



Universidad de San Carlos de Guatemala  
Facultad de Ingeniería  
Escuela de Ingeniería Mecánica Eléctrica

**IMPLEMENTACIÓN DE MIGRACIÓN DE CONTROLADOR LÓGICO  
PROGRAMABLE S5 A S7, DE UNA MÁQUINA LLENADORA DE BEBIDAS**

**Hugo Leonel Tocón Ajsivinac**

Asesorado por el Ing. Lester Guillermo Morales Ruiz

Guatemala, mayo de 2014





UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**IMPLEMENTACIÓN DE MIGRACIÓN DE CONTROLADOR LÓGICO  
PROGRAMABLE S5 A S7, DE UNA MÁQUINA LLENADORA DE BEBIDAS**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA  
FACULTAD DE INGENIERÍA

POR

**HUGO LEONEL TOCÓN AJSIVINAC**

ASESORADO POR EL ING. LESTER GUILLERMO MORALES RUIZ

AL CONFERÍRSELE EL TÍTULO DE

**INGENIERO MECÁNICO ELECTRICISTA**

GUATEMALA, MAYO DE 2014



UNIVERSIDAD DE SAN CARLOS DE GUATEMALA  
FACULTAD DE INGENIERÍA



**NÓMINA DE JUNTA DIRECTIVA**

DECANO	Ing. Murphy Olympto Paiz Recinos
VOCAL I	Ing. Alfredo Enrique Beber Aceituno
VOCAL II	Ing. Pedro Antonio Aguilar Polanco
VOCAL III	Inga. Elvia Miriam Ruballos Samayoa
VOCAL IV	Br. Walter Rafael Véliz Muñoz
VOCAL V	Br. Sergio Alejandro Doniz Soto
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

**TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO**

DECANO	Ing. Murphy Olympto Paiz Recinos
EXAMINADOR	Ing. Julio Rolando Barrios Archila
EXAMINADOR	Ing. Armando Gálvez Castillo
EXAMINADOR	Ing. Carlos Humberto Pérez Rodríguez
SECRETARIO	Ing. Hugo Humberto Rivera Pérez



## HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

### **IMPLEMENTACIÓN DE MIGRACIÓN DE CONTROLADOR LÓGICO PROGRAMABLE S5 A S7, DE UNA MÁQUINA LLENADORA DE BEBIDAS**

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Mecánica Eléctrica, con fecha octubre de 2012.



**Hugo Leonel Tocón Ajsivinac**

Guatemala, 10 de septiembre de 2013

Ingeniero  
Romeo Neftalí López Orozco  
Coordinador Área Electrotecnia  
Escuela de Mecánica Eléctrica

Señor Coordinador:

Atentamente informo a usted que he tenido a bien revisar el Trabajo de Graduación titulado "IMPLEMENTACIÓN DE MIGRACIÓN DE CONTROLADOR LÓGICO PROGRAMABLE S5 A S7, DE UNA MÁQUINA LLENADORA DE BEBIDAS" desarrollado por el estudiante Hugo Leonel Tocón Ajsivinac; y habiéndolo encontrado satisfactorio en su contenido y resultados me permito dar aprobación al mismo en el entendido de que tanto el Autor como el Asesor, somos responsables del desarrollo y conclusiones del mismo.

Sin otro particular quedo de usted,

Atentamente,

  
Ing. Lester Guillermo Morales Ruiz

Colegiado 8762

Asesor

Lester Guillermo Morales Ruiz  
Ingeniero Electricista  
Col. 8762

UNIVERSIDAD DE SAN CARLOS  
DE GUATEMALA



ACULTAD DE INGENIERIA

Ref. EIME 72. 2013

Guatemala, 27 de SEPTIEMBRE 2013.

Señor Director

Ing. Guillermo Antonio Puente Romero  
Escuela de Ingeniería Mecánica Eléctrica  
Facultad de Ingeniería, USAC.

Señor Director:

**Me permito dar aprobación al trabajo de Graduación titulado:  
IMPLEMENTACIÓN DE MIGRACIÓN DE CONTROLADOR  
LÓGICO PROGRAMABLE S5 A S7, DE UNA MÁQUINA  
LLENADORA DE BEBIDAS, del estudiante Hugo Leonel Tocón  
Ajsivinac, que cumple con los requisitos establecidos para tal fin.**

Sin otro particular, aprovecho la oportunidad para saludarle.

Atentamente,  
DIRECCIÓN Y ENSEÑANZA A TODOS

  
Ing. Romeo Nefitali López Orozco  
Coordinador Área Electrotécnica



S/O



REF. EIME 72. 2013.

**El Director de la Escuela de Ingeniería Mecánica Eléctrica, después de conocer el dictamen del Asesor, con el Visto Bueno del Coordinador de Área, al trabajo de Graduación del estudiante; HUGO LEONEL TOCÓN AJSIVINAC titulado: IMPLEMENTACIÓN DE MIGRACIÓN DE CONTROLADOR LÓGICO PROGRAMABLE S5 A S7, DE UNA MÁQUINA LLENADORA DE BEBIDAS, procede a la autorización del mismo.**

  
Ing. Guillermo Antonio Puente Romero



GUATEMALA, 11 DE OCTUBRE 2013.





El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería Mecánica Eléctrica, al trabajo de graduación titulado: **IMPLEMENTACIÓN DE MIGRACIÓN DE CONTROLADOR LÓGICO PROGRAMABLE S5 A S7, DE UNA MÁQUINA LLENADORA DE BEBIDAS**, presentado por el estudiante universitario: **Hugo Leonel Tocón Ajsivinac** y después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, se autoriza la impresión del mismo.

IMPRÍMASE.



Ing. Murphy Olympto Paiz Recinos  
Decano

Guatemala, mayo de 2014





## **AGRADECIMIENTOS A:**

- Dios** Por darme la vida, sabiduría, conocimiento y fuerzas para poder culminar mi carrera.
- Mis padres** Juan José Tocón y Felipa Ajsivinac. Por ser ejemplos de amor y valores, por enseñarme el camino y a seguir a nuestro señor Jesucristo y a su madre la Virgen María.
- Mis hermanos** Dora Leticia, Edgar Geovani, José Alexander, María Elisa, Brenda Marisol, Sonia Patricia y Byron Noé. Por ser esa compañía inseparable y apoyo en mi vida.
- Mis amigos** Por haber estado siempre apoyándome en las buenas y en las malas, y aconsejándome de manera incondicional.



# ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES .....	XI
LISTA DE SÍMBOLOS .....	XIX
GLOSARIO .....	XXI
RESUMEN .....	XXV
OBJETIVOS.....	XXVII
INTRODUCCIÓN .....	XXIX
1. LOS CONTROLADORES LÓGICO PROGRAMABLE SIMATIC S7 Y SOFTWARE STEP7 .....	1
1.1. ¿Qué es un PLC? .....	1
1.2. Visión general de SIMATIC .....	2
1.2.1. El PLC S7-200.....	4
1.2.2. El PLC S7-300.....	8
1.2.3. El PLC S7-400.....	13
1.3. Software STEP 7 .....	17
1.3.1. Dispositivos de programación.....	17
1.3.2. Requerimiento de la PG/PC para la instalación del Software STEP 7 .....	19
1.3.3. Instalación del Software STEP 7 .....	20
1.3.4. Del proceso al proyecto .....	20
1.3.5. Estructura del proyecto del Software STEP 7.....	22
1.3.6. Arrancar el Administrador SIMATIC.....	23
1.3.6.1. Creación de un proyecto.....	24
1.3.6.1.1. Insertar un programa ....	24
1.3.6.1.2. Vista <i>offline/online</i> .....	25

	1.3.6.1.3.	Librerías estándar .....	27
	1.3.6.1.4.	Sistema de ayuda.....	28
	1.3.6.1.5.	Ayuda contextual.....	29
	1.3.6.1.6.	Herramientas de configuración.....	29
1.3.7.		Configuración hardware .....	31
	1.3.7.1.	Configuración hardware y parametrización.....	31
	1.3.7.2.	Insertar un equipo.....	33
	1.3.7.3.	Arrancar HW Config .....	34
	1.3.7.4.	Direccionamiento de los módulos del S7-300.....	35
	1.3.7.4.1.	Direccionamiento libre...37	
	1.3.7.5.	Propiedades de la CPU .....	38
	1.3.7.5.1.	General .....	39
	1.3.7.5.2.	Arranque .....	40
	1.3.7.5.3.	Remanencia .....	42
	1.3.7.5.4.	Protección .....	43
	1.3.7.5.5.	Diagnóstico/reloj.....	45
	1.3.7.5.6.	Comunicación .....	46
	1.3.7.5.7.	Ciclo/marca de ciclo .....	47
1.3.8.		La tabla de símbolos .....	49
	1.3.8.1.	Direccionamiento absoluto y simbólico .....	50
	1.3.8.2.	Editar símbolo .....	51
1.3.9.		Arquitectura de bloques y editor de bloques .....	53
	1.3.9.1.	Tipos de bloques de programa.....	53
	1.3.9.2.	Estructura de programa.....	54
	1.3.9.3.	Imágenes de proceso.....	56

1.3.9.4.	Ejecución cíclica del programa .....	57
1.3.9.5.	Insertar un bloque S7.....	58
1.3.9.6.	Componentes del editor LAD/FBD/STL.....	59
1.3.9.7.	Los lenguajes de programación de STEP 7 .....	60
1.3.9.8.	Llamada a un bloque desde el OB1.....	62
1.3.9.9.	Depurar un programa simple .....	63
1.3.9.10.	Cargar y guardar bloques modificados.....	64
1.3.9.11.	Personalización del editor.....	64
1.3.9.11.1.	Vista.....	65
1.3.9.11.2.	STL.....	66
1.3.9.11.3.	LAD/ FBD .....	67
1.3.9.11.4.	Crear bloque.....	68
1.3.9.11.5.	Fuentes.....	69
1.3.10.	Instrucciones de programación.....	70
1.3.10.1.	Operaciones binarias.....	71
1.3.10.2.	Formato de representación de los números.....	72
1.3.10.2.1.	16 Bits.....	72
1.3.10.2.2.	32 Bits.....	73
1.3.10.3.	Carga y transferencia de datos.....	73
1.3.10.4.	Operaciones de contaje.....	74
1.3.10.5.	Operaciones de temporización .....	75
1.3.10.6.	Operaciones de conversión .....	76
1.3.10.7.	Operaciones de comparación .....	77
1.3.10.8.	Operaciones aritméticas .....	77

	1.3.10.8.1.	Operaciones aritméticas con enteros .....	78
	1.3.10.8.2.	Operaciones aritméticas con números en coma flotante .....	78
	1.3.10.9.	Operaciones de salto.....	79
1.3.11.	Diagnóstico.....		80
	1.3.11.1.	Diagnóstico del sistema.....	81
	1.3.11.2.	Avisos de CPU .....	81
	1.3.11.3.	Información del módulo .....	83
	1.3.11.4.	Observar / forzar variables .....	84
	1.3.11.5.	Datos de referencia .....	86
1.4.	Puesta en marcha del hardware.....		87
1.5.	Convertir archivos de S5 a S7.....		87
1.6.	El PLC S5.....		88
	1.6.1.	Campo de aplicación.....	89
	1.6.2.	Componentes del sistema .....	90
2.	SENSORES Y ACTUADORES .....		91
2.1.	Sensores industriales .....		91
	2.1.1.	Características de los sensores industriales y calificación.....	92
	2.1.1.1.	Clasificación de los sensores industriales según el principio de funcionamiento del elemento sensor ....	93



2.1.1.1.1.	Sensores activos o generadores ( <i>self generating</i> ) .....	93
2.1.1.1.2.	Sensores pasivos o moduladores ( <i>modulating</i> ).....	94
2.1.1.2.	Clasificación de los sensores según el tipo de señal eléctrica que generan.....	95
2.1.1.2.1.	Sensores analógicos ....	95
2.1.1.2.2.	Sensores digitales .....	97
2.1.1.3.	Clasificación de los sensores según el rango de valores .....	100
2.1.1.3.1.	Sensores de medida...	100
2.1.1.3.2.	Sensores todo-nada ...	100
2.1.1.4.	Clasificación de los sistemas sensores según el nivel de integración .....	101
2.1.1.4.1.	Sensores discretos .....	102
2.1.1.4.2.	Sensores integrados...	102
2.1.1.4.3.	Sensores inteligentes .	102
2.1.1.5.	Clasificación de los sensores según la variable física medida .....	105
2.1.2.	Características de entrada de los sensores industriales.....	106
2.1.2.1.	Campo o rango de medida .....	106
2.1.2.2.	Forma de variación de la magnitud de entrada.....	107
2.1.3.	Características eléctricas.....	108

2.1.3.1.	Características eléctricas de salida ....	109
2.1.3.1.1.	Sensores de salida analógica.....	109
2.1.3.1.2.	Sensores de salida digital.....	111
2.1.3.2.	Características de alimentación.....	113
2.1.3.3.	Características de aislamiento .....	113
2.1.4.	Características mecánicas .....	114
2.1.4.1.	Conceptos generales.....	114
2.1.4.2.	Grado de protección ambiental de los sensores industriales .....	114
2.2.	Actuadores .....	117
2.2.1.	Tipos de actuadores.....	117
2.2.2.	Actuadores eléctricos .....	118
2.2.2.1.	Tipos de actuadores eléctricos.....	119
2.2.2.1.1.	Criterio de selección....	120
2.2.3.	Accionamientos continuos.....	120
2.2.3.1.	Motor de características mecánicas rígidas.....	121
2.2.3.2.	Motor de características mecánicas elásticas .....	121
2.2.4.	Accionamientos para movimientos rápidos .....	121
2.2.5.	Elementos de control.....	121
2.2.5.1.	Contactores.....	122
2.2.5.2.	Relés .....	122
2.2.5.3.	Columna de señalización .....	123
2.2.5.4.	Dispositivos variadores de velocidad..	123
2.2.5.5.	Elementos de protección .....	124
2.2.5.5.1.	Disyuntores .....	124

	2.2.5.5.2.	Relé térmico .....	125
	2.2.5.6.	Otros actuadores .....	125
3.	COMUNICACIONES INDUSTRIALES .....		127
3.1.	Comunicaciones digitales .....		127
3.1.1.	Conceptos básicos de las comunicaciones digitales punto a punto.....		128
3.1.1.1.	Normalización de las comunicaciones punto a punto .....		130
	3.1.1.1.1.	Bucle de corriente.....	131
	3.1.1.1.2.	Norma RS-232.....	131
	3.1.1.1.3.	Norma RS-422.....	132
	3.1.1.1.4.	Norma RS-423.....	132
	3.1.1.1.5.	Norma RS-485.....	132
	3.1.1.1.6.	Norma USB .....	133
3.1.2.	Redes informáticas .....		133
3.1.2.1.	Conceptos específicos de las redes de datos.....		134
	3.1.2.1.1.	La topología de la red.	134
	3.1.2.1.2.	La extensión de la red	134
3.1.3.	Subredes en SIMATIC .....		134
	3.1.3.1.	MPI .....	135
	3.1.3.2.	Industrial Ethernet.....	135
	3.1.3.3.	PROFIBUS .....	136
	3.1.3.4.	Enlace punto a punto.....	136
	3.1.3.5.	AS Interfase .....	136
3.1.4.	Métodos de comunicación S7 .....		138
	3.1.4.1.	Datos globales .....	138
	3.1.4.2.	Comunicación básica.....	139

	3.1.4.3.	Comunicación ampliada .....	139
	3.1.5.	Conector de bus .....	140
3.2.		Familia de redes de campo PROFIBUS .....	141
	3.2.1.	Características generales.....	141
		3.2.1.1. Red PROFIBUS-DP .....	142
		3.2.1.2. Red PROFIBUS-PA.....	143
		3.2.1.3. Red PROFIBUS-FMS.....	143
	3.2.3.	Comparación de las redes PROFIBUS y AS-i.....	144
3.3.		Enlaces entre los participantes de la comunicación.....	144
4.		IMPLEMENTACIÓN DE MIGRACIÓN DE CONTROLADOR LÓGICO PROGRAMABLE S5 A S7, DE UNA MÁQUINA LLENADORA DE BEBIDAS .....	147
	4.1.	Pasos de la migración .....	148
		4.1.1. Cómo reemplazar la CPU S5 .....	148
		4.1.1.1. Módulos de interfaz en el bastidor principal .....	149
		4.1.2. Módulos de entradas y salidas .....	149
		4.1.3. Convertir un programa.....	150
		4.1.3.1. Preparación de archivo S5 .....	150
		4.1.3.2. Generar referencias cruzadas .....	150
		4.1.3.3. Utilizando la herramienta de conversión de archivos S5 a S7 .....	155
		4.1.4. Crear un nuevo proyecto de S7.....	160
		4.1.5. Configuración de hardware .....	163
		4.1.5.1. Direccionamiento de señales .....	168
		4.1.6. Generar bloques de programa a partir de archivo convertido.....	174
		4.1.7. Recablear el programa.....	179

4.1.8.	Agregar los bloques de organización para el óptimo funcionamiento.....	182
4.1.9.	Corrección de errores de programa.....	184
4.1.9.1.	Corrección de errores durante la migración.....	184
4.2.	Pruebas de migración.....	190
CONCLUSIONES .....		195
RECOMENDACIONES.....		197
BIBLIOGRAFÍA.....		199
ANEXOS.....		201



# ÍNDICE DE ILUSTRACIONES

## FIGURAS

1.	Posicionamiento del PLC .....	1
2.	Visión general de SIMATIC .....	2
3.	Características del PLC S7-200 .....	4
4.	S7-200: módulos .....	5
5.	S7-200: diseño de la CPU .....	7
6.	El PLC S7-300 .....	8
7.	S7-300: módulos .....	9
8.	S7-300: diseño de la CPU .....	11
9.	Características del PLC S7-400 .....	13
10.	S7-400: módulos .....	14
11.	S7-400: diseño de la CPU .....	16
12.	Dispositivos de programación .....	18
13.	Del proceso al proyecto .....	21
14.	Estructura del proyecto de STEP 7 .....	22
15.	Creación de un proyecto de STEP 7 .....	24
16.	Insertar un programa de STEP 7.....	25
17.	Vista <i>offline/online</i> en el administrador SIMATIC .....	26
18.	Librería estándar en el administrador SIMATIC .....	27
19.	Sistema de ayuda de STEP 7 .....	29
20.	Herramientas de configuración del administrador SIMATIC .....	30
21.	Insertar un quipo .....	33
22.	Arrancar HW Config .....	34
23.	Direccionamiento de los módulos del S7-300 .....	37

24.	Direccionamiento Libre .....	38
25.	Propiedades de la CPU .....	39
26.	Propiedades de la CPU: general .....	40
27.	Propiedades de la CPU: arranque .....	41
28.	Propiedades de la CPU: remanencia .....	42
29.	Propiedades de la CPU: protección .....	43
30.	Propiedades de la CPU: diagnóstico/reloj .....	45
31.	Propiedades de la CPU: comunicación .....	46
32.	Propiedades de la CPU: ciclo/marca de ciclo .....	48
33.	La tabla de símbolos .....	49
34.	Editar símbolo (en el editor LAD/FBD/STL) .....	52
35.	Tipos de bloques .....	53
36.	Estructura de programa .....	55
37.	Imágenes de proceso .....	57
38.	Insertar un bloque S7.....	58
39.	Componentes del editor LAD/FBD/STL .....	59
40.	Selección del lenguaje de programación .....	61
41.	Llamada a un bloque desde el OB1 .....	62
42.	Depurar un programa simple .....	63
43.	Personalización del editor: vista .....	65
44.	Personalización del editor: STL .....	66
45.	Personalización del editor: LAD/ FBD .....	67
46.	Personalización del editor: bloque .....	69
47.	Personalización del editor: fuentes .....	70
48.	Visualización de avisos de la CPU .....	82
49.	Información del módulo .....	84
50.	Observar / forzar variables .....	85
51.	Datos de referencia .....	86
52.	Componentes básicos de un sensor .....	92



53.	Diagrama de bloques de un sensor analógico .....	96
54.	Señales analógicas: a) Señoidal de frecuencia constante; b) Continua .....	96
55.	Señales analógicas: a) Unipolar; b) Bipolar .....	97
56.	Sensor optoelectrónico láser de proximidad 3RG70-1CM00 de Siemens .....	97
57.	Señal digital binaria en formato paralelo .....	99
58.	Señal digital binaria en formato serie .....	99
59.	Diagrama de bloques de un sensor digital con salida en paralelo .....	99
60.	Esquema de bloques típicos de un sensor digital cuyo elemento sensor proporciona una señal analógica .....	100
61.	Esquema de bloque de un sensor todo-nada cuyo elemento sensor proporciona una señal analógica .....	101
62.	Diagrama de bloques de un sensor inteligente .....	103
63.	Conexión de sensores discretos y/o integrados a un autómata programable .....	104
64.	Conexión de un conjunto de sensores inteligentes a un autómata programable mediante un bus de campo .....	104
65.	Sensores optoelectrónico de proximidad BERO 3RG673 01-1RMOO conectable al bus de campo AS-Interface .....	105
66.	Sensor de caudal MG711/A conectable al bus de campo Hart .....	105
67.	Formas de variación a lo largo del tiempo de la magnitud física que debe medir un sensor .....	108
68.	Esquema de bloques típico de un sensor analógico de salida por corriente .....	111
69.	Sensor ultrasónico de proximidad del tipo 3RG61 13-3BF01, que posee una salida analógica por corriente de 4 a 20mA .....	111

70.	Etapas de salida de sensores digitales: a) con transistores NPN y resistencia de carga; b) con transistor NPN en montaje de colector abierto .....	113
71.	Actuadores en una máquina llenadora de bebidas .....	119
72.	Contactador SIRIUS 3R IEC a) y b) Relé SIRIUS 3RH de Siemens .....	122
73.	Columna de señalización 8WD42 de Siemens .....	123
74.	Dispositivo variador de velocidad .....	124
75.	Guarda motor 3RV101 de Siemens .....	124
76.	Relé térmico bimetálico SIRUS 3RU11 de Siemens .....	125
77.	Esquema básico de la comunicación punto a punto en banda base digital entre dos computadores .....	129
78.	Subredes en SIMATIC .....	135
79.	Ejemplo real de la red AS-i .....	138
80.	Conector de bus .....	140
81.	Utilización de la familia de protocolos PROFIBUS .....	142
82.	Enlaces entre los participantes de la comunicación .....	145
83.	Archivo S5.....	150
84.	Abrir STEP 5.....	151
85.	Abrir Set F4.....	151
86.	Abrir el controlador .....	152
87.	Abrir <i>Program file</i> .....	152
88.	Selección de archivo de respaldo LLENA@ST.S5D .....	153
89.	Seleccionar Blocks .....	153
90.	Archivo generado LLENA@XR .....	154
91.	Opción generar referencia cruzada .....	154
92.	Salir de STEP 5 .....	155
93.	Abrir la herramienta convertir archivo S5 a S7.....	156
94.	Abrir el archivo a convertir .....	156
95.	Abrir el archivo LLENA@ST.S5D.....	157

96.	Convirtiendo comparación de bloques S5 a S7.....	158
97.	Errores y advertencias de conversión terminada S5 a S7.....	159
98.	Archivos creados con la herramienta de migración S5 a S7 .....	160
99.	Abrir Administrador SIMATIC S7.....	160
100.	Crear proyecto nuevo .....	161
101.	Elegir ruta y nombre del proyecto .....	161
102.	El nombre del proyecto creado .....	162
103.	Insertar equipo SIMATIC 300.....	162
104.	Creación del equipo .....	163
105.	Posición de módulos en el <i>rack</i> .....	163
106.	Abrir la configuración hardware .....	163
107.	Configuración hardware .....	164
108.	Insertar rack .....	164
109.	Insertar el suministro de energía en <i>rack</i> .....	165
110.	Elegir CPU .....	166
111.	Insertar la CPU en <i>rack</i> .....	167
112.	La CPU en el rack .....	168
113.	Abrir I/QF <i>List</i> .....	169
114.	Chequear I/Q/F List .....	169
115.	Buscar entradas digitales en el programa S5.....	170
116.	Buscar salidas digitales en el programa S5 .....	170
117.	Módulos de entradas y salidas digitales del S5 al S7.....	171
118.	Elegir módulos de entradas y salidas digitales .....	172
119.	Asignación de direcciones de señales digitales .....	173
120.	Guardar y compilar la configuración hardware .....	173
121.	Resultado de la configuración hardware .....	174
122.	Abrir fuente externa .....	174
123.	Insertar fuente externa .....	175
124.	Archivo insertado .....	175

125.	Abrir librería estándar .....	176
126.	Elegir <i>Standard Library</i> .....	176
127.	Copiar bloques .....	177
128.	Pegar bloques .....	177
129.	Compilar archivo insertado .....	178
130.	Bloques generados .....	179
131.	Abrir recableado .....	180
132.	Escribir direcciones viejas y nuevas .....	180
133.	Función de recableado terminado .....	181
134.	Generación de bloc de notas .....	181
135.	Abrir bloque de organización .....	182
136.	Escribir el nombre del bloque de organización .....	183
137.	Bloques de organización insertados .....	183
138.	Código del OB 3.....	184
139.	Parámetros en el DB1 del S5.....	185
140.	Entradas y salidas digitales para el contador en S7 .....	186
141.	Abrir OB 40 .....	186
142.	Copiar código del OB 3.....	187
143.	Pegar código del OB 3 al OB 40.....	188
144.	Acondicionamiento del código del OB 40.....	189
145.	El OB 31 y OB 251 dentro del archivo fuente .....	190
146.	Pruebas de migración de entradas y salidas digitales forzando variables .....	191
147.	Pruebas de migración activación de entradas y salidas digitales en módulos .....	191
148.	Pruebas de migración de los actuadores .....	192
149.	Pruebas de migración de entradas digital .....	193
150.	El tablero antes de la implementación de migración .....	193
151.	El tablero después de la implementación de migración .....	194

152.	Implementación de migración lavado de garrafones .....	194
153.	Implementación de migración llenado de garrafones .....	194

## **TABLAS**

I.	Clasificación de los sensores .....	93
II.	Clasificación de los sensores según el principio de funcionamiento ...	94
III.	Clasificación de los sensores según el tipo de variable física medida . .....	106
IV.	Tabla que indica las variables físicas que se pueden medir con sensores basados en distintos principios de funcionamiento .....	107
V.	Significado de las cifras del grado de protección ambiental de acuerdo con la Norma IEC 144. ....	115
VI.	Características de los actuadores: neumático, hidráulico y eléctrico.	118



## LISTA DE SÍMBOLOS

<b>Símbolo</b>	<b>Significado</b>
<b>Hz</b>	Hertz
<b>kbps</b>	Kilobit por segundos
<b>Mbps</b>	Megabits por segundos
<b><math>\mu\text{V}/^\circ\text{C}</math></b>	Microvoltio/grados celsius
<b>mA</b>	Miliamperio
<b>mV/g</b>	Milivoltio/gramo
<b>EM</b>	Módulos de Expansión
<b>FM</b>	Módulos de Funciones
<b>IM</b>	Módulos de Interfase
<b>SM</b>	Módulos de señal
<b>NO</b>	Normalmente Abierto
<b>NC</b>	Normalmente Cerrado
<b>S5</b>	Paso 5 ( <i>Step 5</i> )
<b>S7</b>	Paso 7 ( <i>Step 7</i> )
<b>V</b>	Voltio





## GLOSARIO

<b>Actuador</b>	Es todo dispositivo que convierte una magnitud eléctrica en una salida, generalmente mecánica, que puede provocar un efecto sobre el proceso automatizado. Algunos actuadores realizan la función de cambiar niveles de energía eléctrica.
<b>ASCII</b>	Es un código de caracteres basado en el alfabeto latino.
<b>CALL</b>	La instrucción para llamar a un bloque.
<b>Ciclo de Scan</b>	La CPU lee las entradas, ejecuta el programa, escribe en las salidas y ejecuta funciones del sistema y de comunicación en un ciclo continuo.
<b>DC5V</b>	Indicador de tensión interna de 5 V DC.
<b><i>Fielbus</i></b>	Es un protocolo de comunicaciones digital de alta velocidad creada para remplazar la clásica señal de 4-20 mA que aún se utiliza en muchos de los sistemas DCS (Sistema de Control Distribuido) y PLC (Controladores Lógicos Programables), instrumentos de medida y transmisión y válvulas de control.

<b>FRCE</b>	<i>FORCE</i> ; indica que, al menos, una entrada o salida está forzada.
<b>HART</b>	Es un bus sensor-actuador de dispositivos inteligentes.
<b>IEC</b>	Es una organización de normalización en los campos eléctrico, electrónico y tecnologías relacionadas.
<b>IEEE</b>	Una asociación técnico-profesional mundial dedicada a la estandarización, entre otras cosas.
<b>Interfase DP</b>	Interfase para la conexión directa de periferia descentralizada a la CPU.
<b>MES</b>	Es un software especialmente destinado a facilitar al máximo los procesos de fabricación en todas sus facetas.
<b>MRES</b>	Función de <i>reset</i> de memoria ( <i>Module Reset</i> ).
<b>Norma DIN</b>	Las normas DIN son los estándares técnicos para el aseguramiento de la calidad en productos industriales y científicos en Alemania.
<b><i>Profibus</i></b>	Es un conjunto de redes de control (o bus de campo) diseñado para resolver las necesidades de comunicación de los procesos industriales tanto discretos (denominados procesos de fabricación)

como continuos y otros procesos distribuidos, como por ejemplo la automatización de edificios.

**RUN** Ejecución del programa, es posible el acceso sólo lectura desde la PG.

**RUN-P** Ejecución del programa, es posible el acceso lectura/escritura desde la PG.

**SCADA** Es un software para ordenadores que permite controlar y supervisar procesos industriales a distancia.

**Sensores Industriales** Los sistemas sensores adecuadamente construidos para trabajar en las condiciones existentes en un entorno industrial (temperatura elevada, presencia de polvo, humedad relativa alta, etc.).

**STOP** Modo *Stop*; el programa no se ejecuta.

**WinCC flexible** Es el innovador software HMI ejecutable en Windows para todas las aplicaciones a pie de máquina en el ámbito de la construcción de maquinaria, maquinaria de serie e instalaciones.



## RESUMEN

Hoy en día se puede encontrar controladores lógicos programables en todas las ramas de la industria, para tareas de automatización de toda orden y tamaño. Debe ser posible iniciar, controlar y supervisar el funcionamiento de cada máquina o proceso. Se tiene nuevas tecnologías de controlador lógico programable como el S7-300 y S7-400 el cual viene a sustituir prácticamente al S5 u otras marcas. En el caso del S5 su ciclo de vida ha llegado a su fin por lo tanto no se cuenta con repuestos y estos deben ser migrados a S7.

En el primer capítulo se detallan todas las características y diseño de la CPU para los controladores lógico programable S7-200, S7-300 y S7-400. También se describe el potente software Administrador SIMATIC S7 y su programación por medio de los tres lenguajes KOP, FUP y AWL. Dentro de las herramientas adicionales que ofrece este software están el convertidor de archivos S5 a S7. Se dan pequeños detalles del S5 con el fin de comprender sus campos de aplicación a migrar. Los sensores y actuadores se describen en el capítulo dos, estos sensores se ven desde el área industrial puesto que requieren un nivel de protección bajo condiciones ambientales de polvo, altas temperaturas, humedad, entre otros y también se mencionan algunos sensores que pueden ser aplicados de acuerdo a las características requeridas. En el capítulo tres se describe las comunicaciones industriales como por ejemplo la interfase punto a punto y multipunto, Industrial Ethernet, PROFIBUS y otros. En el último capítulo se describen los pasos necesarios para llevar a cabo la implementación de migración de archivos S5 a S7 y las pruebas realizadas para verificar el correcto funcionamiento de todo el equipo eléctrico del tablero a migrar.



# OBJETIVOS

## General

Implementar la migración de controlador lógico programable S5 a S7, de una máquina llenadora de bebidas.

## Específicos

1. Presentar los fundamentos de controladores logico programable S5 y S7.
2. Mostrar los sensores y actuadores utilizados en la industria.
3. Enseñar los principios de comunicaciones industriales.
4. Implementar la migración de convertidor de archivos S5 a S7, realizar pruebas y verificar el correcto funcionamiento del ciclo de todo el proceso.





## INTRODUCCIÓN

Actualmente las empresas embotelladoras de bebidas tienen automatizada sus máquinas embotelladoras con el controlador lógico programable (PLC) S5 de Siemens u otras marcas. Sin embargo, el controlador lógico programable S5 ha dejado de existir en el mercado y no se encuentran repuestos después de diez años, por lo tanto si este controlador lógico programable falla toda la maquinaria de la embotelladora pararía su producción.

Migrando el controlador lógico programable S5 a S7 se obtiene una solución, pues actualmente se encuentra en el mercado el S7, con esto se incluye la migración de algunos o todos los equipos eléctricos. Dentro de las herramientas opcionales del software Administrador SIMATIC S7 se tiene el convertidor de archivos S5, el cual se utiliza para convertir el programa original S5 a S7, obteniendo así un nuevo programa que puede ser visualizado por medio del lenguaje de programación AWL dentro del Administrador SIMATIC S7.

Se detalla qué es un S7-300 y un S7-400 para poder comprender esta tecnología, se muestra el potente software Administrador SIMATIC S7 para aprender a utilizarla, una breve descripción del S5 nos ayuda a comprender el campo de aplicación de este. Se estudia los sensores y actuadores para comprender las partes de una máquina llenadora y por último se presenta algunas comunicaciones industriales con el fin de poder conectar con el controlador lógico programable para poder descargar el programa a la CPU de este.

La migración de PLC se facilita conociendo la capacidad del mismo y la de los módulos de señal, esto se busca en de unas tablas proporcionadas en este trabajo de graduación para poder tener con exactitud el controlador lógico programable deseado que puede ser un S7-300 o un S7-400, así como los nuevos módulos de señal requeridos.

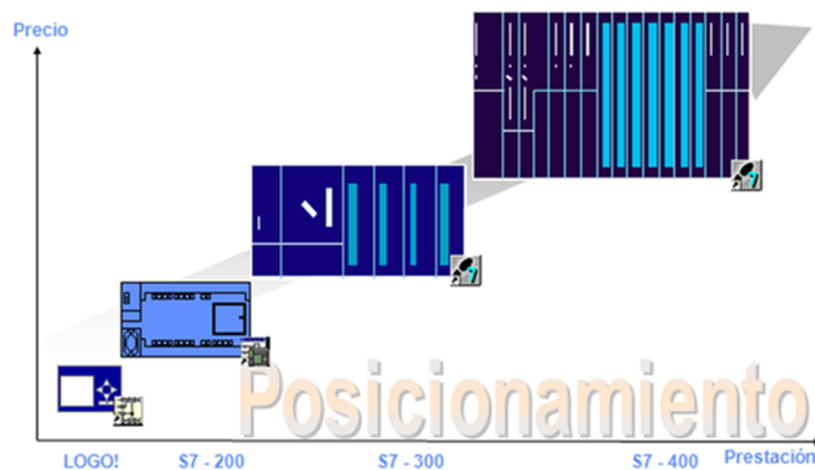
Se realizan pruebas en el taller del panel nuevo con el fin de verificar que todo el equipo esté energizado y designado adecuadamente, esto se realiza también en la parte de la línea de producción de bebidas, en este caso en el llenado de garrafrones, a través del software Administrador Simatic S7.

# 1. LOS CONTROLADORES LÓGICO PROGRAMABLE SIMATIC S7 Y SOFTWARE STEP7

## 1.1. ¿Qué es un PLC?

Los PLC (*Programmable Logic Controller*, Controlador Lógico Programable) surgieron en la producción automotriz y se extendieron luego en todos los ramos de la industria. Al comienzo se limitaron a grandes máquinas, ya que para las tareas de automatización pequeñas la solución electromecánica siempre era bastante mas económica. Eso cambió con el desarrollo de pequeños controladores electrónicos de precio mas accesibles. Hoy en día se pueden encontrar PLCs en todas las ramas de la industria, para tareas de automatización de todo orden y tamaño, como se muestra en la figura 1.

Figura 1. Posicionamiento del PLC

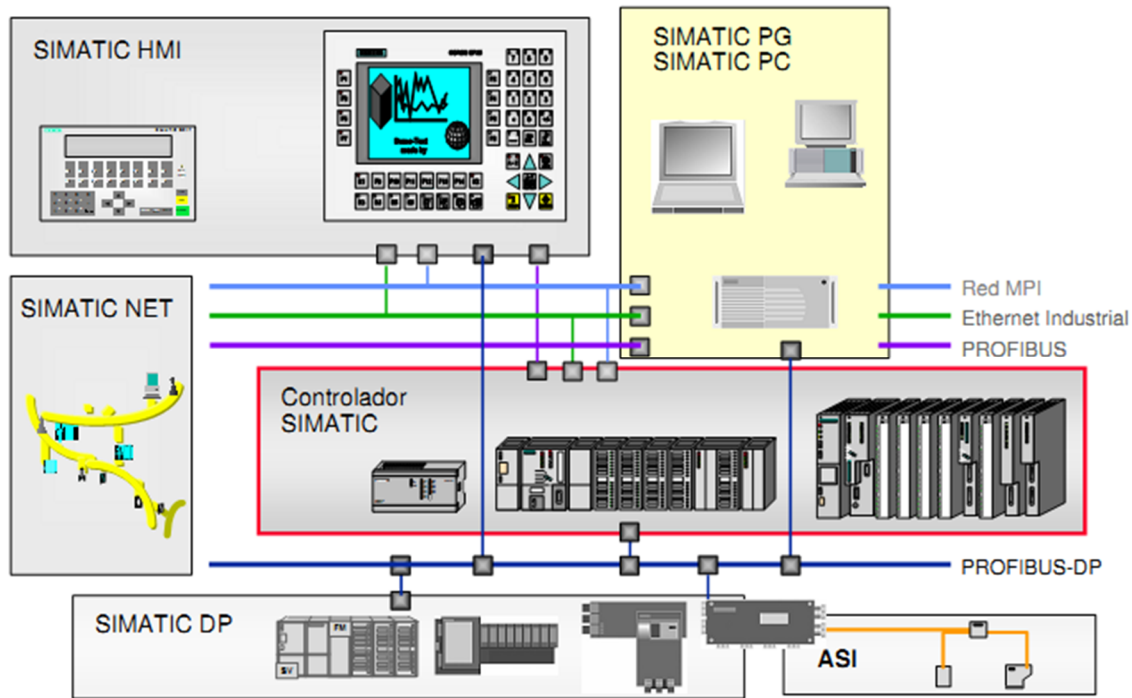


Fuente: SIEMENS, S. A.

## 1.2. Visión general de SIMATIC

La siguiente figura muestra la interconexión por medio de comunicación; red MPI, Ethernet Industrial y PROFIBUS que ayudan a tener comunicación con diferentes equipos como por ejemplo; sensores, actuadores, controladores etc.

Figura 2. Visión general de SIMATIC



Fuente: SIEMENS, S. A.

Introducción: la introducción de la electrónica ha tenido como resultado grandes cambios en la ingeniería de control industrial. Junto con la maquinaria automatizada, cuyas posibilidades de aplicación se han ampliado con los controles electrónicos, estos cambios han llevado también a nuevas tecnologías y ramas.

Controladores: además del suministro de energía, los elementos de control se requieren para el mando de máquinas y procesos en casi todas las áreas de fabricación. Debe ser posible iniciar, controlar y supervisar el funcionamiento de cada máquina o proceso.

En el pasado, las tareas de control se resolvían con la tecnología de control convencional, por cableado de contactores y relés de forma individualizada, es decir, dependiendo de la tarea. Hoy los controladores lógicos programables son ampliamente usados para resolver tareas de automatización.

Automatización integrada totalmente: para que las compañías sigan siendo competitivas, no es suficiente automatizar de forma aislada equipos o máquinas para un proceso individual. La demanda de mayor flexibilidad con mayor productividad solo se puede llevar a cabo cuando las máquinas individuales están integradas en el sistema completo.

El flujo de información entre todos los componentes es esencial para el funcionamiento del sistema completo.

Los procesos de producción ya no se ven como procesos parciales individuales, sino como componentes integrantes de un proceso de producción completo.

Además, el proceso ya no está estructurado de forma centralizada en modo jerárquico. Ahora el proceso se estructura y se distribuye en elementos individuales autónomos.

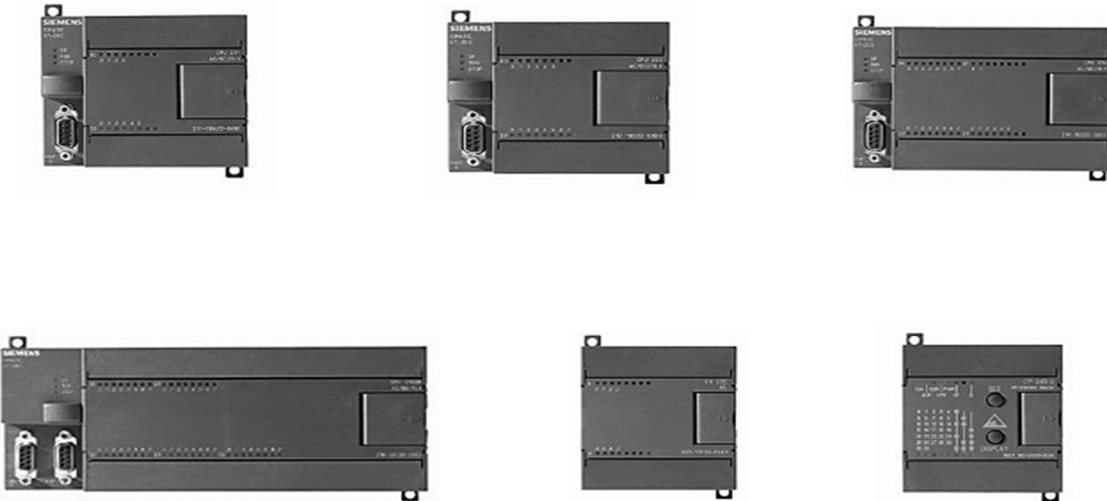
La integración total del entorno de automatización completo se hace posible hoy con la ayuda de:

- Programación y configuración común de los sistemas parciales individuales.
- Manejo de datos común.
- Comunicación común entre todos los componentes de automatización participantes.

### 1.2.1. EI PLC S7-200

La figura siguiente muestra las características del PLC S7-200 en cuanto a su actualización, estos PLC tienen diferentes características, por ejemplo; uno o dos interfases, el número de entradas y salidas, etc.

Figura 3. Características del PLC S7-200

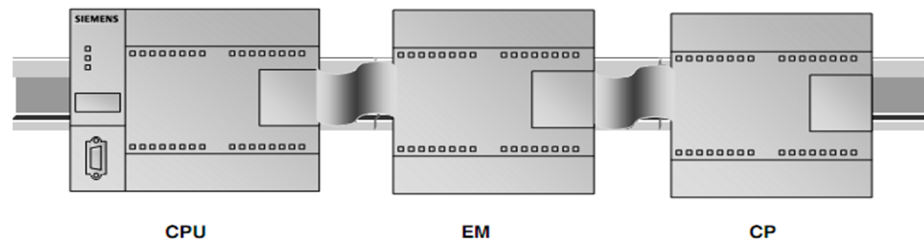


Fuente: SIEMENS, S. A.

### Características:

- Sistema de control pequeño y modular para el rango de prestaciones más bajo.
- CPUs con rango de prestaciones escalonado
- Extensa selección de módulos
- Ampliable hasta con 7 módulos
- Bus integrado en los módulos
- Puede conectarse en red con:
  - La interfase de comunicación RS 485
  - PROFIBUS
- Conexión de PG central con acceso a todos los módulos
- Sin restricciones de slot
- Software propio
- "Paquete completo" con fuente de alimentación, CPU, E/S en una sola unidad.
- "Micro PLC" con funciones integradas

Figura 4. **S7-200: módulos**



Fuente: SIEMENS, S. A.

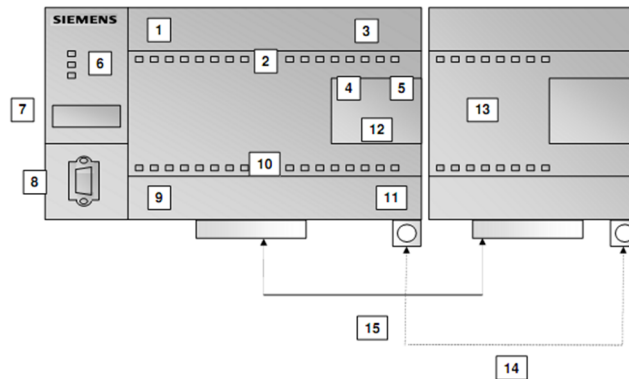
### Módulos de Expansión (EM):

- Módulos de entradas digitales:
  - 24V DC
  - 120/230V AC
- Módulos de salidas digitales:
  - 24V DC
  - Relés
- Módulos de entradas analógicas:
  - Tensión
  - Corriente
  - Resistencia
  - Termopares
- Módulos de salidas analógicas:
  - Tensión
  - Corriente

Procesadores de Comunicación (CP): la CP 243-2 se puede usar para conectar el S7-200 como maestro de una interfase AS-i. Como resultado, hasta 596 elementos binarios pueden controlarse a través de 62 esclavos AS-interfase. Esto aumenta significativamente el número de entradas y salidas para el S7-200. Accesorios: conector de bus.



Figura 5. **S7-200: diseño de la CPU**



Fuente: SIEMENS, S. A.

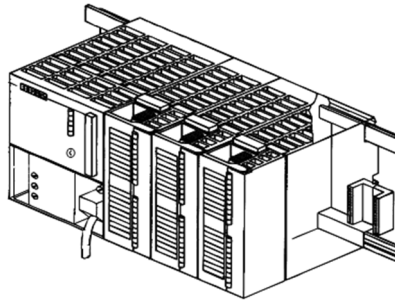
Partes del diseño de la CPU S7-200:

1. Salidas digitales integradas
2. LEDs de estado de las salidas digitales
3. Terminales de alimentación
4. Conmutador *Stop/Run*
5. Conector para el cable de ampliación
6. LEDs de estado de la CPU
7. Ranura para el cartucho de memoria
8. Puerto de comunicaciones (p. ej. PPI)
9. Entradas digitales integradas
10. LEDs de estado de las entradas digitales
11. Fuente de alimentación integrada
12. Potenciómetros integrados
13. Módulo de ampliación
14. Fijadores para tornillo (DIN métrica M4, diámetro 5 mm)
15. Pestaña de fijación

### 1.2.2. EI PLC S7-300

La figura siguiente muestra las características del PLC S7-300, su diseño es diferente a una S7-200, entre las diferencias se encuentran entradas y salidas en módulos, mayor capacidad, mayor velocidad, etc.

