleccionar los bloques de organización (OB) siguientes:

OB1 Bloque cíclico

OB0 Alarma de proceso

OB2 Alarma de diagnóstico

El OB1 es necesario en cada proyecto y se llama de forma cíclica.

El OB0 se llama cuando aparece una alarma de proceso.

El OB2 se llama cuando aparece una alarma de diagnóstico.

Si utiliza módulos diagnosticables y no inserta el OB2, entonces la CPU pasa a STOP cuando aparece una alarma de diagnóstico.

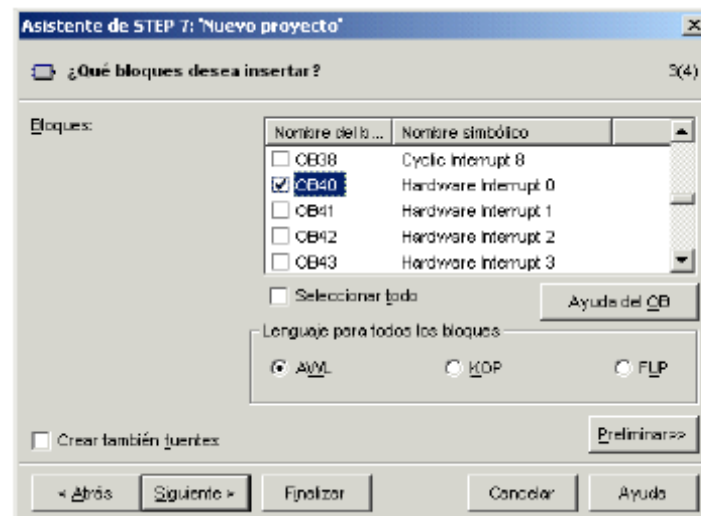


Figura 5-4 Asistente .Nuevo proyecto., insertar bloques de organización

Fuente: SIEMENS-getting-started-teild.pdf.

Hacer clic en .Siguiendo

5.1.3 ASIGNACIÓN DE UN NOMBRE DE PROYECTO

Seleccionar el campo. Nombre del proyecto y sobre escribir, el nombre existente con Getting Started S7-SM331.

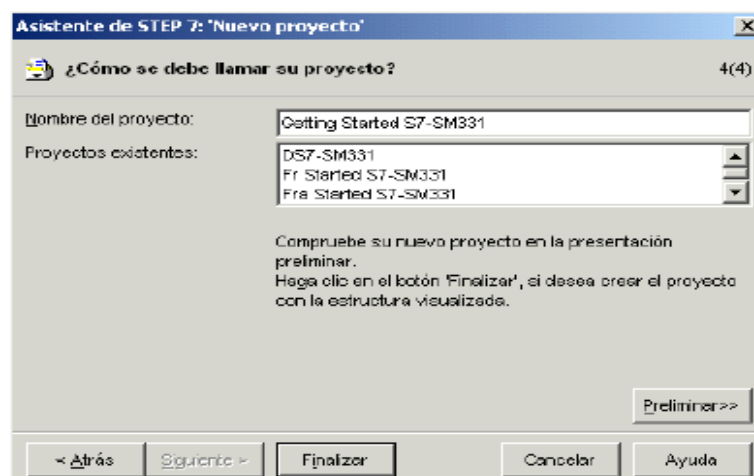


Figura 5-5 Asistente .Nuevo proyecto., denominación del proyecto

Fuente: SIEMENS-getting-started-teild.pdf.

Pulsar en .Finalizar., con ello se crea automáticamente la base del proyecto

5.1.4 PROYECTO S7 RESULTANTE TERMINADO

El asistente ha creado el proyecto .Getting Started S7-SM331.. En la ventana derecha puede ver los bloques de organización insertados.

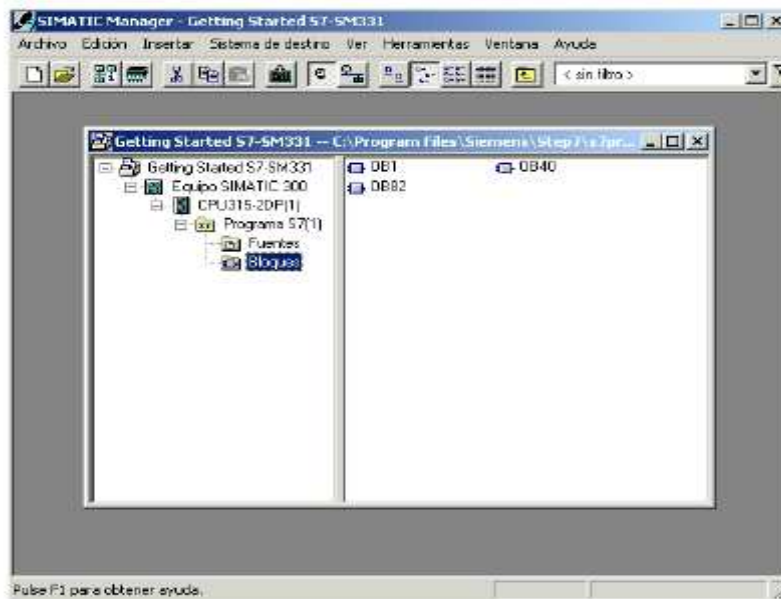


Figura 5-6 Asistente .Nuevo proyecto., resultado

Fuente: SIEMENS-getting-started-teild.pdf.

5.2 CONFIGURACIÓN DEL HARDWARE

El asistente STEP7 crea un proyecto S7 base. Para generar los datos de sistema para la CPU se requiere todavía una configuración completa del hardware.

5.2.1 CREAR CONFIGURACIÓN DEL HARDWARE

La configuración del hardware del sistema del ejemplo se crea con ayuda del Administrador SIMATIC.

Para ello, hacer clic en la ventana izquierda sobre la carpeta .Equipo

SIMATIC 300 e iniciar en la derecha la configuración del hardware haciendo doble clic en la carpeta .Hardware.

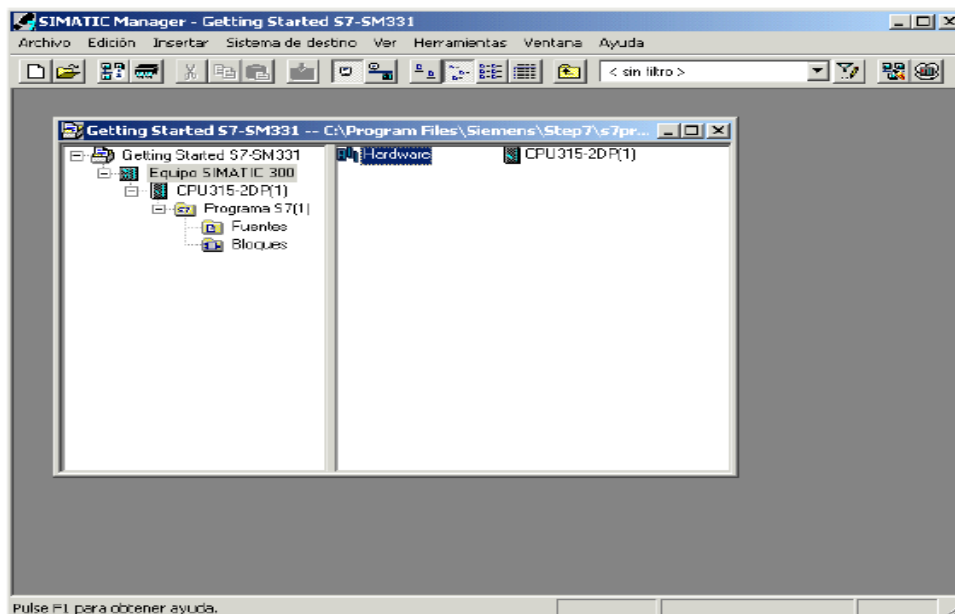


Figura 5-7 Pasos para abrir la configuración del hardware

Fuente: SIEMENS-getting-started-teild.pdf.

5.2.2 AÑADIR COMPONENTES SIMATIC

En primer lugar elegir la fuente de alimentación del catálogo del hardware. Si no está visible el catálogo del hardware, abrir pulsando las teclas Ctrl+K o haciendo clic en el icono del catálogo (flecha azul).

En el catálogo del hardware se puede navegar a través de la carpeta (Equipo SIMATIC 300 hasta la carpeta PS-300).

Hacer clic en PS307 5A y mover hasta el slot 1 (ver flecha roja).

Mover el módulo al primer campo libre del slot 4 de su bastidor (ver flecha roja).

Con esto se añade todos los módulos a la configuración del hardware. El siguiente paso es agregar parámetros en el módulo.

5.2.3 PARAMETROS DEL MÓDULO ANALÓGICO

El Administrador SIMATIC inserta al módulo analógico, con los ajustes predeterminados o por defecto. Modificar los parámetros, para configurar los tipos de sensores y las funciones de diagnóstico y alarma.

5.2.3.1 Funcionalidades del Sistema basados en el ejemplo

La tabla siguiente muestra los parámetros ajustables para el ejemplo:

Funcionalidades	Descripción
Reacciones del proceso	<ul style="list-style-type: none">• Diagnóstico: activo• Alarma de proceso al rebasar límite: activo
Sensor 1	<ul style="list-style-type: none">• Transductor de intensidad a 2 hilos• Diagnóstico colectivo (agrupado)• Vigilancia de rotura de hilo• Límites 6 mA y 18 mA
Sensores 2 & 3	<ul style="list-style-type: none">• Transductor de intensidad a 4 hilos• Diagnóstico colectivo (agrupado)• Vigilancia de rotura de hilo• Límites 6 mA y 18 mA

Tabla 5-1 SM331, funcionalidades del sistema del ejemplo.

www.grupo-maser.com/PAG_Curso/Auto/

5.2.3.2 Recibir Datos de Parámetros

Hacer doble clic en el slot 4 con el SM331.