Figura 6. **Características del PLC S7-300**



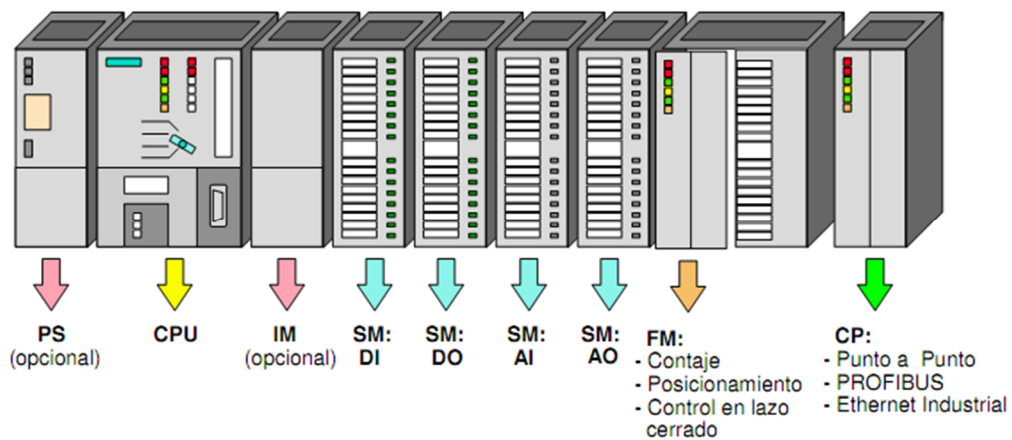
Fuente: SIEMENS, S. A.

Características:

- Sistema de control modular pequeño para el rango de prestaciones bajo/medio.
- Rango de prestaciones de las CPUs escalonado.
- Extensa selección de módulos.
- Ampliable hasta con 32 módulos.
- Bus trasero integrado en los módulos.
- Puede montarse en red:
  - Interfase multipunto (MPI)
  - PROFIBUS
  - Industrial Ethernet

- Conexión central de una PG con acceso a todos los módulos.
- Sin restricciones de slot.
- Configuración y parametrización con ayuda de la herramienta "HWConfig".

Figura 7. S7-300: módulos



Fuente: SIEMENS, S. A.

#### Módulos de Señal (SM):

- Módulos de entradas digitales:
  - 24V DC
  - 120/230V AC
- Módulos de salidas digitales:
  - 24V DC
  - Relé
- Módulos de entradas analógicas:
  - Tensión
  - Corriente

- Resistencia
- Termopares
- Módulos de salidas analógicas:
  - Tensión
  - Corriente

Módulos de Interfase (IM): la IM360/IM361 y la IM365 hacen posible configuraciones multifila. Enlazan el bus a través de una fila a la siguiente.

Módulos *Dummy* (DM): el módulo *dummy* DM 370 reserva un *slot* para un módulo de señal cuyos parámetros aún no han sido asignados. También puede usarse, por ejemplo, para reservar un slot para la instalación de un módulo interfase en el futuro.

Módulos de Función (FM): realizan “funciones especiales”:

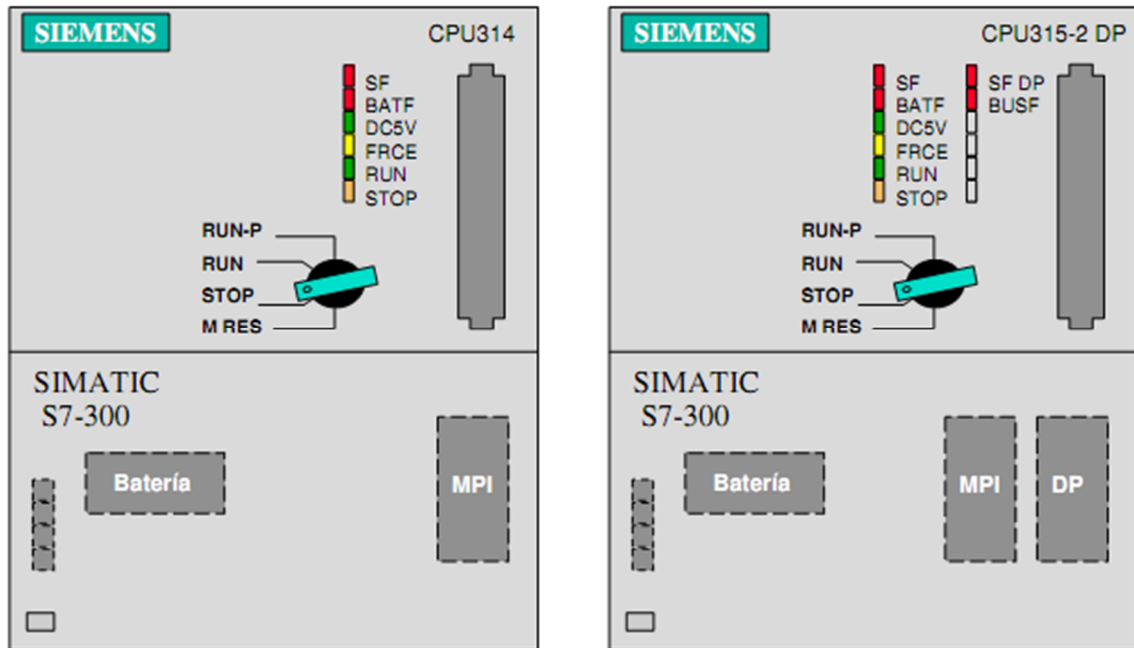
- Contaje
- Posicionamiento
- Control en lazo cerrado

Procesadores de Comunicación (CP): proporciona las siguientes posibilidades de montaje en red:

- Enlaces punto a punto
- PROFIBUS
- Industrial Ethernet

Accesorios: conectores de bus y conectores frontales.

Figura 8. S7-300: diseño de la CPU



Fuente: SIEMENS, S. A.

Selector de modo:

- MRES = función de reset de memoria (Module Reset).
- STOP = modo stop, el programa no se ejecuta.
- RUN = ejecución del programa, es posible el acceso sólo lectura desde la PG.
- RUN-P = ejecución del programa, es posible el acceso lectura/escritura desde la PG.

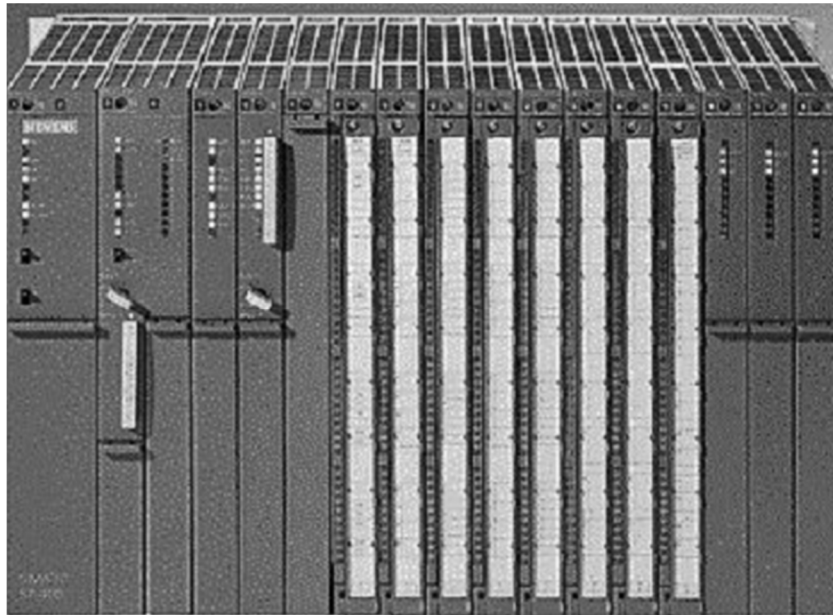
### Indicadores de estado (LEDs):

- SF = error de grupo; fallo interno de la CPU o fallo en un módulo con capacidad de diagnóstico.
- BATF = fallo de batería; Batería vacía o no presente.
- DC5V = indicador de tensión interna de 5 V DC.
- FRCE = FORCE; indica que, al menos, una entrada o salida está forzada.
- RUN = parpadea cuando la CPU está arrancando, luce fija en modo RUN.
- STOP = muestra una luz fija en modo STOP. Parpadea lentamente durante una petición de reset de memoria. Parpadea rápidamente cuando se está efectuando un reset de Memoria. Parpadea lentamente cuando se necesita un reset porque se ha insertado una memory card.
- Memory Card: existe un slot para una memory card. La memory card salva el contenido del programa en caso de caída de alimentación sin necesidad de batería.
- Compartimento de la batería: existe un receptáculo para una batería de litio bajo la cubierta. La batería proporciona energía de respaldo para salvar los contenidos de la RAM ante una supuesta caída de alimentación.
- Conexión MPI: conexión para un dispositivo de programación u otro dispositivo con interfase MPI.
- Interfase DP: interfase para la conexión directa de periferia descentralizada a la CPU.

### 1.2.3. EI PLC S7-400

La figura siguiente muestra las características del PLC S7-400, al igual que el PLC S7-300 sus entradas y salidas se encuentran en módulos, este es un PLC más potente en cuanto a capacidad, velocidad, etc.

Figura 9. **Características del PLC S7-400**



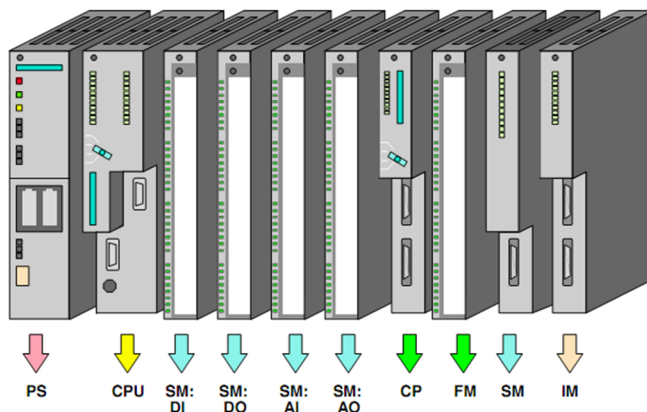
Fuente: SIEMENS, S. A.

Características:

- El PLC Potente para el grado de prestaciones medio/alto.
- Rango de prestaciones de CPUs escalonado.
- Extensa selección de módulos.
- Puede ampliarse hasta 300 módulos.
- Bus trasero integrado en los módulos.

- Puede montarse en red:
  - Interfase multipunto (MPI)
  - PROFIBUS
  - Industrial Ethernet
- Conexión central de PG con acceso a todos los módulos.
- Sin restricciones de slot.
- Configuración y parametrización con ayuda de la herramienta "HWConfig".
- Multiprocesador (hasta 4 CPUs se pueden usar en un bastidor central).

Figura 10. **S7-400: módulos**



Fuente: SIEMENS, S. A.

#### Módulos de Señal (SM):

- Módulos de entradas digitales:
  - 24V DC
  - 120/230V AC
- Módulos de salidas digitales:



- 24V DC
- Relé
- Módulos de entradas analógicas:
  - Tensión
  - Corriente
  - Resistencia
  - Termopares
- Módulos de salidas analógicas:
  - Tensión
  - corriente

Módulos de Interfase (IM): los módulos de interfase IM460, IM461, IM463, IM467 proporcionan la conexión entre varios bastidores:

- UR1 (Universal Rack) con hasta 18 módulos
- UR2 (Universal Rack) con hasta 9 módulos
- ER1 (Extensión Rack) con hasta 18 módulos
- ER2 (Extensión Rack) con hasta 9 módulos

Módulos de Función (FM): realizan "funciones especiales":

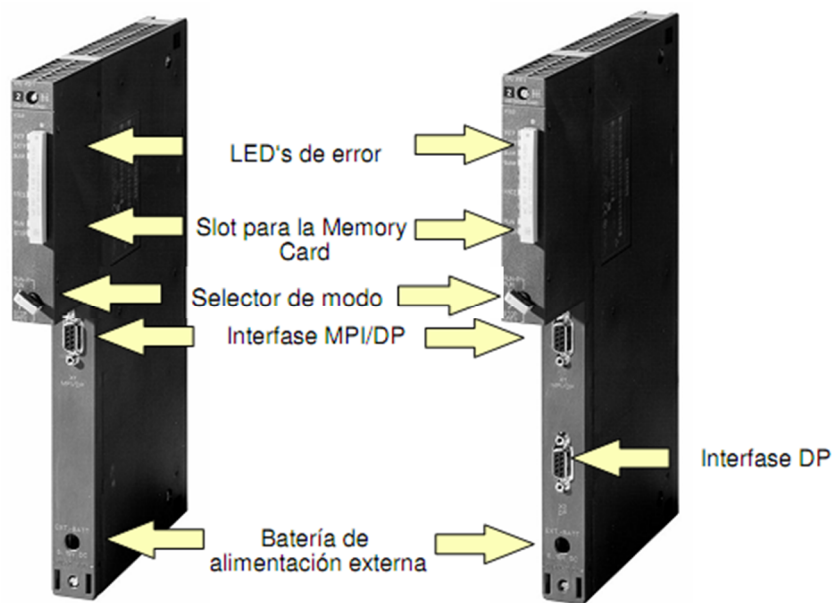
- Contaje
- Posicionamiento
- Control en lazo cerrado

Procesadores de Comunicación (CP): proporcionan las siguientes capacidades de comunicación:

- Enlaces punto a punto

- PROFIBUS
- Industrial Ethernet

Figura 11. S7-400: diseño de la CPU



Fuente: SIEMENS, S. A.

EXT-BATT: suministro de tensión con batería externa adicional (DC 5...15V para salvaguardar la RAM, por ejemplo, cuando se reemplaza la fuente de alimentación).

Conexión MPI/DP: para el dispositivo de programación u otro dispositivo con interfase MPI.

Slot para *Memory Cards*: en el S7-400 CPUs se puede, dependiendo de los requerimientos, insertar tarjetas RAM o Flash EPROM como memoria de carga externa:

- Tarjetas RAM con una capacidad de: 64 kBytes, 256 kBytes, 1 MBytes y 2 MBytes. Los contenidos se salvaguardan a través de la batería de la CPU.
- Tarjetas Flash EPROM con una capacidad de: 64 kBytes, 256 kBytes, 1 MBytes, 2 MBytes, 4 MBytes, 8 MBytes y 16 MBytes. Los contenidos se salvaguardan en las EEPROMs integradas.

Selector de modo:

- MRES = Función de reset de memoria (Module Reset)
- STOP = modo STOP, es decir, no se ejecuta el programa y las salidas están deshabilitadas (modo "OD"= Output Disabled).
- RUN = Ejecución del programa, acceso sólo lectura desde la PG.
- RUN-P = Ejecución del programa, acceso lectura/escritura desde la PG.

### **1.3. Software STEP 7**

El Software STEP 7 como herramienta principal Administrador SIMATIC, es el programa por medio de la cual es posible automatizar de forma organizada y con facilidades de programación por medio del lenguaje KOP, FUP Y AWL.

#### **1.3.1. Dispositivos de programación**

*FIELD PG*: es un dispositivo de programación estándar de la industria, potente y fácil de usar, especialmente para mantenimiento y servicio, pero también para programación y configuración. La herramienta ideal para aplicaciones de planta.

Figura 12. **Dispositivos de programación**

**Power PG**



**Field PG**



Fuente: SIEMENS, S. A.

Características:

- Dimensiones de ordenador portátil
- Fuente de alimentación independiente
- Hardware potente
- Equipado con todos los puertos de interfase SIMATIC necesarios

*POWER PG*: es un dispositivo de programación portátil, ideal para todas las aplicaciones en un proyecto de automatización, además de ser un PC muy potente estándar en la industria.

Características:

- Prestaciones del sistema de alto nivel
- Excelentes posibilidades de ampliación
- *Display* color
- Diseño muy resistente

- Equipado con todos los puertos de interfase SIMATIC necesarios.

### **1.3.2. Requerimiento de la PG/PC para la instalación del Software STEP 7**

Sistema operativo: Microsoft Windows 2000 Professional o Microsoft Windows XP Professional.

Memoria en la unidad de disco duro: dependiendo de la instalación, entre 400 MB y 800 MB.

Interfases:

- CP5611
- CP5512
- Adaptador de PC (USB)
- Industrial Ethernet
- Interfase de programación para tarjeta de memoria (opcional)

Las nuevas PG de la serie SIMATIC S7 proporcionan las condiciones óptimas para el Software STEP 7. Se puede instalar una tarjeta MPI (interfase multipuntos) en los PC que cumplan los requerimientos que se listan más arriba, o se puede conectar al puerto serie COM del PC con un adaptador de PC.

### **1.3.3. Instalación del Software STEP 7**

Instalación:

- Ejecutar "Setup.exe"
- Elegir las opciones
- Elegir el idioma
- Introducir el disco de autorización cuando se pida
- Reiniciar cuando se pida

Como en el STEP 7 V4.0, el software solo está disponible en CD-ROM. Los servicios de paquetes de software se pueden descargar de Internet a través de: <http://www.ad.siemens.de/simatic-cs>.

Protección del Software STEP 7: está protegido contra copia y sólo puede usarse en un Software dispositivo de programación a la vez.

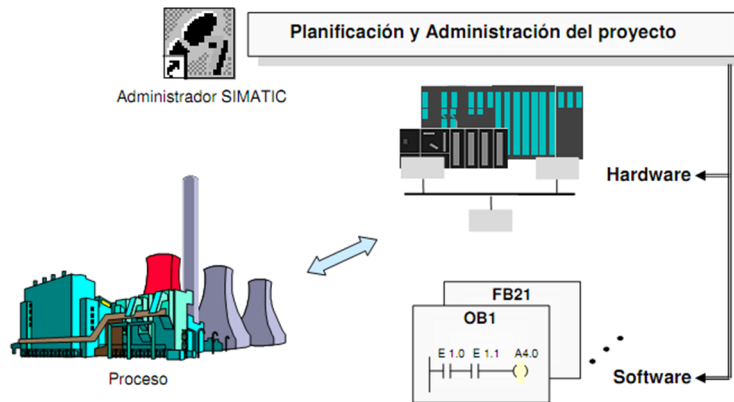
Al igual que el STEP 7 V5.0, el paquete básico STEP 7 puede usarse sin autorización hasta que se haya recibido una nueva autorización. De todos modos, esto no es de aplicación a los paquetes opcionales, tales como *S7 Graph*, si estos han sido instalados.

### **1.3.4. Del proceso al proyecto**

Al examinar un proyecto que se quiera automatizar, se encontrará que está constituido por multitud de secciones y subprocesos más pequeños que están interrelacionados y dependen unos de otros. La primera tarea es, por tanto, dividir el proceso en subtareas más sencillas. Cada subtarea define

ciertos requerimientos hardware y software que debe cumplir el sistema de automatización:

Figura 13. **Del proceso al proyecto**



Fuente: SIEMENS, S. A.

- Hardware:
  - Número y tipo de entradas y salidas
  - Número y tipo de módulos
  - Número de bastidores
  - Capacidad y tipo de CPU
  - Sistemas HMI (interface hombre máquina)
  - Sistemas de comunicación en red
  
- Software:
  - Estructura del programa
  - Manejo de datos para el proceso de automatización

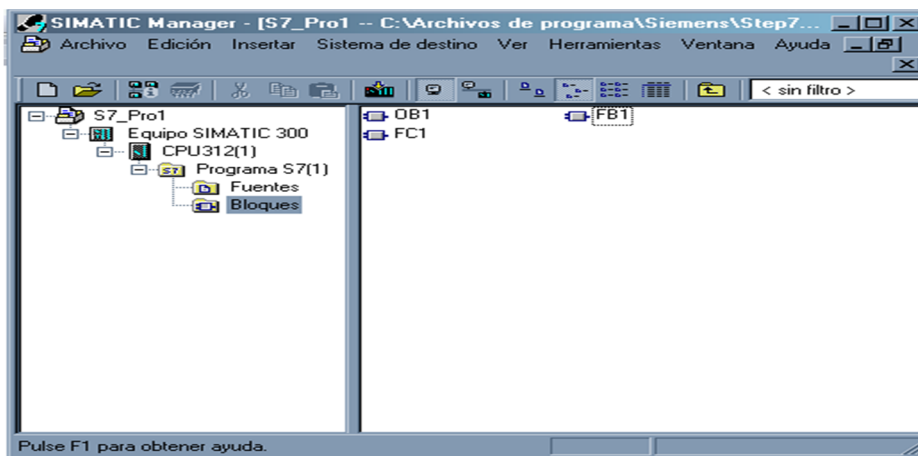
- Datos de configuración
- Datos de comunicación
- Documentación del proyecto y del programa

En SIMATIC S7 todos los requerimientos hardware y software de un proceso de automatización se manejan dentro de un proyecto. Un proyecto incluye todo el hardware necesario (+ configuración), comunicación en red (+ configuración), todos los programas y el manejo de los datos para una solución de automatización.

### 1.3.5. Estructura del proyecto del Software STEP 7

Los datos se almacenan en un proyecto en forma de objetos. Los objetos de un proyecto se organizan en una estructura de árbol (proyecto jerárquico). La estructura de árbol que se visualiza en una ventana de proyecto es similar a la del Explorador de Windows 95. Solo cambian los iconos de los objetos.

Figura 14. Estructura del proyecto de STEP 7



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.



Jerarquía de proyecto:

- El primer nivel contiene el icono del proyecto. Cada proyecto representa una base de datos donde se almacenan todos los datos relevantes del proyecto.
  
- El segundo nivel contiene:
  - Los equipos (por ejemplo, Equipo S7-300) son los que almacenan la información sobre la configuración hardware y la parametrización de los módulos. Los equipos son el punto de partida para la configuración hardware.
  - Las carpetas de Programa S7 son el punto de partida para la edición de programas. Todo el software para un módulo parametrizable de la familia S7 se almacena en una carpeta de Programa S7. Esta contiene más carpetas para los bloques y las fuentes del programa.
  - Las subredes (MPI, Profibus, Industrial Ethernet) forman parte de una red global.
  
- El tercer nivel y siguientes: dependen del tipo de objeto del siguiente nivel más alto.

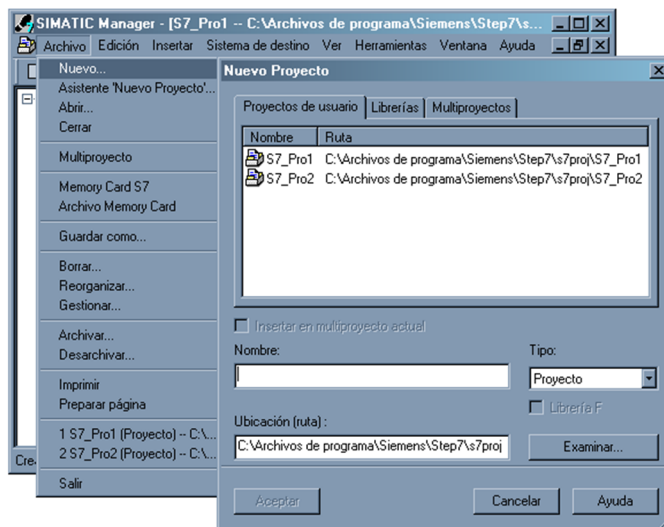
### **1.3.6. Arrancar el Administrador SIMATIC**

Para arrancar el Administrador SIMATIC hacer clic en Inicio, SIMATICA, Administrador SIMATIC, automáticamente se despliega la pantalla en la cual se crea, configura, programa etc. un proyecto nuevo.

### 1.3.6.1. Creación de un proyecto

Seleccionar la opción de menú archivo nuevo para la creación de un nuevo proyecto y confirmar haciendo clic en Aceptar ver figura siguiente.

Figura 15. Creación de un proyecto de STEP 7

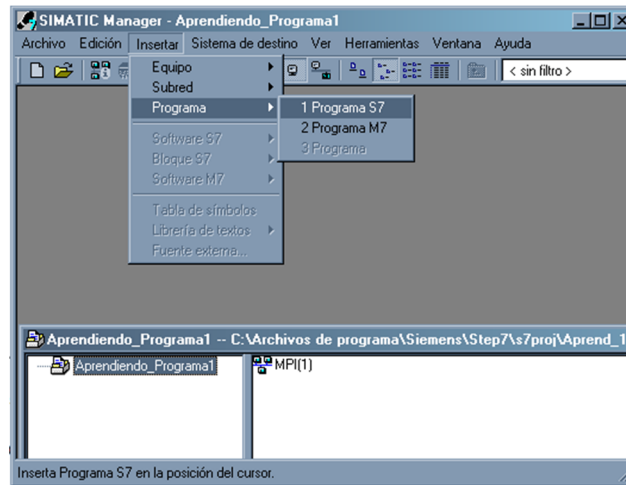


Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

#### 1.3.6.1.1. Insertar un programa

La figura siguiente muestra como insertar un programa nuevo después de haber creado un proyecto de STEP 7, se observa que puede ser un programa S7 o un programa M7.

Figura 16. Insertar un programa de STEP 7



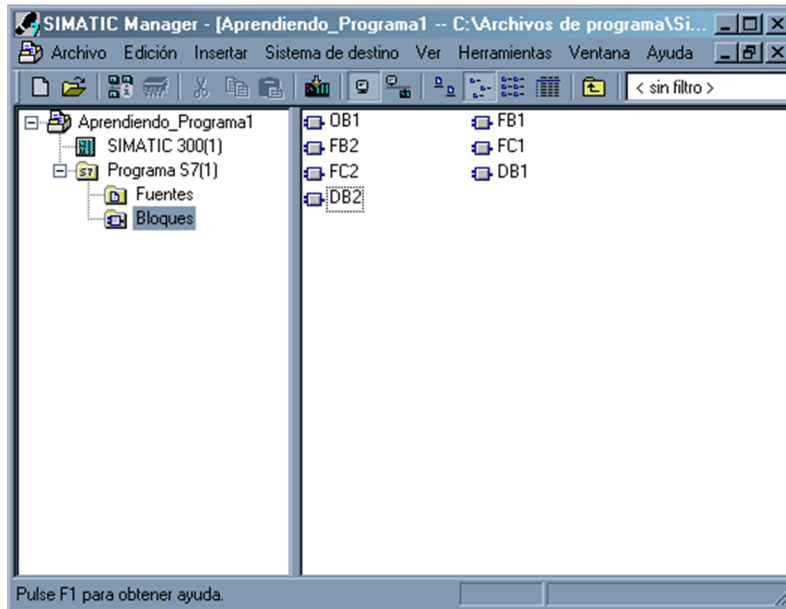
Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

Seleccionar la opción de menú, insertar, programa, programa S7, para introducir un nuevo programa en el proyecto actual. Cuando se inserta un objeto, el sistema le asigna automáticamente un nombre relacionado, por ejemplo, "Programa S7 (1)". Se puede cambiar más tarde este nombre. El método descrito se usa para crear un Programa S7 independiente del hardware.

#### 1.3.6.1.2. Vista offline/online

La figura siguiente muestra la vista *offline/online*, que ofrece el programa Administrador SIMATIC en la cual se diferencia en que la primera muestra los bloques creados y la segunda lo que está almacenado en la CPU.

Figura 17. Vista *offline/online* en el administrador SIMATIC



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

La vista *offline* muestra la estructura del proyecto almacenada en el disco duro del dispositivo de programación. Esta estructura se muestra en la ventana de proyecto del Administrador SIMATIC. La carpeta "Programa S7" contiene los objetos "Fuentes" y "Bloques". La carpeta "Bloques" contiene los datos de sistema creados con HWConfig y los bloques creados con el Editor LAD/FBD/STL.

La vista online muestra la estructura del proyecto almacenado en la CPU. Esta se muestra en la ventana de proyecto del Administrador SIMATIC. La carpeta "Programa S7" sólo contiene la carpeta "Bloques". La carpeta "Bloques" contiene:

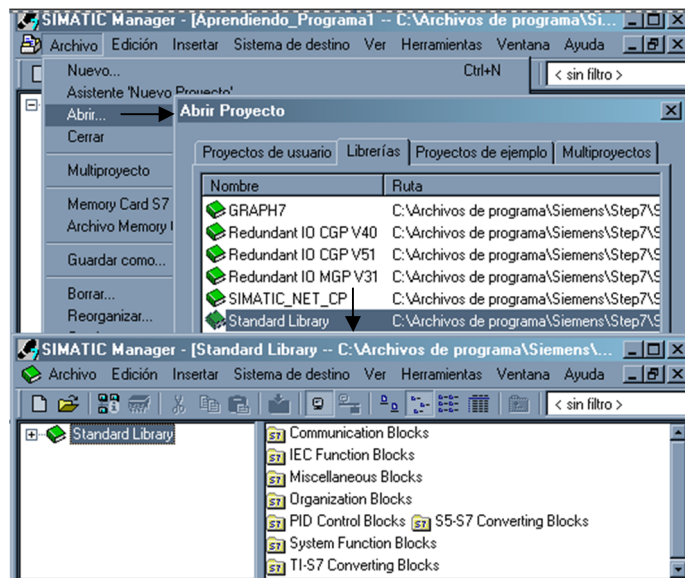
- Bloques de datos de sistema (SDB)

- Bloques de usuario (OB, FC, FB)
- Bloques del propio sistema (SFC, SFB)

### 1.3.6.1.3. Librerías estándar

La figura siguiente muestra como desplegar librería estándar en el Administrador SIMATIC, los pasos son: Archivo, Abrir y Librería estándar en esta ventana se divide en archivos que contienen bloques programados.

Figura 18. Librería estándar en el administrador SIMATIC



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

Las librerías se usan para almacenar bloques que no están asociados a ningún proyecto. Estos bloques se pueden crear en la librería, copiarse en ella o copiarlos de ella, pero no se pueden testear. La estructura de la librería es jerárquica al igual que la de un proyecto. STEP 7 contiene una librería estándar, que está almacenada en una carpeta del Software STEP 7, por ejemplo,

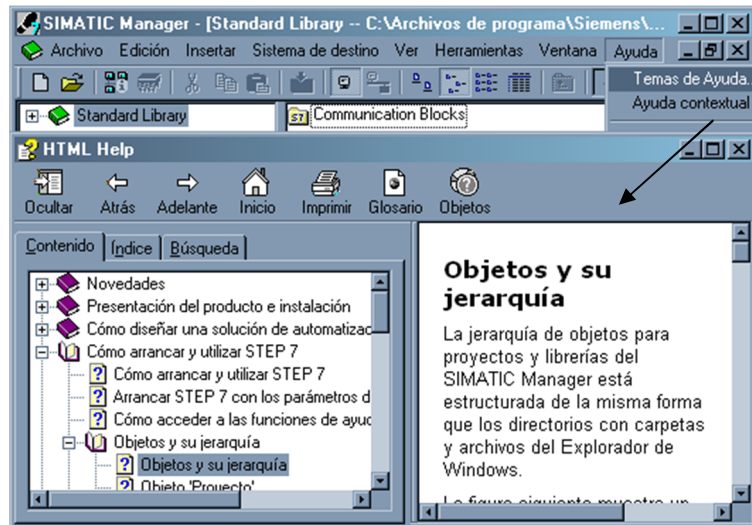
C:\Siemens\Step7\S7libs\stlib30 después de la instalación de STEP 7. Se puede acceder a estos bloques estándar a través del Administrador SIMATIC mediante Abrir, Librerías, en ellas se tiene:

- Communication Blocks: FCs para la comunicación entre la CPU y los procesadores de comunicaciones de periferia descentralizada con el S7-300.
- Organization Blocks: bloques de organización (OBs).
- S5-S7 Converting Blocks: bloques para convertir programas de STEP 5.
- TI-S7 Converting Blocks: funciones estándar de uso general tales como el escalado de valores analógicos.
- IEC Function Blocks: bloques para funciones IEC (IEC: International Electrotechnical Commission), como el procesamiento del día y la hora, operaciones de comparación, procesamiento de cadenas y para la selección de máximo y mínimo.
- PID Control Blocks: bloques de función (FBs) para el control PID en lazo cerrado.
- System Function Blocks: funciones del sistema (SFCs) y bloques de función del sistema (SFBs).

#### **1.3.6.1.4. Sistema de ayuda**

Existe varias formas de ayuda; la ayuda general se obtiene a través de: ayuda, temas de ayuda y la ayuda contextual; se activa con la tecla F1.

Figura 19. Sistema de ayuda de STEP 7



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

#### 1.3.6.1.5. Ayuda contextual

La ayuda contextual da información específica a la aplicación sobre los objetos resaltados, bloques, comandos de menú, ventanas, etc., se puede encontrar información adicional sobre STEP 7 en los manuales electrónicos. Se accede a ellos mediante la opción de menú Inicio, Simatic, Documentación.

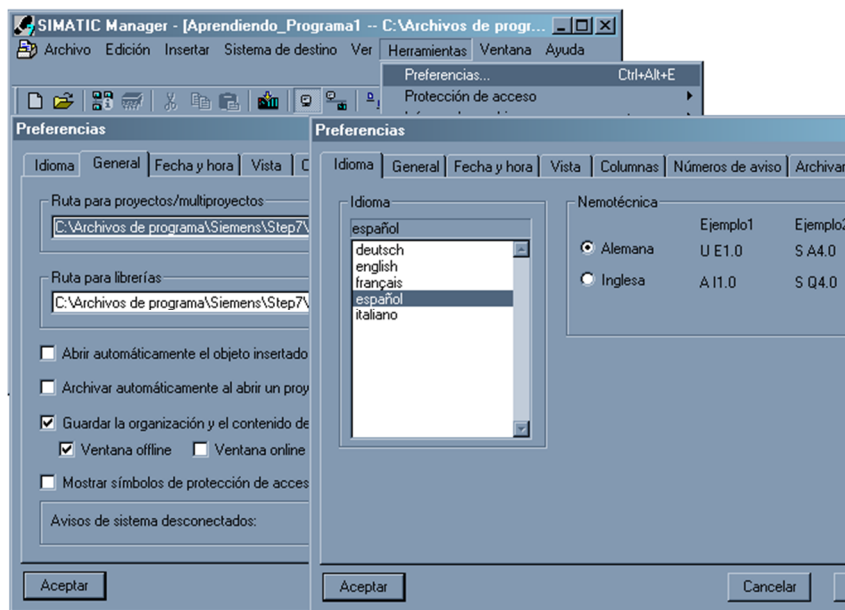
#### 1.3.6.1.6. Herramientas de configuración

Para acceder a la preferencia en las herramientas de Administrador SIMATIC hacer clic en Herramientas, Preferencias y en ella contiene los separadores siguientes ver figura siguiente:

Separador “Idioma”:

- Idioma: se puede seleccionar el idioma que se quiere para el Administrador SIMATIC, los menús, cuadros de diálogo, ayuda, etc. Solo aparecen en la lista los idiomas que se hayan instalado.
- Nemotécnica: se puede seleccionar la nemotécnica que se quiere usar en la programación de los bloques S7.

Figura 20. **Herramientas de configuración del administrador SIMATIC**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

Separador “General”: ajustes básicos para la edición de proyectos y librerías:

- Ruta para proyectos: es donde se especifica el directorio en el que se quiere almacenar los proyectos de usuario.



- Ruta para librerías: es donde se especifica el directorio donde se quiere guardar las librerías de usuario.
- Mensajes del sistema desconectados: presionando el botón “Activar“, se puede activar todos los mensajes de sistema que fueron desconectados en una ventana en la que se eligió previamente la opción “Mostrar siempre este mensaje.... “ .

Separador “Ver”: aquí es donde se especifica lo que debe aparecer en la vista online.

Separador “Columnas”: aquí es donde se especifica qué columnas se van a visualizar cuando se selecciona la vista Detalles.

### **1.3.7. Configuración hardware**

La configuración hardware es una de las ventanas del Administrador SIMATIC que ayuda a insertar una CPU que se requiera para realizar la tarea de automatización y módulos sobre un bastidor teóricamente.

#### **1.3.7.1. Configuración hardware y parametrización**

Configuración HW: los módulos se suministran de fábrica con parámetros predeterminados. Si estos parámetros se adaptan a sus necesidades, no necesita realizar una configuración hardware.

La configuración hardware es necesaria si:

- Si quiere modificar los parámetros predefinidos o las direcciones de un módulo (por ejemplo, habilitar la alarma de proceso de un módulo).

- Si quiere configurar enlaces de comunicación.
- Con equipos con periferia distribuida (PROFIBUS-DP).
- Con equipos S7-400 con varias CPUs (multiprocesador) o bastidores de expansión.
- Con controladores lógicos programables con tolerancia a fallos (paquete opcional).

Configuración teórica: cuando se configura un sistema, se crea la llamada configuración teórica. Contiene un equipo hardware con los módulos planificados y los parámetros asociados. El sistema de PLC se monta de acuerdo a la configuración teórica y durante la puesta en marcha, ésta se carga en la CPU.

Configuración real: en un sistema montado, la configuración existente y la parametrización de los módulos se puede leer de la CPU. Aparece, por tanto, un nuevo equipo HW en el proyecto.

Esto es necesario, por ejemplo, si no se tiene la estructura del proyecto en nuestra PG. Después de leer la configuración real, puede comprobar los parámetros y almacenar los datos del proyecto.

Con el S7-400, la CPU se puede parametrizar de tal forma, que cuando existan diferencias entre la configuración teórica y la real, se interrumpa el arranque de la CPU.

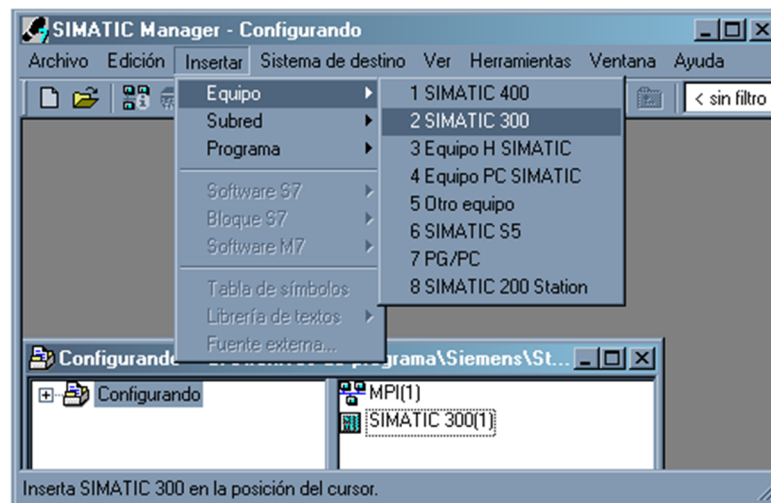
Para llamar a la herramienta HW Config, debe existir un equipo hardware en el Administrador SIMATIC.

Parametrización: establecimiento de las características de los bloques parametrizables, ejemplo: características arranque, áreas remanentes,...

### 1.3.7.2. Insertar un equipo

La figura siguiente muestra como insertar un equipo las cuales pueden ser; SIMATIC 400, SIMATIC 300, SIMATIC S5, etc., al insertar un equipo automáticamente puedo realizar la configuración hardware.

Figura 21. Insertar un quipo



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

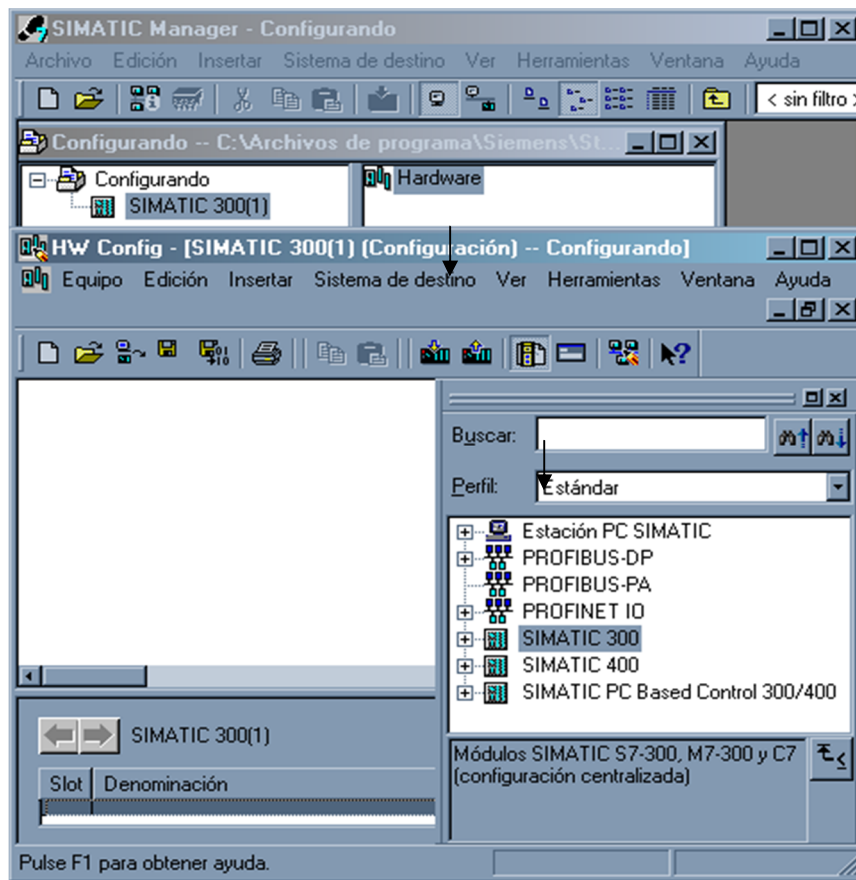
Para insertar un nuevo equipo en el proyecto actual, seleccione la opción de menú, Insertar, Equipo, Equipo SIMATIC 300 o SIMATIC 400. Se puede cambiar el nombre dado a este equipo (“SIMATIC 300 (1)”).

### 1.3.7.3. Arrancar HW Config

Esta herramienta ayuda a configurar, parametrizar y diagnosticar el hardware.

Para arrancar la herramienta HW Config: seleccionar un equipo hardware en el Administrador SIMATIC y elegir la opción de menú Editar, Abrir Objeto o hacer doble clic en el objeto hardware.

Figura 22. Arrancar HW Config



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

“Configuración hardware”: esta es una ventana de la aplicación “HW Config” que se usa para insertar componentes desde la ventana “Configuración Hardware”. La barra de título de esta ventana contiene el nombre del proyecto y el nombre del equipo.

“Catálogo Hardware”: para abrir el catálogo seleccionar la opción de menú Ver, Catálogo o hacer clic en el icono de la barra de tareas.

Si se selecciona el perfil de catálogo “Estándar”, aparecen todos los bastidores, módulos y módulos de interfase en la ventana “Catálogo Hardware”. Puede crear sus propios perfiles de catálogo con los elementos que use más frecuentemente seleccionando la opción de menú herramientas, editar perfiles de catálogo.

Los esclavos Profibus que no existan en el catálogo, se pueden añadir más adelante. Para hacerlo, se usan los archivos GSE que son suministrados por el fabricante del dispositivo esclavo. El archivo GSE contiene una descripción del dispositivo. Para incluir el esclavo en el catálogo hardware, se usa la opción de menú herramientas, instalar nuevo archivo GSE. Encontrará los nuevos dispositivos en el catálogo bajo la cabecera otros aparatos de campo.

#### **1.3.7.4. Direccionamiento de los módulos del S7-300**

Números de slot: los números de slot del S7-300 simplifican el direccionamiento con el entorno del S7-300. La primera dirección de un módulo queda determinada por la posición del módulo en el bastidor.

Slot 1: fuente de alimentación. Por defecto es el primer slot. La fuente de alimentación no es esencial. Un S7-300 se puede alimentar directamente a 24V.

Slot 2: slot para la CPU.

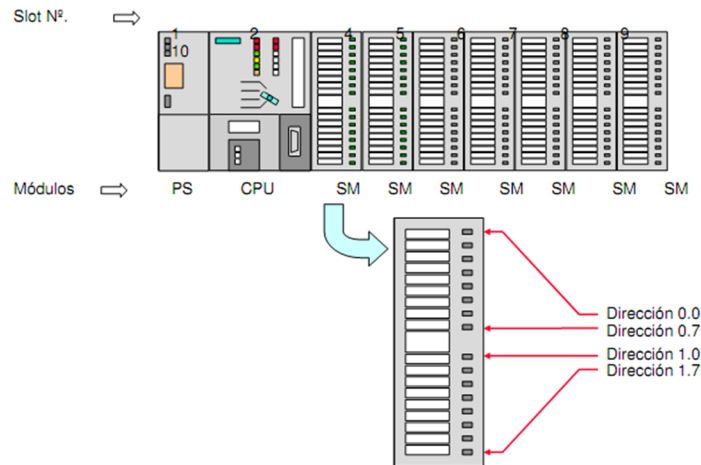
Slot 3: reservados para un módulo interfase (IM) para configuraciones multibastidor usando bastidores de expansión. Incluso si no está instalada la IM, debe ser incluido por temas de direccionamiento. Puede reservar físicamente el slot (por ejemplo, para la instalación de una IM en fechas futuras) insertando un módulo *dummy* DM370.

Slots 4-11: el slot 4 es el primer slot disponible para módulos de E/S, procesadores de comunicación (CP) o módulos de función (FM).

Ejemplos de direccionamiento:

- Un módulo de ED en el slot 4 comienza con el byte de dirección 0.
- Al LED superior de un módulo de SD en el slot 6 se le llama A8.0.

Figura 23. **Direccionamiento de los módulos del S7-300**



Fuente: SIEMENS S. A.

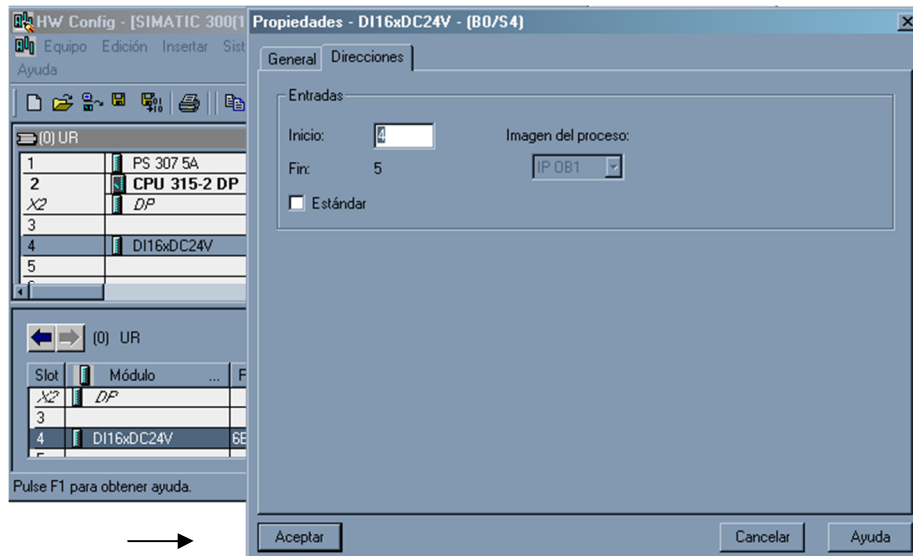
#### 1.3.7.4.1. **Direccionamiento libre**

Con el S7-300 (CPUs sin interfase DP, ni compactas) y S7-400 (sin configuración hardware), a los módulos se les asigna direcciones dependientes del slot en el que están.

Con el S7-300 (CPUs con interfase DP o compactas) y con el S7-400, se puede parametrizar las direcciones de inicio de los módulos como se muestra en la figura siguiente.

Cuando se hace doble clic en un módulo digital o analógico, se abre la pantalla de parametrización. Después de elegir el separador Direcciones, deseleccione la opción Estándar. Ahora puede definir las direcciones de comienzo en la casilla Inicio. Si la dirección ya está en uso, aparece un mensaje de error.

Figura 24. **Direccionamiento libre**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

Las imágenes parciales de proceso solo se pueden definir en el S7-400. De esa forma, puede combinar en un grupo entradas y salidas específicas (por ejemplo, señales críticas en el tiempo). Una función de sistema activa la actualización de una imagen de proceso parcial en el programa de usuario.

Después de un reset de la memoria de la CPU, los parámetros y, en consecuencia, las direcciones se pierden. Esto significa que, son válidas de nuevo las direcciones dependientes del slot con el S7-300 y con el S7-400.

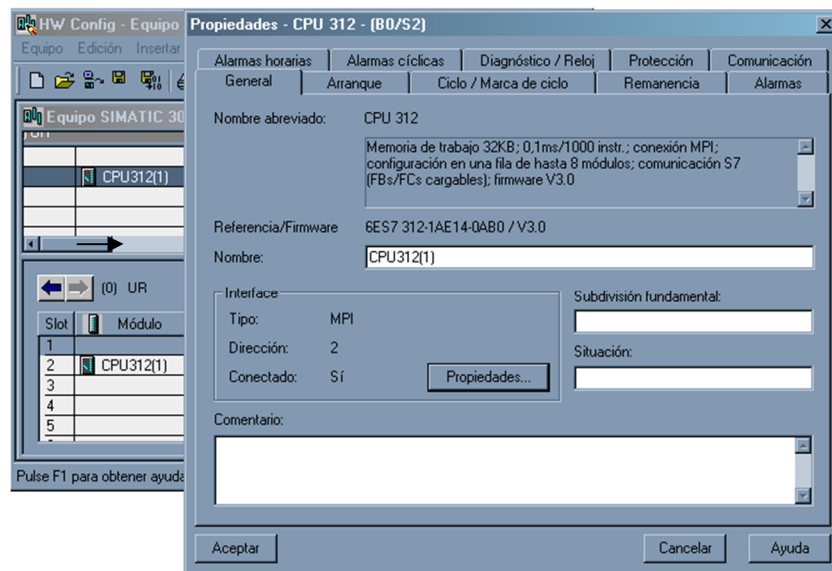
### 1.3.7.5. **Propiedades de la CPU**

Asigne parámetros a los módulos para adaptarlos a los requerimientos del proceso. Realización:



- Seleccionar un módulo en la ventana equipo.
- Hacer doble clic en el módulo seleccionado para abrir la ventana Propiedades.
- Esta ventana contiene 10 separadores en los que puede parametrizar varias características de la CPU.

Figura 25. **Propiedades de la CPU**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

#### 1.3.7.5.1.

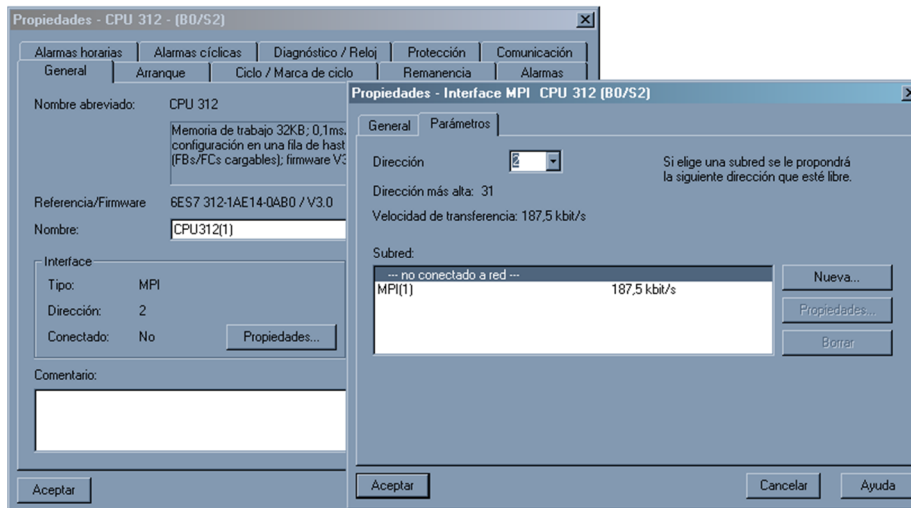
#### 1.3.7.5.2. **General**

La página General proporciona información sobre el tipo de módulo, su localización y, en el caso de módulos programables, su dirección MPI.

Si se quiere montar en red varios PLCs a través de la interfase MPI, se debe asignar una dirección MPI diferente a cada CPU. Hacer clic en el botón

Propiedades para abrir la ventana Propiedades - Interfase MPI, que contiene dos páginas: General y Parámetros.

Figura 26. **Propiedades de la CPU: general**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

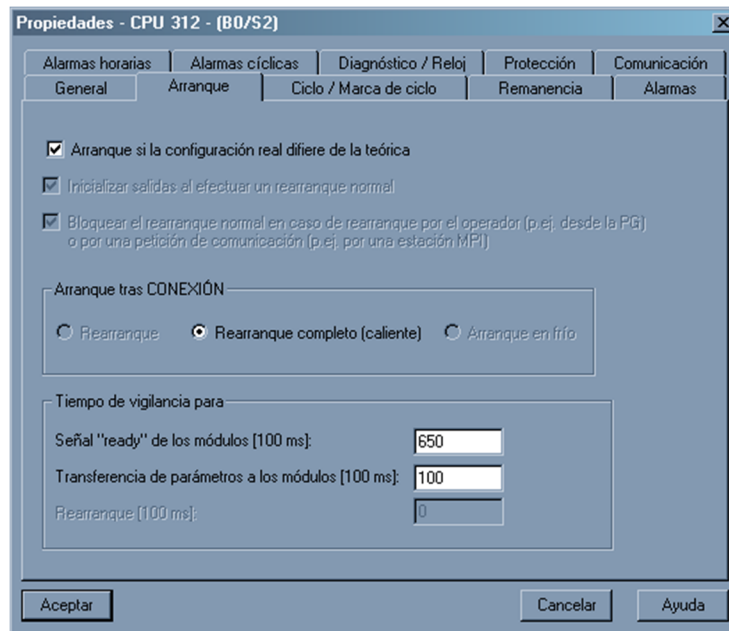
### 1.3.7.5.3. Arranque

Las CPU S7-300 y S7-400 tienen diferentes características de arranque. El S7-300 sólo reconoce el tipo de arranque Rearranque completo. Las CPU más modernas también tienen la posibilidad de Arranque en frío.

Señal “*Ready*” de los módulos (x100ms): tiempo máximo que tienen todos los módulos para entregar un mensaje de *Ready* después de alimentarlos. Si los módulos no envían un mensaje de *Ready* a la CPU dentro de este tiempo, la configuración real difiere de la teórica.

Por ejemplo, en una configuración multibastidor, todas las fuentes de alimentación pueden encenderse dentro de este tiempo sin prestar atención a ninguna secuencia en particular.

Figura 27. **Propiedades de la CPU: arranque**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

Transferencia de parámetros a los módulos (x100ms): tiempo máximo para "distribuir" los parámetros a los módulos parametrizables (el tiempo empieza a contar después del mensaje Señal "Ready" de los módulos). Si, después que pase el tiempo de vigilancia, no se han parametrizado todos los módulos, la configuración real difiere de la teórica.

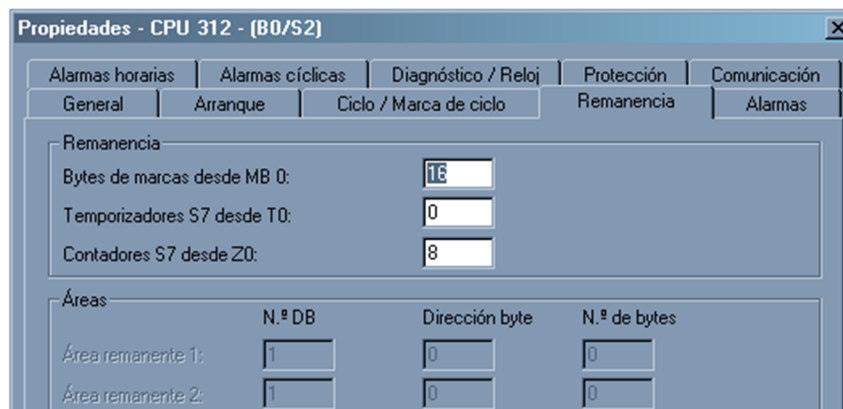
Sólo con las CPUs con interfase DP integrada (y con el S7-400) puede usar la casilla Arrancar si la configuración real difiere de la teórica para decidir si la CPU debería arrancar cuando la configuración real no es igual a la teórica

(número y tipo de módulos instalados). Las otras CPUs S7-300 sí pasan a *RUN* cuando la configuración real difiere de la teórica.

#### 1.3.7.5.4. Remanencia

Dentro de las propiedades de la CPU se tiene la remanencia como lo muestra la figura siguiente, esta muestra los valores asignado el cual se puede modificarlo, la cantidad máxima a asignar dependerá de la CPU elegida.

Figura 28. **Propiedades de la CPU: remanencia**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

La página Remanencia se usa para especificar las áreas de memoria que se deben conservar después de un fallo de alimentación o un paso de *STOP* a *RUN*. En ambos casos se realiza un “rearranque completo” en el S7-300. En el rearranque completo, se conservan los bloques almacenados en la RAM salvaguardada por batería (OB, FC, FB, DB) además de las marcas, temporizadores y contadores definidos como remanentes. Solo las marcas, temporizadores y contadores no definidos como remanentes se resetean.

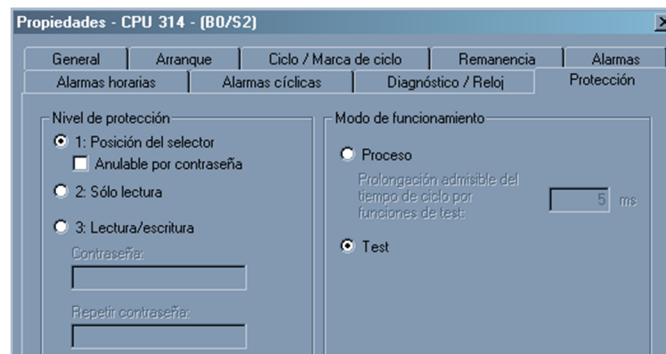
Si la RAM no está salvaguardada con batería, la información en ella se pierde. Sólo las marcas, temporizadores y contadores definidos como remanentes y los bloques de datos remanentes se guardan en el área de memoria RAM no volátil.

Después de un re arranque completo, el programa deber ser cargado de nuevo: Desde la *memory card* (si está insertada) o desde la PG (si no existe una *memory card*).

### 1.3.7.5.5. Protección

La figura siguiente muestra desplegado la opción protección de las propiedades de la CPU, en ella se tiene tres niveles de protección de lectura y escritura el cual solo podrá ser leído y modificado con contraseña.

Figura 29. **Propiedades de la CPU: protección**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

Las características por defecto (nivel de protección 1; sin contraseña): la posición del selector de la CPU determina la protección:

- Selector en la posición RUN-P o en STOP: sin restricciones.
- Selector en la posición RUN: ¡solo es posible el acceso de sólo lectura!

Si asigna un nivel de protección con contraseña (sólo válido hasta un reset de la memoria), una “persona que conozca la contraseña” tiene acceso de lectura y escritura. "La persona que no conozca la contraseña" tiene las siguientes restricciones:

- Nivel de protección 1: corresponde con las características predefinidas.
- Nivel de protección 2: sólo es posible el acceso a lectura, independientemente del selector de modo.
- Nivel de protección 3: no es posible el acceso ni en lectura ni en escritura, independientemente del selector de modo.

Ejemplo: si quiere ejecutar la función "Forzar variable" en un módulo con nivel de protección 2 asignado debe introducir la contraseña.

También puede introducir la contraseña para un módulo protegido en el Administrador SIMATIC:

- Seleccionar el módulo protegido o su programa S7.
- Introducir la contraseña seleccionando la opción de menú Sistema de destino, Permiso de acceso. El permiso de acceso, tras haber introducido la contraseña, sólo es válido hasta que finaliza la última aplicación S7.

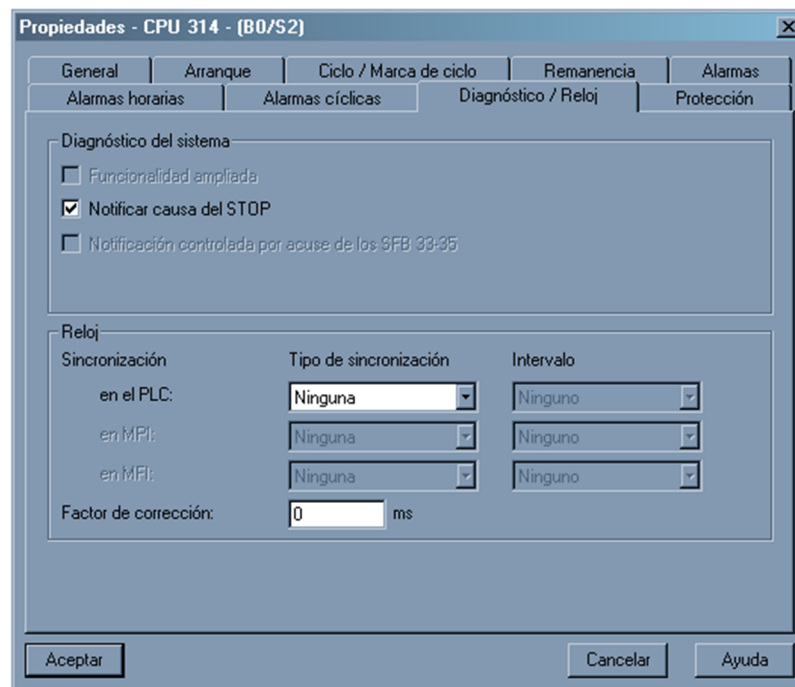
Modo de funcionamiento: el ciclo de carga para las funciones de test se regula con esta opción. En modo proceso, las funciones de test como “Observar” u “Observar/Forzar Variable” están restringidas de modo que el tiempo de vigilancia del ciclo que se haya fijado no se exceda. El test con

puntos de parada y la ejecución paso a paso no se pueden llevar a cabo. En modo Test, se pueden usar todas las funciones de test a través de la PG/PC sin restricciones, incluso si el tiempo de ciclo se excede de forma importante.

### 1.3.7.5.6. Diagnóstico/reloj

Si la casilla Notificar causa del *STOP* está desactivada (no señalada), no se envía ningún mensaje a la PG / OP cuando la CPU pasa a modo *Stop* (Mensajes de la CPU). La causa del *STOP* sí aparece en el buffer de diagnóstico.

Figura 30. **Propiedades de la CPU: diagnóstico/reloj**



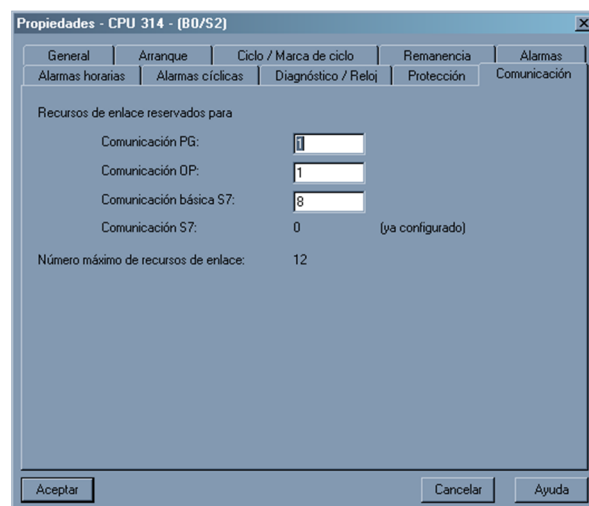
Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

De todos modos, es posible además ajustar de forma automática la hora en el reloj de un dispositivo independiente especificando un factor de corrección. El factor de corrección se usa para corregir la inexactitud del reloj cada 24 horas. El factor de corrección puede ser positivo o negativo. Ejemplo: si el reloj está 3 segundos adelantado tras 24 horas, puede corregirlo con el factor “3000ms”.

### 1.3.7.5.7. Comunicación

Cada enlace de comunicación ocupa un recurso de enlace en la CPU S7. Dependiendo de las especificaciones técnicas, dispondrá de un número específico de recursos de enlace en cada CPU S7 que serán ocupados por distintos servicios de comunicación (comunicación PG/OP, comunicación S7 o comunicación estándar S7).

Figura 31. **Propiedades de la CPU: comunicación**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.



Cuando los servicios de comunicación están presentes, los recursos de enlace se ocupan en la secuencia en que aparecen estos servicios. Para que esta ocupación de los recursos de enlace no dependa solo en la secuencia de aparición de los distintos servicios de comunicación, puede reservar recursos de enlace para los siguientes servicios:

- Comunicación con PG y OP
- Comunicación estándar S7

Se reserva al menos un recurso de enlace para la comunicación PG/OP. Otros servicios de comunicación tales como Comunicación S7 con las funciones PUT/GET no pueden ocupar este recurso de enlace incluso aunque realicen su enlace primero. En vez de esto, se ocupan los recursos de enlace disponibles que no estuvieran reservados específicamente para ningún servicio.

#### **1.3.7.5.8. Ciclo/marca de ciclo**

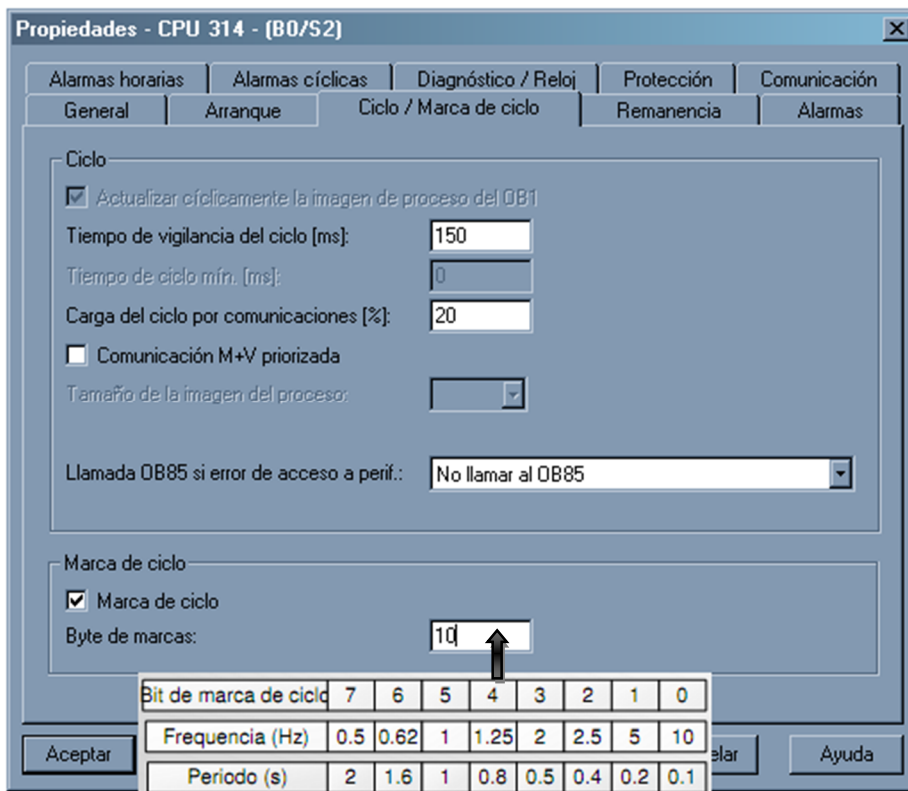
Tiempo de vigilancia del ciclo (ms): si se excede este tiempo la CPU pasa a modo *STOP*. Posibles causas de este desbordamiento del tiempo: Procesos de comunicación, frecuentemente por eventos de interrupción, errores en el programa de la CPU. Si se ha programado un OB de error (OB 80), el tiempo de vigilancia del ciclo se dobla. Después de este tiempo, la CPU también pasa a modo *STOP*.

Carga del ciclo por comunicaciones (%): la comunicación (por ejemplo, transmisión de datos a otra CPU vía MPI o las funciones de test, iniciadas por la PG) está restringida al porcentaje especificado del tiempo de ciclo actual. Restringir la carga del ciclo puede ralentizar la comunicación entre la CPU y la

PG. Ejemplo: restringir la comunicación al 20 % ocasiona una carga máxima del ciclo de 20 ms para un tiempo de vigilancia del ciclo de 100 ms.

Con la CPU 318-2 y varias CPUs S7-400, puede especificar el tamaño de la imagen de proceso (en bytes). El área de imagen de proceso comienza siempre con el byte de entrada o salida 0.

Figura 32. **Propiedades de la CPU: ciclo/marca de ciclo**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

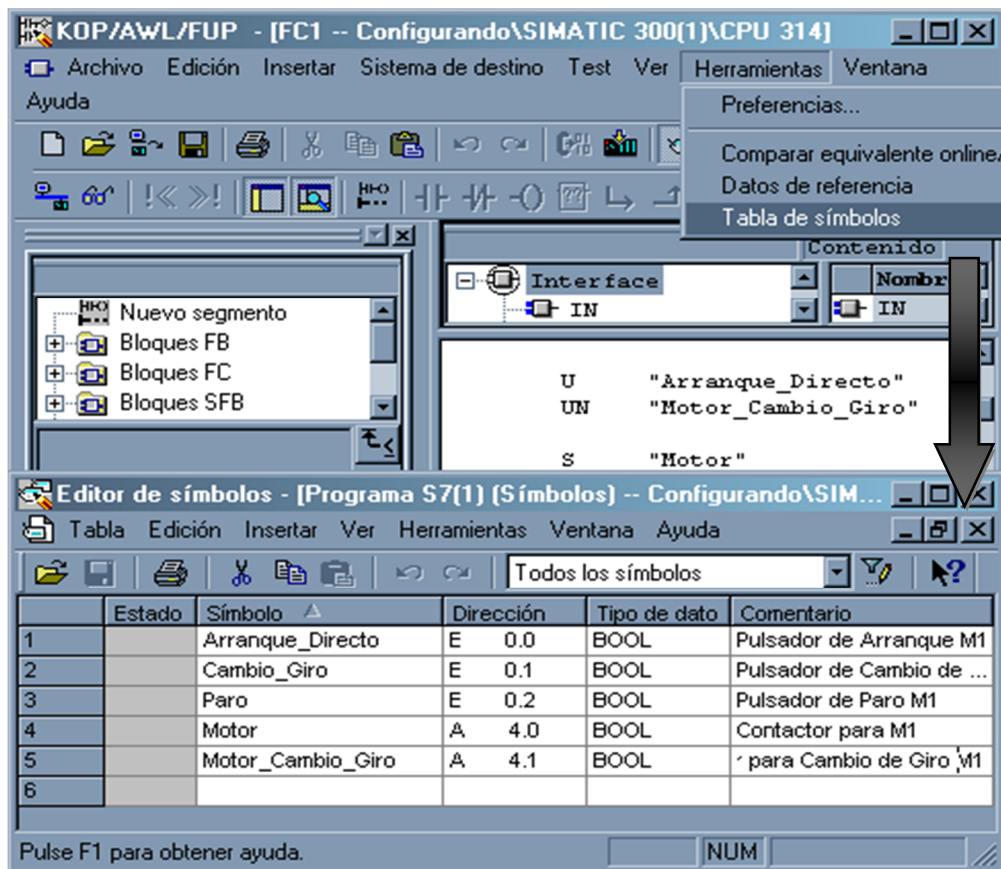
Las marcas de ciclo son marcas que cambian su valor binario periódicamente. Como puede observar en la figura 39 se tiene elegido al byte

de marca 10. Cada bit del byte de marca de ciclo tiene asignado un periodo/frecuencia determinado.

### 1.3.8. La tabla de símbolos



Para abrir la tabla de símbolos seleccione la opción de menú Herramientas, Tabla de símbolos en el editor LAD/FBD/STL. También puede abrir la tabla de símbolos desde el Administrador SIMATIC: seleccionar la ventana en la parte izquierda de la ventana de proyecto y hacer doble clic en el objeto Símbolos.

Figura 33. La tabla de símbolos



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

Cuando abre la tabla de símbolos, se abre una ventana adicional. Se compone de columnas para el nombre del símbolo, el operando, el tipo de datos y un comentario para el símbolo. Cada símbolo ocupa una línea de la tabla. Al final de la tabla se añade automáticamente una línea en blanco para definir un nuevo símbolo. La tabla de símbolos es una base de datos común y puede ser usada por todas las herramientas de STEP 7. Las columnas de la parte izquierda de la tabla tienen el siguiente significado:

- Columna líneas:
  - : si se han ocultado las columnas para las Propiedades especiales del objeto (comando de menú ver, columnas S, MV, A, C, AC desactivado), en la columna de líneas aparece este símbolo si en la línea correspondiente se ha utilizado como mínimo una Propiedad especial del objeto.
- Columna estado:
  - : el nombre simbólico o el operando es idéntico a otra entrada de la tabla de símbolos.
  - S: propiedades de supervisión.
  - MV: manejo y visualización con WinCC.
  - A: propiedades de aviso.
  - C: propiedades de comunicación.
  - AC: accionamiento del contacto.

### **1.3.8.1. Direccionamiento absoluto y simbólico**

En direccionamiento absoluto, especificar el operando directamente (por ejemplo, la entrada E 1.0). En este caso no necesita una tabla de símbolos, pero es más difícil seguir el programa.

En direccionamiento simbólico, se usan símbolos (por ejemplo, MOTOR\_ON) en lugar de direcciones absolutas. Los símbolos para las entradas, salidas, temporizadores, contadores, marcas y bloques se almacenan en la tabla de símbolos. Cuando introduzca nombres simbólicos, no debe incluir comillas. El editor de programas lo hace automáticamente.

Símbolos globales: los símbolos globales declarados en la tabla de símbolos se pueden usar en todos los bloques de un programa. El nombre simbólico asignado debe ser único, es decir, solo debe aparecer una vez en la tabla.

Símbolos locales: los símbolos locales se exponen en la zona de declaración de un bloque. Solo se pueden usar dentro de ese bloque. El nombre simbólico puede usarse de nuevo en la zona de declaración de otro bloque.

En el área de instrucciones de un programa, los símbolos globales se pueden diferenciar de los locales de la siguiente forma:

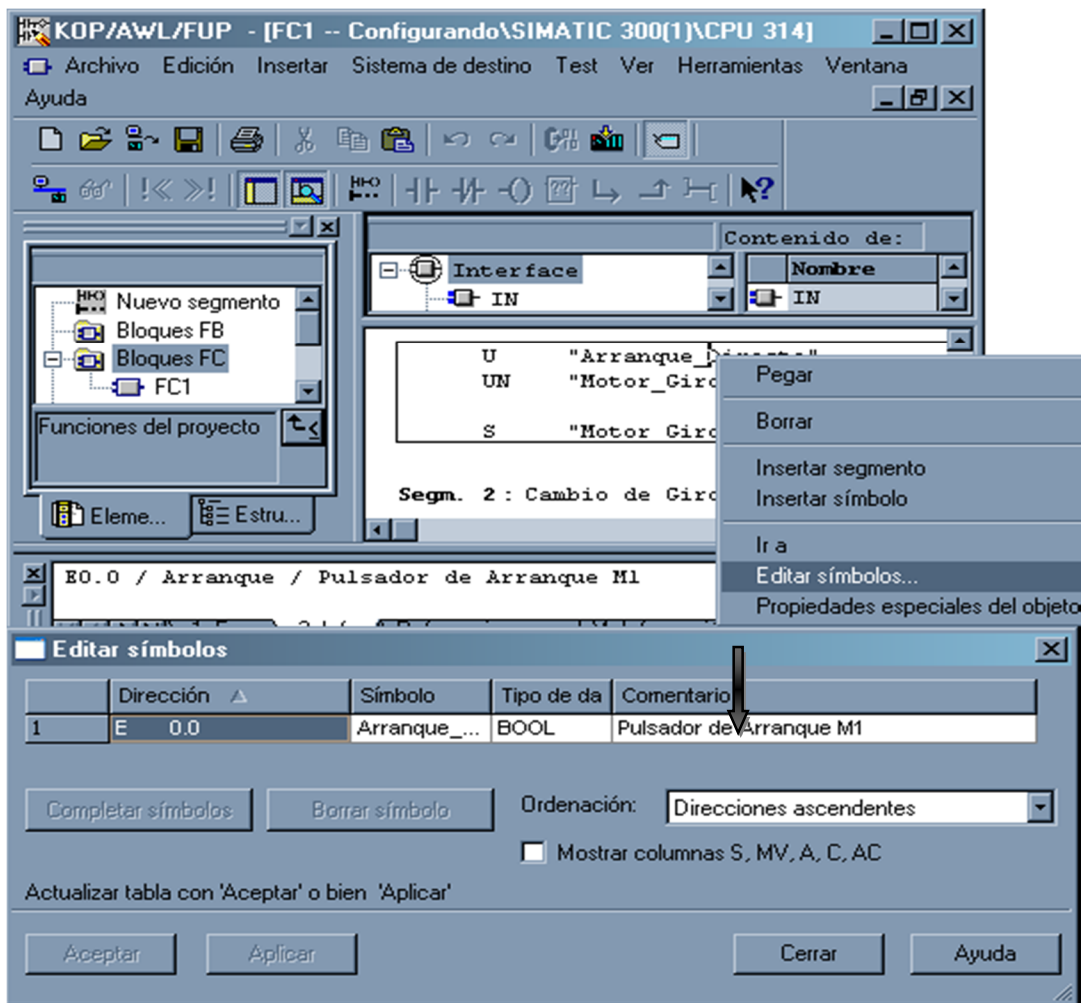
- Los símbolos (globales) de la tabla de símbolos se representan entre comillas "...".
- Los símbolos (locales) de la tabla de declaración de variables se representan precedidos de un "#".

### **1.3.8.2. Editar símbolo**

La opción de menú editar, símbolo, o un clic con el botón derecho del ratón en la dirección y a continuación la opción de menú editar símbolo, le permite asignar, con posterioridad, nombres simbólicos a direcciones absolutas.

Los nombres asignados se introducen automáticamente en la tabla de símbolos. Los nombres que ya estén en la tabla de símbolos aparecen en un color diferente. Estos no pueden ser usados de nuevo en la tabla de símbolos.

Figura 34. **Editar símbolo (en el editor LAD/FBD/STL)**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

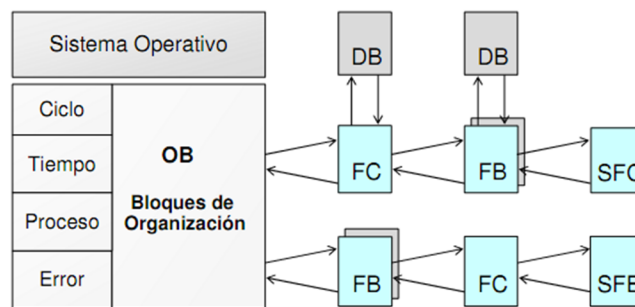
### 1.3.9. Arquitectura de bloques y editor de bloques

En la arquitectura de bloques y editor de bloques tiene los tipos de bloques de programa como; bloques de funciones, funciones, bloque de organización, etc.

#### 1.3.9.1. Tipos de bloques de programa

Bloques: el controlador lógico programable proporciona varios tipos de bloques, donde puede almacenar el programa de usuario y demás datos relacionados. Dependiendo de los requerimientos del proceso, el programa puede ser estructurado en bloques diferentes.

Figura 35. Tipos de bloques



Fuente: SIEMENS, S. A.

Bloques de organización (OB's): los bloques de organización (OB's) constituyen la interfase entre el sistema operativo del PLC y el programa de usuario. El programa completo puede almacenarse en el OB, que es ejecutado cíclicamente por el sistema operativo (programa lineal) o puede dividirse y almacenarse en distintos bloques (programa estructurado).

Funciones (FC Y SFC): una función (FC) contiene parte de la funcionalidad del programa. Es posible programar funciones a las que se les pueda asignar parámetros. Como resultado, las funciones también se pueden utilizar para tareas repetitivas o funcionalidades complejas tales como cálculos. Las Funciones de Sistema (SFC) son funciones integradas en el sistema operativo de la CPU. La cantidad de SFC's y su funcionalidad es fija.

Bloques de función (FB Y SFB): básicamente, los bloques de función (FB) ofrecen la misma funcionalidad que las funciones. La diferencia radica en que los bloques de función poseen su propia área de memoria en forma de bloques de datos de instancia. Como resultado, los bloques de función están concebidos para tareas muy repetitivas o funcionalidades complejas, como tareas de control en lazo cerrado. Los Bloques de Función de Sistema (SFB) son funciones parametrizables integradas en el sistema operativo de la CPU. Su número y funcionalidad es fijo.

Bloques de datos DB: los bloques de datos (DB) son áreas de datos del programa de usuario en las que los datos son distribuidos de forma estructurada.

Operaciones permitidas: se puede utilizar todo el repertorio de instrucciones en todos los bloques (FB, FC y OB).

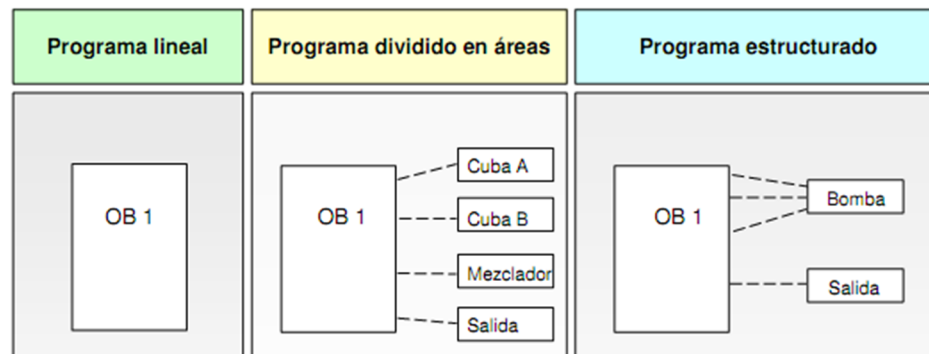
### **1.3.9.2. Estructura de programa**

Programa lineal: todo el programa se encuentra en un módulo (OB1) con todas las instrucciones juntas. Este modelo se asemeja a un esquema de relés, que se reemplaza por un controlador lógico programable. La CPU procesa las instrucciones individuales una detrás de la otra.



Programa dividido: el programa está dividido en bloques, donde cada bloque solo contiene el programa para resolver una tarea parcial. Es posible dividir aún más en segmentos dentro de un bloque. Puede generar plantillas de segmento para segmentos del mismo tipo. El bloque de organización OB 1 contiene instrucciones que llaman a los otros bloques en una secuencia definida.

Figura 36. **Estructura de programa**



Fuente: SIEMENS, S. A.

Programa estructurado: un programa estructurado contiene bloques con parámetros, llamados bloques parametrizables. Estos bloques se diseñan para que puedan usarse de forma universal. Cuando se llama a un bloque parametrizable, le dan los parámetros actuales (las direcciones exactas de entradas y salidas así como los valores de los parámetros). Ejemplo:

- Un "bloque bomba" contiene instrucciones para el control de una bomba.
- Los bloques de programa, que son responsables del control de bombas especiales, llaman al "bloque bomba" y le dan información sobre qué bomba va a ser controlada y con qué parámetros.

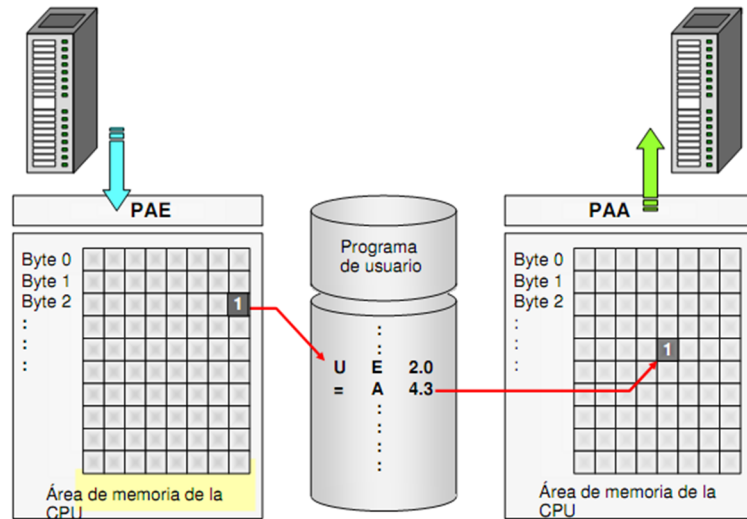
- Cuando el “bloque bomba” haya completado la ejecución de sus instrucciones, el programa retorna al bloque que ha realizado la llamada (por ejemplo, el OB 1), que continua con el procesamiento de sus instrucciones.

### **1.3.9.3. Imágenes de proceso**

La CPU comprueba el estado de las entradas y las salidas en cada ciclo. Existen áreas de memoria específicas en las que se almacenan los datos binarios de los módulos: la PAE y la PAA. El programa accede a estos registros durante el procesamiento. PAE: La tabla de imagen de proceso de entrada se encuentra en el área de memoria de la CPU. Allí se almacena el estado de señal de todas las entradas. PAA: la tabla de imagen de proceso de salida contiene los valores de salida resultantes de la ejecución del programa. Estos se envían a las salidas reales (A) al final del ciclo.

Cuando comprueba las entradas en el programa de usuario, por ejemplo con la instrucción U E 2.0, se evalúa el último estado de señal desde la PAE. Esto garantiza la llegada del mismo estado de señal cuando realiza consultas múltiples de la entrada dentro de un ciclo.

Figura 37. **Imágenes de proceso**



Fuente: SIEMENS, S. A.

#### 1.3.9.4. **Ejecución cíclica del programa**

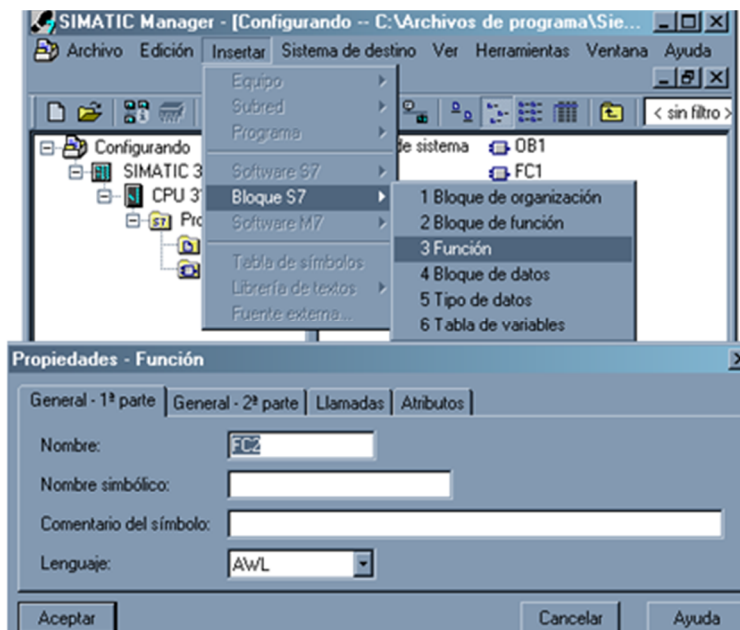
El funcionamiento cíclico de la CPU se compone de tres secciones principales:

- La CPU comprueba el estado de las señales de entrada y actualiza la tabla de imagen de proceso de entrada.
- Ejecuta el programa de usuario con sus respectivas instrucciones.
- Escribe los valores de la tabla de imagen de proceso de salida en los módulos de salidas.

### 1.3.9.5. Insertar un bloque S7

Seleccionar la opción de menú insertar, bloque S7 para visualizar una lista de los tipos de bloques: después de que haya elegido el tipo de bloque, se abre el cuadro de diálogo Propiedades para que pueda introducir el número de bloque y el lenguaje de programación que quiera usar (LAD, FBD o STL). Cuando haya hecho sus ajustes y los haya validado pulsando el botón Aceptar, se inserta el nuevo bloque en el programa actual.

Figura 38. Insertar un bloque S7

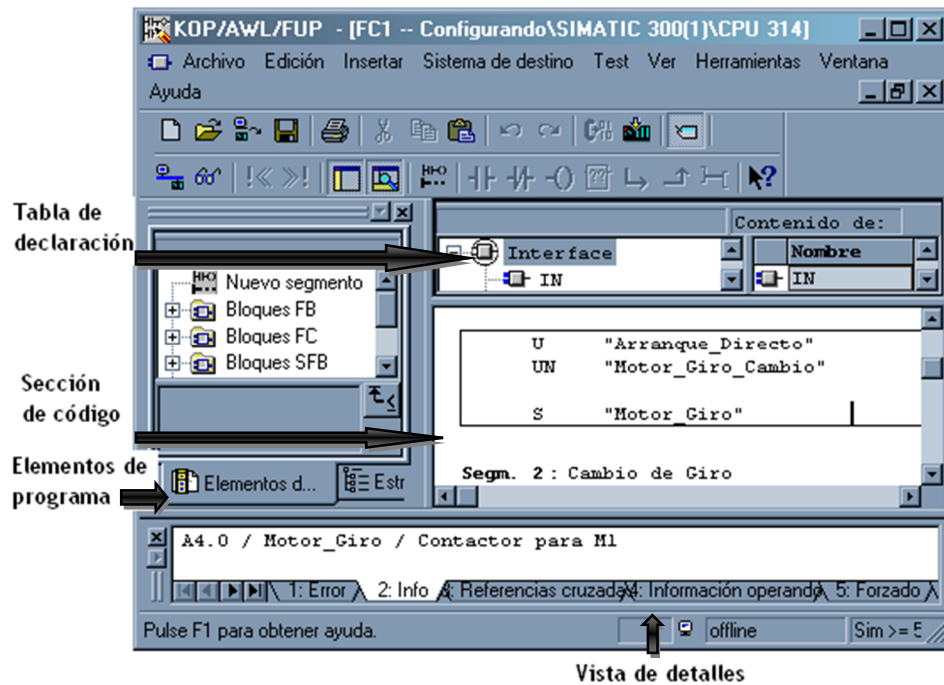


Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

### 1.3.9.6. Componentes del editor LAD/FBD/STL

Hacer doble clic en cualquier bloque para arrancar el editor, cuando se arrancan el editor LAD/FBD/STL, se abren automáticamente dos ventanas: la tabla de declaración y la sección de código.

Figura 39. Componentes del editor LAD/FBD/STL



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

La tabla de declaración pertenece al bloque. Se usa para declarar variables y parámetros para el bloque. La sección de código contiene el programa, dividido en segmentos separados si fuese necesario; se comprueba la sintaxis de las líneas introducidas.

El contenido de la ventana Elementos de Programa depende del lenguaje de programación seleccionado. Puede hacer doble clic en los elementos de la lista ("árbol") para insertarlos en el programa en la posición del cursor. Además puede insertar elementos mediante arrastrar y soltar.

#### **1.3.9.7. Los lenguajes de programación de STEP 7**

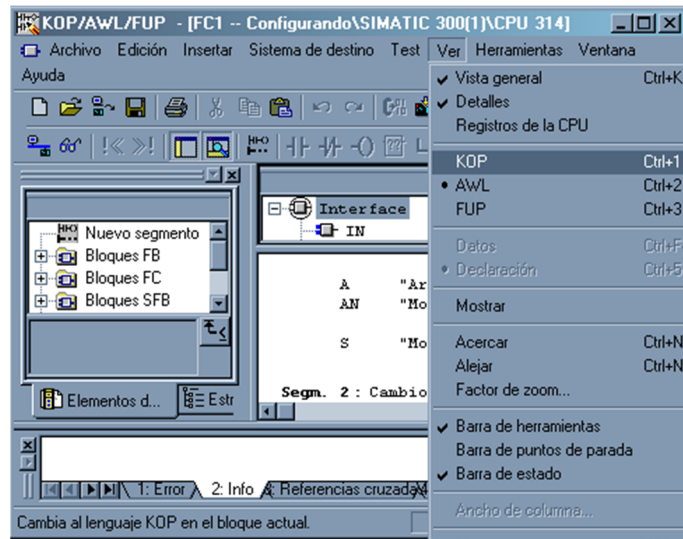
Existen varios lenguajes de programación en STEP 7 que puede usar dependiendo de nuestra preferencia y conocimientos. Adhiriendo a unas reglas específicas, el programa puede crearse en lista de instrucciones y ser convertido a otro lenguaje de programación.

LAD (KOP): el esquema de contactos es muy similar a un esquema eléctrico. Se usan símbolos como contactos y bobinas. Este lenguaje de programación está indicado para aquellos que crecieron con contactores.

STL (AWL): la lista de instrucciones se compone de instrucciones de STEP 7. Se puede programar con bastante libertad usando STL, este lenguaje es el preferido por los programadores que ya están familiarizados con otros lenguajes de programación.

FBD (FUP): el diagrama de funciones utiliza "cajas" para las funciones individuales. El carácter que aparece en la caja nos indica la función (por ejemplo, &=operación lógica AND). Este lenguaje de programación tiene la ventaja de que incluso un no programador, como por ejemplo un ingeniero de proceso, puede trabajar con él.