Seleccionar la ficha **Entradas**.

Analice lo siguiente:

- Alarma de diagnóstico CON
- Alarma de proceso CON

Entrada 0-1:

Tipo de medida: 2DMU
Diagnóstico colectivo CON
Rotura de hilo CON

Entrada 2-3:

Tipo de medida: 2DMU
Diagnóstico colectivo CON
Rotura de hilo CON

Entradas 4-5 y 6-7

Tipo de medición: desactivado (---)

Frecuencia perturbadora

Ajuste la frecuencia de su red (50 Hz ó 60 Hz)

Disparador de alarma de proceso

Límite superior 18 mA
Límite inferior 6 ma

Propiedades - AI8x12Bit - (80/54)

General | Direcciones | Entradas

Habilitar

☒ Alarma de diagnóstico ☒ Alarma de proceso al rebasar límites

Entrada	0 - 1	2 - 3	4 - 5	6 - 7
Diagnóstico				
Diagnóstico colectivo:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Con comprobación de rotura de hilo:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Medición				
Tipo:	TM2H	TM4H	---	---
Margen:	4..20 mA	4..20 mA	---	---
Posición del adaptador de margen:	[D]	[C]		
Frecuencia perturbadora	50 Hz	50 Hz	---	---
Causa de la alarma de proceso	Canal 0	Canal 2		
Límite superior:	18.000 mA	18.000 mA		
Límite inferior:	6.000 mA	6.000 mA		

Aceptar Cancelar Ayuda

Figura 5-10 SM331: parametrización
Fuente: SIEMENS-getting-started-teild.pdf.

5.2.3.3 Explicación de los Diferentes Ajustes

Tipo de medición:

2DMU y 4DMU representan transductores de intensidad a 2 y 4 hilos respectivamente; significa que los canales están desactivados. Al desactivar los canales el resto se procesa de forma más rápida.

Adaptadores de margen de medida

Se muestra la posición necesaria de los adaptadores de margen de medida (apt. 4.2.3).

Frecuencia perturbadora (supresión de frecuencias perturbadoras)

La frecuencia de la red alterna de alimentación puede tener efecto negativo sobre los valores medidos, particularmente en caso márgenes de tensión pequeños y si se usan termopares. Con este parámetro se define la frecuencia de red existente en su sistema.

Este parámetro tiene también efecto sobre la resolución, el tiempo de integración y el tiempo base de ejecución del grupo de canales.

Resolución (precisión)

El valor analógico se memoriza en una palabra de 16 bits.

Tiempo de integración

El módulo requiere un cierto tiempo para medir la señal analógica. Este tiempo se denomina tiempo de integración. Cuanto mayor sea la precisión requerida más tardará el módulo en medir la señal.

Tiempo de ejecución base

Además del tiempo de integración, el módulo requiere un determinado tiempo para su digitalización.

Resolución	Frecuencia perturbadora	Tiempo de integración	Tiempo base de ejecución
9 bits	400 Hz	2,5 ms	24 ms
12 bits	60 Hz	16,6 ms	136 ms
12 bits	50 Hz	20 ms	176 ms
14 bits	10 Hz	100 ms	816 ms

Tabla 5-2 Relación entre resolución, frecuencia perturbadora y tiempo de integración

www.grupo-maser.com/PAG_Curso/Auto/

Alarma de proceso

Sólo los canales 0 y 2 están en condiciones de activar alarma de proceso.

Las alarmas de proceso pueden utilizarse para activar una alarma cuando la señal analógica supera por exceso o por defecto determinados límites ajustad

Finalización de la configuración del hardware

Cerrar la ventana que incluye los parámetros.

Compilar y guardar el proyecto vía Equipo – acceso directo (Ctrl+S).

Con ello queda terminada la configuración del hardware para el proyecto.

5.2.4 PRUEBA DE CONEXIÓN

Para fines de comprobación efectuar un test de conexión y cargar los datos del sistema.

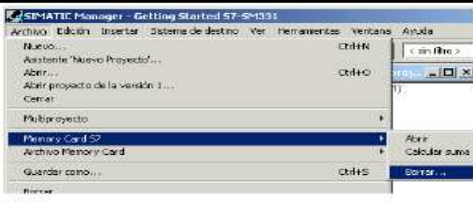




Orden	Gráfico	Descripción
1		Utilizando una Power PG o un PC con prommer externo borre su Micro Memory Card: Haga clic en el Administrador SIMATIC "Archivo -> Memory Card S7 -> Borrar". Con ello se borra la MMC.
2		Desconecte la alimentación de la CPU. Inserte la MMC en la CPU. Conecte la alimentación.
3		Si la CPU está en posición RUN, entonces pásela a la posición STOP.
4		Vuelva a encender la fuente de alimentación. Si parpadea el LED de STOP, la CPU demanda borrado total. En tal caso lleve el selector brevemente a la posición MRES.
5		Utilizando un cable MPI conecte la CPU a su unidad de programación. Para ello, enchufe el cable MPI en el puerto MPI de la CPU y el otro extremo en el puerto correspondiente de su unidad de programación.

Tabla 5-3 Conexión

- **Cargar en la CPU la configuración del hardware**

Utilizando HW Config cargar ahora en la CPU la configuración del hardware.

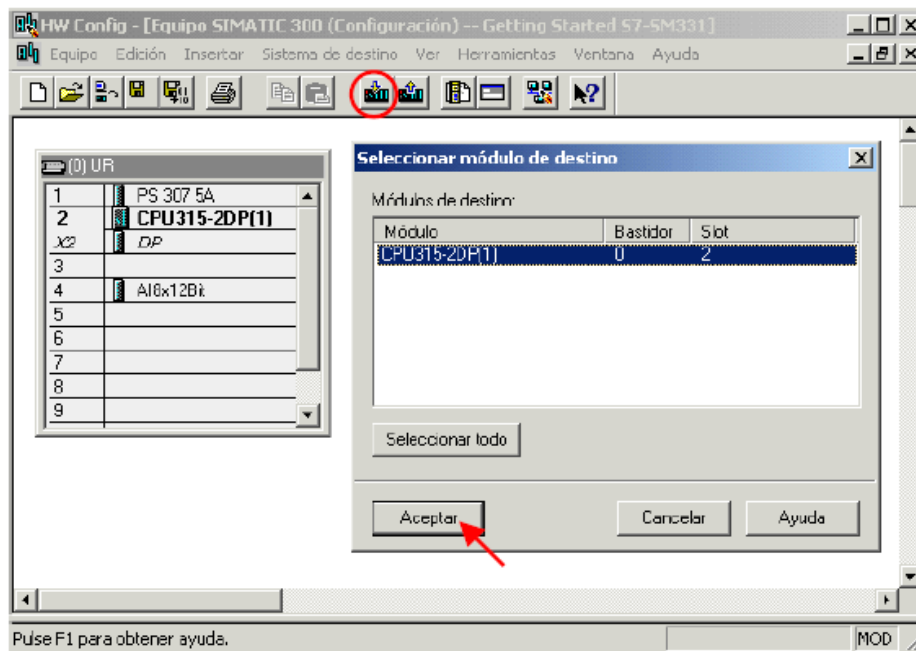


Figura 5-11 Cargar en la CPU la configuración del hardware (1)

Fuente: SIEMENS-getting-started-teild.pdf.

Haga clic en el icono .Cargar en módulo. (ver círculo rojo).

Si aparece la ventana de diálogo. Seleccionar módulo de destino, entonces hacer clic en (Aceptar).

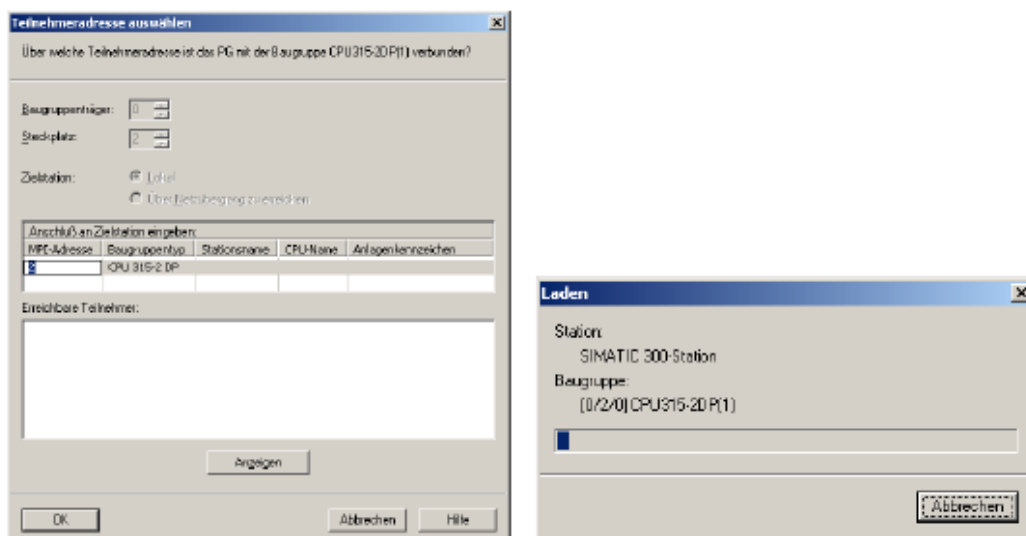


Figura 5-12 Cargar en la CPU la configuración del hardware (2)

Fuente: SIEMENS-getting-started-teild.pdf.