Figura 40. Selección del lenguaje de programación



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

Puede convertir secciones de programa que hayan sido escritas en lenguajes de programación gráfica a STL, como puede ver en la figura anterior. De cualquier modo, debe tener presente que el resultado de esta conversión no es siempre la solución más eficiente en lista de instrucciones. No es posible siempre convertir secciones escritas en STL en LAD o FBD. Las secciones de código que no puedan ser convertidas, permanecen en STL. No se pierden secciones de programa durante la conversión.

Los elementos más usados en LAD y FBD aparecen en forma de iconos en la barra de tareas. Haciendo *clic* en ellos con el ratón, se insertan en la posición seleccionada en el programa.

Puede insertar otros elementos de programa desde la ventana Elementos de programa:

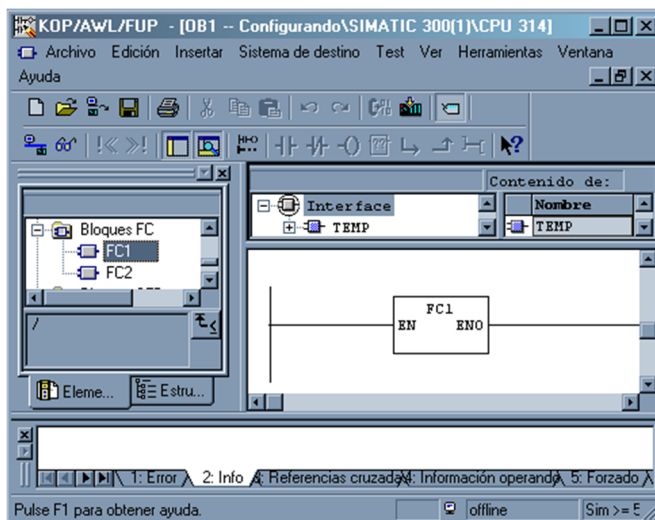
- En cualquier posición usando arrastrar y soltar.
- En la posición seleccionada haciendo doble clic en el elemento de la ventana.

Cuando se hace clic en el icono Nuevo Segmento en la barra de tareas, se añade un nuevo segmento tras el actual.

### 1.3.9.8. Llamada a un bloque desde el OB1

Para que un bloque creado se incluya en la ejecución cíclica del programa de la CPU, debe ser llamado desde el OB1. La manera más simple de llamar a un bloque en los lenguajes de programación gráfica LAD y FBD es a través de la ventana elementos de programa (ver figura siguiente).

Figura 41. Llamada a un bloque desde el OB1



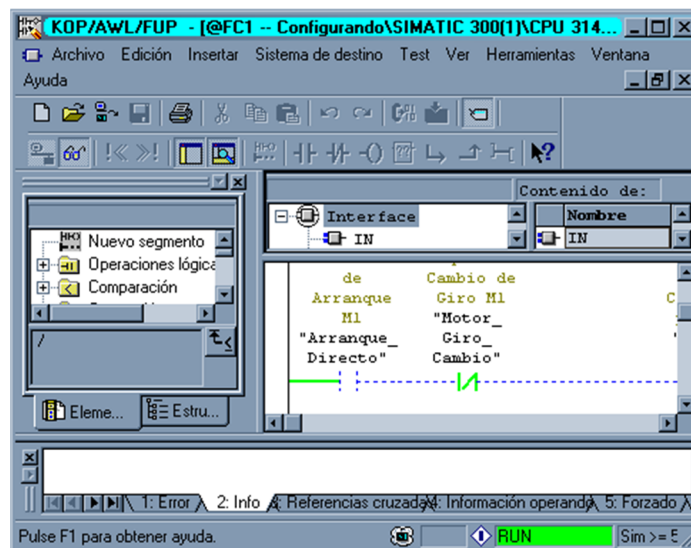
Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.



### 1.3.9.9. Depurar un programa simple

Antes de activar el modo observar, debe abrir el bloque que quiera observar bien *offline* u *online* con el editor LAD/FBD/STL. Para poder realizar el test de un bloque en modo *offline*, debe cargarse en primer lugar en el PLC.

Figura 42. Depurar un programa simple



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

Existen dos formas de activar/desactivar la función de test "Observar":

- Hacer clic en el icono de las Gafas localizado en la barra de tareas.
- Seleccionar la opción de menú test, observar.

El estado del programa se visualiza de distintas formas dependiendo del lenguaje de programación seleccionado (LAD/FBD/STL). Cuando la función

observar está activada, no puede cambiar el lenguaje de programación en el que se ve el bloque (LAD/FBD/STL).

#### **1.3.9.10. Cargar y guardar bloques modificados**

Puede efectuar correcciones en bloques que hayan sido abiertos en *online* u *offline*, pero no en modo test:

- Cargar normalmente el bloque modificado en el PLC, lo testea, realiza más correcciones si fuese necesario y finalmente debe guardarlo en el disco duro cuando haya sido depurado completamente.
- Si no quiere testear el programa de una vez, puede guardar directamente los cambios en el disco duro. De este modo se borra la antigua versión del bloque.
- Si hace correcciones en un número determinado de bloques y todavía no quiere sobrescribir la versión original del programa, puede cargar, en primer lugar los bloques modificados en la CPU, sin guardarlos en el disco duro de la PG. Puede guardar los cambios en el disco duro cuando haya comprobado satisfactoriamente el funcionamiento del programa.

Presionando la tecla Insert (Ins), se activa el modo sobrescribir. Después de esto, puede, por ejemplo, modificar el tipo de temporizador sin recablear las entradas y las salidas.

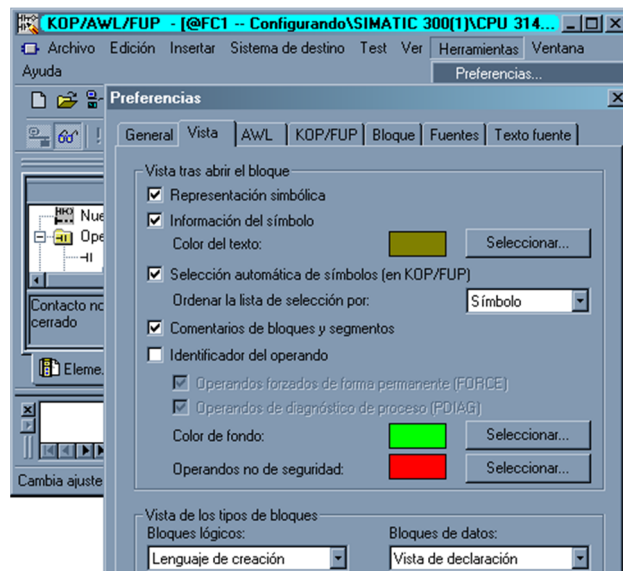
#### **1.3.9.11. Personalización del editor**

En personalización del editor se tiene varias opciones de las cuales se muestra en la figura siguiente, por ejemplo; vista, STL o AWL, LAD/FBD o COP/FUP, bloque y fuente.

### 1.3.9.11.1. Vista

Bloques lógicos: usar las opciones AWL, COP, FUP para seleccionar el lenguaje en el que se quiere escribir un nuevo bloque. Bloques de datos: puede visualizar los bloques de datos con las siguientes vistas: vista de declaración o vista de datos.

Figura 43. Personalización del editor: vista



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

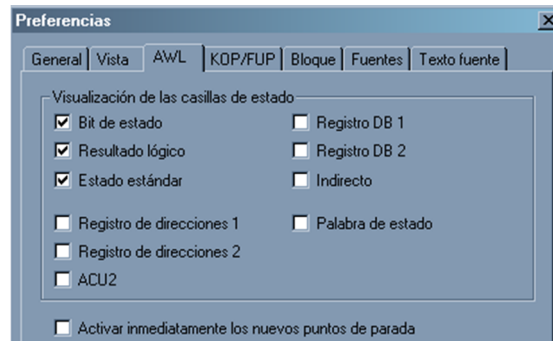
Puede visualizar bloques (ver figura anterior):

- Con direccionamiento absoluto o simbólico.
- Con o sin información del símbolo.
- Con o sin comentarios de bloques y segmentos.
- En el lenguaje en el que fueron escritos o en el lenguaje predeterminado (LAD/FBD/STL).

### 1.3.9.11.2. STL

Cuando observa el estado de un bloque en STL, sólo se visualizarán las casillas de estado que active en este cuadro de diálogo (ver figura siguiente). Están disponibles las siguientes opciones:

Figura 44. Personalización del editor: STL



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- Bit de estado: se visualiza el bit de estado.
- Resultado lógico: se visualiza el resultado lógico de la operación (RLO).
- Estado estándar: se visualiza una palabra de temporizador, de contador o el contenido del ACU 1 dependiendo de la operación usada.
- Registros de direcciones: los registros de direcciones se usan con direccionamiento indirecto.
- ACU2: se visualiza el contenido del acumulador 2.
- Registros DB: se visualiza el contenido del bloque de datos relevante.
- Palabra de estado: se visualiza la palabra de estado.
- Indirecto: esta visualización sólo es posible con direccionamiento indirecto por memoria.

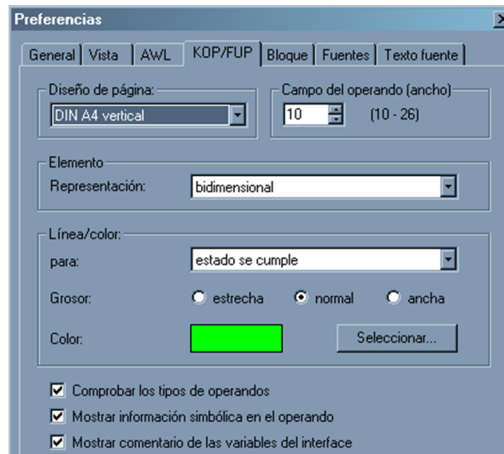
- Estándar: el botón Estándar selecciona el ajuste estándar para las casillas de estado. Se visualizan el bit de estado, el RLO y el estado estándar.
- Activar inmediatamente los nuevos puntos de parada: esta función solo tiene relevancia para la función de test Puntos de parada.

### 1.3.9.11.3. LAD/ FBD

Diseño de página (ver figura siguiente): aquí se selecciona el formato de impresión:

- DIN A4 vertical
- DIN A4 horizontal
- Tamaño máximo

Figura 45. Personalización del editor: LAD/ FBD



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

Campo del operando (ancho): puede ajustar el número máximo de caracteres para el nombre de una dirección a un número comprendido entre 10 y 24. Esto cambia la anchura del elemento de programa en LAD y FBD, con representación simbólica, aparece un recuadro de acuerdo a la anchura del campo de dirección.

Elemento: los elementos de programa se pueden visualizar de varias formas: bidimensional o tridimensional.

Línea/color: usa esta casilla para determinar cómo se van a representar los siguientes elementos:

- Elemento seleccionado (color)
- Contactos (línea)
- Estado se cumple (color y línea)
- Estado no se cumple (color y línea)

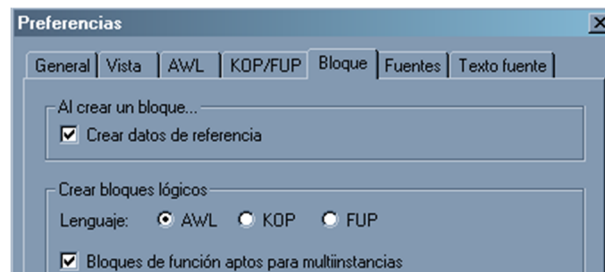
Comprobar los tipos de operando: cuando edita un bloque, se comprueba siempre el tipo de dirección introducido en instrucciones lógicas de bit. Puede desactivar esta opción para comparaciones, operaciones matemáticas, etc. (sólo para usuarios experimentados).

#### **1.3.9.11.4. Crear bloque**

Crear datos de referencia: si activa la opción Crear datos de referencia (ver figura siguiente), los datos de referencia se actualizan de forma automática. Si esta opción no está activada, los datos de referencia no se actualizan inmediatamente. Pero la siguiente vez que abra la ventana Mostrar datos de

referencia, debe decidir si quiere actualizar los datos de referencia y para qué bloques.

Figura 46. **Personalización del editor: bloque**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

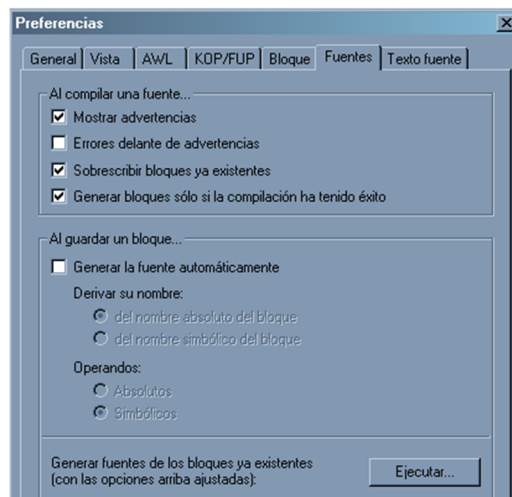
#### 1.3.9.11.5. Fuentes

Es posible introducir un programa o partes de un programa como un fichero fuente en STL y compilar a continuación estos bloques. El fichero fuente puede contener el código de varios bloques que se compilan secuencialmente en bloques. Crear un programa usando ficheros fuente tiene las siguientes ventajas:

- Crear y editar el fichero fuente con el editor ASCII que se desea, importarlo a continuación y, con esta aplicación, compilarlo en bloques individuales. La compilación genera los bloques individuales y los almacena en el programa S7 de usuario.
- Programar varios bloques en un fichero fuente.
- Guardar un fichero fuente incluso con errores de sintaxis. Esto no es posible cuando se crean bloques lógicos con comprobación de sintaxis.

El único problema es que solo detecta los errores de sintaxis cuando compila el fichero fuente.

Figura 47. **Personalización del editor: fuentes**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

El fichero fuente se crea con la sintaxis del lenguaje de programación Lista de instrucciones (STL). La estructuración de un fichero fuente en bloques, declaración de variables o segmentos tiene lugar usando palabras clave.

### **1.3.10. Instrucciones de programación**

El lenguaje de programación KOP/FUP/AWL, en la parte elementos están divididos las diferentes operaciones a programar, por ejemplo; operaciones lógicas con bit, comparación, conversión, contaje, etc.



### 1.3.10.1. Operaciones binarias

Las operaciones lógicas con bits operan con dos dígitos, 1 y 0. Estos dos dígitos constituyen la base de un sistema numérico denominado sistema binario. Los dos dígitos 1 y 0 se denominan dígitos binarios o bits. En el ámbito de los contactos y bobinas, un 1 significa activado (conductor) y un 0 significa desactivado (no conductor).

Las operaciones lógicas con bits interpretan los estados de señal 1 y 0, y los combinan de acuerdo con la lógica de Boole. Estas combinaciones producen un 1 o un 0 como resultado y se denominan resultado lógico (RLO). Las operaciones lógicas con bits permiten ejecutar las más diversas funciones.

Se dispone de las operaciones lógicas con bits siguientes:

- $& Y, \geq 1$  O y XOR O-exclusiva: estas consultan el estado de señal y emiten resultados que se copian en el bit RLO o bien que se enlazan con el mismo.
- Operación O lógica de operaciones Y, y operación Y lógica de operaciones O.
- = Asignación y # Conector: estas asignan el RLO o lo guardan de forma provisional.

Las siguientes operaciones reaccionan ante un RLO de 1:

- R: desactivar salida
- S: activar salida
- RS: flipflop de desactivación/activación
- SR: flipflop de activación/desactivación

Otras operaciones reaccionan ante un cambio de flanco positivo o negativo para ejecutar las siguientes funciones:

- N: detectar flanco negativo (1 -> 0)
- P: detectar flanco positivo (0 -> 1)
- NEG: detectar flanco de señal 1 -> 0
- POS: detectar flanco de señal 0 -> 1

Las restantes operaciones afectan directamente al RLO:

- Insertar una entrada binaria
- Invertir una entrada binaria
- SAVE: cargar resultado lógico (RLO) en el registro RB

### **1.3.10.2. Formato de representación de los números**

Al llamar una operación en el lenguaje de programación KOP/FUP/AWL, está la realiza de forma binaria y pueden ser de 16 bits y 32 bits representando a una palabra o doble palabra sucesivamente.

#### **1.3.10.2.1. 16 Bits**

Código BCD: cada dígito de un número decimal es codificado con 4 posiciones de bit. Se usan 4 bits porque el dígito más alto en esta representación, 9, necesita 4 bits para ser codificado en binario (1001). Los dígitos decimales del 0 al 9 se representan en código BCD de igual forma que los números binarios del 0 al 9.

Entero: el tipo de datos INT es un entero (16 bits). El signo (bit 15) indica si el número es positivo o negativo (0 = positivo, 1 = negativo). Un entero de 16 bits puede tomar un valor entre -32 768 y +32 767. En formato binario, se representa la forma negativa de un entero como el complemento a dos del número entero positivo. (El complemento a dos se obtiene cambiando ceros por unos (y viceversa), y sumando uno). Para evaluar el patrón de bits de un número negativo, se ponderan las posiciones de los ceros, después sumar uno al resultado y poner un signo menos delante.

#### **1.3.10.2.2. 32 Bits**

DINT: los enteros de 32 bits con signo son denominados también doble enteros (double integers) o enteros largos (long integers). Cubren el rango desde L# -2147483648 a L#+2147483647.

REAL: un número real (también llamado número en coma flotante) es un número positivo o negativo en el rango de  $-1.175495 \cdot 10^{-38}$  a  $3,402823 \cdot 10^{38}$ . Ejemplos: +10.339 o +1.0339E1 y -234567 o -2.34567E5. Un número real ocupa dos palabras en memoria. El bit más significativo indica el signo. Los otros bits representan la mantisa y el exponente en base dos.

#### **1.3.10.3. Carga y transferencia de datos**

Las operaciones de carga (L) y transferencia (T) permiten programar un intercambio de información entre módulos de E/S y áreas de memoria, o bien entre áreas de memoria. La CPU ejecuta estas operaciones en cada ciclo como operaciones incondicionales, es decir, independientemente del resultado lógico de la operación.

Se dispone de las operaciones de cargar y transferencias siguientes:

- L: cargar.
- L STW: cargar palabra de estado en ACU 1.
- LAR: cargar registro de direcciones 1 con contenido del ACU 1.
- LAR1 <D>: cargar registro de direcciones 1 con puntero (formato de 32 bits).
- LAR1 AR2: cargar registro de direcciones 1 con contenido del registro de direcciones 2.
- LAR2: cargar registro de direcciones 2 con contenido del ACU 1.
- LAR2 <D>: cargar registro de direcciones 2 con puntero (formato de 32 bits).
- T: transferir.
- T STW: transferir ACU 1 a la palabra de estado.
- TAR: intercambiar registro de direcciones 1 y registro de direcciones 2.
- TAR1: transferir registro de direcciones 1 a ACU 1.
- TAR1 AR2: transferir registro de direcciones 1 a registro de direcciones 2.
- TAR1 <D>: transferir registro de direcciones 1 a dirección de destino (puntero de 32 bits).
- TAR2: transferir registro de direcciones 2 a ACU 1.
- TAR2 <D>: transferir registro de direcciones 2 a dirección de destino (puntero de 32 bits).

#### **1.3.10.4. Operaciones de contaje**

Los contadores tienen reservada un área de memoria propia en la CPU. Dicha área de memoria reserva una palabra de 16 bits para cada contador. En los datos técnicos de la CPU encontrará la cantidad de contadores que puede

disponer. Las operaciones de contaje son las únicas funciones que tienen acceso al área de memoria reservada para contadores.

Se dispone de las operaciones de contaje siguientes:

- ZAEHLER: parametrizar e incrementar / decrementar contador.
- Z\_VORW: parametrizar e incrementar contador.
- Z\_RUECK: parametrizar y decrementar contador.
- SZ: posicionar el contador en preselección.
- ZV: incrementar contador.
- ZR: decrementar contador.

#### **1.3.10.5. Operaciones de temporización**

Se dispone de las operaciones de temporización siguientes:

- S\_IMPULS: parametrizar y arrancar temporizador como impulso.
- S\_VIMP: parametrizar y arrancar temporizador como impulso prolongado.
- S\_EVERZ: parametrizar y arrancar temporizador como retardo a la conexión.
- S\_SEVERZ: parametrizar y arrancar temporizador como retardo a la conexión con memoria.
- S\_AVERZ: parametrizar y arrancar temporizador como retardo a la desconexión.
- SI: arrancar temporizador como impulso.
- SV: arrancar temporizador como impulso prolongado.
- SE: arrancar temporizador como retardo a la conexión.
- SS: arrancar temporizador como retardo a la conexión con memoria.
- SA: arrancar temporizador como retardo a la desconexión.

### 1.3.10.6. Operaciones de conversión

Las operaciones de conversión leen el contenido del parámetro *IN* y lo convierten o le cambian el signo. El resultado se puede recoger en el parámetro *OUT*. Las siguientes operaciones se utilizan para convertir números decimales codificados en binario y enteros a otros tipos de números:

- BCD\_I: convertir de BCD a entero.
- I\_BCD: convertir de entero a BCD.
- BCD\_DI: convertir de BCD a entero doble.
- I\_DI: convertir de entero a entero doble.
- DI\_BCD: convertir de entero doble a BCD.
- DI\_R: convertir de entero doble a número en coma flotante.

Para formar complementos de números enteros o para cambiar el signo de un número en coma flotante se utilizan las siguientes operaciones:

- INV\_I: complemento a 1 de entero.
- INV\_DI: complemento a 1 de entero doble.
- NEG\_I: complemento a 2 de entero.
- NEG\_DI: complemento a 2 de entero doble.
- NEG\_R: cambiar el signo de un número en coma flotante.

Para convertir un número en coma flotante de 32 bits IEEE 754 en un entero de 32 bits (entero doble) se utilizan las operaciones descritas a continuación. Las operaciones difieren en el método de redondeo:

- ROUND: redondear a entero doble.
- TRUNC: truncar a entero doble.

- CEIL: redondear número en coma flotante a entero superior.
- FLOOR: redondear número en coma flotante a entero inferior.

### 1.3.10.7. Operaciones de comparación

Las operaciones comparan las entradas IN1 e IN2 según los tipos de comparación siguientes:

- == IN1 es igual a IN2.
- <> IN1 es diferente a IN2.
- > IN1 es mayor que IN2.
- < IN1 es menor que IN2.
- >= IN1 es mayor o igual a IN2.
- <= IN1 es menor o igual a IN2.

Si la comparación es verdadera, el resultado lógico (RLO) de la función es 1, en otro caso 0. Se pueden usar diversas funciones de comparación para comparar los siguientes pares de valores numéricos:

- I: compara enteros (en base de 16 bits en coma fija).
- D: compara enteros (en base de 32 bit en coma fija).
- R: compara números en coma flotante (en base de número real de 32 bit = números IEEE en coma flotante).

### 1.3.10.8. Operaciones aritméticas

El juego de instrucciones del S7-300/400 soporta una multitud de funciones matemáticas. Todas las instrucciones tienen el mismo formato. Las operaciones son; con entero y con números en coma flotante.

### **1.3.10.8.1. Operaciones aritméticas con enteros**

Las operaciones aritméticas con enteros sirven para ejecutar las siguientes operaciones aritméticas con dos enteros (16 y 32 bits):

- ADD\_I: sumar enteros
- SUB\_I: restar enteros
- MUL\_I: multiplicar enteros
- DIV\_I: dividir enteros
- ADD\_DI: sumar enteros dobles
- SUB\_DI: restar enteros dobles.
- MUL\_DI: multiplicar enteros dobles
- DIV\_DI: dividir enteros dobles
- MOD\_DI: obtener el resto de división de enteros dobles
- 

### **1.3.10.8.2. Operaciones aritméticas con números en coma flotante**

Los números de 32 bits IEEE en coma flotante pertenecen al tipo de datos denominado REAL. Las operaciones aritméticas con números en coma flotante sirven para ejecutar las siguientes operaciones aritméticas con dos números en coma flotante IEEE de 32 bits:

- ADD\_R: sumar números en coma flotante
- SUB\_R: restar números en coma flotante
- MUL\_R: multiplicar números en coma flotante
- DIV\_R: dividir números en coma flotante



Con las operaciones aritméticas de números en coma flotante se pueden ejecutar las siguientes funciones con un número en coma flotante (32 bit, IEEE 754):

- ABS: calcular el valor absoluto de un número en coma flotante.
- SQR: calcular el cuadrado de un número en coma flotante.
- SQRT: calcular la raíz cuadrada de un número en coma flotante.
- EXP: calcular el valor exponencial de un número en coma flotante.
- LN: calcular el logaritmo natural de un número en coma flotante.
- Calcular funciones trigonométricas de ángulos representados mediante números en coma flotante:
  - Seno (SIN) y arcoseno (ASIN).
  - Coseno (COS) y arcocoseno (ACOS).
  - Tangente (TAN) y tangente (ATAN).

#### **1.3.10.9. Operaciones de salto**

Esta operación la puede aplicar en todos los bloques lógicos, p. ej., en bloques de organización (OB), en bloques de función (FB) y en funciones (FC). Se dispone de las operaciones de salto siguientes:

- JMP: salto incondicionado
- JMP: salto condicionado a 1 en el bloque
- JMPN: salto condicionado a 0

El operando de una operación de salto es una meta. La meta indica el destino a donde se desea saltar en el programa. La meta se introduce encima del cuadro JMP. Una meta se compone de cuatro caracteres como máximo. El

primer carácter ha de ser una letra alfabética; los restantes caracteres pueden ser letras o números (p. ej. SEG3).

### **1.3.11. Diagnóstico**

La diagnosis normalmente tiene lugar cuando un problema (perturbación) nos lleva a un bloqueo o a un funcionamiento incorrecto del sistema o máquina. Debido al coste asociado al tiempo que una máquina está parada o a funciones fallidas, la causa asociada a la perturbación debe ser encontrada y eliminada rápidamente. Los errores ocurridos pueden dividirse en dos categorías, dependiendo de si son detectados o no por el PLC:

- Normalmente, los errores que son detectados por el sistema operativo del PLC, hacen que la CPU pase al estado de STOP.
- Los errores funcionales, es decir, aquellos en los que la CPU procesa el programa normalmente, pero la función deseada no se ejecuta o la hace de forma incorrecta.

La búsqueda de este tipo de errores es mucho más difícil, ya que inicialmente es difícil encontrar la causa del error. Unas posibles causas podrían ser:

- un error de programación lógica (error software), que no fue detectado durante la creación y arranque del programa de usuario y probablemente sólo tenga lugar en raras ocasiones.
- un fallo del proceso, que se disparó por el funcionamiento incorrecto de componentes asociados directamente con el control del proceso, tales los cables de los sensores/actuadores o por un defecto en el propio sensor/actuador.

### **1.3.11.1. Diagnóstico del sistema**

Todas las CPU S7 poseen un sistema de diagnóstico inteligente. Está integrada en el sistema operativo de la CPU y en otros módulos con posibilidad de diagnóstico, y se ejecuta de forma automática. Los errores ocurridos se almacenan (temporalmente) en el buffer de diagnóstico por las CPU y, por tanto, permite una diagnosis del error rápida y precisa por el personal de mantenimiento.

### **1.3.11.2. Avisos de CPU**

Con esta función, puede visualizar inmediatamente un mensaje de error, para errores esporádicos del sistema en un dispositivo de programación o en un dispositivo HMI. Se despliega una ventana de mensajes en la PG o el OP tan pronto como la CPU conectada pase a *STOP* por causa de un error. También se pueden visualizar mensajes de usuario con una función de sistema.

Módulos registrados: la lista contiene todas las CPUs llamadas con el administrador SIMATIC con la opción de menú sistema de destino, mensajes de la CPU. La lista se divide en cuatro columnas (ver figura siguiente):

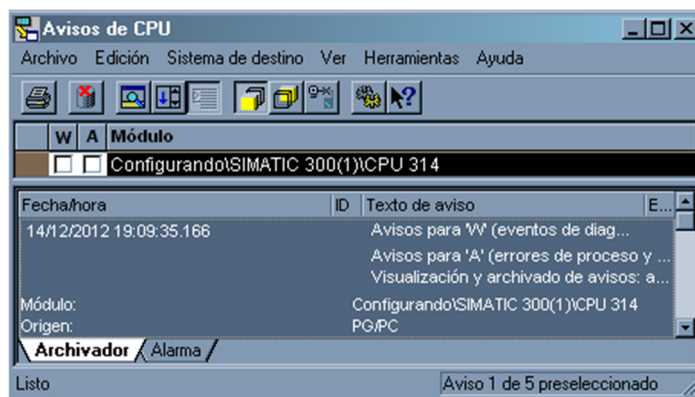
- En la primera columna, aparece un icono si se interrumpió la conexión por parte de un equipo externo.
- En la columna W, se activan/desactivan los mensajes de diagnóstico del sistema y los mensajes de diagnóstico del usuario.
- En la columna A, se activan/desactivan los mensajes de alarma. La aplicación Mensajes CPU comprueba incluso si el módulo en cuestión soporta diagnóstico y mensajes de alarma. Si no es este el caso, aparece un mensaje.

- En la columna Módulo, se introduce el nombre del módulo o la ruta del programa S7.

Mensajes entrantes:

- Resaltar: tan pronto como se recibe el mensaje, la ventana Mensajes de la CPU se despliega en la parte superior, aparece el mensaje y al mismo tiempo se almacena en el archivo de mensajes.
- Segundo plano: la recepción de mensajes tiene lugar en segundo plano. Los mensajes aparecen en la ventana, pero la ventana permanece oculta. Los mensajes quedan archivados y pueden visualizarse cuando se desee.
- Ignorar: los mensajes no son visualizados ni archivados.

Figura 48. Visualización de avisos de la CPU



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

### 1.3.11.3. Información del módulo

La información de la CPU le ayuda en la diagnosis del sistema sin tener que programar nada, y hace posible una detección, localización y eliminación rápida de errores.

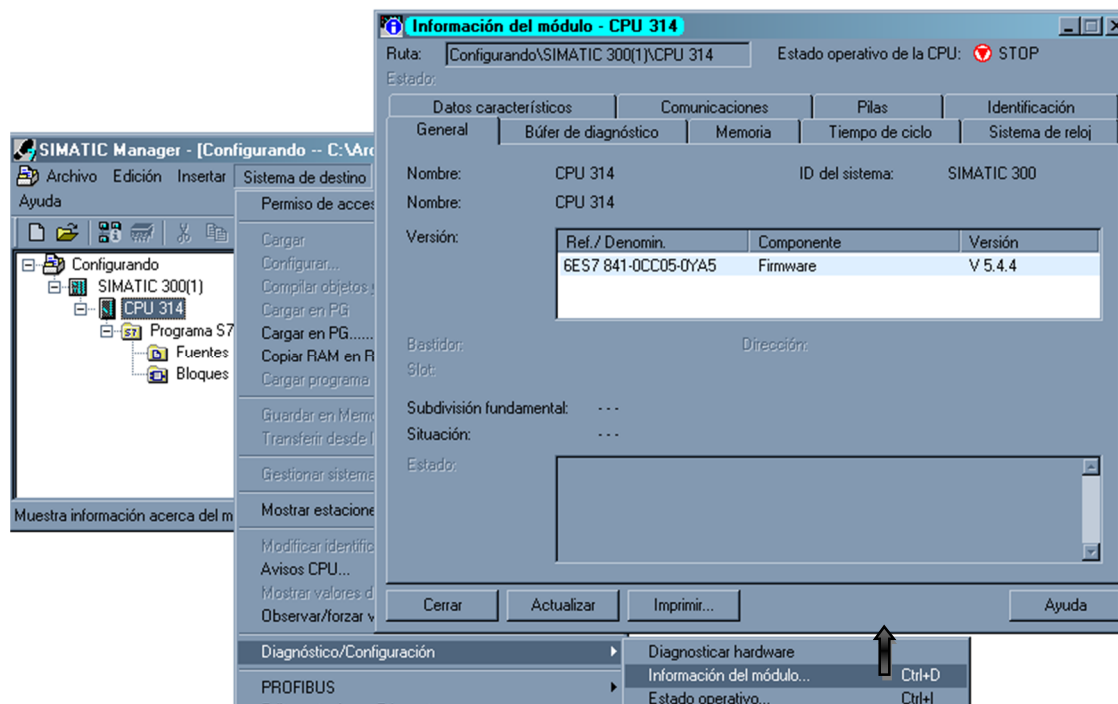
La función información del módulo (ver figura 49) lee los datos más importantes del módulo directamente conectado. Encontrara información adicional en los separadores adicionales:

- General: entre otras cosas, la descripción del módulo, revisiones de firmware y hardware.
- Buffer de diagnóstico: contiene todos los sucesos de diagnóstico en el orden en el que ocurrieron. Todos los sucesos se listan en la pantalla en texto normal y en el orden en el que ocurrieron.
- Memoria: tamaño y uso de la memoria de carga EPROM, la memoria de carga RAM y la memoria de trabajo.
- Tiempo de ciclo: se visualiza el tiempo de supervisión seleccionado, el más corto, el más largo y el tiempo de ciclo actual.
- Sistema de reloj: muestra el reloj en tiempo real y el medidor integrado en tiempo real.
- Datos característicos: muestra los bloques de sistema integrados y aquellos bloques de organización disponibles así como las áreas de direcciones (E, A, M, T, Z, L).
- Comunicaciones: muestra los datos de rendimiento de las interfases de comunicación y una visión general de la conexión.
- Pilas: la información de los contenidos de la U Stack, B Stack y L Stack. Para esto, la CPU debe estar en estado de STOP o haber alcanzado un punto de parada.

#### 1.3.11.4. Observar / forzar variables

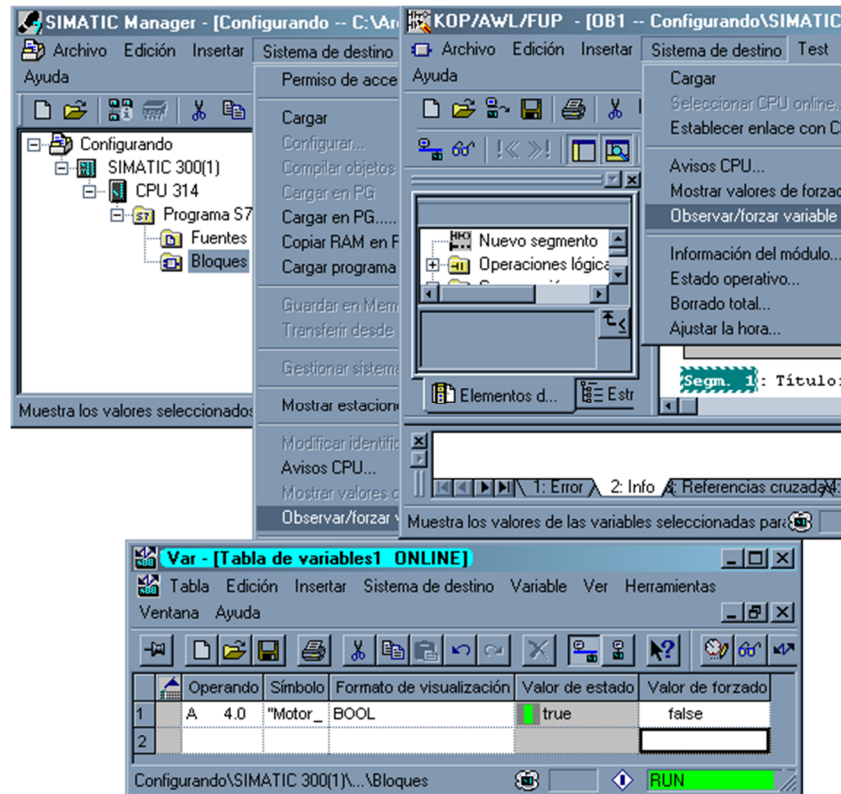
La función de test Observar/Forzar Variables puede arrancarse desde el Administrador SIMATIC o desde el editor LAD/STL/FBD (ver figura 50). Se usa para observar y / o modificar variables en el formato que se elija. Las tablas de variables ofrecen la ventaja de que pueden guardar distintos entornos de test. Así se pueden reproducir sin esfuerzo tests y observaciones durante la puesta en marcha o para operaciones de servicio técnico o de mantenimiento. El número de tablas de variables que se pueden almacenar es ilimitado.

Figura 49. Información del módulo



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

Figura 50. Observar / forzar variables



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

Se puede observar y/o modificar todas las variables y direcciones, con la excepción de variables locales y temporales. Las columnas de la tabla de variables que puede visualizar, pueden seleccionarse usando el menú Ver y tienen los siguientes significados:

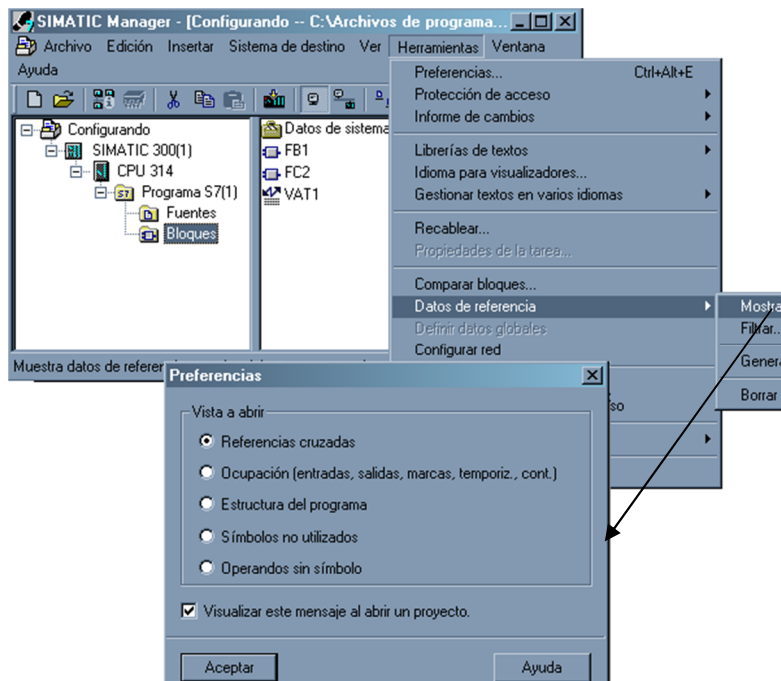
- Dirección: dirección absoluta de la variable.
- Símbolo: nombre simbólico de la variable.
- Comentario de símbolo: comentario sobre la variable visualizada.

- Formato de visualización: el formato de datos, que puede elegir mediante el clic del ratón (por ejemplo, binario, decimal...), en el cual serán visualizados los contenidos de la tabla de variables.
- Valor de estado: valor de la variable en el formato seleccionado.
- Valor de forzado: valor que se va a asignar a la variable.

### 1.3.11.5. Datos de referencia

Para programas extensos, es especialmente necesario para el diagnóstico tener una visión general de dónde está escaneada o asignada cada dirección, qué entradas y salidas están en uso, o cómo está estructurado el programa de usuario desde el punto de vista de la jerarquía de llamadas.

Figura 51. Datos de referencia



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.



La herramienta Datos de Referencia le da una visión general de la estructura del programa de usuario además de las direcciones usadas (ver figura anterior). Los datos de referencia son generados desde el programa de usuario abierto offline. Si, por ejemplo, una operación lógica no se lleva a cabo porque una marca no está activada, se puede usar los Datos de Referencia para averiguar dónde está asignada esa marca.

Los datos de referencia consisten en varias listas que son visualizadas como datos filtrados individualmente. Cuando se selecciona Mostrar Datos de referencia, se puede elegir en el cuadro de diálogo herramientas qué lista va a ser visualizada en primer lugar. Entonces puede elegir cualquiera de las diferentes listas.

#### **1.4. Puesta en marcha del hardware**

Los pasos individuales para ejecutar la puesta en marcha del hardware son las siguientes: ejecutar un *reset* de memoria CPU, llevar a cabo un rearranque en caliente de la CPU, comprobar los LED en los módulos, arrancar el Administrador SIMATIC, cargar la configuración hardware, comprobar entradas con la herramienta observar/forzar variables y comprobar salidas con la herramienta observar/forzar variables.

#### **1.5. Convertir archivos de S5 a S7**

El objetivo de la aplicación convertir archivo S5 es convertir a S7 el conjunto de los comandos de programas S5 ya existentes del modo más completo posible. A estos efectos la conversión de los programas S5 existentes se efectúa siempre a AWL S7.

El proceso de conversión completo se divide en 4 pasos:

- A. Prepare los archivos S5 a convertir.
- B. Realice la conversión con ayuda de la aplicación convertir archivo S5. En la misma:
  - Se sustituyen comandos S5 por comandos o secuencias de comandos (macros) S7 equivalentes.
  - Se sustituyen llamadas de bloques de función estándar S5 por las correspondientes funciones S7.
  - Se transfieren comentarios S5 al programa S7.
  - Se modifica la lista de asignación S5 de acuerdo con la conversión de los bloques.
- C. Reprocese el programa convertido. Tras efectuar la conversión, en una ventana de mensajes aparte se muestra en qué lugar del programa han aparecido errores durante la conversión. A continuación deberá tomar medidas de corrección para eliminarlos.
- D. Compile el archivo S7 concluido en el programa editor (AWL).

Después de instalar el Software STEP 7 en la PG, arranque el programa Convertir archivo S5 partiendo del botón de comando Inicio de la barra de tareas de Windows: la vía de acceso al comando de menú es *Simatic/STEP7/Convertir archivo S5*.

## **1.6. EI PLC S5**

El autómatas programable SIMATIC S5 se utiliza a nivel mundial en casi todos los sectores, y para las aplicaciones más diversas. Tiene estructura modular, y las diversas funciones de automatización son realizadas por diferentes tarjetas. El sistema le ofrece diferentes posibilidades de

comunicación y una gama escalonada de aparatos de operación, observación y programación. El lenguaje STEP 5 y un extenso catálogo de software posibilitan una programación muy sencilla.

Con el lenguaje de programación STEP 5 unificado para la familia SIMATIC S5 son posibles las siguientes formas de representación: KOP, FUP Y AWL. Cada forma de representación tiene sus particularidades. Por ello, un módulo de programa que haya sido escrito en AWL, no puede sacarse así como así en FUP o KOP. Las formas de representación gráficas tampoco son compatibles entre sí, sin embargo, siempre es posible traducir a AWL los programas escritos en FUP o KOP.

### **1.6.1. Campo de aplicación**

Campos de aplicación son, entre otros:

- Industria automovilística: unidades automáticas de taladrado y prueba, de montaje, líneas de pintura y bancos de prueba de amortiguadores.
- Industria alimentaria: equipos cerveceros y centrífugas.
- Industria del plástico: máquinas sopladoras, máquinas de inyección, máquinas de termoconformado y fabricación de fibras sintéticas.
- Industria pesada: líneas de moldes, hornos industriales, trenes de laminación, plantas de incineración y sistemas de regulación de temperatura en pozos.
- Industria química: instalaciones dosificadoras y instalaciones de mezcla.
- Construcción de máquinas: unidades de control de máquinas, máquinas empaquetadoras, máquinas-herramienta, mandrinadoras, máquinas de elaboración de la madera, centrales de señalización de averías, máquinas de soldadura y máquinas especiales.

- Servicios de edificios: control de ascensores, aire acondicionado, ventilación e iluminación.
- Sistemas de transporte: almacenes automatizados, dispositivos de transporte y clasificación, dispositivos de transporte y grúas.
- Energía, gas, agua, aire: controles de bombas, tratamiento del agua, filtros, estaciones de bombeo, tratamiento del aire, recuperadores de gas y grupos electrógenos.

### **1.6.2. Componentes del sistema**

El sistema tiene estructura modular. Sus diferentes componentes son:

- Fuentes de alimentación.
- Tarjetas centrales (CPU).
- Tarjetas de entrada y salida.
- Tarjetas preprocesadoras de señal (IP) y procesadores de comunicaciones (CP).

## **2. SENSORES Y ACTUADORES**

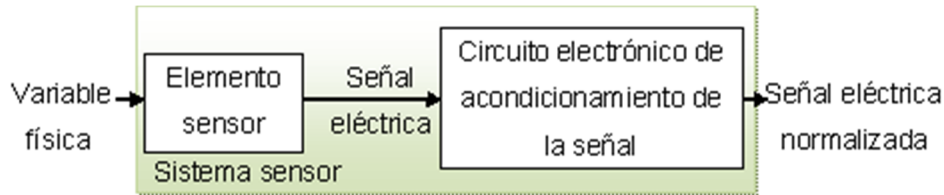
### **2.1. Sensores industriales**

Es un sensor todo dispositivo que tiene algún parámetro (valor de resistencia, desplazamiento, etc.) que es función del valor de una determinada variable física del medio en el cual está situado (presión, nivel, temperatura etc.). Por lo tanto el dispositivo sensor convierte una variable física en otra en general diferente y, por ello, se le suele denominar transductor. Además, la existencia de sistemas que memorizan, amplifican y, en general, procesan señales eléctricas, hace que la mayoría de los transductores conviertan variables no eléctricas en eléctricas.

Por otra parte, el tipo de señal eléctrica portadora de información y sus parámetros varían de un dispositivo sensor a otro, por lo que es necesario acoplar la salida del sensor a un circuito que, de acuerdo con las características de aquella, amplifica, filtra, corrige y/o convierte la señal proporcionada por el elemento sensor. Dicho circuito recibe el nombre de circuito de acondicionamiento. Se obtiene así un sistema como el representado en la figura siguiente que puede ser denominado Sistema sensor, aunque normalmente se lo denomina simplemente Sensor.

Dicho sistema genera una señal normalizada ya sea por el fabricante o siguiendo pautas establecidas por organismos de normalización, como por ejemplo la IEC (acrónimo de International Electrotechnical Commission), el IEEE (acrónimo de Institute of Electrical and Electronic Engineers), etc.

Figura 52. **Componentes básicos de un sensor**



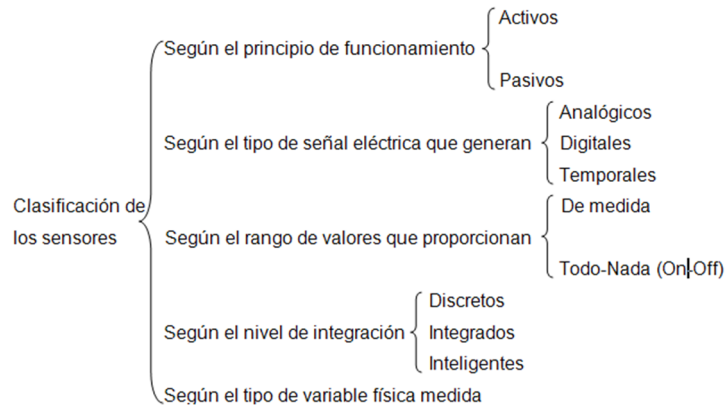
Fuente: elaboración propia.