Al aparecer la ventana de diálogo, seleccionar la dirección de estación.

Hacer clic en (Aceptar).

Con ello se bajan en la CPU los datos del sistema.

- **Arrancar la CPU**

Pasar la CPU a RUN, si ha ejecutado correctamente la configuración del hardware, entonces en la CPU deben lucir dos LEDs verdes (RUN y DC5V).



Figura 5-13 CPU funcionando sin error

Fuente: SIEMENS-getting-started-teild.pdf.

5.3 PROGRAMA DE USUARIO STEP 7

5.3.1 TAREAS DEL PROGRAMA DE USUARIO

En el ejemplo se deben grabar en un bloque de datos los valores procedentes del sensor y en una palabra de marcas, la información de estado sobre las alarmas de proceso. Las informaciones de estado deberán poderse confirmar (acusar) con ayuda de un bit. Además también se memorizarán en otro bloque de datos los valores de los canales (valores de las palabras de entrada).

En el programa de usuario deben ejecutarse las tareas siguientes:

1. Memorización cíclica en un bloque de datos (DB1) de los valores de las entradas analógicas.
2. Convertir cíclicamente los valores de las entradas analógicas en valores en coma flotante (FC1) y guardarlos en un bloque de datos (DB2).
3. En caso de estado lógico TRUE en la marca de confirmación booleana (M200.0), acusar las informaciones de estado de las alarmas de proceso.
4. Si aparece una alarma de proceso, guardar el estado en una palabra de marcas (MW100).

Tipo de llamada	Bloque de organización competente	Tarea a programar	Bloques y marcas usados
Llamada cíclica	OB1	Guardar entradas analógicas	DB1
		Convertir y memorizar los valores de los sensores	FC1, DB2
		Acusar alarma de proceso	M200.0
Llamada controlada por alarma de proceso	OB40	Memorizar estado	MW100
Llamada controlada por alarma de diagnóstico	OB82	Sólo existe porque se utilizan módulos diagnosticables	---

Tabla 5-4 Estructura del programa de usuario

www.grupo-maser.com/PAG_Curso/Auto/

El OB2 se utiliza para módulos diagnosticables. Si está habilitada la alarma de diagnóstico en un módulo diagnosticable entonces; cuando detecta un error, el OB2 presenta a la CPU una demanda de alarma de diagnóstico (tanto cuando aparece como cuando desaparece la incidencia).

En el ejemplo el OB2 sólo se utiliza para que la CPU no pase a STOP.

En el OB2 pueden programarse reacciones a las alarmas de diagnóstico.

5.3.2 CREAR PROGRAMA DE USUARIO

Existen dos formas de crear un programa de usuario:

- Si tiene conocimientos en el lenguaje AWL de STEP7, entonces podrá crear y programar los bloques y funciones necesarios en la carpeta de bloques.
- También tiene la posibilidad de insertar en su proyecto un programa de usuario procedente de una fuente AWL. En estos Primeros pasos, describimos esta forma de proceder.

Para crear el programa de usuario usando STEP7 son necesarios tres pasos:

1. Descargar del archivo fuente, directamente de la página HTML
2. Importar el archivo fuente
3. Compilar la fuente

5.3.3 DESCARGA DEL ARCHIVO FUENTE

El archivo fuente puede descargarse directamente de la página HTML desde la que ha descargado también estos Primeros pasos. El archivo fuente de la versión alemana se denomina GSSM331T1DE.AWL.

Guarde la fuente en su disco duro.

5.3.4 IMPORTAR ARCHIVO FUENTE

El archivo fuente puede importarse con el Administrador SIMATIC de la forma siguiente.

Con la tecla derecha del ratón hacer clic en la carpeta .Fuente.

Seleccionar .Insertar nuevo objeto. -> Fuente externa

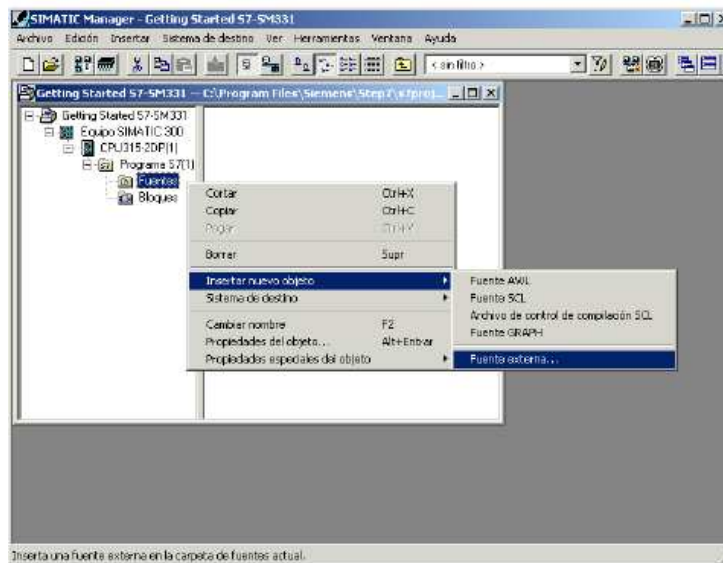


Figura 5-14 Importar fuente externa

Fuente: SIEMENS-getting-started-teild.pdf.

En el diálogo .Insertar fuente externa. Navegar hasta el archivo fuente GSSM331T1DE.AWL que ya ha descargado y guardado en su disco duro. Seleccione el archivo fuente GSSM331T1DE.AWL (flecha roja).



Figura 5-15 Importar fuente externa

Fuente: SIEMENS-getting-started-teild.pdf.

Pulsar en .Abrir.

El Administrador SIMATIC ha cargado la fuente. En la ventana derecha puede ver la fuente insertada.

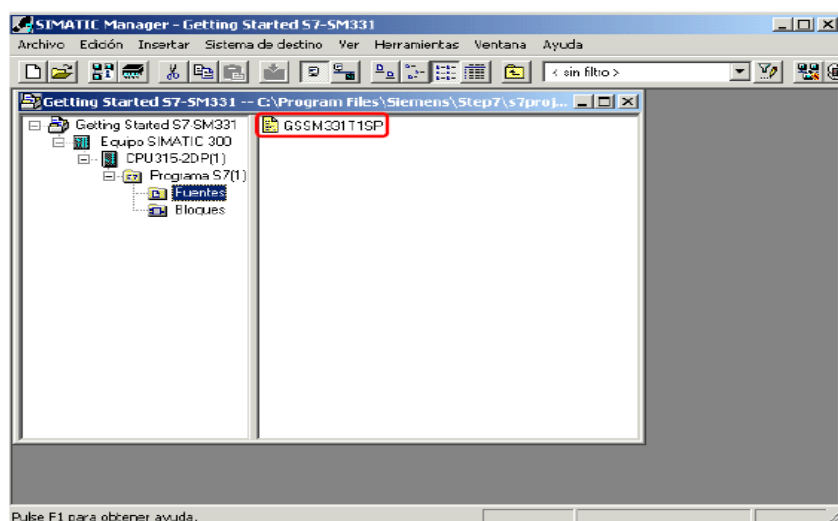


Figura 5-16 Lugar de almacenamiento del archivo fuente

Fuente: SIEMENS-getting-started-teild.pdf.

5.3.5 COMPILAR CÓDIGO FUENTE

Para crear un programa STEP7 ejecutable es necesario compilar la fuente AWL.

Haga doble clic sobre la carpeta fuente en GSSM331T1DE. Con ello se abre el editor de código fuente.

En la ventana del editor del código fuente puede verse éste (código de cap. 10).

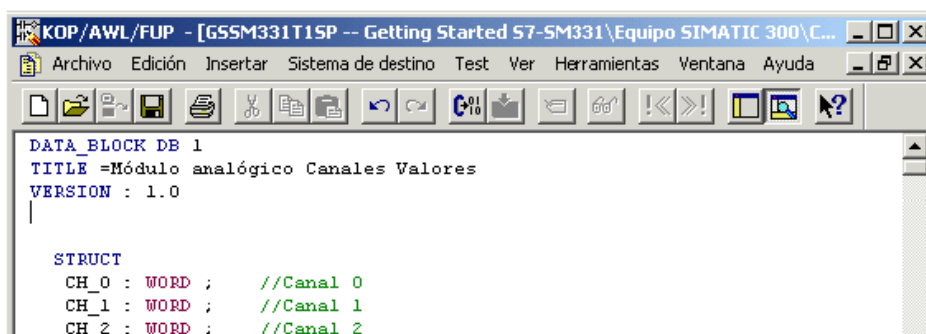


Figura 5-17 Editor de código

Fuente: SIEMENS-getting-started-teild.pdf.

Después de cargar el código fuente es necesario iniciar la compilación.

Para ello pulse las teclas Ctrl+K o haga clic en Archivo -> Compilar. Con ello se inicia inmediatamente la compilación.