A los sistemas sensores adecuadamente contruidos para trabajar en las condiciones existentes en un entorno industrial (temperatura elevada, presencia de polvo, humedad relativa alta, etc.) se los denomina en este trabajo de graduación Sensores Industriales. Los sensores industriales son, en la gran mayoría de los casos, sistemas sensores, como el representado en la figura anterior o sensores inteligentes, como los descritos en el apartado 2.1.1.4.3., encapsulado adecuadamente.

### **2.1.1. Características de los sensores industriales y clasificación**

Tal como se indica en el apartado anterior, son muy numerosas las variables o magnitudes físicas susceptibles de ser transformadas en señales eléctricas. Pero, además, las señales eléctricas pueden contener la información en un número elevado de parámetros diferentes (la amplitud de la tensión, la frecuencia, etc.) y por ello los sensores se pueden clasificar de acuerdo con un conjunto de características diferentes y no excluyentes que se indican en la tabla I y se analizan en sucesivos apartados.

Tabla I. **Clasificación de los sensores**



Fuente: elaboración propia.

### 2.1.1.1. **Clasificación de los sensores industriales según el principio de funcionamiento del elemento sensor**

La clasificación de los sensores industriales según el principio de funcionamiento del elemento sensor son dos: sensores activos o generadores y sensores pasivos o moduladores.

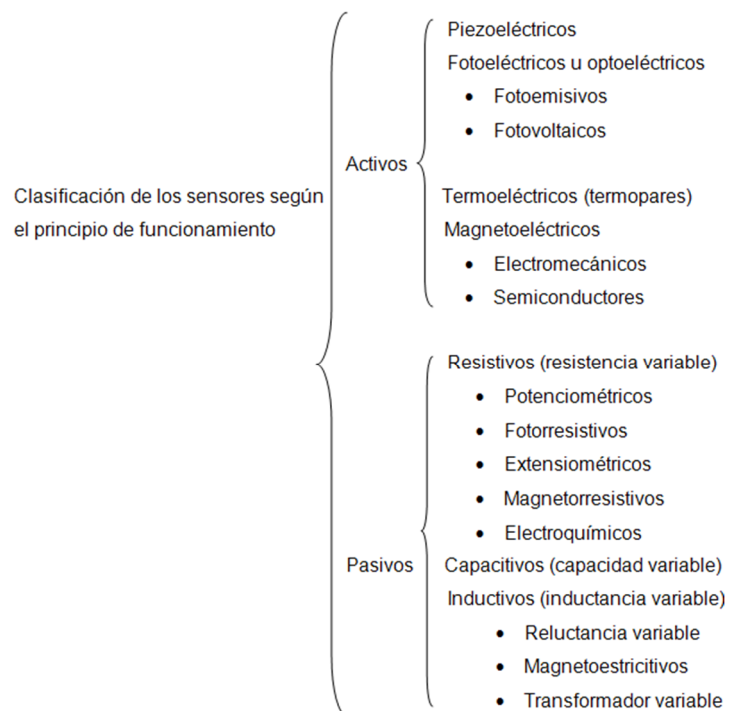
#### 2.1.1.1.1. **Sensores activos o generadores (*self generating*)**

Se consideran activos los sensores en los que la magnitud física a medir proporciona la energía necesaria para la generación de la señal eléctrica de salida. Son un ejemplo de sensores activos los basados en los efectos piezoeléctrico y termoeléctrico.

### 2.1.1.1.2. Sensores pasivos o moduladores (*modulating*)

Son pasivos los sensores en los que la magnitud física a medir se limita a modificar alguno de sus parámetros eléctricos característicos, por ejemplo la resistencia, la capacidad, etc. Los sensores de este tipo se caracterizan por necesitar una tensión de alimentación externa. Son ejemplo de sensores pasivos los basados en las resistencias cuyo valor depende de la temperatura (termorresistivos) o de la luz (fotorresistivos).

Tabla II. **Clasificación de los sensores según el principio de funcionamiento**



Fuente: elaboración propia.



En la tabla II se indican los principales sensores activos y pasivos. Es conveniente resaltar que una determinada magnitud física puede ser convertida, en general, mediante sensores que utilizan principios de funcionamiento distintos.

#### **2.1.1.2. Clasificación de los sensores según el tipo de señal eléctrica que generan**

Los sensores pueden clasificarse según el tipo de señal eléctrica que generan, los más importantes son: sensores analógicos y sensores digitales.

##### **2.1.1.2.1. Sensores analógicos**

Los sensores analógicos generan señales eléctricas denominadas análogas, que pueden tomar cualquier valor dentro de unos determinados márgenes y que llevan la información en su amplitud. Su diagrama de bloques es el representado en la figura siguiente. Las señales eléctricas analógicas se pueden clasificar a su vez en variables o continuas.

Las señales analógicas variables son aquellas que equivalen a la suma de un conjunto de senoides de frecuencia mínima mayor que cero. Un caso típico es una señal senoidal de frecuencia constante que representa la información mediante su amplitud (figura 54a) y las señales de audiofrecuencia. Las señales analógicas continuas son aquellas que se pueden descomponer en una suma de senoides cuya frecuencia mínima es cero. Se trata de señales que pueden tener un cierto nivel fijo durante un tiempo indefinido (figura 54b) y que representan también la información mediante su amplitud.

Las señales analógicas pueden ser también unipolares o bipolares. Las unipolares sólo pueden ser positivas o negativas con respecto a un terminal de referencia, mientras que las bipolares pueden ser tanto positivas como negativas (figura 55).

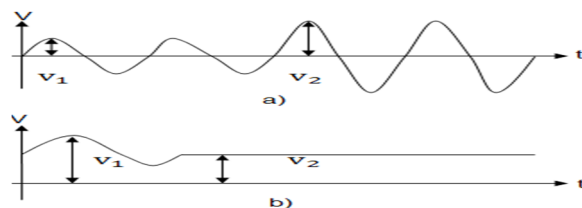
Las señales eléctricas generadas por los elementos de los sensores analógicos adolecen de problemas relacionados con la presencia de ruido, interferencias y distorsión, sobre todo si han de transmitirse a distancias relativamente grandes, y por ello es imprescindible el circuito electrónico de acondicionamiento de la señal indicado en la figura 54. La figura 56 muestra un sensor que genera una variable analógica en el rango de 0 a 10 V.

Figura 53. **Diagrama de bloques de un sensor analógico**



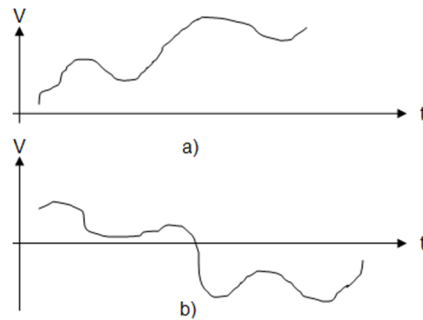
Fuente: elaboración propia.

Figura 54. **Señales analógicas: a) Señoidal de frecuencia constante; b) Continua**



Fuente: elaboración propia.

Figura 55. **Señales analógicas: a) Unipolar; b) Bipolar**



Fuente: elaboración propia.

Figura 56. **Sensor optoelectrónico láser de proximidad 3RG70-1CM00 de Siemens**



Fuente: SIEMENS, S. A.

#### **2.1.1.2.2. Sensores digitales**

Los sensores digitales generan señales eléctricas que solo toman un número finito de niveles o estados entre un máximo y un mínimo, y por ello reciben el nombre de digitales. Las más utilizadas son las binarias que solo pueden tener dos niveles de tensión, que se asignan a los números binarios 0 y 1. Una variable binaria recibe el nombre de “bit”. Para representar una

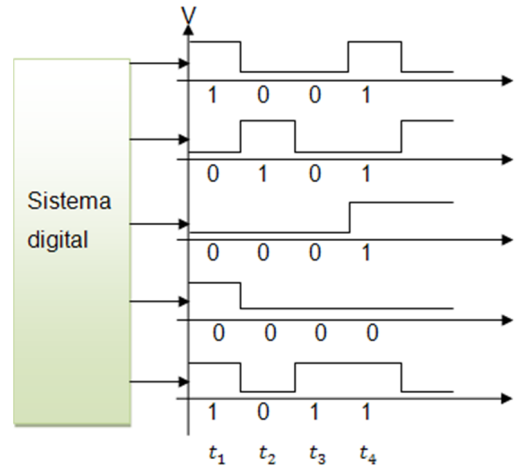
información se necesita un cierto número  $n$  de variables binarias cuyo valor depende de la precisión que se desea. Las  $n$  variables binarias se pueden representar de dos formas diferentes:

- Mediante señales binarias independientes. Se tiene así el número 10011 en un único instante  $t_1$ . En instantes sucesivos se pueden tener números diferentes (por ejemplo, 01000 en  $t_2$ , en la figura 57). Este formato recibe el nombre de paralelo.
- Mediante una secuencia de niveles cero y uno de una señal digital. En la figura 58 se indica una señal digital binaria que representa el número 1001 en el sistema de numeración binaria. Este formato recibe el nombre de serie.

El formato serie se utiliza para transmitir a distancia una información digital, mientras que el paralelo es el utilizado por los procesadores digitales y el que proporcionan algunos tipos de sensores, por ejemplo los codificadores lineales y angulares de tipo absoluto.

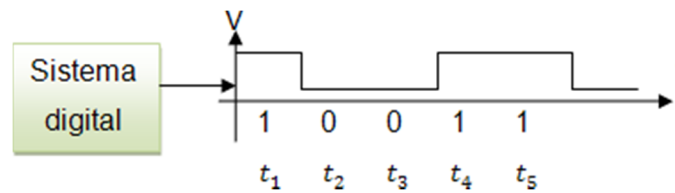
En la figura 59 se representa el diagrama de bloques de un sensor digital de salida en paralelo. Cuando el elemento sensor proporciona una señal analógica el esquema de bloques típico es el representado en la figura 60. El circuito de acondicionamiento adapta las características de la señal de salida del elemento sensor a las que necesita el convertidor en su entrada. Si al sensor digital de la figura 59 se le añade un procesador de comunicaciones, se convierte en un sensor con salida serie. Son sensores de este tipo los inteligentes que se describen el apartado 2.1.1.4.

Figura 57. **Señal digital binaria en formato paralelo**



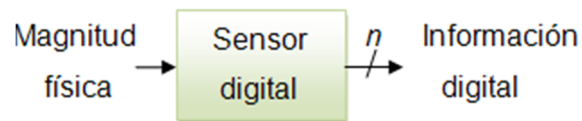
Fuente: elaboración propia.

Figura 58. **Señal digital binaria en formato serie**



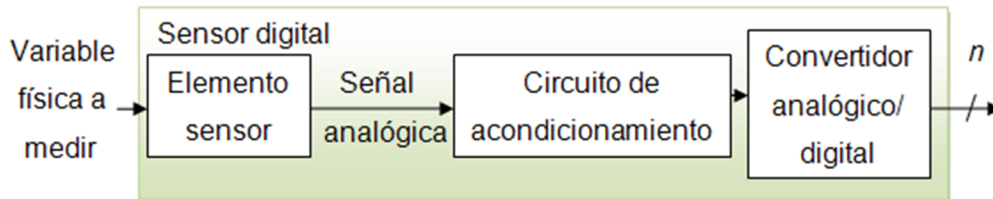
Fuente: elaboración propia.

Figura 59. **Diagrama de bloques de un sensor digital con salida en paralelo**



Fuente: elaboración propia.

Figura 60. **Esquema de bloques típicos de un sensor digital cuyo elemento sensor proporciona una señal analógica**



Fuente: elaboración propia.

### **2.1.1.3. Clasificación de los sensores según el rango de valores**

La clasificación de los sensores según el rango de valores que se mencionan en este trabajo de graduación y que se ajustan a lo que se implementa son: sensores de medida y sensores todo-nada.

#### **2.1.1.3.1. Sensores de medida**

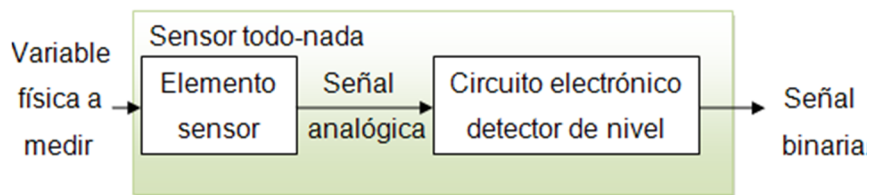
Son los que proporcionan a la salida todos los valores posibles correspondientes a cada valor de la variable de entrada dentro de un determinado rango. Pueden ser analógicos, digitales o temporales. Ejemplos de sensores de medida son un sensor analógico resistivo de temperatura y un sensor temporal incremental de posición.

#### **2.1.1.3.2. Sensores todo-nada**

Son los que solamente detectan si la magnitud de la variable de entrada está por encima o por debajo de un determinado valor. Proporcionan a la salida

una señal eléctrica que sólo puede tomar dos valores y por ello a veces se les denomina, con un exceso de simplificación, sensores digitales. Son ejemplos típicos los sensores que indican si el valor de una magnitud física está por encima o por debajo de un cierto nivel cuyo diagrama de bloques típico se representa en la figura siguiente.

Figura 61. **Esquema de bloque de un sensor todo-nada cuyo elemento sensor proporciona una señal analógica**



Fuente: elaboración propia.

Son también sensores todo-nada los que indican si se ha producido o no una determinada circunstancia, como la presencia o ausencia de un objeto en las proximidades del sensor.

#### **2.1.1.4. Clasificación de los sistemas sensores según el nivel de integración**

La clasificación de los sistemas sensores según el nivel de integración pueden ser discretos, integrados e inteligentes, estos son expuestos a continuación.

#### **2.1.1.4.1. Sensores discretos**

Reciben esta denominación los sistemas sensores en los que el circuito de acondicionamiento se realiza mediante componentes electrónicos separados e interconectados entre sí.

#### **2.1.1.4.2. Sensores integrados**

Los sensores integrados se caracterizan por el elemento sensor y el circuito acondicionador, o al menos este último, están construidos en un único circuito integrado monolítico o híbrido. Son ejemplos típicos muchos sensores, basados en las características de los semiconductores, que miden temperatura, humedad, presión etc.

#### **2.1.1.4.3. Sensores inteligentes**

No existe un consenso generalizado en relación con la definición de sensor inteligente. En numerosas ocasiones, la salida del circuito acondicionador debe ser modificada para llevar a cabo una o más de las siguientes tareas:

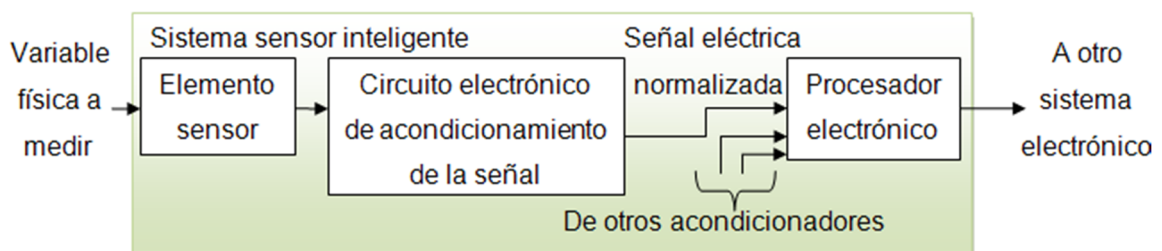
- Corregir no linealidades.
- Verificar el correcto funcionamiento del elemento sensor y el circuito acondicionador asociado.
- Transmitir la información a distancia.

Para ello es de gran utilidad añadir al sistema sensor de la figura 52 un procesador electrónico que realiza un algoritmo de corrección, diagnóstico, verificación del funcionamiento y/o comunicación. Cuando dicho conjunto se



realiza en un solo circuito integrado monolítico o híbrido se tiene un sensor con capacidad de proceso de información al que se suele denominar Sensor Inteligente (*Smart sensor*), cuyo diagrama de bloques se representa en la figura siguiente.

Figura 62. **Diagrama de bloques de un sensor inteligente**



Fuente: elaboración propia.

Un sensor inteligente tiene capacidad para realizar la mayoría de las funciones siguiente:

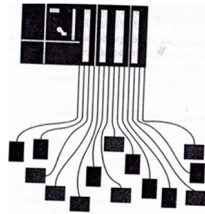
- Cálculos numéricos
- Comunicación en red (no una simple conexión punto a punto)
- Autocalibración y autodiagnóstico
- Múltiples medidas con identificación del sensor

Los sensores discretos e integrados deben conectarse a un procesador digital como, por ejemplo, un autómata programable a través de conexiones independientes tal como se indica en la figura siguiente. Ello hace que se complique excesivamente el cableado cuando el número de sensores y/o su distancia al autómata programable son elevados. Por ello los fabricantes de equipos de automatización comercializan sensores industriales inteligentes que

poseen un procesador electrónico digital con capacidad de comunicación con otros procesadores a través de una línea única de comunicaciones que suele recibir el nombre genérico de bus de campo (*field bus*) (figura 68).

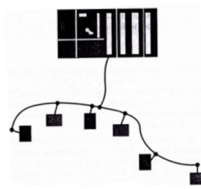
Son ejemplos de sensores con capacidad de conexión a un bus de campo, el sensor optoelectrónico de proximidad BERO 3RG673 01-1RM00 con interfaz al bus de campo AS-i incorporado (figura 69) y el sensor de caudal MG711/A de la serie SITRANS FM de Siemens conectable al bus Hart (figura 70).

Figura 63. **Conexión de sensores discretos y/o integrados a un autómata programable**



Fuente: SIEMENS, S. A.

Figura 64. **Conexión de un conjunto de sensores inteligentes a un autómata programable mediante un bus de campo**



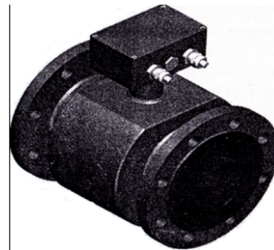
Fuente: SIEMENS, S. A.

Figura 65. **Sensores optoelectrónico de proximidad BERO 3RG673 01-1RMOO conectable al bus de campo AS-Interface**



Fuente: SIEMENS, S. A.

Figura 66. **Sensor de caudal MG711/A conectable al bus de campo Hart**

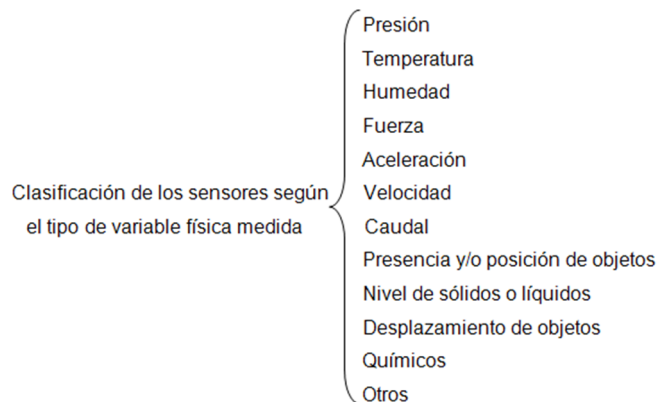


Fuente: SIEMENS, S. A.

#### **2.1.1.5. Clasificación de los sensores según la variable física medida**

Otra clasificación útil de los sensores es la realizada en función del tipo de variable física que convierten en eléctrica. En la tabla III se representa dicha clasificación, en la que se incluyen las variables físicas más importantes que es necesario medir en los procesos y productos industriales. Mediante la combinación de la tabla II y la tabla III se obtiene la tabla IV que constituye una guía general para seleccionar el sensor utilizable en diferentes aplicaciones.

Tabla III. **Clasificación de los sensores según el tipo de variable física medida**



Fuente: elaboración propia.

## 2.1.2. Características de entrada de los sensores industriales

Las características de entrada de los sensores industriales son: el campo o rango de medida y forma de variación de la magnitud de entrada, cada uno de esta subdivisión se explica a continuación.

### 2.1.2.1. Campo o rango de medida

El campo o rango de medida de un sensor se define como el conjunto de valores de la magnitud a medir que están comprendidos dentro de los límites superior e inferior de la capacidad de medida del sensor. Se indica mediante la especificación de los valores extremos.

Tabla IV. **Tabla que indica las variables físicas que se pueden medir con sensores basados en distintos principios de funcionamiento**

		Variable física medida										
		Posición	Desplazamiento	Velocidad	Aceleración	Tamaño	Nivel	Presión	Fuerza	Proximidad	Temperatura	Radiación luminosa
Principio de funcionamiento	Microrruptores	X				X						
	Finales de carrera	X										
	Extensiómetros	X	X	X	X			X	X			
	Termorresistivos										X	
	Magnetorresistivos	X	X	X								
	Capacitivos	X	X		X		X	X	X	X		
	Inductivos	X	X	X	X			X	X	X		
	Optoelectrónicos	X	X	X						X		
	Piezoeléctricos		X	X	X			X	X			
	Fotovoltaicos											X
	Ultrasónicos	X					X					

Fuente: elaboración propia.

El rango puede ser unidireccional (por ejemplo: 0 a 5 cm), bidireccional simétrico (por ejemplo:  $\pm 45$  °C), bidireccional asimétrico (por ejemplo: -20 a +80 °C) o desplazado (por ejemplo: 50 a 100 kg/cm<sup>2</sup>). La diferencia algebraica entre los valores máximo y mínimo se denomina amplitud de la medición.

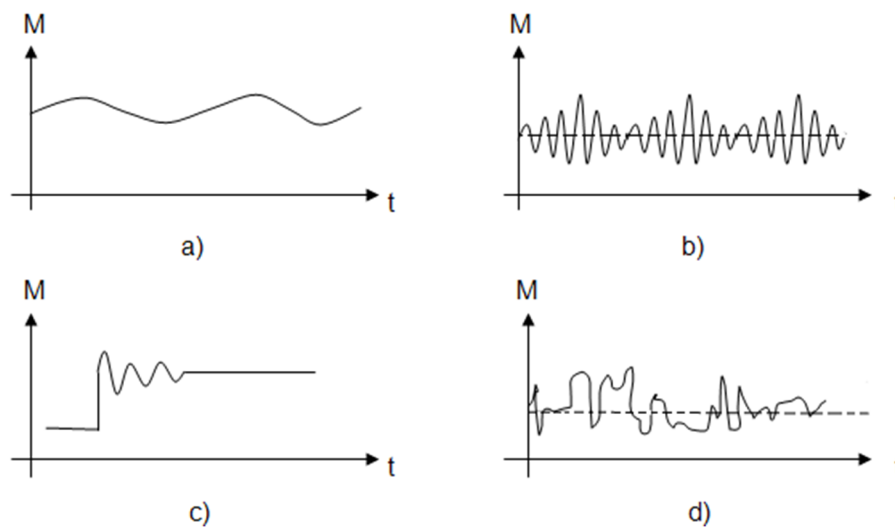
#### 2.1.2.2. Forma de variación de la magnitud de entrada

Otra característica a tener en cuenta en la elección de un sensor es la forma de variación en el tiempo de la magnitud que se desea medir. La importancia de esta característica reside en que pueden producirse grandes errores si la velocidad de los cambios de la variable a medir se sale del margen aceptado por el sensor.

Según el principio de medida en que se basan, hay sensores que sólo detectan magnitudes de variación lenta y hay otros que responden a variaciones rápidas.

De acuerdo con el modo de variación en función del tiempo, las variables o datos que se aplican a un sensor se pueden clasificar (figura siguiente) en a) estáticos, b) dinámicos, c) transitorios y d) aleatorios, que se analizan seguidamente.

Figura 67. **Formas de variación a lo largo del tiempo de la magnitud física que debe medir un sensor**



Fuente: elaboración propia.

### 2.1.3. Características eléctricas

En las características eléctricas se tiene: características eléctricas de salida la cual está dividida en; sensores de salida analógica y sensores de salida digital, características de alimentación y características de aislamiento.

### **2.1.3.1. Características eléctricas de salida**

En las características eléctricas de salida se tiene, como se había mencionado anteriormente; sensores de salida analógica y sensores de salida digital, que son las que se aplican a este trabajo de graduación.

#### **2.1.3.1.1. Sensores de salida analógica**

El parámetro de la señal de salida de un sensor analógico al que está asociada la información puede ser la tensión o la corriente. En el primer caso la impedancia de salida ( $Z_o$ ) del sensor, que se define como la impedancia a través de sus terminales de salida, debe ser mucho menor que la impedancia de entrada ( $Z_i$ ) de la carga conectada a él, que puede ser un aparato de medida o un procesador electrónico. Se logra así que resulte despreciable la caída de tensión en los cables que unen el sensor con la carga, pero surge el inconveniente que se tiene una baja inmunidad al ruido y además se puede producir la destrucción de la carga debido a la presencia de señales espúreas de elevada tensión en los cables, en especial si su longitud es elevada.

Los márgenes de tensión de salida más comunes de los sensores analógicos son: de 0 a 10 V, de 1 a 5 V, de -5 a +5 V y de -10 a +10 V. En ocasiones, el circuito de acondicionamiento forma parte del circuito de interfaz entre el sensor y el procesador digital. En este caso el sensor (por ejemplo, un termopar) está formado solamente por el elemento sensor que suele proporcionar una señal analógica de un nivel de tensión del orden del milivoltio.

Los inconvenientes de los métodos de transmisión en tensión hacen más adecuado técnicamente que la salida del sensor constituya una fuente de corriente que transmita a los cables una corriente proporcional a la magnitud a

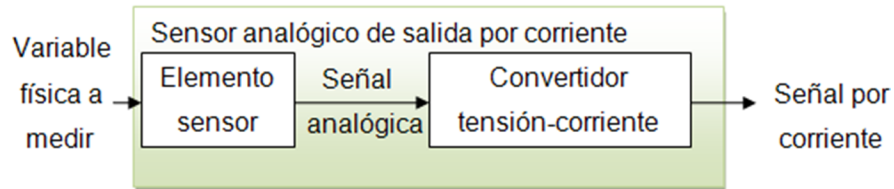
medir. Para ello la impedancia de salida del sensor ha de ser mucho más elevada que la de entrada de la carga.

La ventaja de este método consiste en que, al ser la corriente constante para cada valor de dicha magnitud, la transmisión no está influenciada por la variación de la impedancia de los cables porque la caída de tensión en ellos no tiene ningún efecto ya que la corriente en todos los puntos de un circuito serie es la misma. Por otra parte, el hecho que la impedancia de entrada de la carga sea mucho menor que la de salida del sensor hace que las señales parásitas (ruido) que aparecen en ella debido a la presencia de interferencia (ruido) sobre los cables, sean de muy pequeña amplitud. Los sensores analógicos con salida por corriente están contruidos de tal manera que admiten un cortocircuito permanente en su salida.

El interés de la transmisión por corriente propició la realización de convertidores tensión-corriente y su desarrollo en circuito integrado, así como su incorporación en el propio sensor, formando parte del circuito electrónico acondicionador (figura 68). Se han normalizado los niveles de corriente -20 a +20 mA, 4 a 20 mA y 0 a 20 mA. En la figura 69 se representa el sensor ultrasónico de proximidad con salida analógica de corriente de 4 a 20mA, modelo 3RG61 13-3BF01 de Siemens.



Figura 68. **Esquema de bloques típico de un sensor analógico de salida por corriente**



Fuente: elaboración propia.

Figura 69. **Sensor ultrasónico de proximidad del tipo 3RG61 13-3BF01, que posee una salida analógica por corriente de 4 a 20mA**



Fuente: SIEMENS, S. A.

### 2.1.3.1.2. **Sensores de salida digital**

El parámetro más importante de los sensores cuya señal de salida posee el formato digital es la corriente de carga máxima, que se define como la máxima corriente que puede circular, en uno u otro sentido, a través del terminal de salida.

Los niveles de tensión correspondientes a los estados diferenciados que puede adoptar la salida constituyen otra característica de gran importancia cuando la señal de salida del sensor se aplica a la entrada de otros circuitos electrónicos digitales (TTL, CMOS, etc.), en los que los niveles del 0 y del 1 solo

pueden tomar valores comprendidos dentro de un determinado margen. Para asegurar la compatibilidad entre los niveles de tensión y corriente de la salida del sensor y la entrada del circuito electrónico conectado a él es necesario conocer también la configuración de su etapa de salida que puede presentar diversas variantes, describiéndose a continuación las más utilizadas.

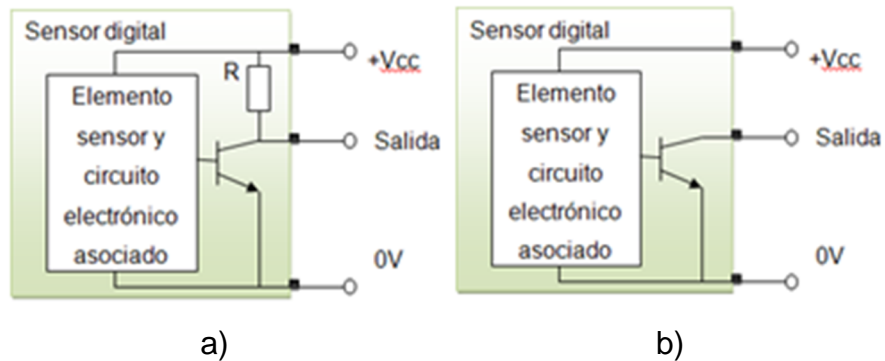
- Salida con transistor NPN y resistencia de carga

La configuración de cada terminal de salida es la representada en la figura 70 a) se caracteriza por proporcionar niveles de tensión y de corriente compatibles con las dos familias lógicas de uso más extendido que son la TTL y la CMOS.

- Salida con transistor NPN y colector abierto

Corresponde al esquema de la figura 70 b) que se diferencia de la anterior en que no tiene incorporada la resistencia R, por lo que recibe la denominación de colector abierto (*open collector*). Mediante la colocación de una resistencia R externa de valor adecuado, la salida se puede conectar a circuitos lógicos de diferentes tecnologías (TTL, CMOS, etc.). Se tiene también sensores de dos, tres y cuatro hilos que pueden ser PNP o NPN.

Figura 70. **Etapas de salida de sensores digitales: a) con transistores NPN y resistencia de carga; b) con transistor NPN en montaje de colector abierto**



Fuente: elaboración propia.

### 2.1.3.2. Características de alimentación

La tensión de alimentación puede ser continua o alterna, aunque generalmente es continua. Cuando la alimentación se realiza mediante una tensión continua es necesario conocer también el máximo rizado u ondulación admisible en ella. Se define la ondulación residual como la máxima tensión alterna, pico a pico, superpuesta a la tensión continua de alimentación, admisible para que el sensor industrial funcione correctamente. Se indica en forma porcentual y en muchos casos es admisible una ondulación residual del 10 %.

### 2.1.3.3. Características de aislamiento

Los parámetros que proporcionan dicha información suelen ser la resistencia de aislamiento y la tensión de ruptura o rigidez dieléctrica. La resistencia de aislamiento es la resistencia entre las partes, medida mediante la

aplicación de una tensión continua de un determinado valor. La tensión de ruptura o rigidez dieléctrica se puede definir como la máxima tensión que se puede aplicar entre las partes aisladas sin que se produzca arco eléctrico o sin que la corriente que circule entre ambas supere un valor determinado.

#### **2.1.4. Características mecánicas**

Las características mecánicas son; conceptos generales y grado de protección ambiental de los sensores, estos temas son importantes para la adecuada aplicación de un sensor dependiendo de las condiciones del lugar.

##### **2.1.4.1. Conceptos generales**

Generalmente los fabricantes especifican siempre:

- La configuración constructiva y sus dimensiones externas.
- Las instrucciones de montaje.
- El tipo, tamaño y localización de las conexiones eléctricas y mecánicas.
- La forma de realizar los ajustes externos (en caso de ser necesarios).
- El material de la carcasa.
- El grado de protección ambiental de la carcasa ante agentes externos.

##### **2.1.4.2. Grado de protección ambiental de los sensores industriales**

El grado de protección ambiental de los sensores industriales (también denominado grado de sellado) es una característica mecánica de especial importancia en todas aquellas aplicaciones, que constituyen los casos más frecuentes, en las que el sensor forma parte de un sistema hermético o está

sometido a determinadas condiciones de humedad, polvo, etc. Una de las normas desarrolladas por la IEC, aplicable a los sensores industriales, es la IEC 144, que especifica el grado de protección a la entrada de agentes externos sólidos o líquidos.

El grado de protección ambiental o sellado se indica mediante las siglas IP (*Ingress Protection*) seguidas de dos cifras decimales. La primera cifra indica el grado de protección frente al contacto y entrada de cuerpos sólidos externos y la segunda cifra el grado de protección frente a la entrada de líquidos. Cuanto más alto es el número de la primera y segunda cifra, mayor es el grado de protección de la carcasa. Por ejemplo, un grado de protección ambiental o sellado IP55 engloba a todos los grados inferiores, tales como IP22, IP23, IP34 e IP 54, por citar algunos de los más utilizados. En la tabla V se indica el tipo de protección que corresponde a los distintos valores de cada una de las cifras.

Tabla V. **Significado de las cifras del grado de protección ambiental de acuerdo con la norma IEC 144**

1.ª cifra	Grado de protección	2.ª cifra	Grado de protección
0	El equipo no está protegido contra la entrada de cuerpos sólidos externos	0	Sin protección
1	Protección contra la entrada de cuerpos sólidos externos grandes (mayores de 50 mm de diámetro)	1	Protección contra la condensación de gotas de agua
2	Protección contra la entrada de cuerpos sólidos externos de tamaño medio (mayores de 12 mm de diámetro)	2	Protección contra gotas de líquido; la caída de gotas de líquido no tiene efectos perjudiciales si la carcasa tiene una inclinación de hasta 15° desde la vertical

Continuación de la tabla V.

3	Protección contra la entrada de cuerpos sólidos externos mayores de 2,5 mm de diámetro	3	Protección contra lluvia o agua en forma de lluvia, para un ángulo menor o igual a 60° con respecto a la vertical
4	Protección contra la entrada de cuerpos sólidos externos pequeños (mayores de 1 mm de diámetro)	4	Protección contra salpicaduras de líquido en cualquier dirección
5	Protección contra depósitos perjudiciales de polvo. La entrada de polvo no se evita totalmente, pero este no puede entrar en cantidades suficientes para interferir el adecuado funcionamiento del equipo	5	Protección contra chorros de agua. El agua no produce efectos perjudiciales cuando es proyectada por un inyector en cualquier dirección bajo condiciones especificadas
6	Protección contra la entrada de polvo. Protección total frente al contacto con partes móviles situadas dentro de la carcasa	6	Protección contra condiciones del tipo de las de cubierta de barco (equipos herméticos de cubierta). El agua procedente de un fuerte oleaje no entra en la carcasa bajo condiciones especificadas
7		7	Protección contra la inmersión en agua bajo condiciones especificadas de presión y tiempo
8		8	Protección contra la inmersión indefinida en agua bajo condiciones especificadas de presión

Fuente: elaboración propia.

## **2.2. Actuadores**

Para que un sistema electrónico de control pueda comprobar un proceso o producto es necesario que pueda actuar sobre el mismo. Los dispositivos que realizan esta función reciben diversos nombres, entre ellos: accionamientos y actuadores. No existe una única definición de actuador aceptada de manera universal. Se considera, en general, que es todo “dispositivo que convierte una magnitud eléctrica en una salida, generalmente mecánica, que puede provocar un efecto sobre el proceso automatizado”<sup>1</sup>. Algunos actuadores realizan la función de cambiar niveles de energía eléctrica.

### **2.2.1. Tipos de actuadores**

Los actuadores, en función de la energía que transformen, se puede clasificar en: eléctricos, neumáticos, hidráulicos, térmicos, elásticos, gravitacional y de origen animal.

Los actuadores sobre los que se va a interesar en esta parte, por ser los que se utilizan fundamentalmente en las máquinas modernas, son los eléctricos, los neumáticos y los hidráulicos. En la tabla siguiente se resumen las principales características de estos tres tipos de actuadores.

---

<sup>1</sup> Sensores y Actuadores Industriales. Actuadores Industriales [en línea]. [http://tv.uvigo.es/uploads/material/video/1709/ISAD\\_tema7\\_2.pdf](http://tv.uvigo.es/uploads/material/video/1709/ISAD_tema7_2.pdf). [Consulta: 25 de marzo de 2013].

Tabla VI. **Características de los actuadores: neumático, hidráulico y eléctrico**

	<b>Neumático</b>	<b>Hidráulico</b>	<b>Eléctrico</b>
<b>Energía</b>	Aire a presión (5-10 bar)	Aceite mineral (50-100 bar)	Corriente eléctrica
<b>Opciones</b>	Cilindros Motor de paletas Motor de pistón	Cilindros Motor de paletas Motor de pistones axiales	Corriente continua Corriente alterna Motor paso a paso
<b>Ventajas</b>	Baratos Rápidos Sencillos Robustos	Rápidos Alta relación potencia-peso Autolubricantes Alta capacidad de carga Estabilidad frente a cargas estáticas	Precisos Fiables Fácil control Sencilla instalación Silenciosos
<b>Desventajas</b>	Dificultad de control continuo Instalación espacial (compresor, filtros) Ruidosos	Difícil mantenimiento Instalación especial (filtros, eliminación aire) Frecuentes fugas Caros	Potencia limitada

Fuente: elaboración propia.

### 2.2.2. Actuadores eléctricos

Los actuadores eléctricos son los más extendidos y los que poseen un mayor campo de aplicación dada la fácil disponibilidad de la energía eléctrica a través de las redes de distribución.

De hecho, hay actuadores que dependen de una etapa previa realizada por un accionamiento eléctrico, como son los actuadores neumáticos o hidráulicos, existen una gran cantidad de modelos y es fácil utilizarlos con motores eléctricos estandarizados según la aplicación (ejemplo, una máquina llenadora de bebidas, ver figura siguiente). En muchos casos es necesario utilizar reductores, debido a que los motores son de operación continua.

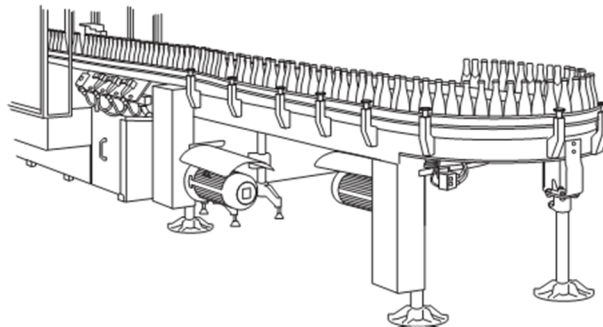


### 2.2.2.1. Tipos de actuadores eléctricos

Entre los tipos de actuadores eléctricos se pueden mencionar:

- Motores de corriente alterna
  - Monofásicos
  - Trifásicos
- Motores de corriente continúa
  - Bobinados en derivación
  - Bobinados en serie
  - Excitación independiente
- Motores paso a paso
- Servomotores
- Motor universal

Figura 71. **Actuadores en una máquina llenadora de bebidas**



Fuente: SIEMENS, S. A.

### **2.2.2.1.1. Criterio de selección**

Los factores que debe tener en cuenta para elegir un tipo de motor u otro son los siguientes:

- Par resistente de la carga.
- Velocidad de funcionamiento.
- Potencia (hay que tener en cuenta que la potencia será igual al par nominal de la aplicación por la velocidad nominal de la aplicación).
- Variaciones en la carga y en la velocidad (hay que tener en cuenta que hay motores que pueden llegar a pararse si la variación de la carga es excesiva).
- Par de arranque (en ocasiones hará falta el uso de un arrancador para poner en marcha un motor).
- Ciclo de trabajo (arranque, funcionamiento y parada).
- Voltaje de operación.
- Factores ambientales.
- Tipo de energía disponible.

### **2.2.3. Accionamientos continuos**

En el diseño de cualquier accionamiento intervienen dos elementos: el motor o accionamiento y el elemento receptor. Este último incluye la transmisión que transforma la energía suministrada por el motor para hacerla más adecuada a las necesidades del receptor.

Entre ambos elementos se puede dar cualquiera de estas tres fases: arranque, funcionamiento en régimen y parada. El funcionamiento del accionamiento durante estas tres fases se podrá determinar a partir de las

características mecánicas del motor y del receptor, siendo éstas sus curvas de par-velocidad.

#### **2.2.3.1. Motor de características mecánicas rígidas**

Aquel en el que la velocidad disminuye muy poco al aumentar el par, por ejemplo, motor de corriente continua en derivación, motor asíncrono cerca de su punto nominal, motor síncrono.

#### **2.2.3.2. Motor de características mecánicas elásticas**

Aquel en el que se da una variación importante de la velocidad al variar el par, por ejemplo, motor de corriente continua en serie, motor de gasolina.

#### **2.2.4. Accionamientos para movimientos rápidos**

En determinadas aplicaciones (robótica y automatización) es interesante que los accionamientos efectúen movimientos rápidos y precisos, por lo que usualmente se utilizan servomotores, cuya característica mecánica (par motor) es aproximadamente constante con la velocidad y regulable con la señal de mando.

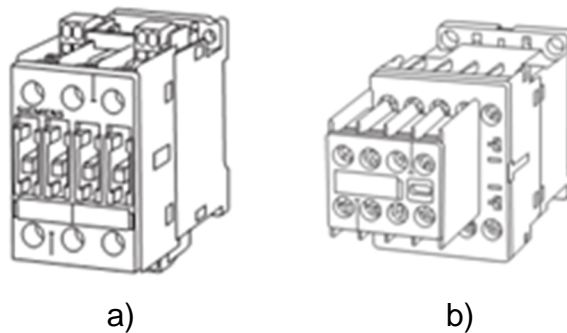
#### **2.2.5. Elementos de control**

En los elementos de control se pueden mencionar algunos: contactores, relés, columna de señalización, dispositivos variadores de velocidad, elementos de protecciones y otros actuadores.

### 2.2.5.1. Contactores

Son actuadores que sirven de interfase entre los mandos de control y los actuadores eléctricos de mayor potencia. Por medio de la excitación eléctrica de una bobina, el magnetismo creado por ella, atrae un dispositivo mecánico que a su vez conmuta uno o varios interruptores mecánicos que pueden manejar corrientes elevadas. Un ejemplo de contactor es el SIRIUS 3R IEC de Siemens, ver figura siguiente inciso a). Los contactores más comunes poseen bobinas de control a 110 o 220 voltios y contienen un juego de 4 interruptores conmutables, uno como auxiliar y los otros tres utilizados para las fases de la corriente trifásica utilizada comúnmente en la industria.

Figura 72. **Contactor SIRIUS 3R IEC a) y b) Relé SIRIUS 3RH de Siemens**



Fuente: SIEMENS, S. A.

### 2.2.5.2. Relés

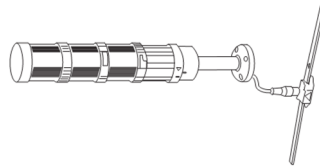
Son mecanismos electromagnéticos, que conmutan uno o varios contactos eléctricos por medio de la fuerza electromagnética, generada por paso de la corriente de control a través de su bobina. El relé es utilizado ampliamente en

circuitos de control. Un ejemplo de relé es el SIRIUS 3RH de Siemens, ver figura anterior inciso b).

### **2.2.5.3. Columna de señalización**

Las columnas de señalización son utilizadas para indicar al personal operador el monitoreo de máquinas o la operación de un proceso a distancia. Un ejemplo de columna de señalización es el 8WD42 de Siemens, ver figura siguiente.

Figura 73. **Columna de señalización 8WD42 de Siemens**

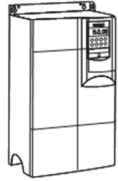


Fuente: SIEMENS, S. A.

### **2.2.5.4. Dispositivos variadores de velocidad**

La velocidad de un motor puede ser controlado usando algún tipo de dispositivo electrónico, referido como dispositivo de velocidad variable o ajustable. Los dispositivos de velocidad variable usados para controlar motores DC son llamados dispositivos DC. Los dispositivos de velocidad variable usados para controlar motores AC son llamados dispositivos AC (ver figura siguiente). El termino inversor es también usado para describir un dispositivo AC variador de velocidad. El inversor es solo una parte de un dispositivo AC, sin embargo, es común en la práctica referir a un dispositivo AC como un inversor.

Figura 74. **Dispositivo variador de velocidad**

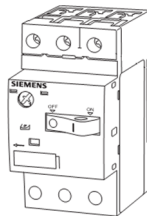


Fuente: SIEMENS, S. A.

### 2.2.5.5. Elementos de protección

Los elementos de protección pueden ser: los disyuntores y relé térmico, estos protegen a los equipos eléctricos por sobrecarga y sobrecorriente, evitando que estos dañen a todo un conjunto de equipos eléctricos dentro de un tablero.

Figura 75. **Guarda motor 3RV101 de Siemens**



Fuente: SIEMENS, S. A.

#### 2.2.5.5.1. Disyuntores

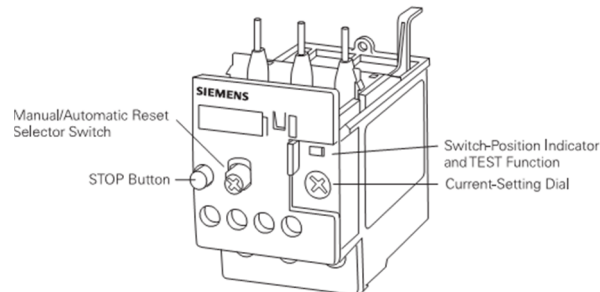
Son simplemente interruptores o bloqueadores, que se encargan de aislar la corriente de potencia de entrada, de los diferentes sistemas eléctricos controlados. Estos son instalados en serie con las líneas de potencia y vienen

diseñados para soportar determinadas corrientes de tal manera que si sobrepasa el límite, este se activa y aísla la corriente eléctrica de potencia. Deben ser instalados en el circuito antes del contactor. Un ejemplo es el guarda motor 3RV101 de Siemens, ver figura siguiente.

#### 2.2.5.5.2. Relé térmico

Su función es la de proteger los diferentes dispositivos de las sobrecorrientes. Deben ser instalados después del contactor, de tal manera que brinde seguridad en caso de que uno de sus contactos se quede pegado o no funcione correctamente, corriendo el riesgo de dejar solo dos fases, lo que ocasionaría grandes daños a los sistemas conectados a este. Un ejemplo de relé térmico bimetálico es el SIRIUS 3RU11 de Siemens, ver figura siguiente.