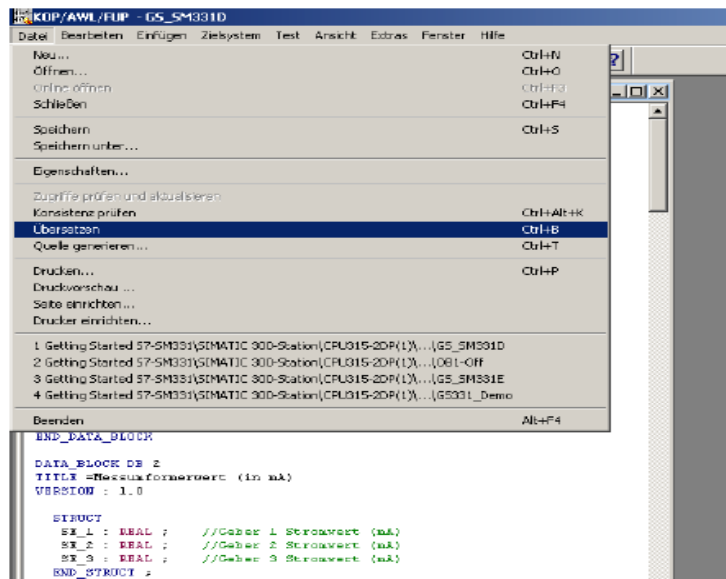


Figura 5-18 Compilar fuente AWL

Fuente: SIEMENS-getting-started-teild.pdf.

En caso de mensaje de error o advertencia, compruebe su fuente.

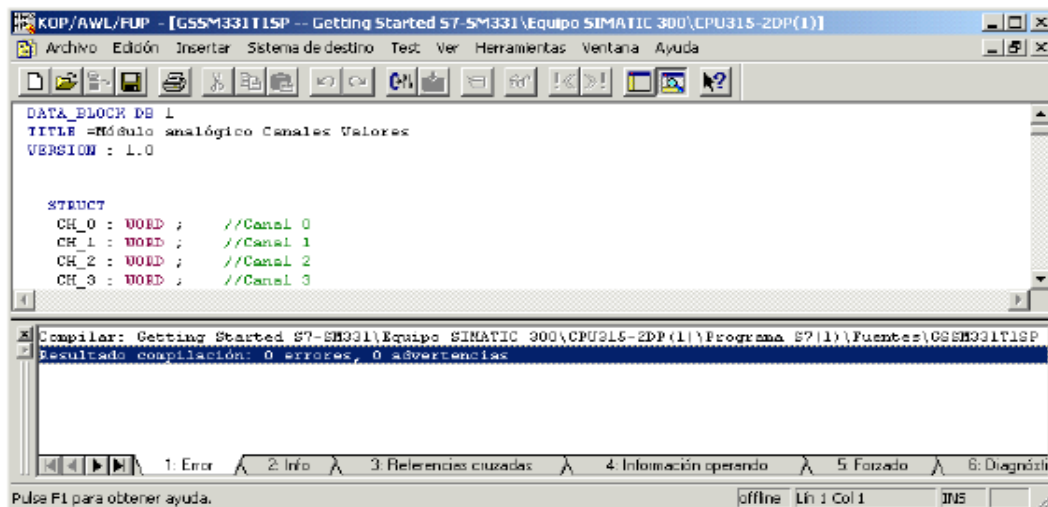


Figura 5-19 Editor de código fuente, mensajes tras la compilación

Fuente: SIEMENS-getting-started-teild.pdf.

Cierre el editor de código fuente.

Si se ha compilado sin errores la fuente AWL, la carpeta de bloques incluye los bloques siguientes:OB1, OB40, OB82, FC1, DB1 y DB2

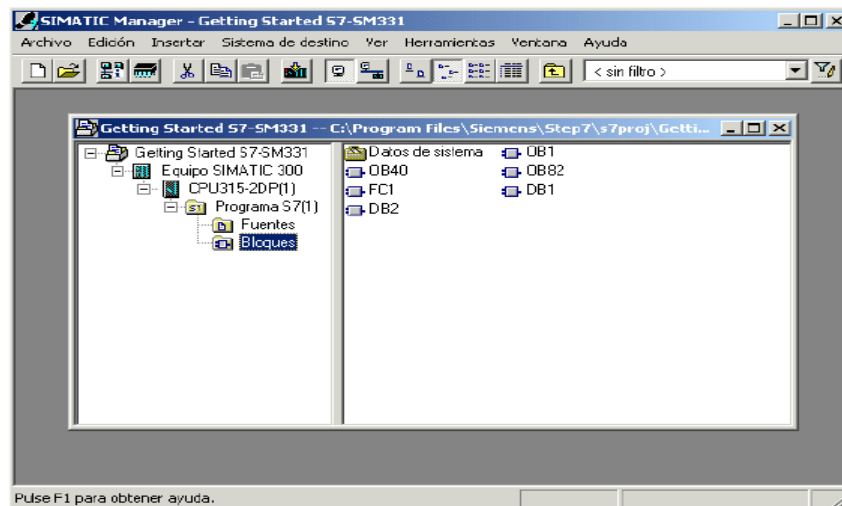


Figura 5-20 Bloques generados

Fuente: SIEMENS-getting-started-teild.pdf.

5.4 EJEMPLOS EN S7-300

Semáforo

Banda transportadora

Variación de voltaje

5.4.1 SEMÁFORO

Realizar un programa para un Plc que automatice el funcionamiento de un semáforo de 3 focos por lado.

Orden:

Prende rojo 1-verde2

Prende rojo1-verde2-amarillo1

Prende rojo2-verde1

Prende rojo 2-verde1-amarillo2

5.4.2 Banda transportadora

Realizar un programa para un Plc que automatice el funcionamiento de una banda transportadora para el empaque de cigarrillos.

Orden:

- Prende el dispensador 1 y todas las bandas
- Luego contar 3 cajetillas se apaga el dispensador 1 y la banda 1
- Prende el pistón 1 durante 2 seg
- Prende nuevamente dispensador 1, la banda 1, dispensador 2
- Cuenta 3 cajetillas, se apaga la banda 2 y se prende el piston 2
- Se repite nuevamente

5.4.3 Variación de voltaje

Realizar un programa para un Plc que de una salida análoga donde se visualice la variación de voltaje

Orden:

- Entra un señal análoga
- Sale una señal análoga

5.5 APLICACIONES DEL S7-300

Las áreas de aplicación del SIMATIC S7-300 incluyen:

5.5.1 SISTEMAS DE TRANSPORTE:

Gracias a su sencillez, permite programar y monitorear rápidamente aplicaciones como por ejemplo cintas transportadoras. La programación basada en " arrastrar y soltar " ayuda a configurar lógica de marcha/paro para motores con mando por pulsador y permite asimismo seleccionar contadores para supervisar el número de piezas producidas.

Controles de entrada y salida: Gracias a su diseño compacto, permite además una integración fácil en dispositivos de espacio reducido, como por ejemplo en barreras de aparcamientos o entradas. Como por ejemplo se puede detectar un vehículo tanto a la entrada como a la salida, abriendo o cerrando la barrera automáticamente. La cantidad de vehículos estacionados resulta fácil de comprobar programando simplemente un contador.

5.5.2 SISTEMAS DE ELEVACIÓN:

El potente juego de instrucciones de un PLC, permite que controle una gran variedad de sistemas de elevación de material. La vigilancia de secuencias de control (arriba/abajo) así como la capacidad de tomar decisiones eficientes en cuanto a tareas de control complejas son algunas de las tareas asistidas por todas las instrucciones residentes en el PLC.

5.5.3 OTRAS APLICACIONES:

Además de los ejemplos representados arriba, cabe considerar algunas de tantas otras tareas de automatización, para las que este PLC constituye la solución ideal:

- Líneas de ensamblaje
- Sistemas de embalaje
- Máquinas expendedoras
- Controles de bombas
- Mezclador
- Equipos de tratamiento y manipulación de material
- Maquinaria para trabajar madera
- Paletizadoras
- Máquinas textiles
- Máquinas herramientas

5.5.4 CAMPO DE APLICACIÓN

Los autómatas programables no se limitan a funciones de control lógico sino que también permiten

Regular,

Posicionar,

Contar, dosificar,

Mandar válvulas y mucho más

CAPITULO VI

6.1 EQUIPOS DE PRUEBA

Destacando los equipos de pruebas más utilizados están:

El multímetro

Multímetro es un instrumento muy importante debido a que sirve para realizar pruebas de salida de voltaje y corriente en sus terminales.



La PC es fundamental para la simulación de programas antes de cargar al PLC.

6.2 PRUEBAS REALIZADAS

6.2.1 PROBAR EL PROGRAMA DE USUARIO

Para realizar pruebas se siguen los siguientes pasos:

6.2.1.1 Descargar Datos de Sistema y Programa de Usuario

Tanto el hardware como el software están ya perfectamente preparados.

El siguiente paso es descargar a la CPU los datos del sistema y el programa de usuario. Para ello proceder de la siguiente forma:



Paso	Descripción
1	<p>Usando el Administrador SIMATIC cargue en la CPU los datos del sistema (incluye la configuración del hardware) y el programa de usuario.</p> 
2	 <p>Siga las instrucciones en pantalla.</p> <p>Si todos los sensores se han conectado correctamente, ni en la CPU ni en el SM331 luce ninguna lámpara roja de señalización.</p> <p>El funcionamiento de la CPU se señala con la lámpara verde "RUN".</p>

Tabla 6-1 Cargar en CPU datos de sistema y programa de usuario

Fuente: SIEMENS-getting-started-teild.pdf

6.2.1.2 Visualización de los Valores de los Sensores

Para visualizar los valores de los sensores, insertar en el proyecto la tabla de variables siguiente. Para ello abra el menú contextual en la carpeta Bloques y elija:

Insertar nuevo objeto -> Tabla de variables.

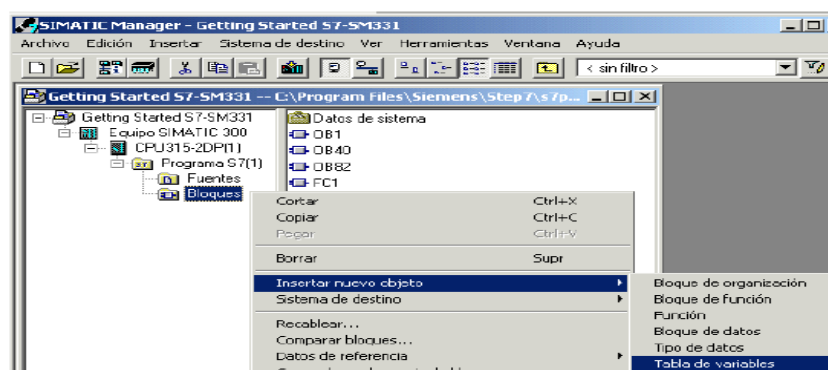


Figura 6-2 Insertar tabla de variables

Fuente: SIEMENS-getting-started-teild.pdf

Rellenar la tabla de variables creada de la forma siguiente:

	Operando	Forma	Valor de estado	Valor de forzado
1	//Valores de canales			
2	DB1.DBW/ 0	HEX		
3	DB1.DBW/ 2	HEX		
4	DB1.DBW/ 4	HEX		
5	DB1.DBW/ 6	HEX		
6	DB1.DBW/ 8	HEX		
7	DB1.DBW/ 10	HEX		
8	DB1.DBW/ 12	HEX		
9	DB1.DBW/ 14	HEX		
10				
11	//Valores analógicos			
12	DB2.DBD 0	REAL		
13	DB2.DBD 4	REAL		
14	DB2.DBD 8	REAL		
15				
16	//Estado del proceso			
17	MW 100	HEX		
18	M 200.0	BOOL		
19	M 101.0	BOOL		
20	M 101.1	BOOL		
21	M 101.2	BOOL		
22	M 101.3	BOOL		
23				

En esta zona puede observar los valores en los canales

En esta zona puede observar los valores analógicos

En esta zona puede observar y forzar las señales de estado

Figura 6-3 Tabla de variables Control_Display

Fuente: SIEMENS-getting-started-teild.pdf

6.2.1.2.1 Observación de valores

Para observar los valores pase, en modo online, a la CPU pulsando el icono con las gafas. Con ello podrá ver en pantalla los valores contenidos en los bloques de datos y marcas.

	Operando	Forma	Valor de estado	Valor de forzado
//Valores de canales				
1	DB1.DBW 0	HEX	W#16#02B8	
2	DB1.DBW 2	HEX	W#16#2570	
3	DB1.DBW 4	HEX	W#16#3350	
4	DB1.DBW 6	HEX	W#16#2416	
5	DB1.DBW 8	HEX	W#16#7FFF	
6	DB1.DBW 10	HEX	W#16#7FFF	
7	DB1.DBW 12	HEX	W#16#7FFF	
8	DB1.DBW 14	HEX	W#16#7FFF	
//Valores analógicos				
11	DB2.DBD 0	REAL	4.402778	
12	DB2.DBD 4	REAL	11.60185	
13	DB2.DBD 8	REAL	9.347222	
//Estado del proceso				
16	Mw 100	HEX	W#16#0000	
17	M 200.0	BOOL	false	
18	M 101.0	BOOL	false	
19	M 101.1	BOOL	false	
20	M 101.2	BOOL	false	
21	M 101.3	BOOL	false	
22				
23				

Figura 6-4 Vista online de la tabla de variables

Fuente: SIEMENS-getting-started-teild.pdf

6.2.1.2.2 Forzado de valores

Para forzar el acuse del proceso escriba en la columna "Valor a forzar" el valor deseado ("TRUE o FALSE" dependiendo se desea activar o desactivar el acuse) y pulse el icono con la flecha doble:

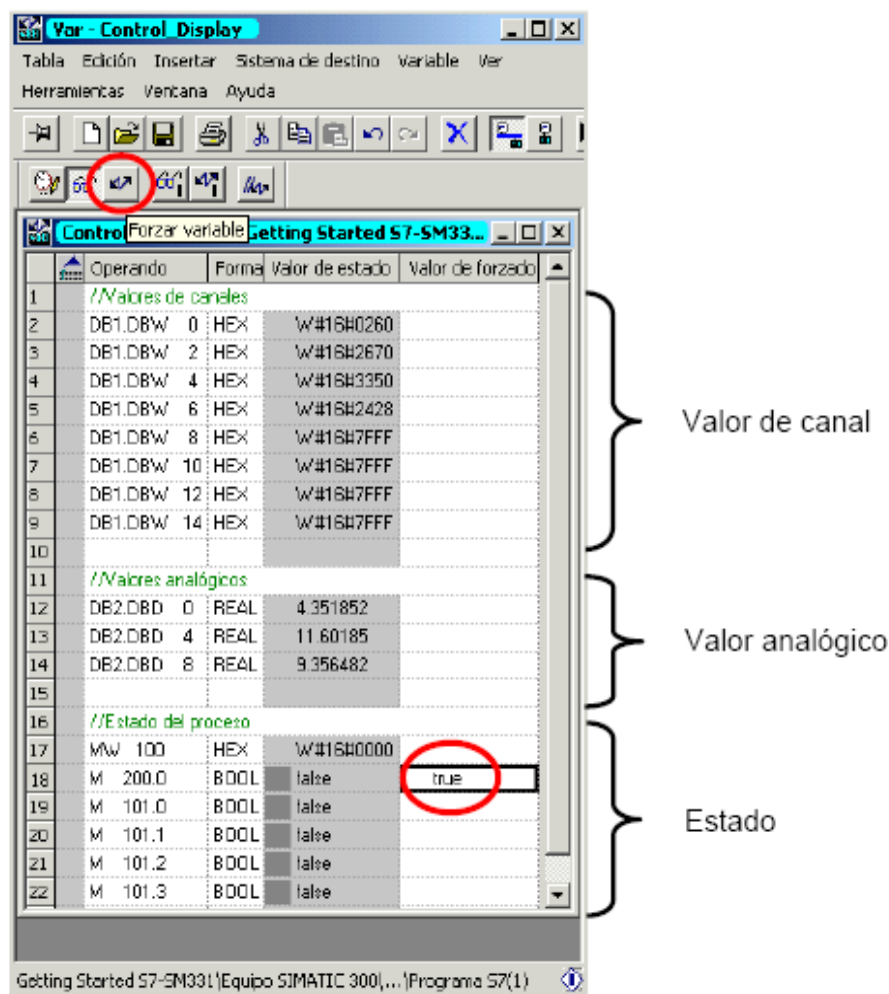


Figura 6-5 Forzado de variables

Fuente: SIEMENS-getting-started-teild.pdf

6.2.1.2.3 Particularidad al observar los valores

Al observar los valores le ha llamado seguramente la atención el hecho de que los valores de canal no coinciden con los valores analógicos. La causa de ello es que el módulo analógico únicamente emite el formato binario.Palabra. (16 bits). Es decir, es necesario convertir los valores del módulo analógico.

6.2.1.3 Representación de Valores Analógicos

La CPU sólo puede procesar valores analógicos digitalizados. Por ello los módulos de entrada analógica transforman la señal analógica de proceso en formato digital (palabra de 16 bits).

A la hora de convertir valores digitales en analógicos es necesario considerar cinco zonas:

Hexadecimal	Rango medida intensidad	Comentario	Significado
7FFF	22,96 mA	Rebase por exceso	A partir del valor hexadecimal 16#7F00 el valor leído del sensor se encuentra fuera del margen de medida parametrizado, con lo que no es válido.
7F00			
7EFF	22,81 mA	Zona de saturación por exceso	Esta zona equivale a una banda de tolerancia antes de que se produzca desbordamiento. Sin embargo, dentro de esta zona la resolución ya no es óptima.
6C01			
6C00	20 mA	Zona nominal	Es la zona normal para leer los valores medidos. En esta zona la resolución es óptima.
5100	15 mA		
1	4 mA + 578,7 nA		
0	4 mA		
FFFF		Zona de saturación por defecto	Zona que equivale a la zona de saturación por exceso sólo que para valores bajos.
ED00	1,1185 mA		
ECFF		Rebase por defecto	A partir del valor hexadecimal 16#ECFF el valor leído del sensor se encuentra por debajo del margen de medida parametrizado, por lo que no es válido.
8000			

Tabla 6-3 Representación de valores analógicos en margen de medida de intensidad 4 a 20 mA

Es decir, es necesario convertir la representación de valores binarios para poder visualizar valores analógicos del proceso. En el programa de nuestro ejemplo se visualizan los valores en mA. Esto se realiza utilizando una función programada (la FC1) encargada de convertir en mA la representación del valor analógico.

En nuestro ejemplo observaremos los valores a la salida del transductor.

Utilizando un amperímetro es posible comparar los valores medidos por él con los valores en la representación analógica. Los valores serán idénticos.

6.2.1.4 Alarma de Diagnóstico

Las alarmas de diagnóstico sirven para reaccionar en el programa de usuario frente a averías en el hardware.

Son los módulos diagnosticables que están en condiciones de señalar alarmas de diagnóstico.

En el OB82 se programan las reacciones a las alarmas de diagnóstico.