Figura 76. Relé térmico bimetálico SIRUS 3RU11 de Siemens



Fuente: SIEMENS, S. A.

#### 2.2.5.6. Otros actuadores

Otros actuadores que pueden mencionarse son: electroválvulas, servoválvulas, válvulas proporcionales, motores hidráulicos, etc.





### **3. COMUNICACIONES INDUSTRIALES**

Se pueden definir las comunicaciones industriales como el área de la tecnología que estudia la transmisión de información entre circuitos y sistemas electrónicos utilizados para llevar a cabo tareas de control y gestión del ciclo de vida de los productos industriales.

Las comunicaciones industriales deben, por lo tanto, resolver la problemática de la transferencia de información entre los equipos de control del mismo nivel y entre los correspondientes a los niveles contiguos.

En la década de 1980, las comunicaciones industriales comenzaron a realizarse mediante comunicaciones digitales punto a punto para facilitar la utilización de los computadores como sistemas de diseño asistido por computador del programa de control. Pero las conexiones punto a punto dan lugar a múltiples canales de comunicación, con el consiguiente incremento de complejidad. Por ello, para llevar a cabo dicha transferencia de información de la manera más eficaz posible, los equipos deben compartir canales de comunicación, lo cual incentivó el desarrollo de redes de comunicaciones que posean las características adecuadas.

#### **3.1. Comunicaciones digitales**

Las comunicaciones digitales (*data communications*), conocidos actualmente como telemática (combinación abreviada de las palabras Telecomunicación e Informática). Nacieron como resultado del progreso de la informática, que, a comienzos de la década de 1960, necesitó, para ampliar su

campo de aplicación, dotar a los computadores de terminales remotos y enviarle o recibir la información procedente de ellos a través del mínimo número de hilos. Para ello fue necesario convertir el formato paralelo al formato serie tanto la información generada en paralelo por la unidad central de proceso del computador como por los terminales.

Nació así el concepto de transmisión de datos, utilizado tradicionalmente para referirse a la comunicación de información digital en el formato serie a una cierta distancia superior a decenas de centímetros, y se inició el teleproceso o proceso de datos a distancia.

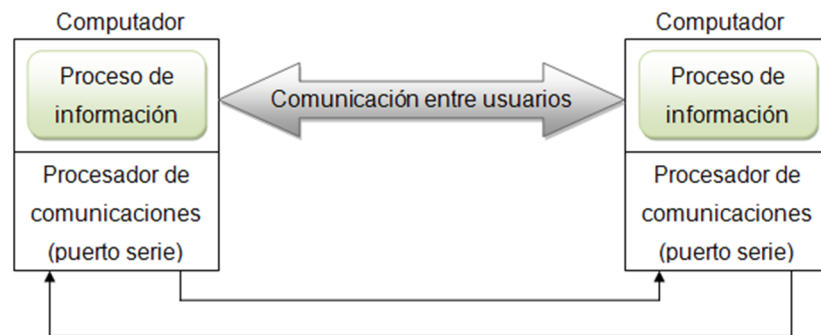
Por otra parte, la gran cantidad de conceptos asociados con las comunicaciones digitales hace que no resulte fácil su análisis y que sea necesario ir de lo particular a lo general, y analizar primero cada concepto por separado. Por ello es conveniente estudiar primero la comunicación entre un computador y un periférico, que recibe la denominación de comunicación punto a punto (*Point to Point*) y permite comprender los conceptos básicos relativos a las comunicaciones digitales. A partir de ello se procede al estudio de las redes de comunicación digitales, también denominadas redes de datos o redes informáticos (*Data Network Communications*) que constituye conexiones multipunto (*Multipoint or Multidrop*).

### **3.1.1. Conceptos básicos de las comunicaciones digitales punto a punto**

La comunicación punto a punto se puede definir como la transmisión de información en serie entre un computador y un periférico u otro computador, que no está compartida con ningún otro elemento de proceso de datos. En la figura siguiente se representa su esquema básico en banda base digital en el

que cada computador está asociado a una interfaz serie que se denomina procesador de comunicaciones, que algunos fabricantes de autómatas programables, como por ejemplo Siemens, denominan CP (acrónimo de *Communication Processor*) porque realizan las acciones adecuadas para llevar a cabo la comunicación.

Figura 77. **Esquema básico de la comunicación punto a punto en banda base digital entre dos computadores**



Fuente: elaboración propia.

La transmisión punto a punto está asociada al siguiente conjunto de conceptos básicos interrelacionados, que son además comunes a las comunicaciones multipunto:

- Los modos de transmisión de las señales
  - Transmisión en banda base
  - Transmisión en banda ancha
- Los modos de comunicación
  - La transmisión simple
  - La transmisión semidoble o doble no simultánea
  - La transmisión doble simultánea

- Los modos de sincronización
  - Transmisión asíncrona
  - Transmisión síncrona
    - Protocolos síncronos orientados al carácter
    - Protocolos síncronos orientados al contaje de octetos
    - Protocolos síncronos orientados al bit
- Los métodos de detección de errores
  - Detección de errores en la transmisión asíncrona
  - Detección de errores en la transmisión síncrona
- Los medios de transmisión
  - Medios guiados
    - Conductores
    - Cable coaxial
    - Fibra óptica
  - Medios no guiados
    - La atmósfera (transmisión inalámbrica)
- Las características mecánicas y eléctricas

#### **3.1.1.1. Normalización de las comunicaciones punto a punto**

La normalización de las comunicaciones punto a punto se tiene; bucle de corriente, Norma RS-232, Norma RS-422, Norma RS-423, Norma RS-485 y Norma USB.

#### **3.1.1.1.1. Bucle de corriente**

El bucle de corriente asigna, como su nombre indica, el cero y el uno lógicos a dos corrientes distintas como son la ausencia de corriente al cero y la corriente de 20 mA al uno. Nació para acoplar a un computador digital los periféricos electromecánicos como, por ejemplo, el teletipo.

Se caracteriza porque:

- Transmite la información en banda base digital
- Tiene una velocidad de transmisión baja
- Se puede realizar con aislamiento galvánico

En la actualidad el bucle de corriente no se utiliza en el diseño de las comunicaciones serie punto a punto porque ha sido sustituido por los bucles de tensión correspondientes a las Normas RS 232, RS 423, RS 422 Y RS 485.

#### **3.1.1.1.2. Norma RS-232**

La Norma RS-232 define los conceptos asociados con toda comunicación serie, tales como el protocolo de la capa de enlace, que establece la forma de sincronizar el transmisor y el receptor, la velocidad de transmisión, las características eléctricas y las características mecánicas.

En relación con las características eléctricas hay que destacar que la Norma RS-232 define un margen de tensión de +3 V a +15 V para el cero lógico y -3 V a -15 V para el uno lógico.

#### **3.1.1.1.3. Norma RS-422**

La Norma RS-422 representa la información mediante el estado cero o uno de un terminal y por ello se dice que es de salida única. La Norma RS-422, que implementa una comunicación asíncrona doble simultánea en banda base digital, utiliza en el transmisor un amplificador (*driver*) que representa la información mediante la diferencia de nivel entre sus dos salidas y en el receptor un amplificador diferencial. Se consigue así una mayor inmunidad al ruido, al igual que una mayor distancia y velocidad de transmisión, que mediante la Norma RS-232.

#### **3.1.1.1.4. Norma RS-423**

La Norma RS-423, es una solución intermedia entre la RS-232 y la RS-422, porque utiliza una salida única en el transmisor y una entrada diferencial en el receptor.

#### **3.1.1.1.5. Norma RS-485**

Las normas RS-232, RS-422 y RS-423 se desarrollaron para realizar conexiones punto a punto entre dos procesadores digitales o entre un procesador digital y un periférico. Pero el avance de la microelectrónica, que disminuyó el coste de los procesadores digitales, propició el desarrollo de una norma que permitiese conectar a dos hilos un número elevado de procesadores para realizar una comunicación serie entre ellos que recibe el nombre de bus serie, que en las comunicaciones industriales recibe la denominación de bus de campo. Dicha norma es la RS-485, que facilita la implementación de un bus serie con un máximo de 32 transmisores y otros tantos receptores.

#### **3.1.1.1.6. Norma USB**

El bus serie universal, conocido como USB (acrónimo de *Universal Serial Bus*), sustituye con ventaja a los descritos en los apartados anteriores y sus principales características son:

- Es una conexión punto a punto. Sustituye a la Norma RS-232 y por ello ha sido diseñado para conectar periféricos a un computador principal (host) o conectar este último a otros procesadores subordinados del mismo.
- Combina la transmisión síncrona y asíncrona.
- Permite el envío de paquetes de pequeña y gran longitud.
- Utiliza cuatro hilos.

#### **3.1.2. Redes informáticas**

Se define una red de datos o red informática (*Data Network Communications*) como un conjunto de computadores que pueden intercambiar información entre sí a través de un sistema de comunicaciones digitales.

La Organización Internacional de Normalización (ISO) creó el concepto de sistema informático abierto como aquel que es capaz de comunicarse con otros de acuerdo con unas normas determinadas y que estableciese un modelo denominado OSI (acrónimo de *Open System Interconnection*) para definirlo.

### **3.1.2.1. Conceptos específicos de las redes de datos**

Hay dos conceptos específicos de las redes de datos, que también deben ser tenidos en cuenta:

#### **3.1.2.1.1. La topología de la red**

La topología de una red de comunicaciones establece la forma en que se interconectan los diferentes procesadores que la constituyen.

#### **3.1.2.1.2. La extensión de la red**

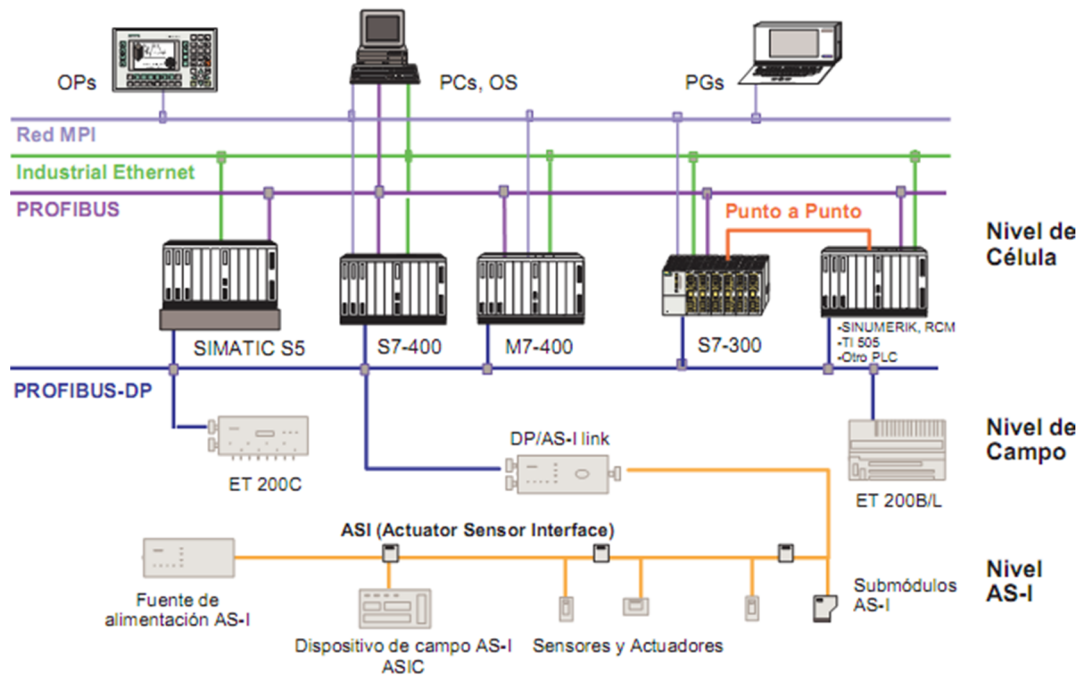
Según su extensión las redes informáticas se pueden clasificar en locales, metropolitanas y de área extensa. La extensión no es un parámetro independiente de la topología porque de ella depende el tipo de topología más adecuado para una aplicación determinada.

### **3.1.3. Subredes en SIMATIC**

SIEMENS ofrece las siguientes subredes para cumplir los distintos requerimientos de comunicación (ver figura siguiente) a nivel de célula (no críticos en el tiempo) y a nivel de campo (críticos en el tiempo).



Figura 78. **Subredes en SIMATIC**



Fuente: SIEMENS, S. A.

### 3.1.3.1. **MPI**

La subred MPI está diseñada para su uso a nivel de célula y es la interfase multipunto en SIMATIC S7. La interfase MPI es básicamente una interfase PG, es decir, diseñada para la conexión de PGs (para puesta en marcha y test) y OPs (interfase hombre-máquina). De todos modos, la subred MPI puede usarse para conectar en red un pequeño número de CPU.

### 3.1.3.2. **Industrial Ethernet**

Industrial Ethernet es la red para el control de planta y niveles de célula en el sistema de comunicación abierta e independiente del fabricante SIMATIC.

Está diseñado para la transmisión, no crítica en el tiempo, de grandes cantidades de datos y proporciona facilidades para la conexión a redes remotas a través de las puertas de enlace.

### **3.1.3.3. PROFIBUS**

PROFIBUS es la red para los niveles de comunicación de célula y campo en el sistema de comunicación abierto e independiente del fabricante SIMATIC. Existen dos versiones:

- PROFIBUS es para comunicaciones no críticas en el tiempo entre nodos inteligentes iguales a nivel de célula.
- PROFIBUS DP es el bus de campo para intercambio de datos cíclicos, críticos en el tiempo, entre maestros inteligentes y dispositivos de campo.

### **3.1.3.4. Enlace punto a punto**

Las conexiones punto a punto se usan principalmente para intercambio de datos, no críticos en el tiempo, entre dos estaciones o para conectar dispositivos tales como OP, impresoras, lectores de códigos de barras, lectores de tarjetas de banda magnética, etc. a un equipo.

### **3.1.3.5. AS Interfase**

El Actuador-Sensor-Interfase es una subred para el nivel de proceso más bajo en un sistema de automatización, permite conectar en red sensores y actuadores binarios. Son ejemplos de aplicaciones industriales en las que la red AS-i resulta especialmente indicada los sistemas de transporte, manutención,

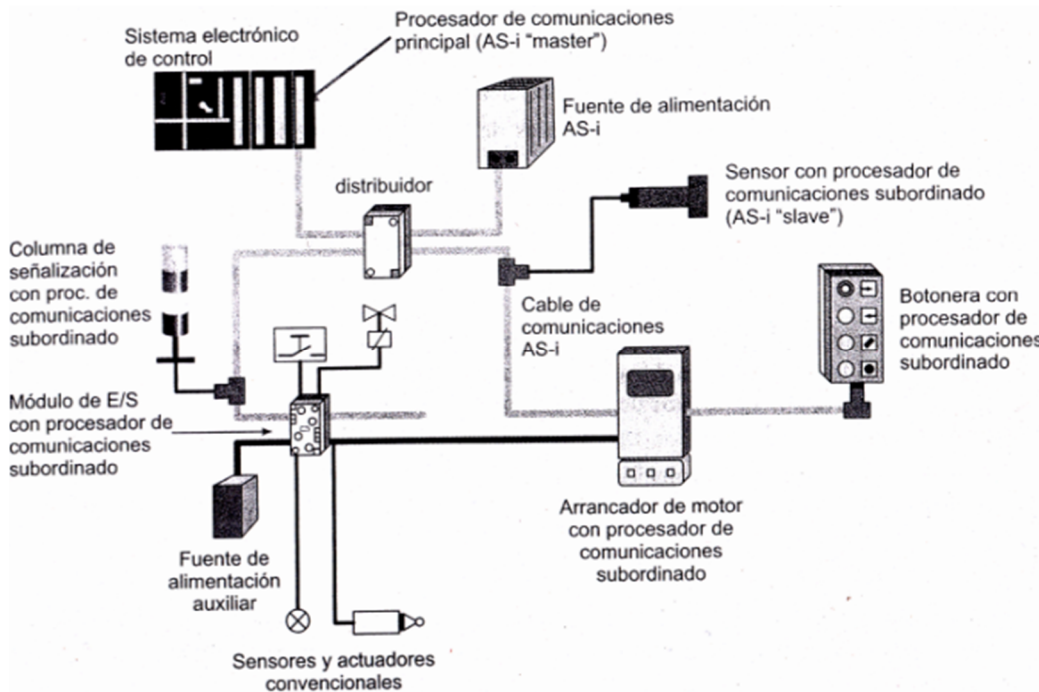
embalaje, ensamblaje, mecanizado, ventilación, climatización, aparcamiento, alumbrado, etc.

En la figura siguiente se representa un ejemplo de red AS-i real. El conjunto formado por un procesador de comunicaciones subordinado y los sensores conectados a él, se denomina estación o módulo subordinado. La distancia máxima de una red AS-i es de 100 metros, pero se pueden alcanzar distancias de 300 metros aproximadamente mediante elementos amplificadores de señal denominados repetidores AS-i. Además de las funciones de comunicación, la red AS-i proporciona, a través de un único cable de dos hilos, alimentación a los sensores y actuadores de bajo consumo.

Las principales características de la red AS-i son las siguientes:

- Reduce al mínimo el cableado necesario para conectar un autómata programable con los dispositivos de campo del proceso.
- Garantiza un ciclo de exploración de los sensores-actuadores todo-nada conectados a todos los nodos de la red cuya duración está limitada y depende del número de nodos.
- Es fácil de instalar, porque no necesita programas de configuración.
- Está normalizada tanto a nivel eléctrico como mecánico, lo que garantiza la modularidad y la intercambiabilidad de los productos.
- Sus nodos se pueden implementar con un elevado nivel de sellado IP67, lo que permite su instalación a pie de máquina (sin necesidad de protección adicional).
- Posee funciones complementarias de diagnóstico, que le proporcionan una elevada fiabilidad.

Figura 79. Ejemplo real de la red AS-i



Fuente: SIEMENS, S. A.

### 3.1.4. Métodos de comunicación S7

Los métodos de comunicación del S7 son; datos globales, comunicación básica y comunicación ampliada, estas se diferencian en el tamaño de datos a transmitir y el tipo de CPU que se puede conectar.

#### 3.1.4.1. Datos globales

Este método de comunicación permite el intercambio de datos entre CPU's, de forma cíclica, a través de la interfase MPI sin necesidad de programar. Los datos se intercambian en el punto del ciclo de scan en el que se actualiza la imagen de proceso. En el S7-400, el intercambio de datos también

puede iniciarse usando SFC. Los Datos Globales pueden ser entradas, salidas, marcas, temporizadores, contadores y áreas de bloques de datos.

#### **3.1.4.2. Comunicación básica**

Este método de comunicación puede usarse con todas las CPU S7 300/400 transmitiendo datos a través de la subred MPI o dentro de un equipo en su bus K. Las funciones de sistema (SFCs), por ejemplo, X\_SEND en el emisor y X\_RCV en el receptor, son llamadas en el programa de usuario. La máxima cantidad de datos de usuario es 76 bytes. Cuando se llama a la función de sistema, se establece y se rompe dinámicamente un enlace con el interlocutor. Se requiere un enlace libre en la CPU. La comunicación básica se implementa con SFCs que están contenidas en el sistema operativo de la CPU. (La comunicación con SFC funciona sin enlace configurado).

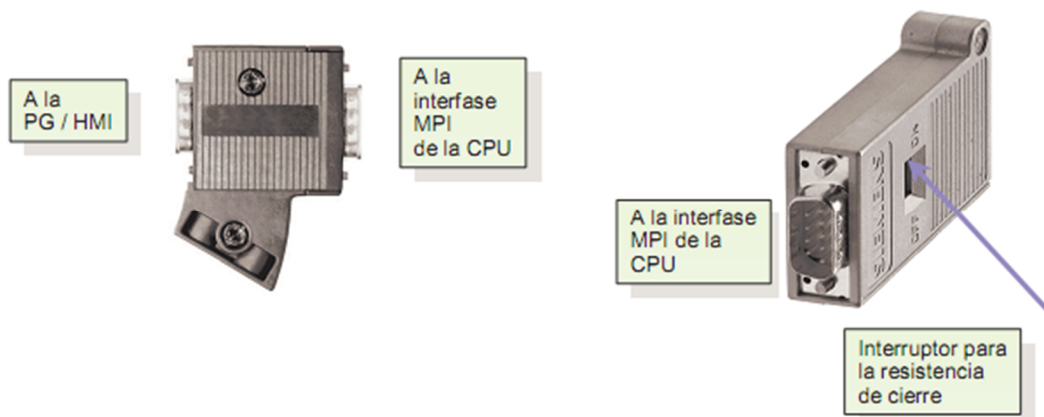
#### **3.1.4.3. Comunicación ampliada**

Es posible usar este método de comunicación con todas las CPUs S7-400. Se pueden transmitir hasta 64KBytes de datos a través de cualquier subred (MPI, Profibus, Industrial Ethernet). Esto se lleva a cabo con funciones de sistema (SFBs), que permiten además comunicación con reconocimiento. Los datos también pueden leerse o escribirse en un S7-300 (bloques PUT/GET). No sólo es posible transmitir datos, sino también realizar funciones de control, como *Stop* o *Start*, en el interlocutor. Se requieren enlaces configurados (tabla de enlaces) para la comunicación, estos enlaces se establecen en un rearranque completo del equipo y normalmente permanecen establecidos. Se necesitan enlaces libres en la CPU. La comunicación ampliada tiene lugar mediante enlaces configurados con la ayuda de SFBs (S7-400 Cliente/Servidor; S7-300 solo Servidor).

### 3.1.5. Conector de bus

El conector con el zócalo para la PG mostrado a la izquierda de la figura siguiente es el conector estándar usado para enlazar nodos MPI uno con otro, permitiendo al mismo tiempo la conexión de una PG. El conector sin el zócalo para PG mostrado a la derecha se usa cuando no se necesita la facilidad de conexión de una PG. En el último nodo del bus, el cable de bus de salida debe reemplazarse por una resistencia terminal.

Figura 80. Conector de bus



Fuente: SIEMENS, S. A.

Para conectar un dispositivo de programación / PC a la interfase MPI del PLC, se necesita:

- Un módulo MPI en la PG/PC y un cable de conexión.
- Un PC adaptador (un cable de conexión con un convertidor MPI integrado, si no hay un slot libre para una tarjeta en la PG/PC). El PC adaptador tiene las siguientes especificaciones:
  - Longitud 5 m.

- Velocidad de transmisión a partir del adaptador 187,5 kbps.
- Del adaptador a la PG 19,2 o 38,4 kbps (ajustable).

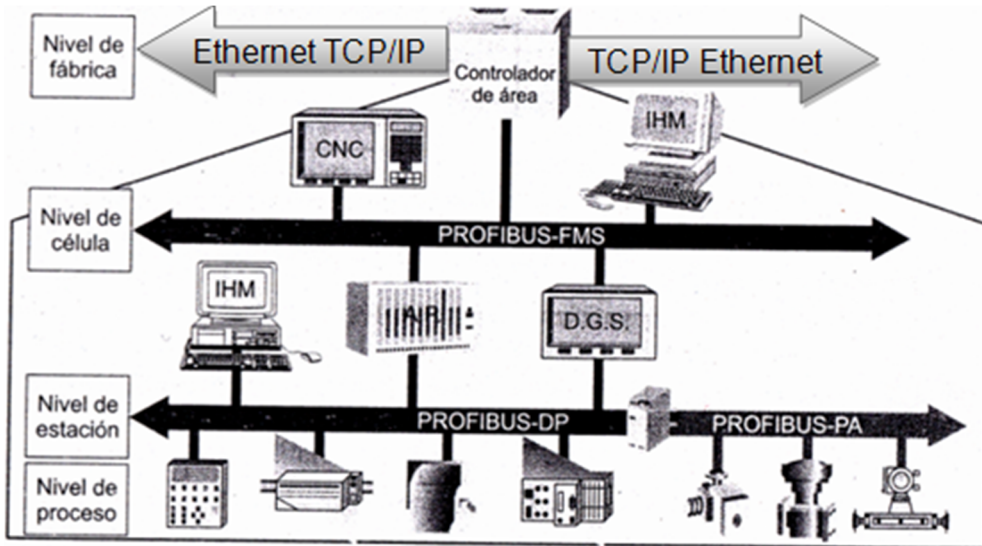
### **3.2. Familia de redes de campo PROFIBUS**

PROFIBUS (abreviatura de *PROcess Field BUS*) es un conjunto de redes de control (o buses de campo) diseñado para resolver las necesidades de comunicación de los procesos industriales tanto discretos (denominados procesos de fabricación) como continuos y otros procesos distribuidos, como la automatización de edificios.

#### **3.2.1. Características generales**

Uno de los objetivos iniciales de PROFIBUS fue el de proporcionar una solución adecuada, mediante una única norma, para las comunicaciones de los niveles de proceso, estación y célula (e incluso de los niveles de área y fábrica). La familia de protocolos PROFIBUS cubre todo el espectro de las redes de control (a diferencia del bus AS-i, que fue diseñado únicamente para el nivel de proceso), e incluso, en determinados casos, puede utilizarse como red de datos con capacidad de comunicación determinista.

Figura 81. Utilización de la familia de protocolos PROFIBUS



Fuente: SIEMENS, S. A.

PROFIBUS posee actualmente tres miembros denominados PROFIBUS-DP, PROFIBUS-PA y PROFIBUS-FMS, cada uno de los cuales satisface, tal como se indica en la figura anterior. Las principales características de cada miembro son las siguientes:

### 3.2.1.1. Red PROFIBUS-DP

PROFIBUS-DP (*Decentralised Periphery*) es una red o bus de campo cuyo protocolo está optimizado para realizar las transferencias de información en el nivel de proceso, tal como se indica en la figura anterior, que necesita alta velocidad de transmisión y bajo coste. Se diseñó, al igual que AS-i, para realizar la comunicación entre un sistema electrónico de control (por ejemplo, un autómatas programable, un robot, un sistema de control numérico, etc.) y la denominada Periferia Distribuida (*Decentralised Periphery*), constituida por los



dispositivos sensores y actuadores (dispositivos de campo) que se deben conectar a él.

### **3.2.1.2. Red PROFIBUS-PA**

PROFIBUS-PA (*Process Automation*), es una red o bus de campo cuyo protocolo está optimizado para realizar las transferencias de información necesarias entre los sistemas electrónicos de control y los sistemas de instrumentación utilizados en la industria de los procesos continuos (reguladores de caudal, temperatura, presión, válvulas proporcionales, etc.), que se caracterizan por realizar medidas de variables analógica y actuar sobre el proceso en función de valor de dicha medida.

Su principal diferencia con PROFIBUS-DP es que la capa física permite su utilización en zonas de seguridad intrínseca. Para ello, PROFIBUS-PA posibilita la comunicación de datos (a una velocidad máxima inferior a la de PROFIBUS-DP) y la alimentación de los procesadores de comunicaciones a través de un único par de hilos mediante fuentes de alimentación especiales. En muchas instalaciones se combinan los dispositivos de campo, conectados mediante una red PROFIBUS-DP, con los sistemas de instrumentación, conectados mediante una red PROFIBUS-PA. En este caso, ambas redes se enlazan mediante un acoplador de segmentos DP-PA (*DP-PA segment coupler*) que constituye un puente (*bridge*) de comunicaciones (ver figura anterior).

### **3.2.1.3. Red PROFIBUS-FMS**

PROFIBUS-FMS (*Fieldbus Message Specification*) es una red o bus de campo optimizada para realizar las transferencias de información en los niveles de estación, célula e incluso fábrica, tal como se indica en la figura anterior.

Para ello, dispone de un conjunto de funciones [que constituyen un subconjunto de las establecidas por la Norma ISO/IEC 9506-1, denominada MMS (acrónimo de *Manufacturing Mesage Specification*)] que le proporcionan una gran flexibilidad.

### **3.2.2. Comparación de las redes PROFIBUS y AS-i**

Se deducen las siguientes conclusiones desde un punto de vista práctico:

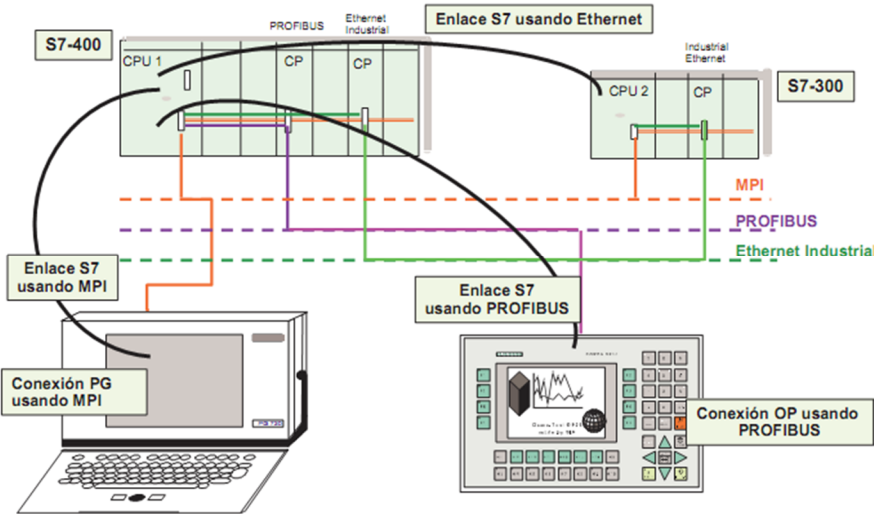
- La red PROFIBUS-DP y PROFIBUS-PA son redes de sensores-actuadores de capacidad funcional elevada y la red AS-i es una red de sensores-actuadores de capacidad funcional limitada.
- En la práctica, en PROFIBUS-DP y PROFIBUS-PA es posible configurar sistemas de control distribuido en los que el volumen global de datos de entrada/salida intercambiados con un procesador principal de comunicaciones (*master*) sea de 16 koctetos/16 kbyte. Por el contrario, en el bus AS-i sólo se puede intercambiar un máximo de 248 octetos.
- La combinación de una red PROFIBUS-DP y una red AS-i constituye una solución adecuada en instalaciones complejas en las que coexisten máquinas que sólo tienen sensores todo-nada con otras que incorporan sensores inteligentes, por ejemplo codificadores angulares absolutos, cámaras digitales, etc.

### **3.3. Enlaces entre los participantes de la comunicación**

Un enlace (conexión) es una asignación lógica de dos interlocutores de comunicación para llevar a cabo servicios de comunicación (figura siguiente). Los enlaces pueden ocupar uno o varios recursos de conexión en los módulos con capacidad de comunicación participantes (CPU, CP y FM) por cada

posición final. Para garantizar una configuración de los enlaces ordenada, los enlaces deben ser activos en una posición final y pasivos en la otra posición final. En caso contrario, el enlace no puede establecerse.

Figura 82. **Enlaces entre los participantes de la comunicación**



Fuente: SIEMENS, S. A.



#### **4. IMPLEMENTACIÓN DE MIGRACIÓN DE CONTROLADOR LÓGICO PROGRAMABLE S5 A S7, DE UNA MÁQUINA LLENADORA DE BEBIDAS**

La migración de programas controladores SIEMENS es posible desde una CPU de la serie S5 a una CPU de la serie S7-300/400, en esta migración se crea la CPU nueva en el proyecto y el programa convertido se implementa en la nueva CPU. La migración del programa se adapta automáticamente a la nueva familia de CPU dentro de las posibilidades. El programa probablemente se podría convertir sin demasiados errores mediante la conversión del programa con el paquete de programación STEP 7. Las ventajas de migrar son:

- Tener varios bastidores locales
- Tener bastidores remotos
- Utiliza ET 200 I / O
- Sistemas redundantes
- Sustitución/mejora de la interfaz visual (HMI)
- Adición de más I/O
- Actualizar y mejorar la red de comunicación
- Mayor capacidad de la CPU, en cuanto a memoria, velocidad de respuesta, diagnóstico y otros.

Aparte del aumento de la productividad, en general eficiencia y disponibilidad, modernización y un sistema de automatización de control moderno SIMATIC proporciona muchos más beneficios tecnológicos y financieros como:

- El cumplimiento de nueva normas de fabricación.
- El acceso a una gama más amplia de servicios de funcionalidades de automatización integrados.
- Integración del sistema de control y planta de fabricación.
- La integración de nuevas funcionalidades tales como la visualización, la tecnología, o el archivo de datos.
- Ampliación flexible utilizando los módulos de E / S, de función y módulos de comunicación.

#### **4.1. Pasos de la migración**

Las razones principales de migración desde el controlador de S5 a un controlador de S7 pueden ser por falta de memoria en la CPU S5 y el ciclo de vida del controlador S5 que ha llegado a su fin. Otras razones incluyen tomar ventaja de una o más de las muchas nuevas tecnologías disponibles en S7, pasar a un entorno de programación estructurado, o tomar ventaja de los lenguajes de programación adicionales y/o paquetes de opciones S7.

##### **4.1.1. Cómo reemplazar la CPU S5**

El factor crítico es elegir la CPU S7 correcta, hay dos parámetros principales. El primero es el tamaño de la memoria de la CPU, debe elegirse una CPU S7-300 o S7-400 con un tamaño de memoria equivalente al tamaño de la memoria del controlador S5 más cualquier memoria adicional para cubrir la expansión o el exceso de capacidad del 20 %, el exceso variará dependiendo de la cantidad de la memoria de la CPU S5 que utiliza realmente el programa. El segundo parámetro importante es la CPU I / O frente a la capacidad, consultar el anexo A para una tabla de referencia de CPU que muestra la

cantidad de memoria en la CPU S5 y la CPU S7 comparativo. En la migración se implementó la CPU 314-2 DP con memoria de trabajo de 192KB.

#### **4.1.1.1. Módulos de interfaz en el bastidor principal**

Al reemplazar la CPU del controlador también se requiere cambiar el panel entero del PLC, los módulos que fueron utilizados anteriormente en el bastidor S5 también deben ser reemplazado. Esto incluye todas las E / S, módulos de función y de comunicación.

#### **4.1.2. Módulos de entradas y salidas**

En el anexo B y C proporcionan una referencia cruzada de S5 a S7 I / O, estas tablas de referencias cruzadas se establecen como sugerencia para la mayoría de las características del módulo original S5. En la migración se implementó los siguientes módulos; dos de entrada digital con 32 bit 6ES7 321-1BL00-0AA0 y dos de salida digitales con 32 bit 6ES7 322-1BL00-0AA0.

La migración de controladores S7 requieren *STEP 7* que incluye la herramienta de conversión S5. *STEP 7* comprende tres lenguajes de programación básicos; esquema de contactos, diagrama de bloques de funciones y lista de instrucciones. Además de estos idiomas, hay varios paquetes de opciones disponibles para ampliar las capacidades del software básico.

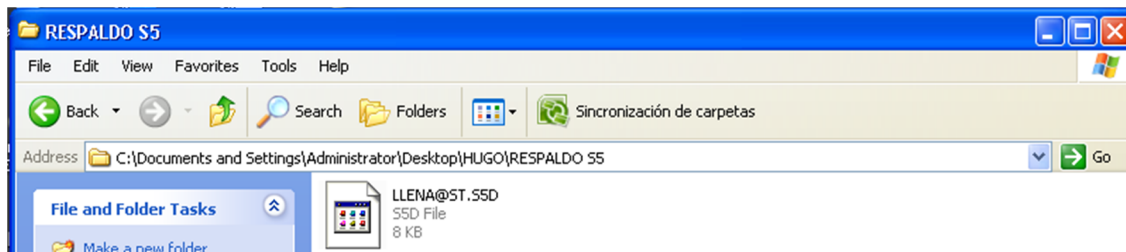
### 4.1.3. Convertir un programa

La conversión de la lógica del programa S5 a la lógica del programa S7 consiste en utilizar la herramienta de conversión S5 a S7 que se incluye con *STEP 7*.

#### 4.1.3.1. Preparación de archivo S5

En esta investigación se trabajó en una máquina llenadora de bebidas. El archivo del programa S5 de respaldo descargado de la CPU S5-95U del controlador se llama LLENA@ST.S5D, como muestra la figura siguiente, este archivo es necesario tenerlo para poder realizar la migración, debe estar instalado el programa S5 para poder descargar el archivo.

Figura 83. Archivo S5



Fuente: elaboración propia.

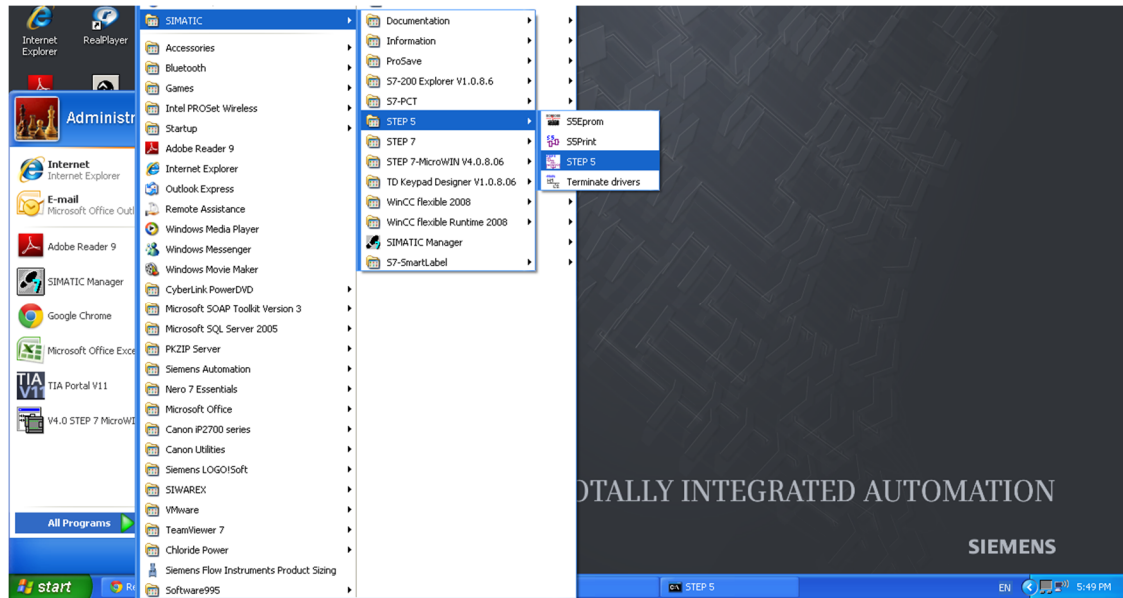
#### 4.1.3.2. Generar referencias cruzadas

Existen dos formas de generar referencias cruzadas, las cuales se describen en los siguientes pasos.

- A. Se abre el programa *STEP 5* haciendo clic en *Start* y seleccionar *SIMATIC STEP 5*, como muestra la figura siguiente.



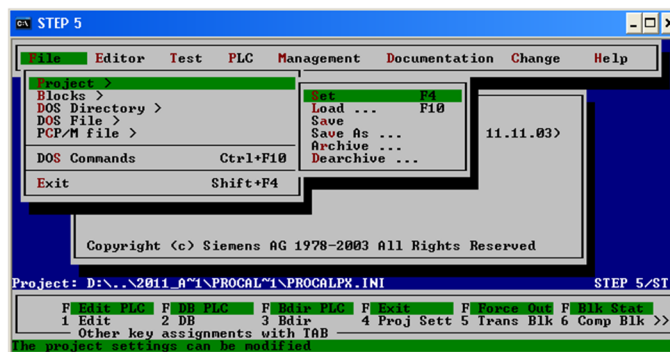
Figura 84. Abrir STEP 5



Fuente: elaboración propia.

- B. Se abre Set F4 haciendo clic izquierdo sobre este, como muestra la figura siguiente.

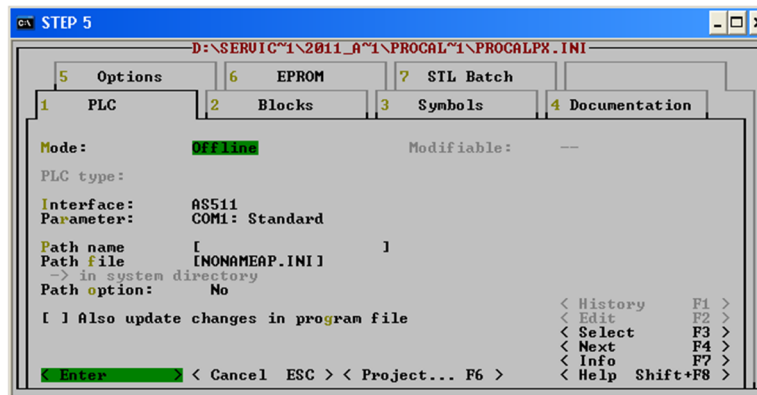
Figura 85. Abrir Set F4



Fuente: elaboración propia, con base en el programa SIMATIC STEP 5.

C. Hacer clic izquierdo en *Blocks*.

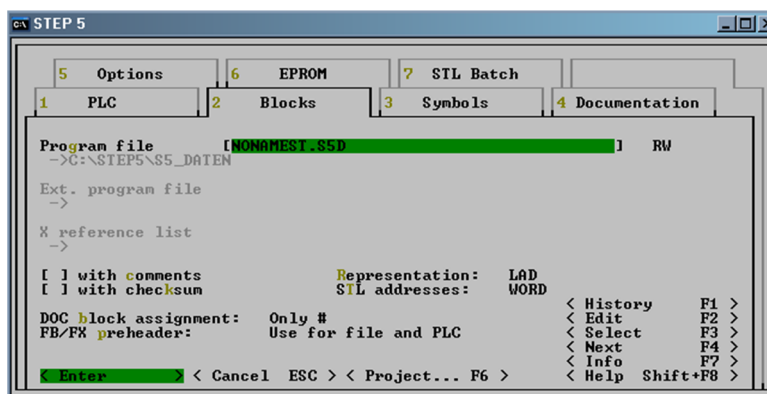
Figura 86. Abrir el controlador



Fuente: elaboración propia, con base en el programa SIMATIC STEP 5.

D. Hacer doble clic izquierdo en *Program file*.

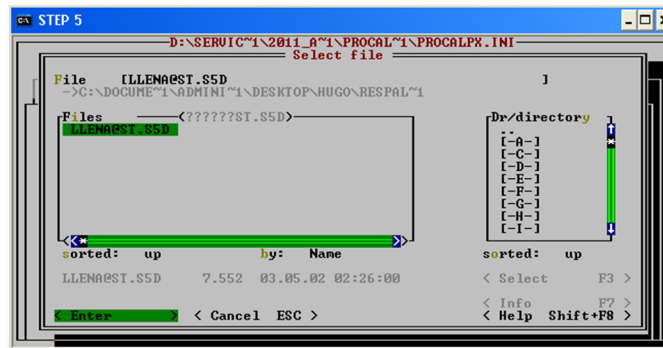
Figura 87. Abrir *Program file*



Fuente: elaboración propia, con base en el programa SIMATIC STEP 5.

- E. Se busca el archivo de respaldo LLENA@ST.S5D en Dr/directory donde este guardado luego hacer doble clic izquierdo sobre el archivo LLENA@ST.S5D que aparece en Files.

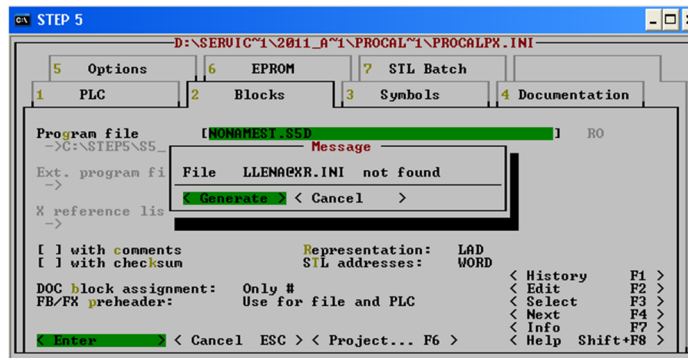
Figura 88. Selección de archivo de respaldo LLENA@ST.S5D



Fuente: elaboración propia, con base en el programa SIMATIC STEP 5.

- F. Del paso anterior se tiene la siguiente figura en el cual hacer clic en Generate.

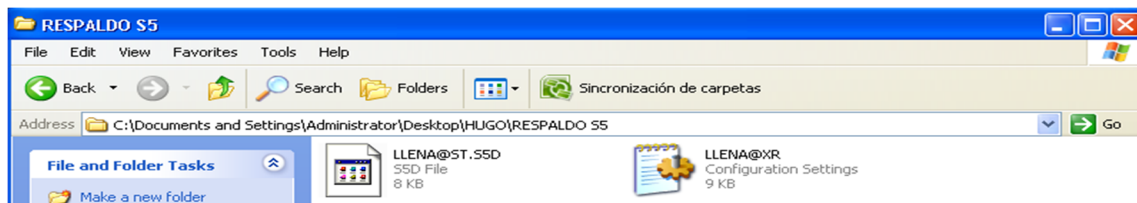
Figura 89. Seleccionar Blocks



Fuente: elaboración propia, con base en el programa SIMATIC STEP 5.

- G. Con el paso anterior se generó el archivo LLENA@XR.INI, como se muestra en la figura siguiente.

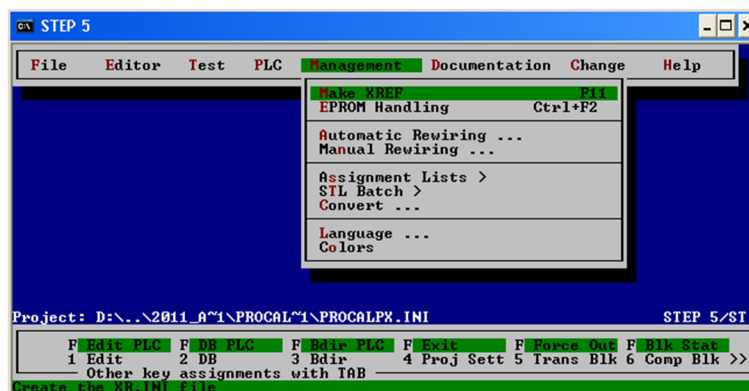
Figura 90. Archivo generado LLENA@XR



Fuente: elaboración propia.

- H. Si no se generó el archivo realizando los pasos anteriores la segunda opción es generar el archivo de la manera como lo muestra la figura siguiente.

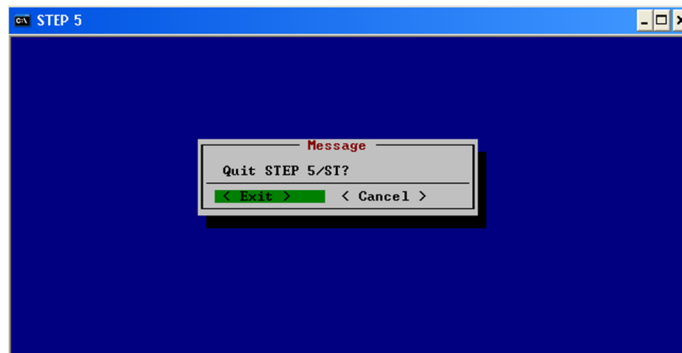
Figura 91. Opción generar referencia cruzada



Fuente: elaboración propia, con base en el programa SIMATIC STEP 5.

- I. Se ha generado satisfactoriamente el archivo, por lo tanto hacer clic en *File* para luego salir de STEP 5, como muestra la figura siguiente.

Figura 92. **Salir de STEP 5**



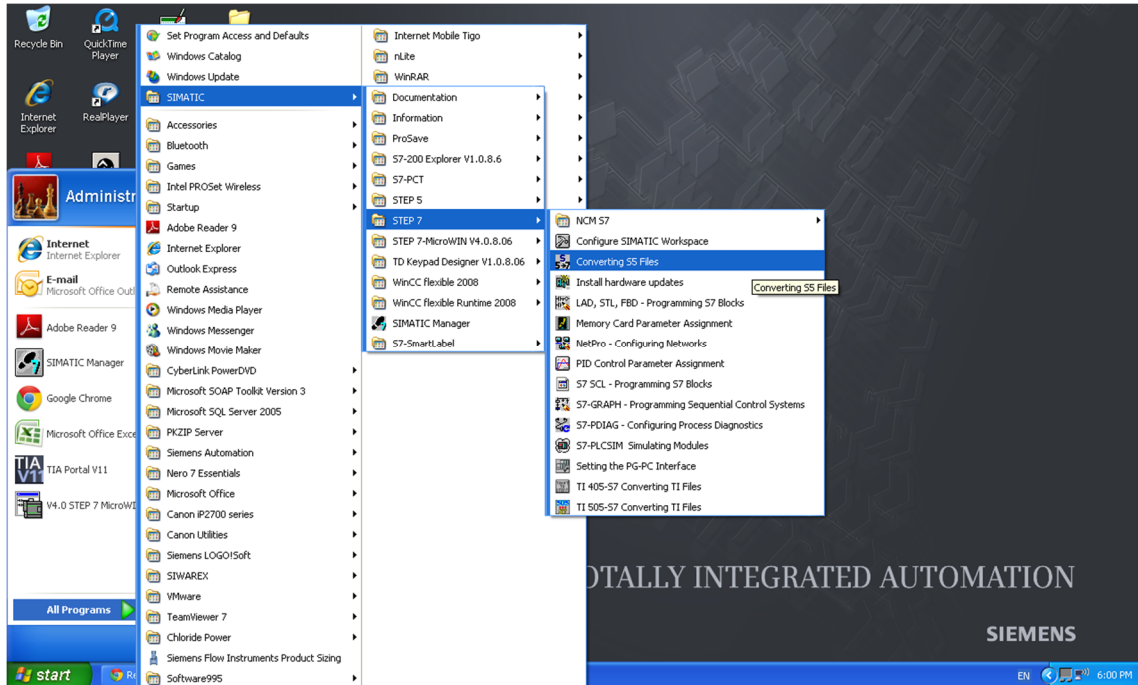
Fuente: elaboración propia, con base en el programa SIMATIC STEP 5.

#### **4.1.3.3. Utilizando la herramienta de conversión de archivos S5 a S7**

Luego que se ha generado el archivo de referencia cruzada se debe seguir los siguientes pasos para convertir un archivo S5 a S7.

- A. Abrir la herramienta convertir archivos S5 a S7 que se encuentra en las herramientas de SIMATIC, como muestra la figura siguiente.

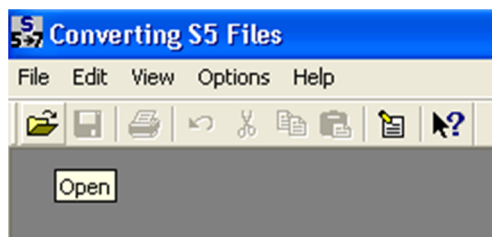
Figura 93. **Abrir la herramienta convertir archivo S5 a S7**



Fuente: elaboración propia.

- B. Hacer clic en abrir para buscar el archivo a convertir, como muestra la figura siguiente.

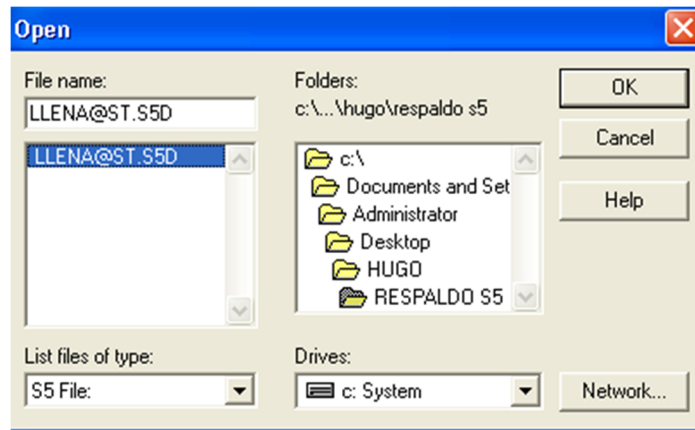
Figura 94. **Abrir el archivo a convertir**



Fuente: elaboración propia, con base en el programa SIMATIC Convertir Archivo S5.

- C. Buscar el archivo de respaldo LLENA@ST.S5D seleccionando S5 File en *List files of tipe*, también el *Drive* en la que esté guardado, buscando en *Folders*, luego hacer clic en *OK*, como muestra la figura siguiente.

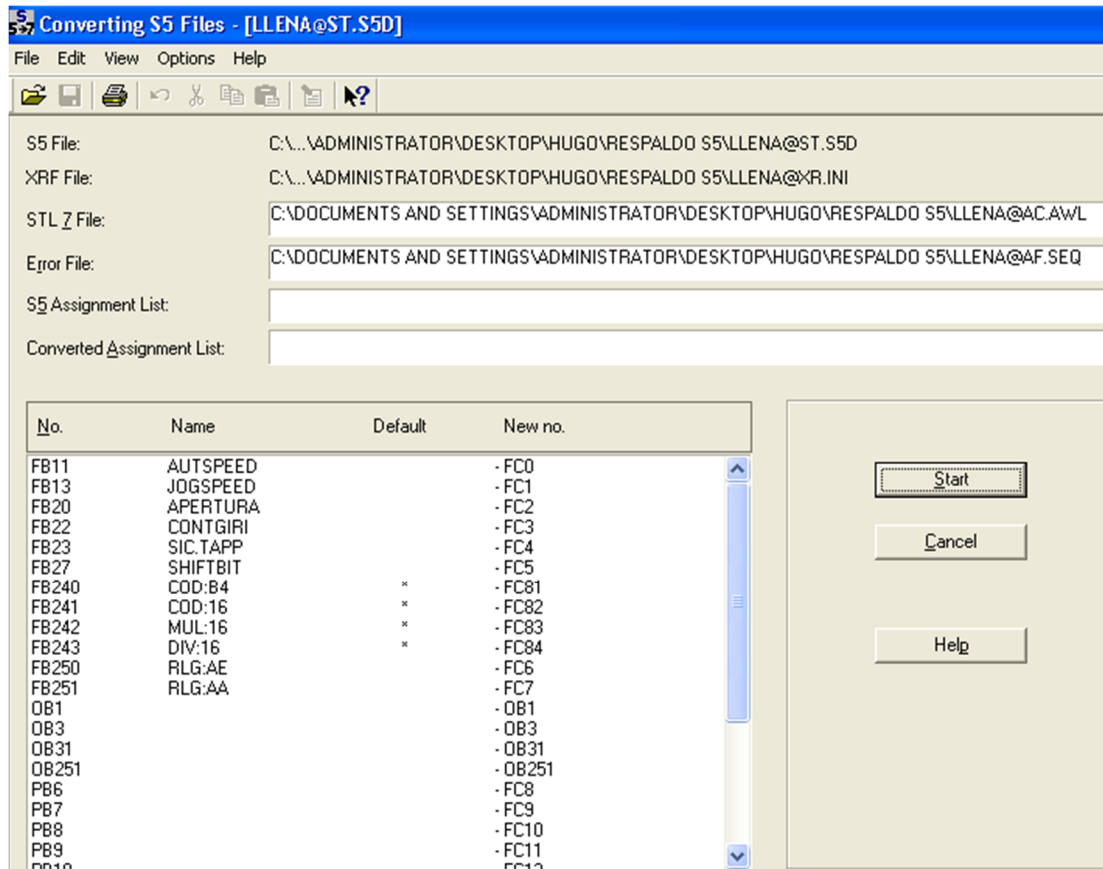
Figura 95. **Abrir el archivo LLENA@ST.S5D**



Fuente: elaboración propia, con base en el programa SIMATIC Convertir Archivo S5.

- D. La siguiente figura muestra los bloques equivalentes antes y después. Primero hacer clic en *Start* para iniciar la conversión. Se recomienda guardar las imágenes necesarias de esta pantalla para posteriormente utilizarla como referencia al realizar los cambios manualmente de los bloques que tienen un asterisco. También se debe tener en cuenta que a veces se necesita investigar sobre los bloques equivalentes en caso de tener errores durante la conversión, lo cual se explica más adelante.

Figura 96. **Convirtiendo comparación de bloques S5 a S7**

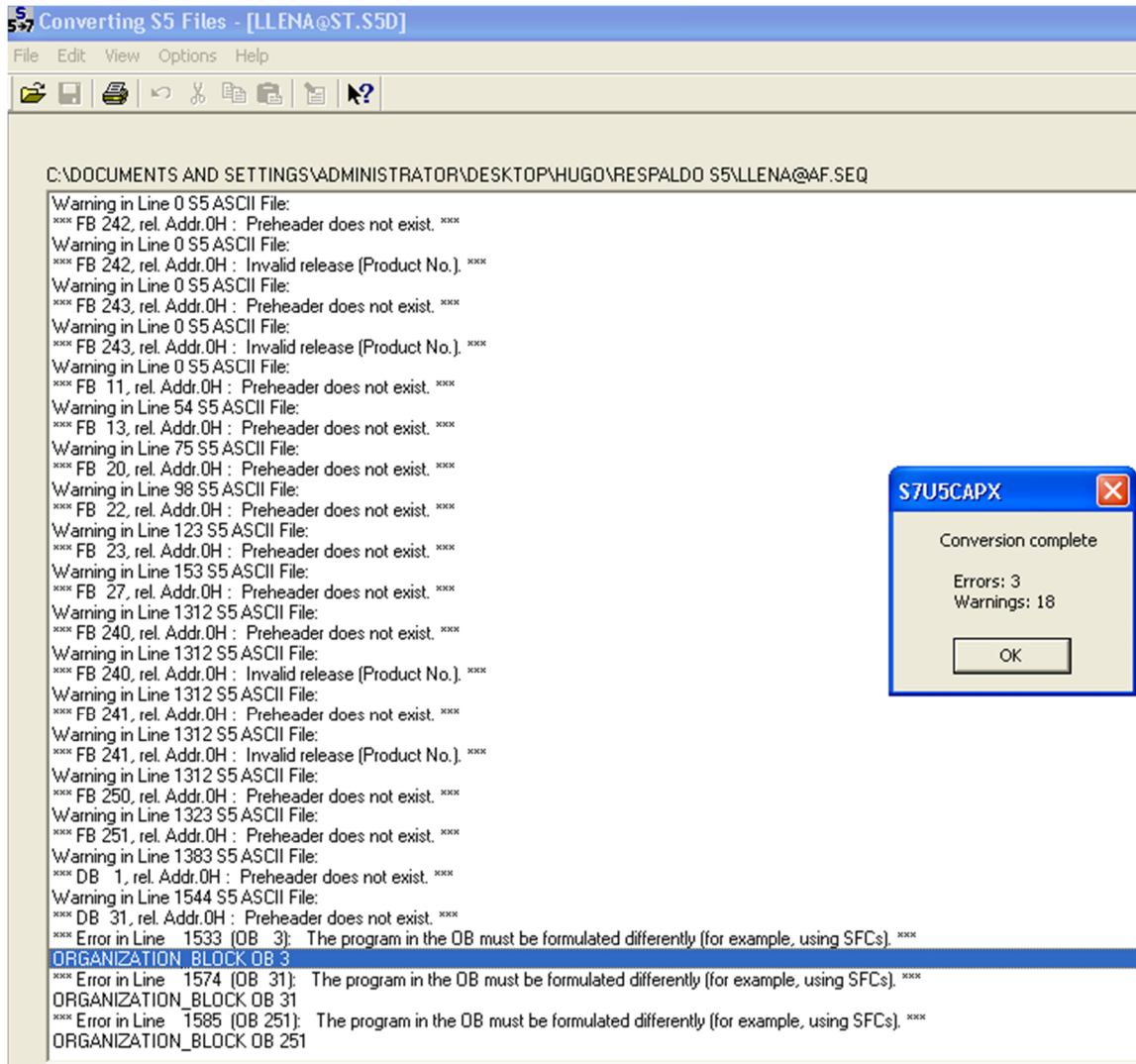


Fuente: elaboración propia, con base en el programa SIMATIC Convertir Archivo S5.

- E. Al completarse la conversión se muestra en la pantalla tres errores que son parte del programa y que serán corregidos más adelante, también aparecen dieciocho advertencias. Se recomienda guardar las imágenes necesarias de esta pantalla para posteriormente utilizarla como referencia al realizar los cambios manualmente de los bloques que tienen error.



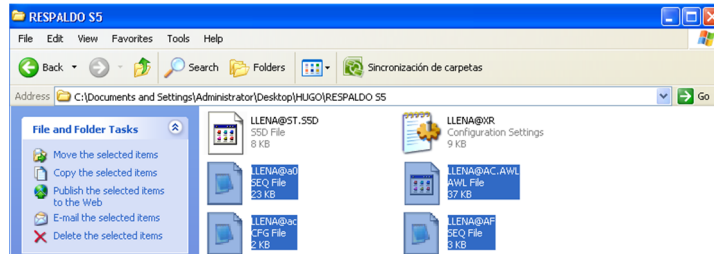
Figura 97. Errores y advertencias de conversión terminada S5 a S7



Fuente: elaboración propia, con base en el programa SIMATIC Convertir Archivo S5.

- F. Ahora se han creado nuevos archivos de la herramienta de migración dentro de la carpeta Respaldo S5, como se muestra en la figura siguiente.

Figura 98. Archivos creados con la herramienta de migración S5 a S7

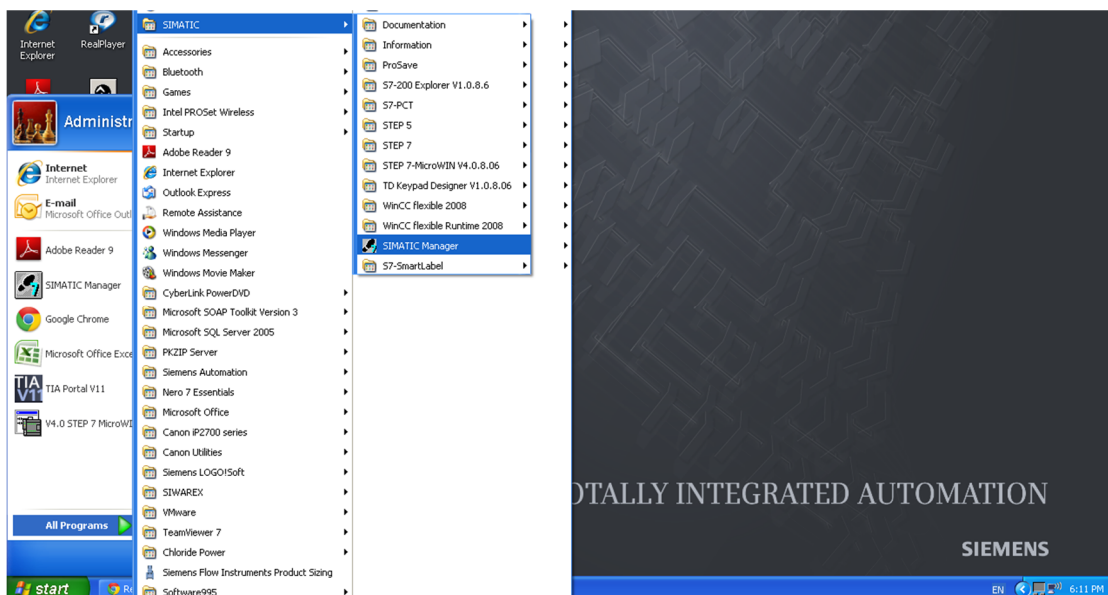


Fuente: elaboración propia.

#### 4.1.4. Crear un nuevo proyecto de S7

- A. Abrir el programa Administrador SIMATIC S7, como muestra la figura siguiente.

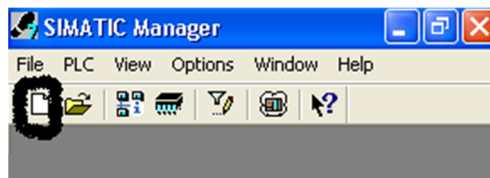
Figura 99. Abrir Administrador SIMATIC S7



Fuente: elaboración propia.

- B. Para crear un proyecto nuevo, hacer clic en la hoja blanca marcada como aparece en la siguiente figura.

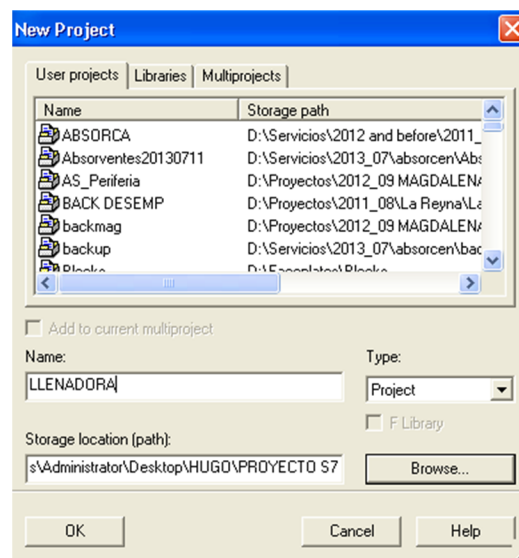
Figura 100. **Crear proyecto nuevo**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- C. Debe crear una carpeta donde guardar el proyecto nuevo, entonces elegir la ruta donde colocarla y el nombre del proyecto nuevo.

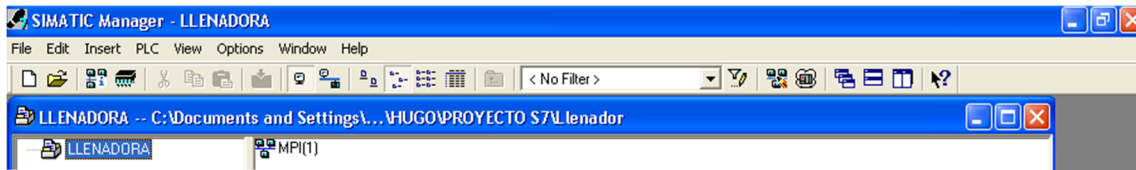
Figura 101. **Elegir ruta y nombre del proyecto**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- D. Del paso anterior hacer clic en *OK*, entonces se tiene el proyecto nuevo en pantalla.

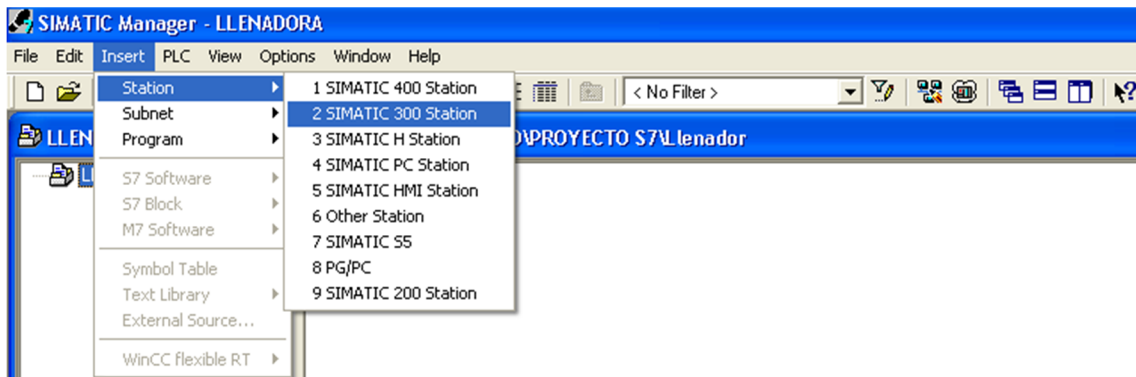
Figura 102. El nombre del proyecto creado



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- E. Insertar un equipo SIMATIC 300 como muestra la figura siguiente.

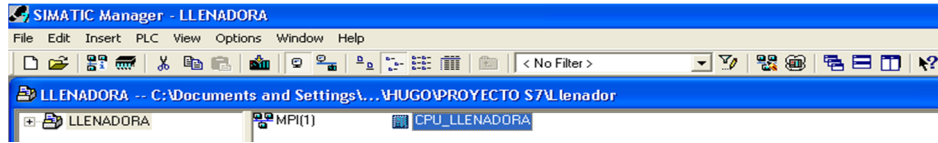
Figura 103. Insertar equipo SIMATIC 300



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- F. Del paso anterior se creó el equipo con el nombre CPU\_LLENADORA el cual está marcado como muestra la figura siguiente.

Figura 104. **Creación del equipo**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

#### 4.1.5. Configuración de hardware

- A. La adaptación de hardware nuevo es información necesaria para los pasos siguientes.

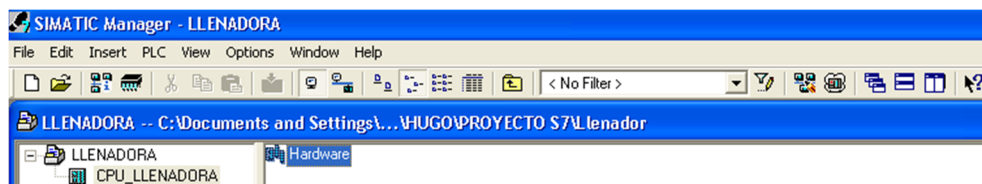
Figura 105. **Posición de módulos en el rack**



Fuente: elaboración propia.

- B. Hacer doble clic en hardware.

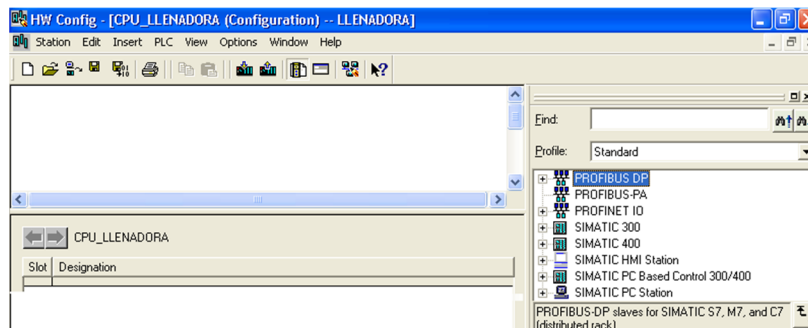
Figura 106. **Abrir la configuración hardware**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- C. La configuración hardware debe hacerse como muestra la figura siguiente.

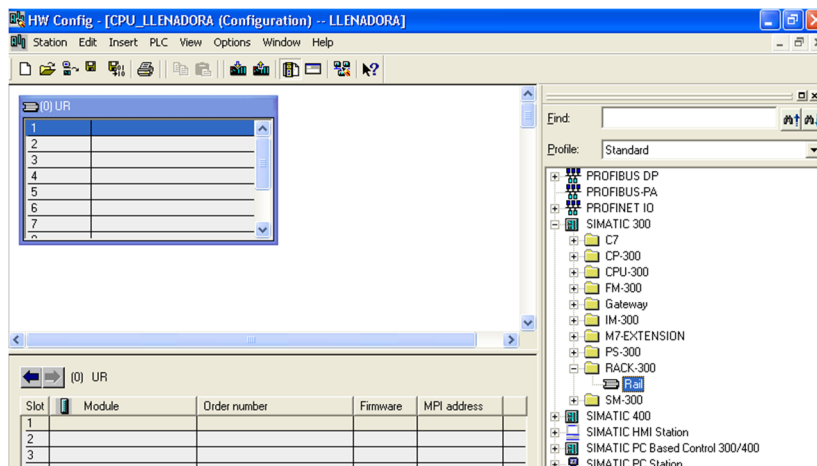
Figura 107. Configuración hardware



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- D. Insertar el rack haciendo doble clic en *rack*, esto es para distribuir los módulos.

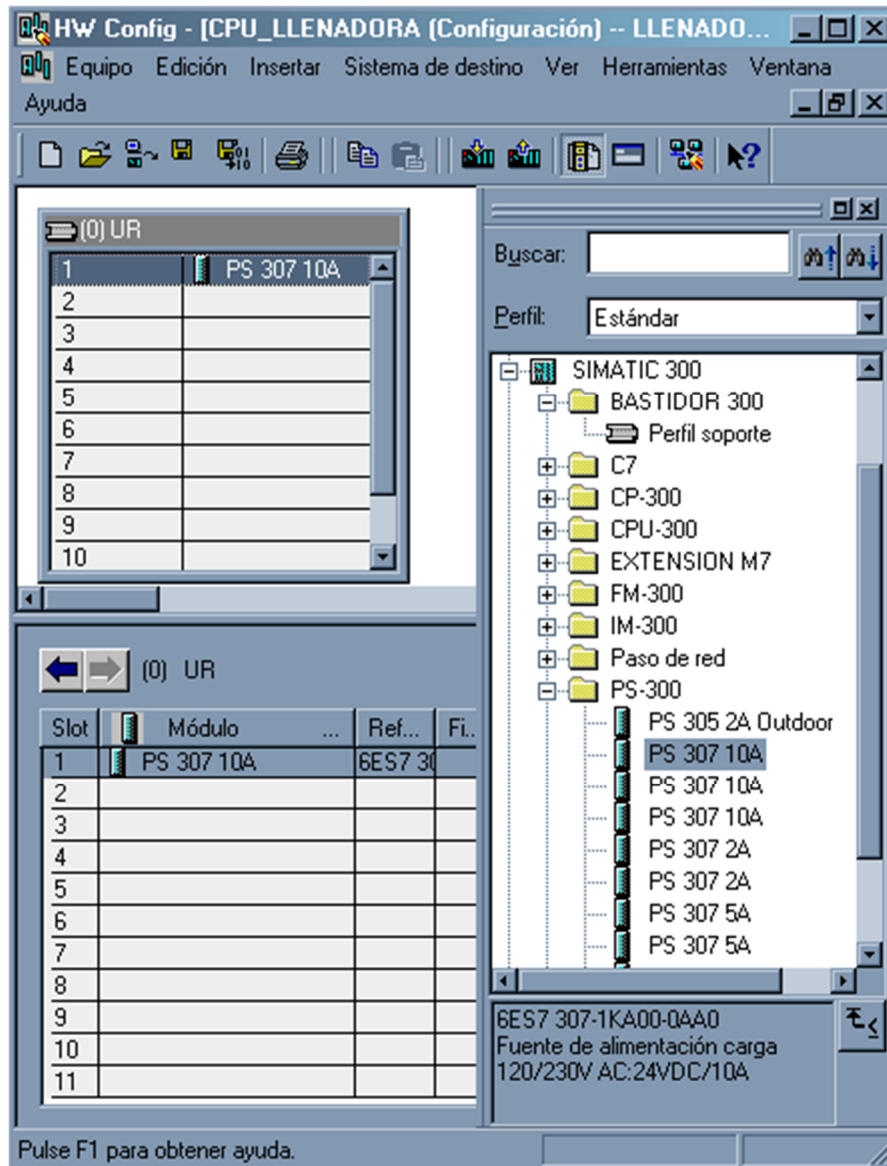
Figura 108. Insertar rack



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- E. Arrastrar el suministro de energía en la posición número uno del *rack*, como muestra la figura siguiente.

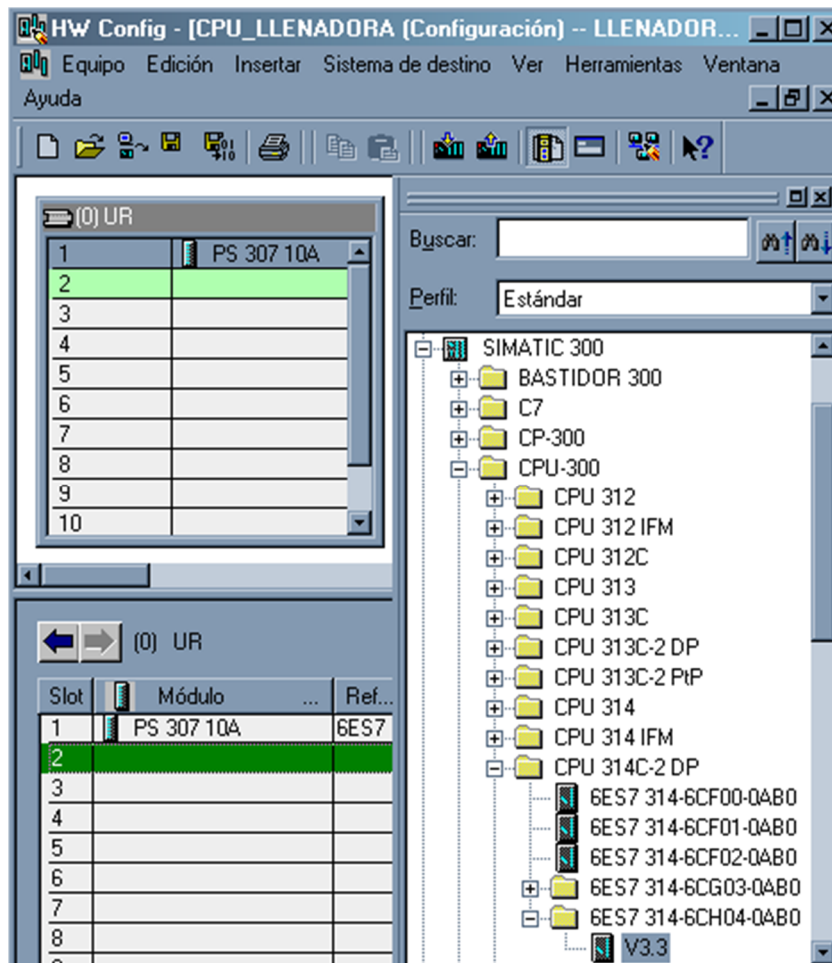
Figura 109. Insertar el suministro de energía en *rack*



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

F. Elegir la CPU que se necesita como muestra la figura siguiente.

Figura 110. **Elegir CPU**

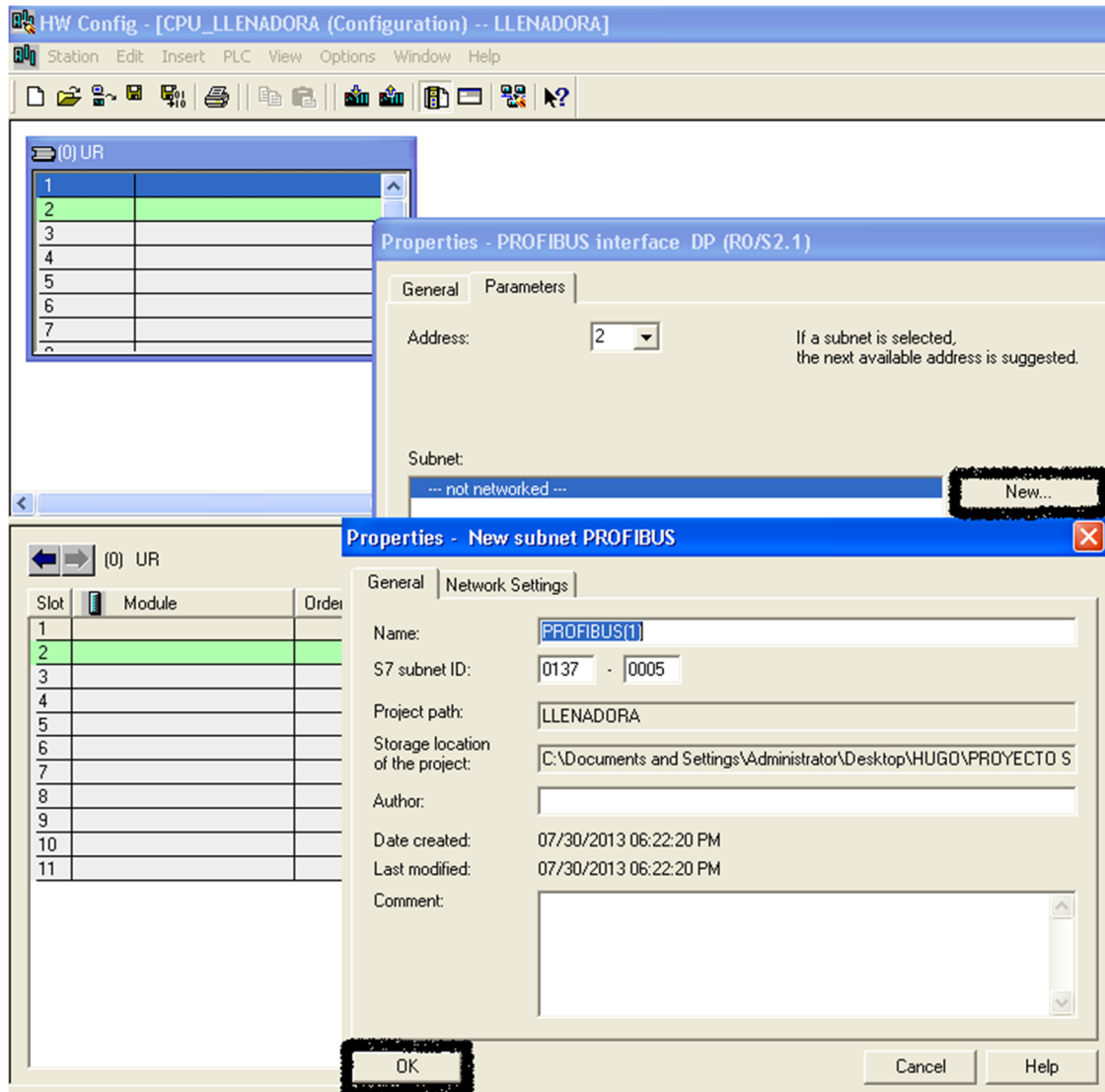


Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

G. Insertar la CPU arrastrándola en la posición número dos del *rack*, entonces se desplegará automáticamente una ventana. Hacer clic en *New*, luego hacer clic en *OK*, esto es para establecer PROFIBUS interface DP, como muestra la figura siguiente.



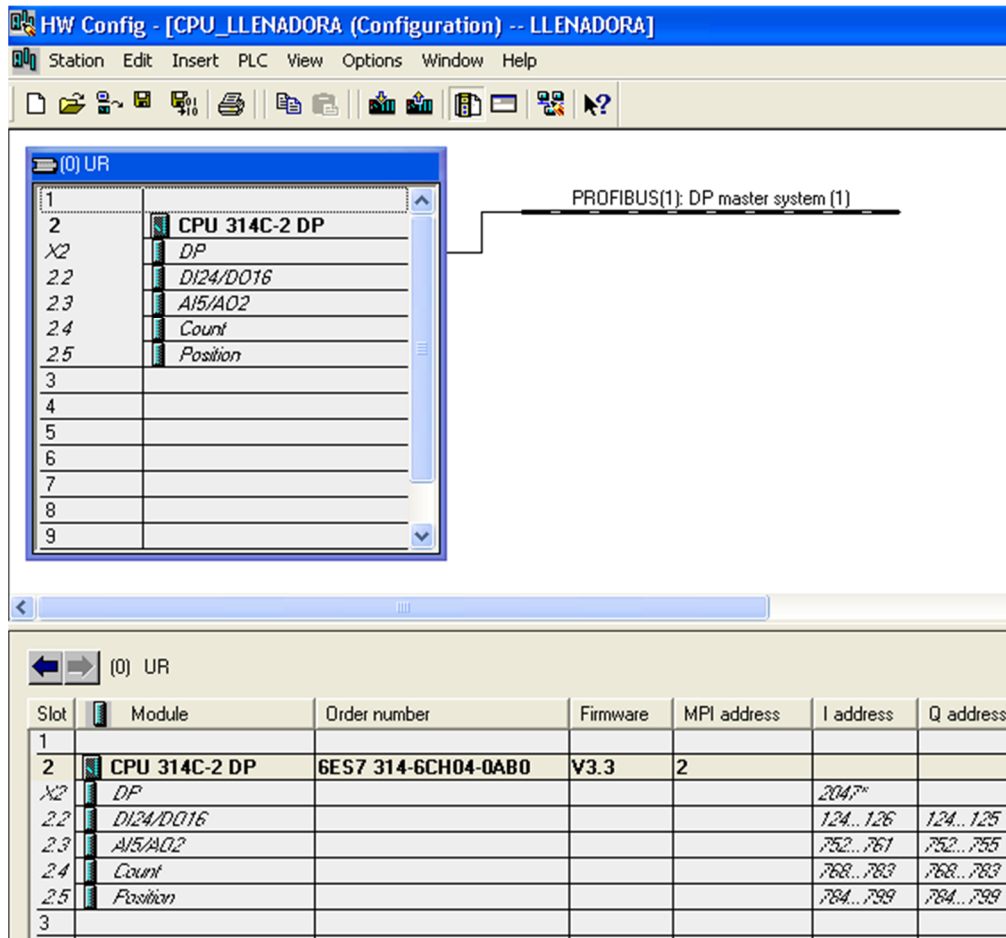
Figura 111. Insertar la CPU en rack



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- H. Como muestra la figura siguiente, se ha colocado la CPU elegida, la cual tiene varias funciones integradas necesarias para completar el ciclo completo del proyecto.

Figura 112. La CPU en el rack



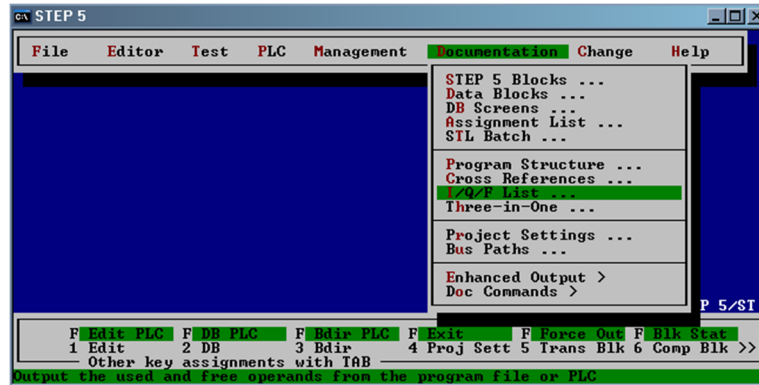
Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

#### 4.1.5.1. Direccionamiento de señales

Para poder direccionar los módulos de entradas y salidas digitales al insertarlas en el *rack* se necesita realizar los siguientes pasos:

- A. Abrir archivo S5 y hacer clic en I/Q/F List, como muestra la figura siguiente.

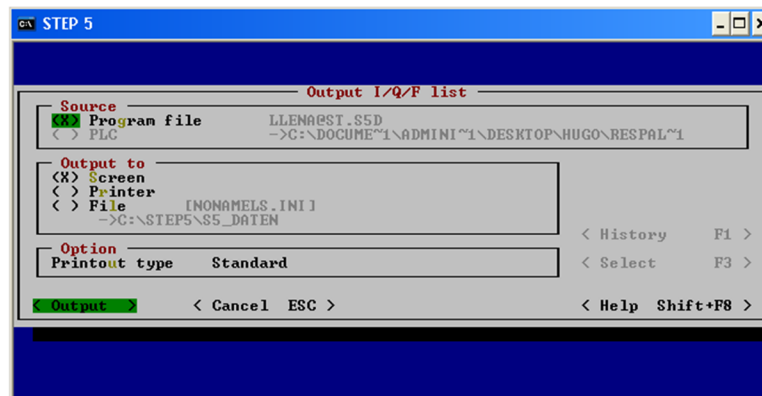
Figura 113. Abrir I/QF List



Fuente: elaboración propia, con base en el programa SIMATIC S5.

- B. Observar que estén marcados con una equis *Program file* y *Screen*, entonces hacer clic en *Output*, como muestra la figura siguiente.

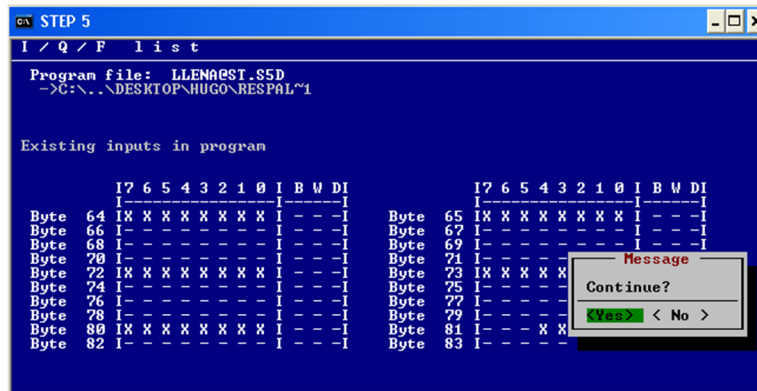
Figura 114. Chequear I/Q/F List



Fuente: elaboración propia, con base en el programa SIMATIC S5.

- C. Hacer clic en *Yes* para buscar las entradas digitales existentes en el programa S5, como muestra la figura siguiente.

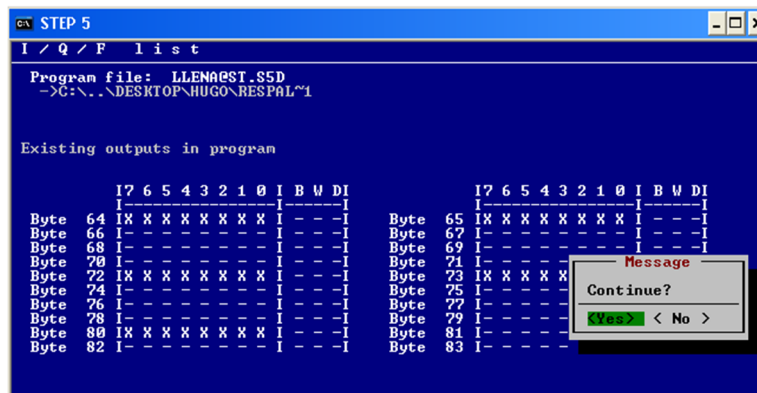
Figura 115. **Buscar entradas digitales en el programa S5**



Fuente: elaboración propia, con base en el programa SIMATIC S5.

- D. Hacer clic en Yes para buscar las salidas digitales existentes en el programa S5, como muestra la figura siguiente.

Figura 116. **Buscar salidas digitales en el programa S5**



Fuente: elaboración propia, con base en el programa SIMATIC S5.

- E. La figura siguiente muestra el direccionamiento de los módulos de entradas y salidas digitales de la CPU del S5 que se buscó anteriormente

y se reacomoda al S7 para la configuración nueva de las entradas y salidas digitales.