6.2.1.4.1 Leer desde PG información de diagnóstico

El módulo de entrada analógica SM331 AI8x12bit es diagnosticable.

La alarma de diagnóstico aparecida se señala en el módulo SM331 y en la CPU porque luce el LED rojo .SF..

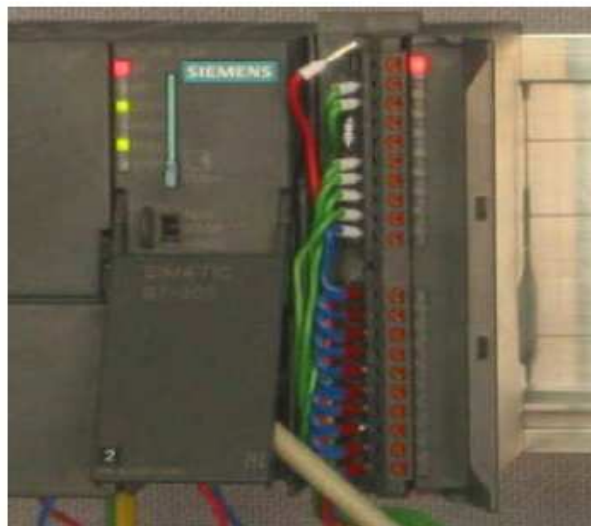


Figura 6-6 Avería hardware

Fuente: SIEMENS-getting-started-teild.pdf

La causa del fallo puede detectarse .online. Consultando el estado del módulo.

Para poder ver "online" el estado del módulo, proceda de la forma siguiente.

En la configuración del hardware, haga clic en SM331 y abra el diagnóstico, del hardware en el menú Sistema de destino / Estado del módulo.

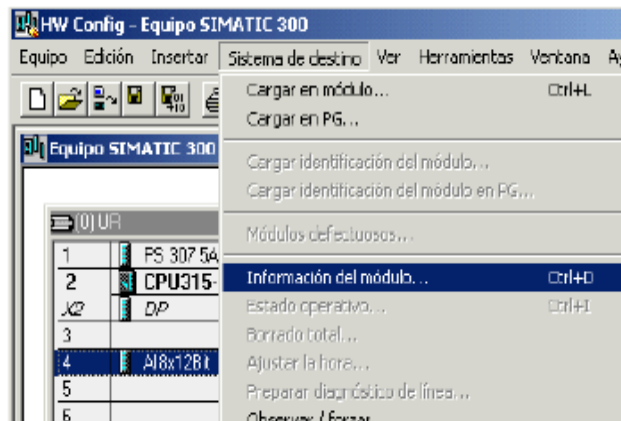


Figura 6-7 Estado del módulo

Fuente: SIEMENS-getting-started-teild.pdf

6.2.1.4.2 Mensajes Generales de Diagnóstico

En la ficha Alarma de diagnóstico figura información sobre el error notificado.

Las alarmas no dependen de un determinado canal y afectan a todo el módulo.

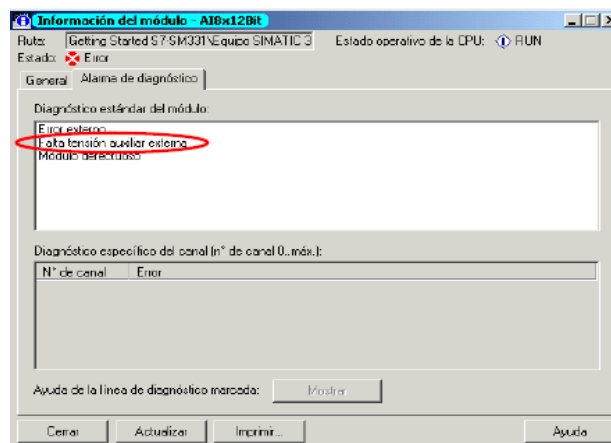


Figura 6-8 Diagnóstico del SM331

Fuente: SIEMENS-getting-started-teild.pdf

6.2.1.4.3 Mensajes de Diagnóstico por Canal

Existen cinco mensajes de diagnóstico por canal:

- Error de configuración/parametrización
- Error de modo común

Rotura de hilo
Rebase por defecto
Rebase por exceso

6.2.1.4.3.1 Error de Configuración/parametrización

La posición de los adaptadores de margen de medida no coincide con el tipo de medición configurado en la configuración del hardware.

6.2.1.4.3.2 Error de Modo Común

La diferencia de potencia U_{cm} entre las entradas (M-) y el potencial de referencia del circuito de medida (M_{ana}) es excesiva.

En nuestro ejemplo no puede aparecer este error ya que en un transductor a 2 hilos M_{ana} está unido con M (sin aislamiento galvánico).

6.2.1.4.3.3 Rotura de hilo

Si está parametrizada la rotura de hilo en transductores de intensidad no se comprueba directamente si se ha roto el hilo sino que esta función de diagnóstico actúe cuando se rebase por defecto una determinada intensidad límite.

En un transductor de 4 a 20mA, cuando la intensidad baja de 3,6 mA se visualiza el mensaje .Rotura de hilo entrada analógica. en la pantalla de diagnóstico del módulo.

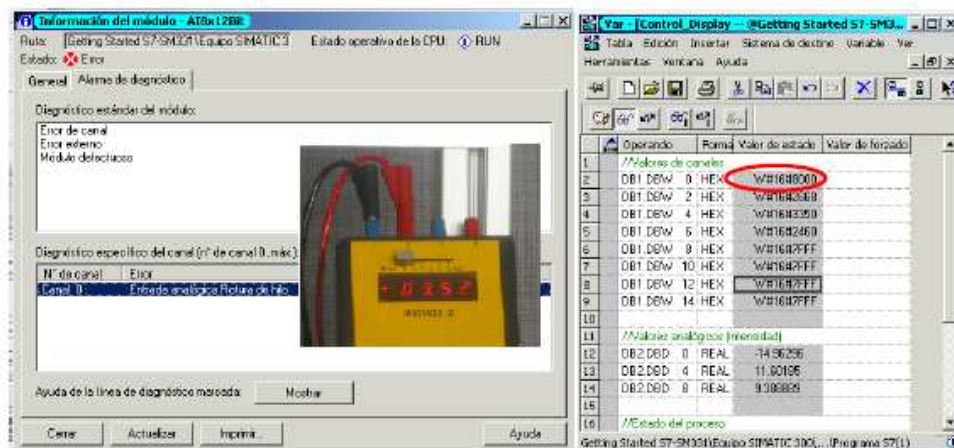


Figura 6-9 Izda.: pantalla de diagnóstico con rotura de hilo/dcha.: tabla de variables

Fuente: SIEMENS-getting-started-teild.pdf

La representación del valor analógico pasa inmediatamente a la zona de rebase por defecto (HEX 8000), a pesar de que la intensidad medida supera ampliamente 1,1185 mA (ver apt. 7.3).

Un rebase por defecto de 3,6 mA sólo es posible si se ha desactivado el parámetro Detección de rotura de hilo.

6.2.1.4.3.4 Rebase por defecto

El mensaje de rebase por defecto sólo se activa si está desactivada la función de diagnóstico de rotura de hilo y la intensidad es inferior a 1,185 mA.

6.2.1.4.3.5 Rebase por exceso

Si la intensidad supera el valor de 22,81 mA en la ventana de diagnóstico se presenta el mensaje con el texto .Entrada analógica, margen de medida/ límite superior sobrepasado..

La representación de valor analógico (HEX 7FFF) está dentro de la zona de rebase por exceso.

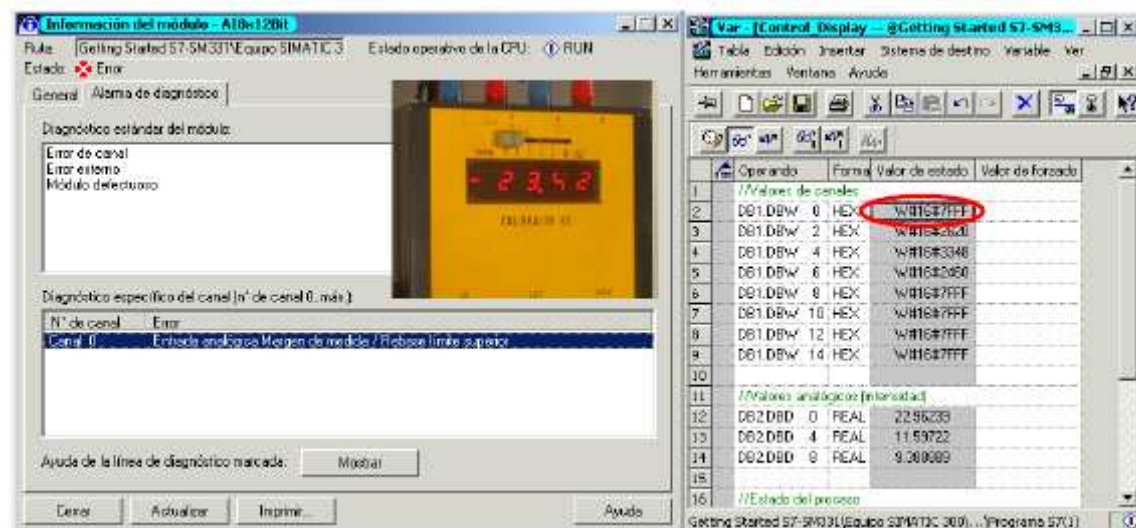


Figura 6-10 Izda.: diagnóstico en zona de rebase por exceso/dcha.: tabla de variables

Fuente: SIEMENS-getting-started-teild.pdf

6.2.1.5 Alarma de Proceso

El SM331 AI8x12Bit tiene como particularidad su capacidad para disparar alarmas de proceso. Para ello pueden configurarse correspondientemente dos canales (el 1 y 2).

Por regla general las alarmas de proceso llaman un bloque de organización de alarmas en la CPU. En nuestro ejemplo se llama el OB40.

Para transformadores de intensidad los límites de las alarmas de proceso deben definirse en mA

Si aparece una alarma de proceso, entonces se llama el OB40. En el programa de usuario del OB40 puede definirse qué funciones debe ejecutar el PLC debido a como consecuencia de la alarma de proceso.

En el programa de usuario del ejemplo la causa de la alarma de proceso se lee en el OB40. Ésta figura en la estructura de variables temporal OB40_POINT_ADDR (palabras locales 8 a 11).

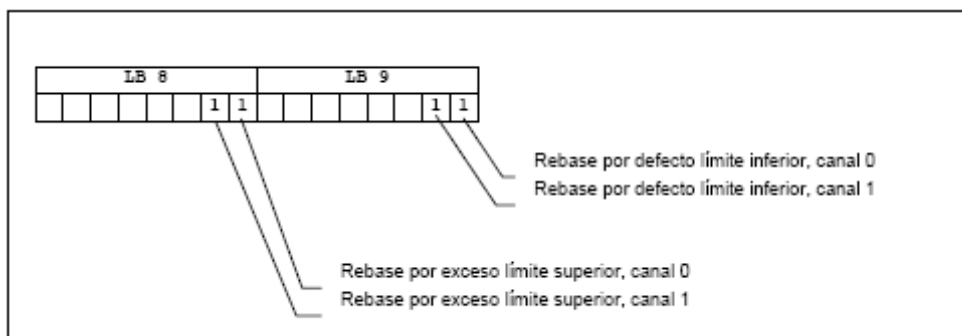


Figura 6-11 Información de arranque del OB40: qué incidencia ha activado la alarma de proceso al rebasarse un límite

Fuente: SIEMENS-getting-started-teild.pdf

En el ejemplo, en el OB40 sólo se transfieren a una palabra de marcas (MW100) el LD8 y el LD9. La palabra de marcas se muestra en la tabla de variables ya generada. La palabra de marcas en el OB1 se acusa con la marca M200.0 ó forzando la marca en la tabla de variables con "TRUE".

Si usando un calibrador alimenta el canal 0 con 5,71 mA, entonces en el MW100 de la tabla de variables aparece el valor HEX 0001. Esto significa que se ha llamado el OB40 y que en el canal 0 se ha rebasado el límite inferior a 6 mA.

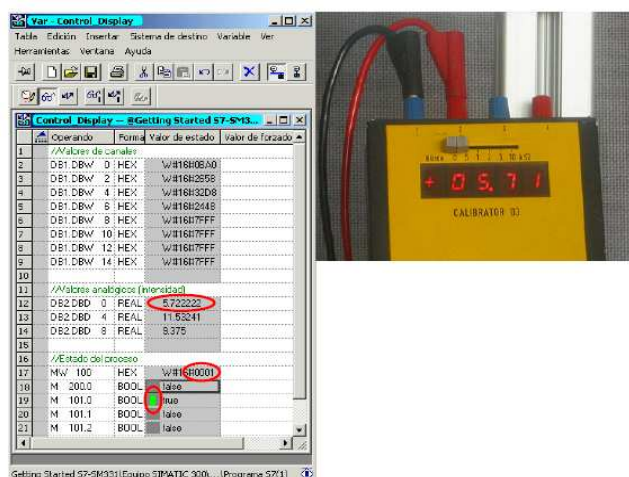


Figura 6-12 Alarma de proceso: rebase por defecto el límite inferior en canal 0

Fuente: SIEMENS-getting-started-teild.p

6.2.2 ADVERTENCIA TAREAS DE SIMULACIÓN

6.2.2.1 Iniciar la Simulación

Para iniciar S7-PLCSIM, utilice uno de los métodos siguientes:

En el menú Inicio de Windows, elija el comando **Simatic > STEP 7 > S7-PLCSIM Simular módulos**.

En la barra de herramientas del Administrador SIMATIC, haga clic en el botón "Simulación ON/OFF" o elija el comando de menú **Herramientas > Simular módulos**.

Si está activado el botón "Simulación ON/OFF" en el Administrador SIMATIC, todos los nuevos enlaces se establecerán automáticamente con el PLC simulado. Cuando active la simulación, todos los programas se cargarán en el PLC simulado, si la dirección MPI del PLC simulado coincide con la del proyecto de STEP 7 que contiene el programa. Si hace clic en el botón "Estaciones accesibles", en la ventana "Estaciones accesibles" aparecerá la dirección de estación del PLC simulado que se ha creado.

Si está desactivado el botón "Simulación ON/OFF" , todos los nuevos enlaces se establecerán automáticamente con el PLC real. Si hace clic en el botón "Estaciones accesibles", en la ventana de "Estaciones accesibles" aparecerá la red de PLCs reales.

Tras iniciar S7-PLCSIM puede abrir un PLC simulado u observar el programa simulado. Puede continuar simulando el PLC hasta finalizar la sesión de simulación.

6.2.2.2 Abrir un PLC simulado

Cuando arranque S7-PLCSIM, se abrirá un nuevo PLC simulado sin título. Si es la primera vez que utiliza S7-PLCSIM, aparecerá únicamente una

subventana CPU. En caso contrario, S7-PLCSIM visualizará las subventanas de la sesión anterior de trabajo.

Ahora podrá elegir una de las siguientes alternativas:

- Puede comenzar a trabajar en S7-PLCSIM utilizando el nuevo PLC simulado.
- Puede abrir un PLC simulado que haya guardado anteriormente. Elija el comando de menú **Archivo > Última simulación**, o bien **Archivo > Abrir PLC** para seleccionar el archivo .PLC deseado. El trabajo realizado con este PLC (p. ej. cargar un programa o una configuración de hardware, o bien asignar valores en una subventana) se guarda en la simulación almacenada.

Por tanto, cuando abra la simulación no tendrá que repetir esos pasos. Puede comenzar a trabajar donde concluyó la sesión anterior.

Tras haber decidido si desea trabajar con un PLC simulado nuevo o existente, puede también agregar o cambiar las subventanas visualizadas en S7-PLCSIM. Si la organización se visualiza cuando abra S7-PLCSIM, tiene la posibilidad de cerrarla. Para agregar nuevas subventanas, utilice la barra de herramientas o los comandos de los menús Ver e Insertar. Asimismo, puede elegir el comando de menú **Archivo > Abrir organización** para seleccionar una organización distinta (si ha guardado más de una). Los cambios de la organización no afectan a la información del programa, sino únicamente a la forma como se visualiza la información.

.

6.2.2.3 Elegir el modo de simulación

S7-PLCSIM ofrece diversas opciones para ejecutar el programa simulado:

6.2.2.3.1 Ciclo individual:

La CPU ejecuta un ciclo y espera hasta que Ud. inicie el siguiente ciclo. En cada ciclo, la CPU lee las entradas de periferia (PI), ejecuta el programa y escribe luego los resultados en las salidas de periferia (PQ). La CPU espera

hasta que Ud. inicie el siguiente ciclo (utilizando el comando de menú **Ejecutar > Siguiente ciclo** o haciendo clic).

6.2.2.3.2 Ciclo continuo:

La CPU ejecuta un ciclo completo e inicia luego otro ciclo. En cada ciclo, la

CPU lee las entradas de periferia (PI), ejecuta el programa y escribe luego los resultados en las salidas de periferia (PQ).

Para elegir la opción "Ciclo individual", haga clic en o elija el comando de menú **Ejecutar > Ciclo> Individual**. Ejecutando un programa con la opción "Ciclo individual" se pueden visualizar los cambios en cada ciclo. En tanto que una CPU real puede ejecutar el programa a una mayor velocidad de la que el editor visualiza los datos, la opción "Ciclo individual" de S7-PLCSIM permite "congelar" el estado del programa de ciclo en ciclo.

Si quiere elegir la opción "Ciclo continuo", haga clic en o elija el comando de menú **Ejecutar >Ciclo > Continuo**. (El ajuste estándar para el simulador es la opción "Ciclo continuo".)

6.2.2.4 Cambiar el modo de operación de la CPU

Es posible cambiar el modo de operación de la CPU. El PLC simulado reacciona de la misma manera que un PLC real. En las casillas de verificación que aparecen en la subventana del PLC simulado se indica el modo de operación actual. Para cambiar el modo de operación, haga clic en la casilla de verificación en cuestión o utilice los comandos de menú **Ejecutar > Posición del selector de modo [modo]**.

Los modos de operación de la CPU simulada funcionan como el selector de modo de una CPU real. Si se utilizan herramientas de STEP 7 para cambiar el modo de operación o caso de que la CPU cambie automáticamente de modo (por ejemplo, si se presenta una condición de error y la CPU cambia de RUN a STOP), no se modifican las casillas de verificación en la subventana de la CPU simulada. El indicador LED cambia, más no la posición del selector. Ello le

indica que ha cambiado el modo de operación de la CPU, probablemente debido a un error en el programa.

6.2.2.5 Observar el programa simulado

Es posible visualizar diferentes tipos de subventanas para visualizar y modificar el programa que se está ejecutando en el PLC simulado.

Las siete subventanas siguientes se activan a partir del menú **Insertar**.

6.2.2.5.1 Entrada:

Permite acceder a los datos almacenados en la imagen del proceso de las entradas (I). La dirección estándar es el byte 0 (IB0).

6.2.2.5.2 Salida:

Permite acceder a los datos almacenados en la imagen del proceso de las salidas (Q). La dirección estándar es el byte 0 (QB0).

6.2.2.5.3 Marca:

Permite acceder a los datos almacenados en el área de marcas (M). La dirección estándar es el byte 0 (MB0).

6.2.2.5.4 Temporizador:

Permite acceder a los temporizadores utilizados por el programa. El temporizador estándar es T0.

6.2.2.5.5 Contador:

Permite acceder a los contadores utilizados por el programa. El contador estándar es C0.

6.2.2.5.6 General:

Permite acceder a cualquiera de las áreas de memoria de la CPU simulada, incluyendo los bloques de datos (DBs) del programa.

6.2.2.5.7 Bits verticales:

Permite visualizar la dirección simbólica o absoluta de todos los bits, así como observar y modificar datos. La subventana "Bits verticales" se puede utilizar para representar los bits de las variables de las entradas y salidas de periferia, de las variables de la imagen del proceso de las entradas y de las salidas, de la memoria de marcas y de los bloques de datos.

El direccionamiento simbólico también se puede utilizar para acceder a cualquiera de las subventanas mencionadas.

Las tres subventanas siguientes se activan a partir del menú **Ver**:

6.2.2.5.7.1 Acumuladores:

Permite acceder a los datos de los diversos acumuladores de la CPU, así como a la palabra de estado y a los registros de direcciones. En esta subventana se visualizan cuatro campos de acumuladores para la CPU S7-400. Los programas para la CPU S7-300 utilizan sólo dos acumuladores.

6.2.2.5.7.2 Registros de bloques:

Permite visualizar el contenido de los registros de direcciones de los bloques de datos depositados en la CPU simulada. Además, se visualizan los números del bloque lógico que se está ejecutando y del bloque lógico precedente, así como el número de la operación (contador de direcciones SAZ) que se está ejecutando.

6.2.2.5.7.3 Pilas:

Permite visualizar los datos almacenados en la pila de anidado y en la pila MCR (Master Control Relay) del PLC simulado.

El programa se puede observar simultáneamente con la aplicación de STEP 7 "KOP/AWL/FUP":

1. En el Administrador SIMATIC, haga clic en o elija el comando de menú **Ver > Online** para conmutar al modo online.
2. Navegue hasta el objeto "Bloques" del proyecto de ejemplo S7_ZEBRA y abra FC1.
3. En la aplicación "KOP/AWL/FUP" se visualizará el programa que se está ejecutando en el PLC simulado. Utilice los comandos de menú para ver el estado de las operaciones.

6.2.2.6 Utilizar el regulador

El regulador permite simular valores que cambian gradualmente o que están comprendidos en un margen específico, como p. ej. los valores analógicos. El regulador se puede utilizar en las subventanas siguientes:

Entrada

Salida

Marca

Al área de memoria se puede acceder bien sea a través de una dirección en la memoria, o bien mediante una dirección simbólica. Si desea especificar un regulador para una de las subventanas, elija la representación del regulador en la lista desplegable Seleccionar formato numérico. Al seleccionar el regulador se elige al mismo tiempo el formato para representar los valores: decimal (enteros positivos), entero (enteros positivos y negativos) o real. Las opciones seleccionables dependen del tamaño de la dirección de la memoria a la que se accede:

Byte (B): decimal

Palabra (W): decimal y entero

Palabra doble (D): decimal, entero y real

Utilice el ratón o las teclas con flecha para graduar la posición del regulador. Cambiando su posición se modifica el valor de la variable almacenada en la correspondiente dirección de la memoria. También es posible introducir un valor exacto en el campo "Valor". Para el regulador se puede configurar también un valor mínimo y uno máximo. La selección de un margen no afecta a los valores almacenables en la variable: los valores mínimo y máximo afectan únicamente a los valores que se pueden introducir o visualizar mediante el regulador. La selección de un margen de valores aporta las siguientes ventajas:

Se puede simular un margen determinado de valores. Ello permite, por ejemplo, simular el margen de valores que generaría un módulo analógico en particular.

Limitando el margen de valores entre uno mínimo y uno máximo, se ofrece una mejor resolución para introducir datos con el regulador. En el campo "Valor" de la subventana se puede introducir siempre un valor exacto.

Si desea seleccionar un valor mínimo para el regulador, seleccione "Mín" en la lista desplegable

"Visualizar valor", "Mín" o "Máx" de la subventana. A continuación, introduzca el valor numérico mínimo en el campo "Mín".

Si desea seleccionar un valor máximo para el regulador, seleccione "Máx" en la lista desplegable

"Visualizar valor", "Mín" o "Máx" de la subventana. A continuación, introduzca el valor numérico máximo en el campo "Máx".

6.2.2.7 Consideraciones relativas a las entradas y salidas

Para poder simular OBs de alarma es preciso cargar una configuración de hardware que contenga las entradas y salidas (E/S). En algunos casos, las E/S

se incluyen automáticamente en los datos de sistema que STEP 7 carga en S7-PLCSIM. De lo contrario, deberá modificar la configuración que contenga las E/S.

6.2.2.8 Conectar y desconectar la alimentación de la CPU simulada

Elija los comandos **PLC > ON** u **OFF** para simular que se conecta o desconecta la fuente de alimentación de la CPU, respectivamente.

6.2.2.8.1 Utilizar direcciones simbólicas

Para utilizar direcciones simbólicas en el programa simulado:

Elija el comando de menú **Herramientas > Opciones > Asignar símbolos** para abrir el correspondiente cuadro de diálogo.

Seleccione la tabla de símbolos de STEP 7 que desea utilizar.

Haga clic en el botón "Aceptar".

Cree una subventana para las variables que desea direccionar simbólicamente.

Para visualizar las direcciones simbólicas en todas las subventanas, elija el comando de menú **Herramientas > Opciones > Mostrar símbolos**.

Para ocultarlas, seleccione nuevamente dicho comando.

En la subventana "Bits verticales", los valores de bit se visualizan verticalmente y las direcciones simbólicas o absolutas aparecen junto a los bits correspondientes. En todas las demás subventanas, se dispone de rótulos informativos para los campos de direcciones. Para ver la dirección simbólica y el comentario (separados entre sí mediante dos puntos) de un campo determinado, apunte a éste con el puntero del ratón. Entonces aparecerá un rótulo con la información en cuestión.

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

Con la finalización de este proyecto se cumplió a cabalidad con los objetivos planteados, debido a que es un tablero de fácil manejo para realizar prácticas de automatización.

El diseño mecánico es independiente de la tarea de control y permanece constante.

El cableado se restringe a la conexión de los sensores y actuadores ubicados en el proceso.

Permiten un diagnóstico directo de fallas.

Los PLC's tienen una amplia gama de módulos para una adaptación óptima a la tarea de automatización en particular.

El PLC ofrece alto rendimiento para automatizar máquinas e instalaciones en la industria manufacturera. No requiere mantenimiento, ofrece numerosas funciones integradas.

Es de fácil manejo gracias a un montaje simple y sencillo en cuanto a conexiones.

Recomendaciones

Se recomienda que el proyecto presentado se lo siga reacondicionando y se lo utilice como material didáctico, para la optimización o creación de sistemas de control de nivel automatizado.

Cablear el S7-300 únicamente cuando esté desconectada la tensión.

No es posible cambiar la dirección de estación del PLC simulado mientras esté utilizando aplicaciones de STEP 7 para observar el mismo.

El modo de simulación no estará disponible si existe algún enlace con un PLC real.

Puede entrar en contacto con cables bajo tensión si está encendida la fuente de alimentación o está conectado a la red el cable de la alimentación.

Para el correcto mantenimiento de este tablero de automatización se recomienda visualizar la numeración respectiva de entradas y salidas.

El PLC SIEMENS puede ser sustituido por cualquier otra marca, si y solo si tiene las especificaciones técnicas relativamente iguales.

Verifique el diagrama de circuito que viene impreso en el mismo PLC (levantando la tapa de cada módulo), porque le facilitara el conexionado.

BIBLIOGRAFÍA

J. Gißler, M. Schmid: Vom Prozeß zur Regelung. Analyse, Entwurf, Realisierung in der Praxis. Siemens AG. ISBN 3-8009-1551-0. (libro técnico en alemán).

Álvaro Aguinaga: Manual/EPN/Automatización/

Horta Santos José: Técnicas de Automación Industrial

Fipetrol Tecno Lat.: Controladores Lógicos Programables (PLC)

Mc Graw Hill: Manual de Mantenimiento Industrial

SIEMENS: Manual/SIEMENS-getting-started-teild.pdf

Manual: *Autómata programable S7-300*

Configuración, instalación y datos de las CP

Manual de referencia: *Sistemas de automatización S7-300 y M7-300*

Datos de los módulos

Manual del usuario: *Software estándar para SIMATIC S7 y M7,*

STEP 7

Manual de programación: *Software de sistema para SIMATIC S7-300/400*

Diseño de programas

www.es.wikipedia.org/Automatización_industrial

www.deremate.com.ar/accdb/viewitem.asp.

www.siemens.com.co/SiemensDotNetClient_Andina

www.grupo-maser.com/PAG_Curso/Auto/

ANEXOS I
PLANOS Y FOTOS
DEL TABLERO