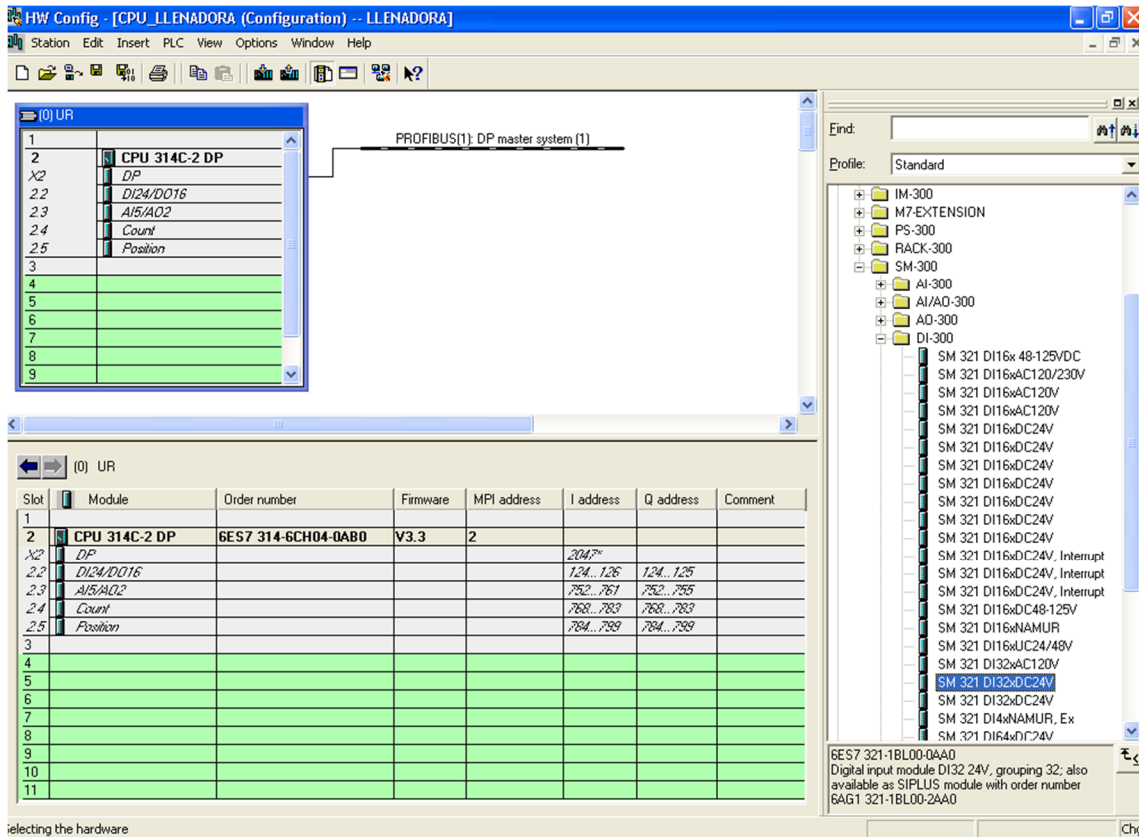
Figura 117. Módulos de entradas y salidas digitales del S5 al S7

S5			S7			S5			S7			S5			S7				
MODULO INTEGRADO	17	I32.0	I0.0	MODULO 2	81	I73.0	I4.0	MODULO INTEGRADO	1	Q32.0	Q8.0	MODULO 3	65	Q73.0	Q12.0	MODULO 4	112	Q80.7	Q15.7
	18	I32.1	I0.1		82	I73.1	I4.1		2	Q32.1	Q8.1		66	Q73.1	Q12.1		111	Q80.6	Q15.6
	19	I32.2	I0.2		83	I73.2	I4.2		3	Q32.2	Q8.2		67	Q73.2	Q12.2		110	Q80.5	Q15.5
	20	I32.3	I0.3		84	I73.3	I4.3		4	Q32.3	Q8.3		68	Q73.3	Q12.3		109	Q80.4	Q15.4
	21	I32.4	I0.4		85	I73.4	I4.4		5	Q32.4	Q8.4		69	Q73.4	Q12.4		108	Q80.3	Q15.3
	22	I32.5	I0.5		86	I73.5	I4.5		6	Q32.5	Q8.5		70	Q73.5	Q12.5		107	Q80.2	Q15.2
	23	I32.6	I0.6		87	I73.6	I4.6		7	Q32.6	Q8.6		71	Q73.6	Q12.6		106	Q80.1	Q15.1
	24	I32.7	I0.7		88	I73.7	I4.7		8	Q32.7	Q8.7		72	Q73.7	Q12.7		105	Q80.0	Q15.0
	25	I33.0	I1.0		89	I72.0	I5.0		9	Q33.0	Q9.0		73	Q72.0	Q13.0		104	Q81.7	Q14.7
	26	I33.1	I1.1		90	I72.1	I5.1		10	Q33.1	Q9.1		74	Q72.1	Q13.1		103	Q81.6	Q14.6
	27	I33.2	I1.2		91	I72.2	I5.2		11	Q33.2	Q9.2		75	Q72.2	Q13.2		102	Q81.5	Q14.5
	28	I33.3	I1.3		92	I72.3	I5.3		12	Q33.3	Q9.3		76	Q72.3	Q13.3		101	Q81.4	Q14.4
29	I33.4	I1.4	93	I72.4	I5.4	13	Q33.4		Q9.4	77	Q72.4	Q13.4	100	Q81.3	Q14.3				
30	I33.5	I1.5	94	I72.5	I5.5	14	Q33.5		Q9.5	78	Q72.5	Q13.5	99	Q81.2	Q14.2				
31	I33.6	I1.6	95	I72.6	I5.6	15	Q33.6		Q9.6	79	Q72.6	Q13.6	98	Q81.1	Q14.1				
32	I33.7	I1.7	96	I72.7	I5.7	16	Q33.7		Q9.7	80	Q72.7	Q13.7	97	Q81.0	Q14.0				
MODULO 1	43	I65.0	I2.0	MODULO 3	113	I81.0	I6.0	MODULO 1	33	Q65.0	Q10.0	MODULO 3	97	Q81.0	Q14.0	MODULO 4	112	Q80.7	Q15.7
	50	I65.1	I2.1		114	I81.1	I6.1		34	Q65.1	Q10.1		98	Q81.1	Q14.1		111	Q80.6	Q15.6
	51	I65.2	I2.2		115	I81.2	I6.2		35	Q65.2	Q10.2		99	Q81.2	Q14.2		110	Q80.5	Q15.5
	52	I65.3	I2.3		116	I81.3	I6.3		36	Q65.3	Q10.3		100	Q81.3	Q14.3		109	Q80.4	Q15.4
	53	I65.4	I2.4		117	I81.4	I6.4		37	Q65.4	Q10.4		101	Q81.4	Q14.4		108	Q80.3	Q15.3
	54	I65.5	I2.5		118	I81.5	I6.5		38	Q65.5	Q10.5		102	Q81.5	Q14.5		107	Q80.2	Q15.2
	55	I65.6	I2.6		119	I81.6	I6.6		39	Q65.6	Q10.6		103	Q81.6	Q14.6		106	Q80.1	Q15.1
	56	I65.7	I2.7		120	I81.7	I6.7		40	Q65.7	Q10.7		104	Q81.7	Q14.7		105	Q80.0	Q15.0
	57	I64.0	I3.0		121	I80.0	I7.0		41	Q64.0	Q11.0		105	Q80.0	Q15.0		104	Q81.7	Q14.7
	58	I64.1	I3.1		122	I80.1	I7.1		42	Q64.1	Q11.1		106	Q80.1	Q15.1		103	Q81.6	Q14.6
	59	I64.2	I3.2		123	I80.2	I7.2		43	Q64.2	Q11.2		107	Q80.2	Q15.2		102	Q81.5	Q14.5
	60	I64.3	I3.3		124	I80.3	I7.3		44	Q64.3	Q11.3		108	Q80.3	Q15.3		101	Q81.4	Q14.4
61	I64.4	I3.4	125	I80.4	I7.4	45	Q64.4	Q11.4	109	Q80.4	Q15.4	100	Q81.3	Q14.3					
62	I64.5	I3.5	126	I80.5	I7.5	46	Q64.5	Q11.5	110	Q80.5	Q15.5	99	Q81.2	Q14.2					
63	I64.6	I3.6	127	I80.6	I7.6	47	Q64.6	Q11.6	111	Q80.6	Q15.6	98	Q81.1	Q14.1					
64	I64.7	I3.7	128	I80.7	I7.7	48	Q64.7	Q11.7	112	Q80.7	Q15.7	97	Q81.0	Q14.0					

Fuente: elaboración propia.

- F. Elegir los módulos de entradas y salidas digitales, como muestra la figura siguiente, en las posiciones siguientes del *rack* marcados. El direccionamiento de estos módulos de señal digital en la configuración S7 se puede realizar arreglando las direcciones S7 como estaban en S5 o recableando el programa para pasar direcciones de S5 a S7, lo cual se explica más adelante.

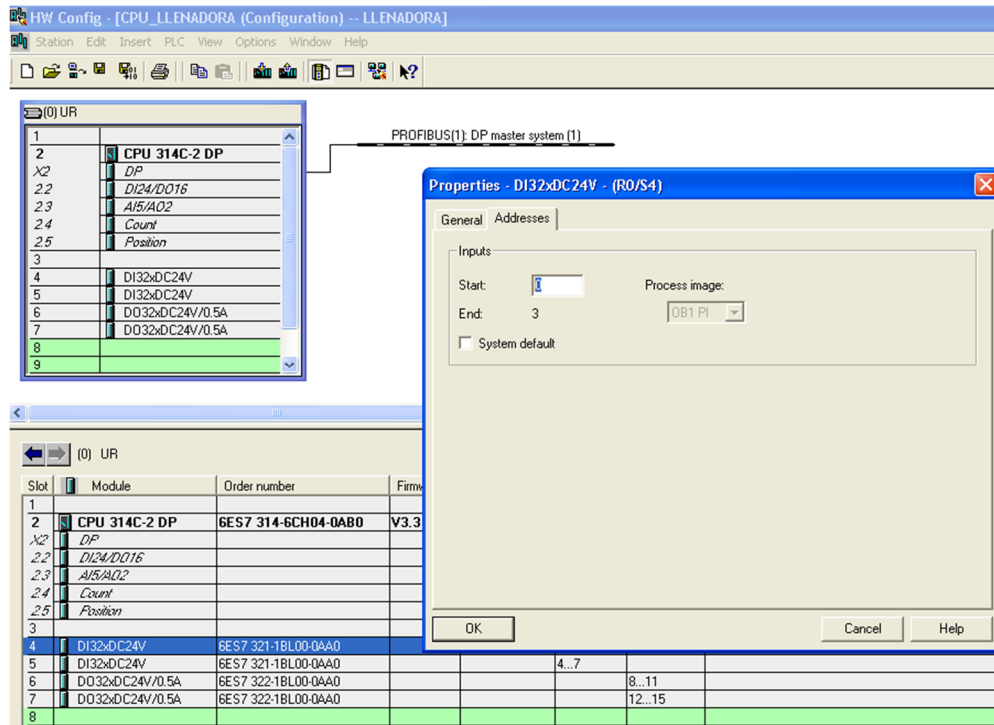
Figura 118. Elegir módulos de entradas y salidas digitales



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- G. Insertar módulos de señal digital y direccionar cada uno de ellos de acuerdo a la figura 117, haciendo doble clic en cada módulo de señal en el rack para desplegar la pantalla que permite cambiar el direccionamiento o simplemente dejar el estándar que el programa asigna automáticamente, como muestra la figura siguiente.

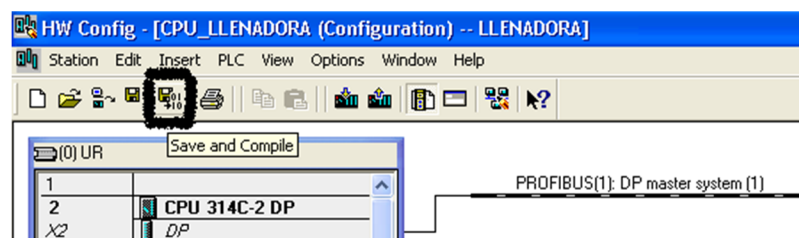
Figura 119. Asignación de direcciones de señales digitales



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- H. Hacer clic en guardar y compilar al terminar de direccionar los módulos de señal, como muestra la figura siguiente.

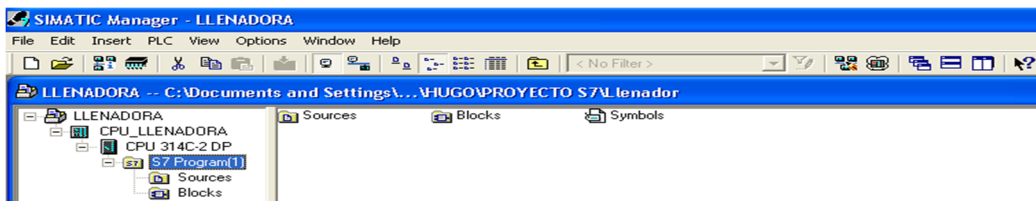
Figura 120. Guardar y compilar la configuración hardware



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- I. La siguiente figura, muestra el resultado de la configuración del hardware en el programa *Manager SIMATIC*.

Figura 121. **Resultado de la configuración hardware**

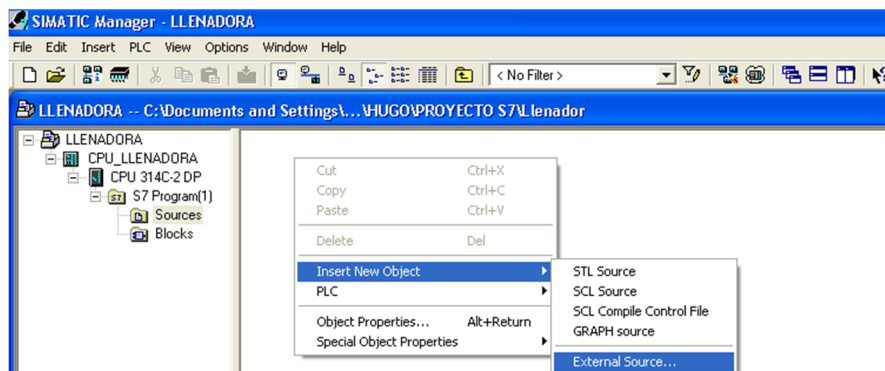


Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

#### 4.1.6. **Generar bloques de programa a partir de archivo convertido**

- A. Hacer clic en *Sources*, luego en la pantalla hacer clic derecho y elegir la opción *External Source*, como muestra la figura siguiente.

Figura 122. **Abrir fuente externa**

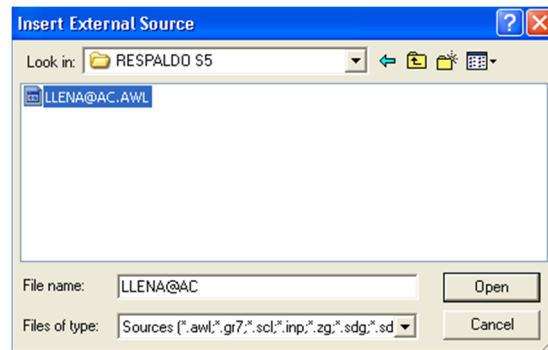


Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.



- B. Buscar el directorio de archivo generado LLENA@AC.AWL para insertarlo en el proyecto, haciendo clic en *Open* como muestra la figura siguiente.

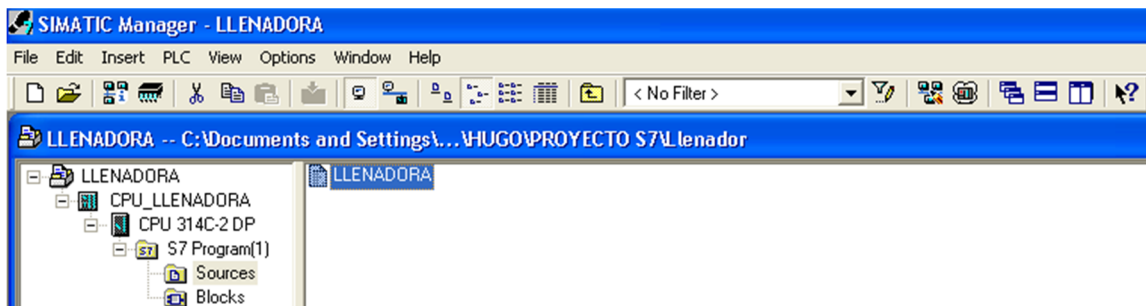
Figura 123. Insertar fuente externa



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- C. La figura siguiente muestra el archivo insertado al cual se le cambió el nombre a LLENADORA.

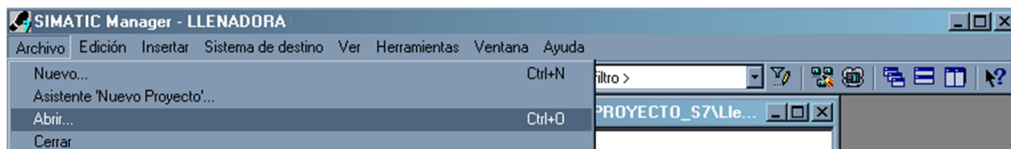
Figura 124. Archivo insertado



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- D. Pegar los bloques; FC 81, FC 82, FC 83 y FC 84 en *Blocks*. Estos son los bloques que tenían un asterisco durante la conversión de archivo S5 a S7, estos se copian desde la librería estándar del programa. Esto es necesario para que el archivo insertado LLENADORA no tenga errores a la hora de compilarlo. Abrir la librería estándar como muestra la figura siguiente.

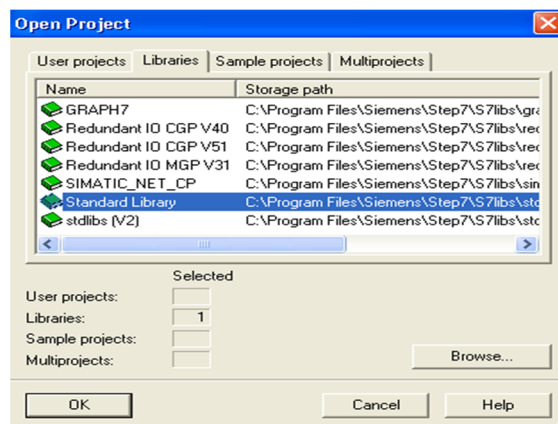
Figura 125. **Abrir librería estándar**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- E. Elegir *Libraries* y seleccionar *Standard Library*, luego hacer clic en *Ok*, como muestra la figura siguiente.

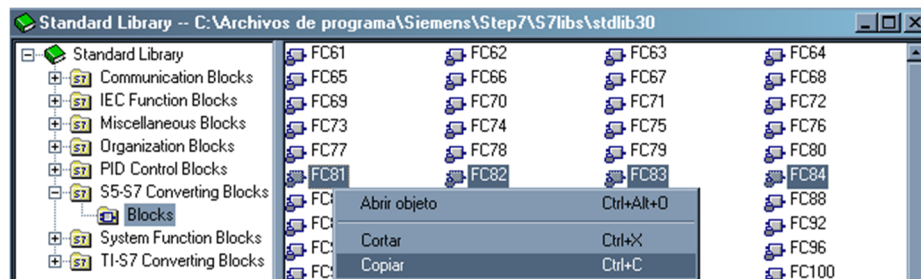
Figura 126. **Elegir *Standard Library***



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- F. Seleccionar y copiar haciendo clic derecho sobre los bloques, como muestra la figura siguiente.

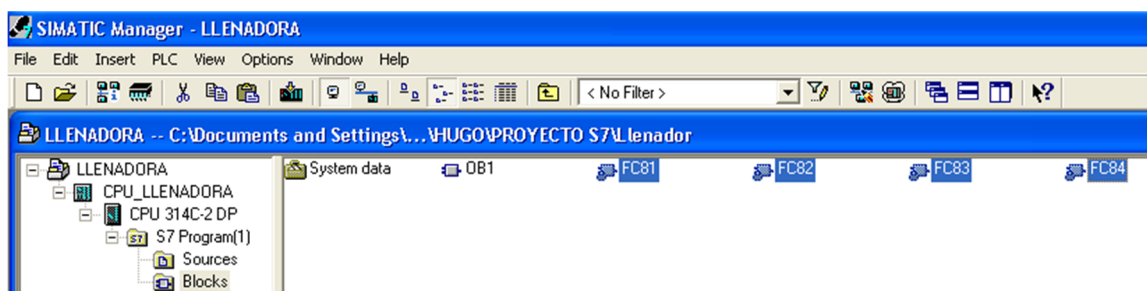
Figura 127. Copiar bloques



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- G. Hacer clic en *Blocks* y luego hacer clic derecho en la pantalla para pegar los bloques elegidos, como muestra la figura siguiente.

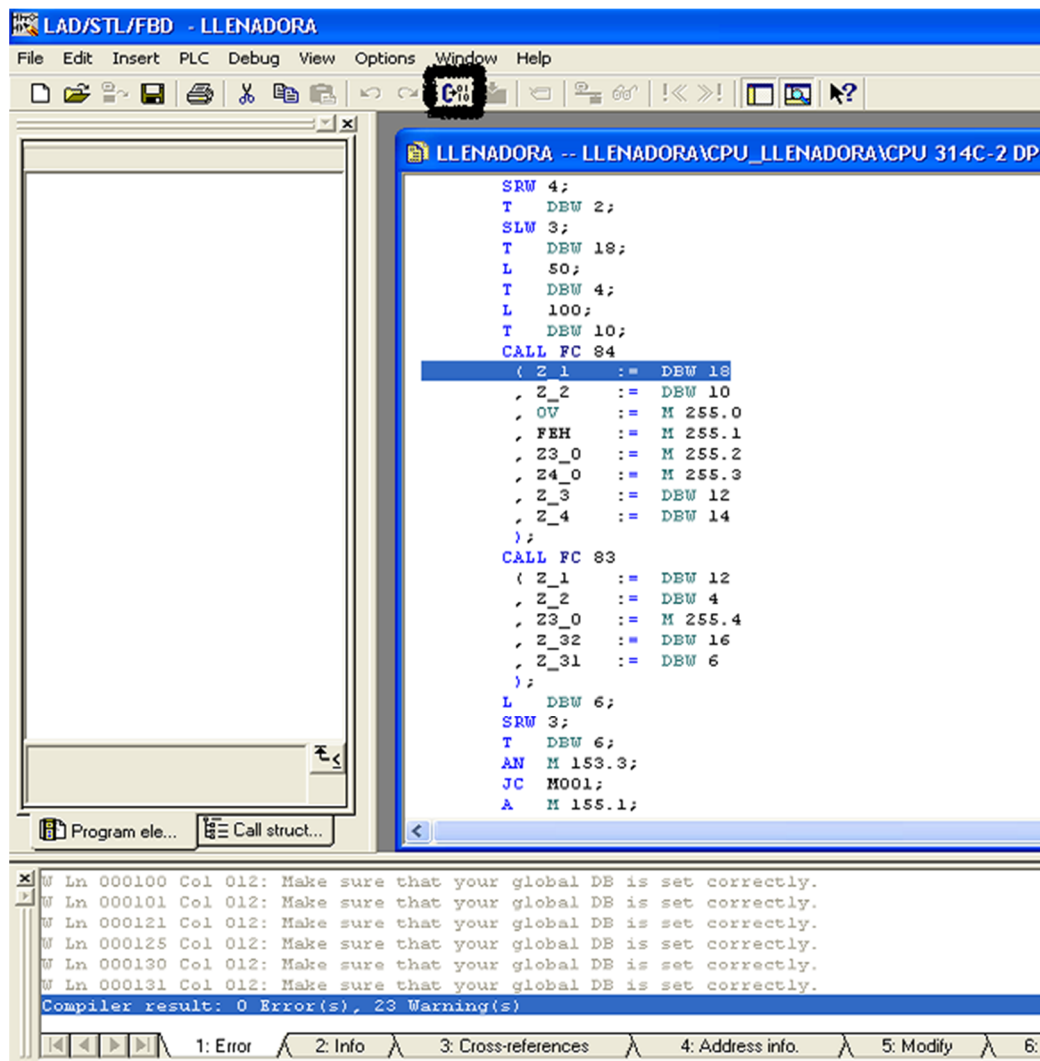
Figura 128. Pegar bloques



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- H. Hacer doble clic sobre el archivo insertado LLENADORA, al abrirse hacer clic en compilar y observar en la parte inferior de la pantalla que no tengan errores.

Figura 129. **Compilar archivo insertado**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- I. Aparecen luego todos los bloques en la parte del proyecto llamada *Blocks*, como muestra la figura siguiente.

Figura 130. **Bloques generados**

The screenshot shows the SIMATIC Manager interface with a project tree on the left and a table of generated blocks on the right. The project tree shows 'LLENADORA' containing 'CPU\_LLENADORA' (CPU 314C-2 DP) and 'S7 Program(1)' with subfolders 'Sources' and 'Blocks'. The table lists the following blocks:

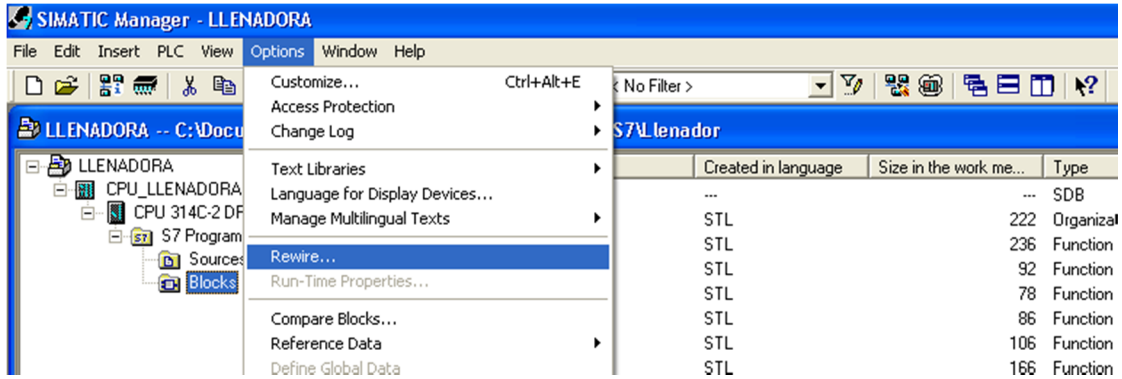
Object name	Symbolic name	Created in language	Size in the work me...	Type	Version (Header)	Name (Header)	Un
System data	---	---	---	SDB	---	---	---
OB1	---	STL	222	Organization Block	0.0	---	---
FC0	---	STL	236	Function	0.0	AUTSPEED	---
FC1	---	STL	92	Function	0.0	JOGSPEED	---
FC2	---	STL	78	Function	0.0	APERTURA	---
FC3	---	STL	86	Function	0.0	CONTGIRI	---
FC4	---	STL	106	Function	0.0	SIC_TAPP	---
FC5	---	STL	166	Function	0.0	SHIFTBIT	---
FC6	---	STL	38	Function	0.0	RLG_AE	---
FC7	---	STL	38	Function	0.0	RLG_AA	---
FC8	---	STL	208	Function	0.0	---	---
FC9	---	STL	148	Function	0.0	---	---
FC10	---	STL	530	Function	0.0	---	---
FC11	---	STL	168	Function	0.0	---	---
FC12	---	STL	266	Function	0.0	---	---
FC13	---	STL	196	Function	0.0	---	---
FC14	---	STL	172	Function	0.0	---	Function
FC15	---	STL	312	Function	0.0	---	---
FC16	---	STL	156	Function	0.0	---	---
FC17	---	STL	108	Function	0.0	---	---
FC18	---	STL	140	Function	0.0	---	---
...	---	---	20	Function	0.0	---	---

Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

#### 4.1.7. Recablear el programa

- A. El recableado de programa es necesario para cambiar el direccionamiento del programa de acuerdo a la figura 117, esto es para cambiar los direccionamientos del S5 al actual S7 que se encuentra en la configuración hardware. Hacer clic en *Blocks*, luego seleccionar *Options*, en la parte *Rewire* hacer clic, como muestra la figura siguiente.

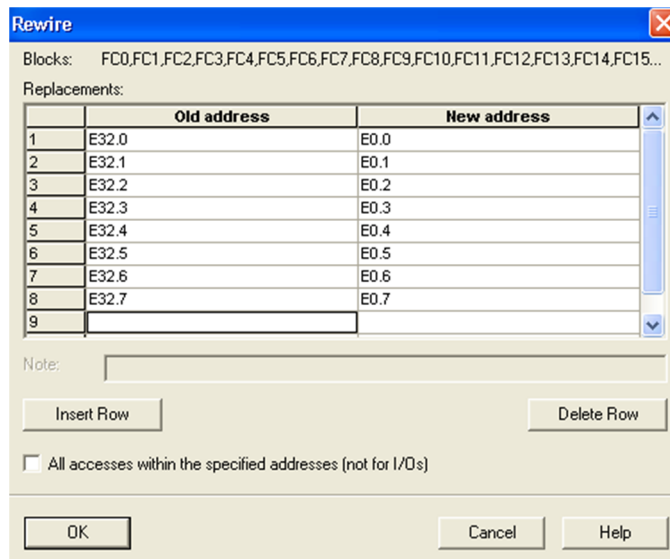
Figura 131. Abrir recableado



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- B. Escribir las direcciones viejas y las nuevas de acuerdo a la figura 117, como muestra la figura siguiente.

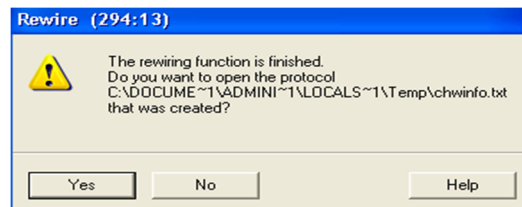
Figura 132. Escribir direcciones viejas y nuevas



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- C. Al terminar de direccionar, hacer clic en *Ok*, luego hacer clic en *Yes*, como muestra la figura siguiente.

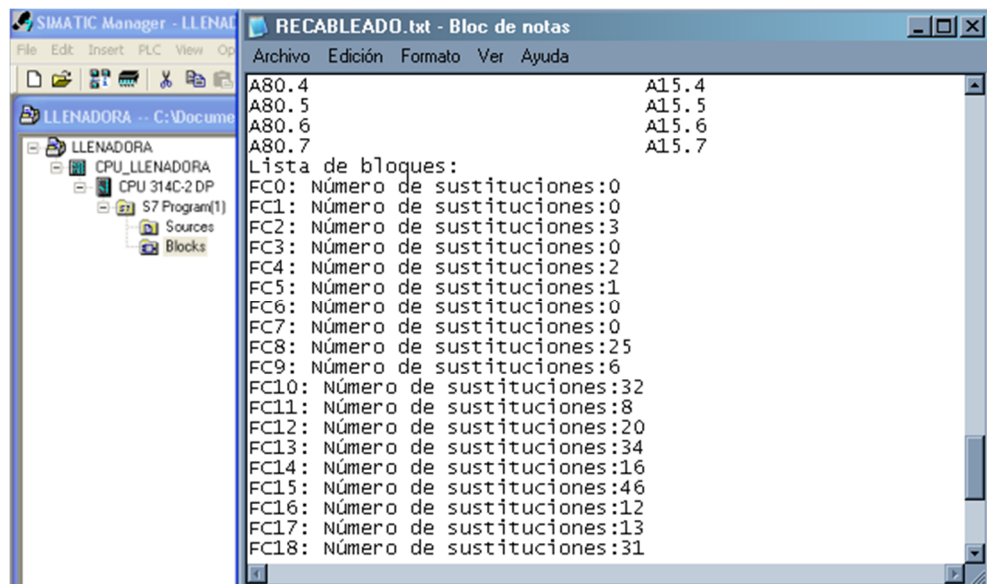
Figura 133. **Función de recableado terminado**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- D. Automáticamente se genera un archivo en bloc de notas, como muestra la figura siguiente.

Figura 134. **Generación de bloc de notas**

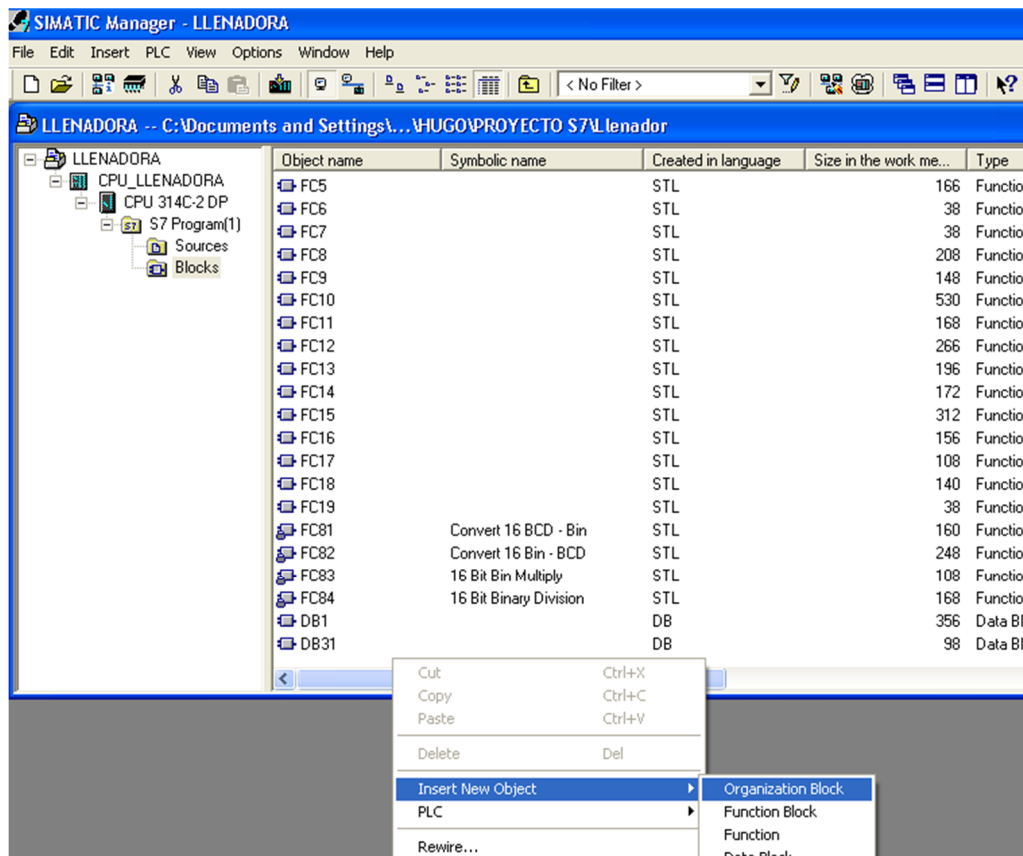


Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

#### 4.1.8. Agregar los bloques de organización para el óptimo funcionamiento

- A. Para que el proyecto funcione adecuadamente y así evitar que la CPU pase a modo *STOP* por errores o alarmas, se debe agregar los siguientes bloques de organización: 40, 82, 83, 85, 86, 87, 100, 121 y 122. Abrir el bloque de organización para elegir los bloques a insertar, haciendo clic derecho en pantalla de *Blocks*, como muestra la figura siguiente.

Figura 135. Abrir bloque de organización

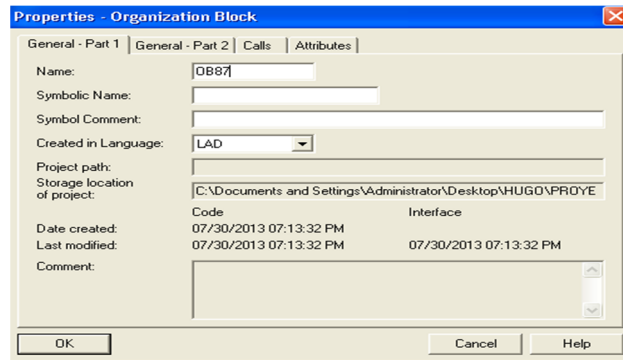


Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.



- B. Escribir el nombre del OB que se necesita insertar y luego hacer clic en **OK** como muestra la figura siguiente.

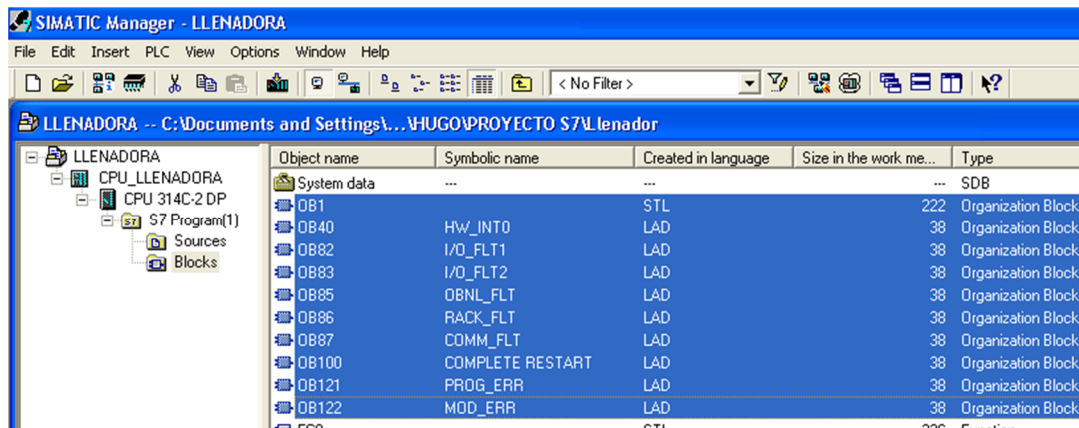
Figura 136. **Escribir el nombre del bloque de organización**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- C. Bloques de organización insertados en el proyecto y mostrados en la figura siguiente.

Figura 137. **Bloques de organización insertados**



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

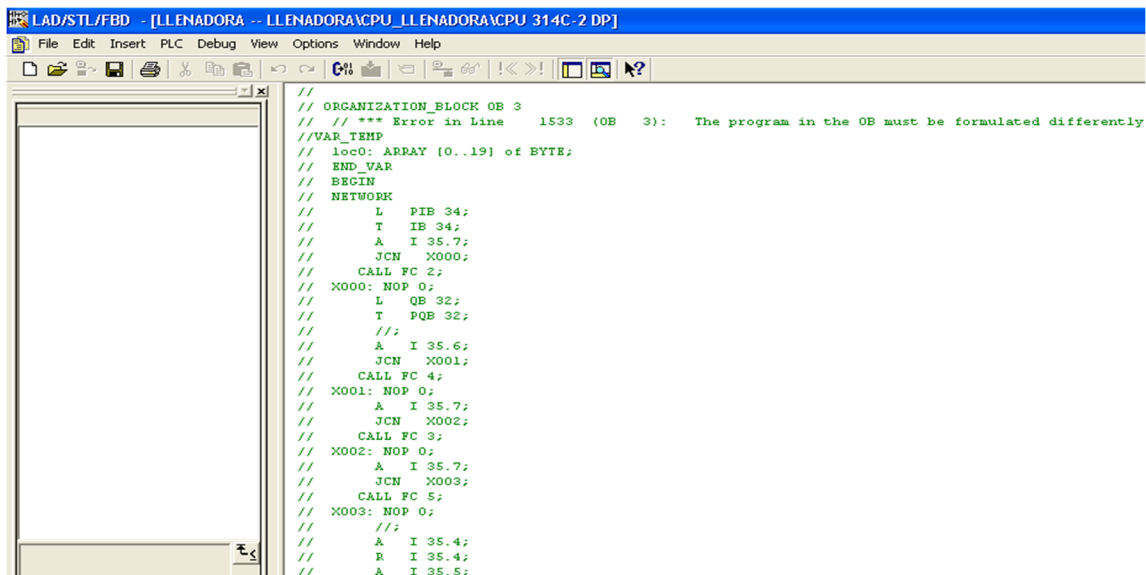
## 4.1.9. Corrección de errores de programa

Los errores de bloque de organización OB 3, OB 31 y OB 251 que se mostraron durante la conversión de archivo S5 a S7, debe de investigarse en el manual S5-95U para comprender su funcionalidad dentro del programa y ver los equivalentes bloques de organización en S7.

### 4.1.9.1. Corrección de errores durante la migración

- A. Abrir archivo fuente (*Sources*) LLENADORA y observar que el único que tiene código de programación es el OB 3, como muestra la figura siguiente, este bloque de organización en el S5 es una interrupción en señal de conteo rápido por lo tanto su equivalente es el OB 40 en S7 que es una alarma de proceso.

Figura 138. Código del OB 3



```
// ORGANIZATION_BLOCK OB 3
// *** Error in Line 1533 (OB 3): The program in the OB must be formulated differently
//VAR_TEMP
// loc0: ARRAY [0..19] of BYTE;
// END_VAR
// BEGIN
// NETWORK
// L PIB 34;
// T IB 34;
// A I 35.7;
// JCN X000;
// CALL FC 2;
// X000: NOP 0;
// L QB 32;
// T PQB 32;
// ///
// A I 35.6;
// JCN X001;
// CALL FC 4;
// X001: NOP 0;
// A I 35.7;
// JCN X002;
// CALL FC 3;
// X002: NOP 0;
// A I 35.7;
// JCN X003;
// CALL FC 5;
// X003: NOP 0;
// ///
// A I 35.4;
// R I 35.4;
// A I 35.5;
```

Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- B. Uno de los requisitos para la ejecución del OB 3 en S5 es que estén habilitadas las entradas del contador en el DB 1. Entonces se debe abrir el programa S5, seleccionar en Editor el *Data Block* y observar que en la parte OBI se tiene parametrizado IP 2 e IP 3 como muestra la figura siguiente. Esto significa, según el manual de usuario del S5-95U, que las entradas 34,2 y 34,3 son las que generan interrupción en el flanco positivo.

Figura 139. Parámetros en el DB1 del S5

```

STEP 5
DB1 C:\LLENACST.S5D LEN=165 /4
0: C = 'AI 1 OBI: IP';
12: C = ' 2 IP 3 OBC: ';
24: C = '#SL1: SLN 1 SF';
36: C = 'DB2 DW0 EF DB3 DW0';
48: C = 'KBE MB100 KBS MB1';
60: C = '01 PGN 1 ;# SDP: N';
72: C = 'T 128 PBUS N ; TFB: OB13';
84: C = '100 ; #CLP: STW MW10';
96: C = '2 CLK DB5 DW0';
108: C = 'SET 3 01.10.91 12:00:00';
120: C = '00 OHS 000000:00:00';
132: C = 'TIS 3 01.10. 12:00:00';
144: C = 'STP Y SAU Y CF 00';
156: C = '; #END';
160:

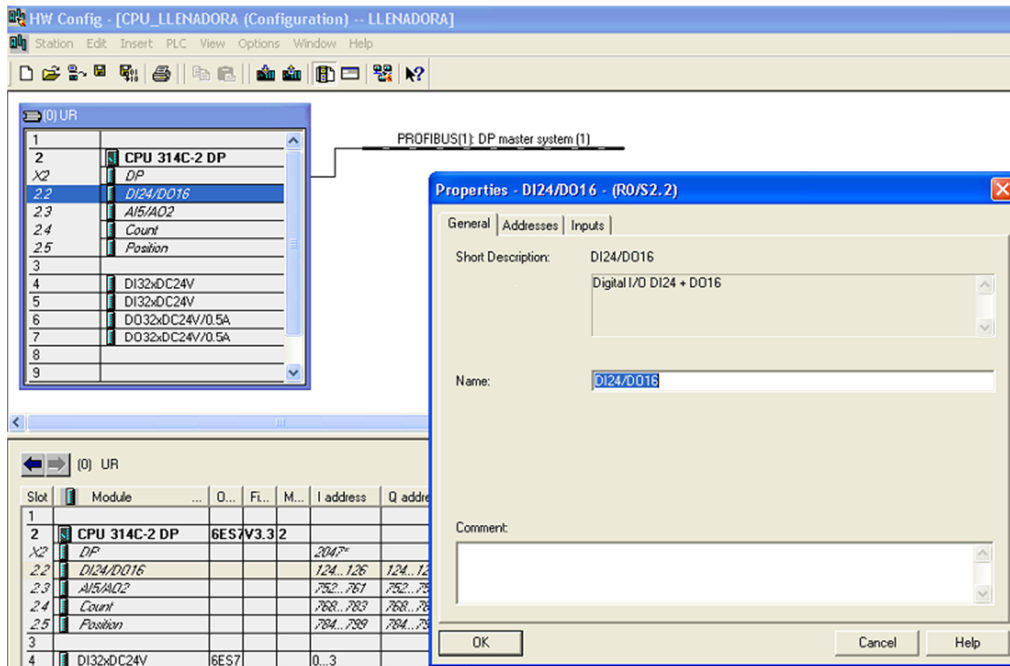
```

Lib No Line Fwd Line Back Title Comment Help  
Expand DC2 Delete DC3 Expand DF4 Delete DF5 KG Test 6 7 Enter 8 Cancel

Fuente: elaboración propia, con base en el programa SIMATIC S5.

- C. Abrir HW Config para abrir DI 24/DO 16 de la CPU haciendo doble clic sobre este, para activar las entradas del contador que estaban en S5, puede cambiar el tiempo de retardo para modificar el pulso en la parte *Input*.

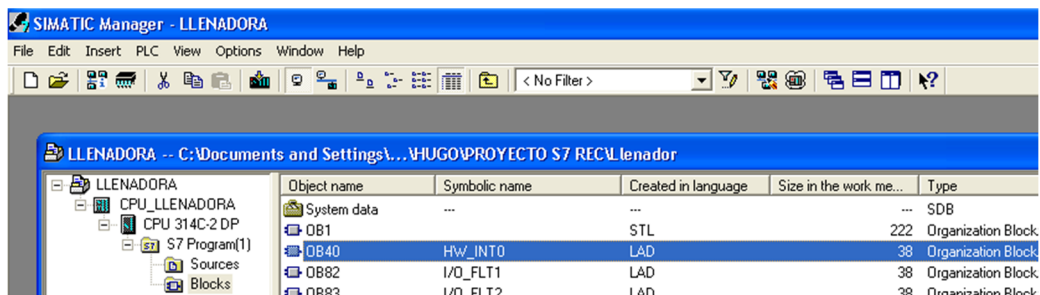
Figura 140. Entradas y salidas digitales para el contador en S7



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- D. Abrir el OB 40 haciendo doble clic sobre el bloque, como muestra la figura siguiente.

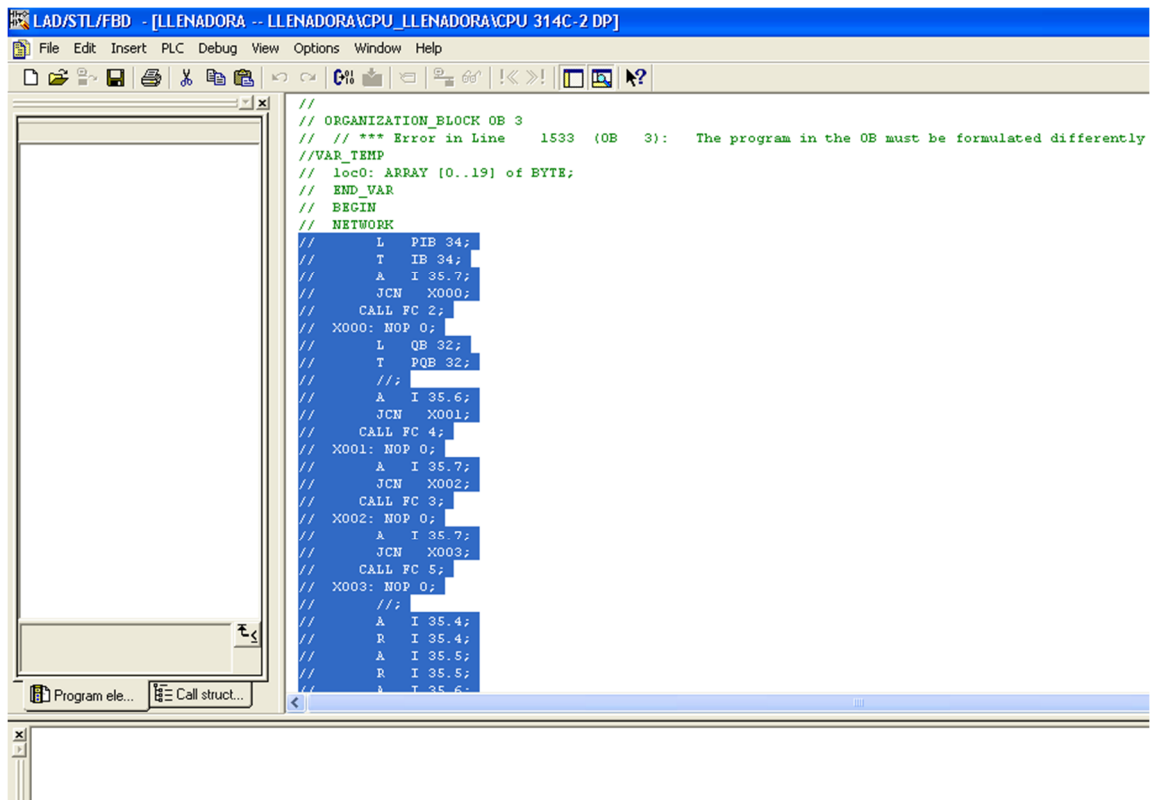
Figura 141. Abrir OB 40



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- E. Copiar el código del OB 3 desde el archivo fuente LLENADORA como muestra la figura siguiente.

Figura 142. Copiar código del OB 3

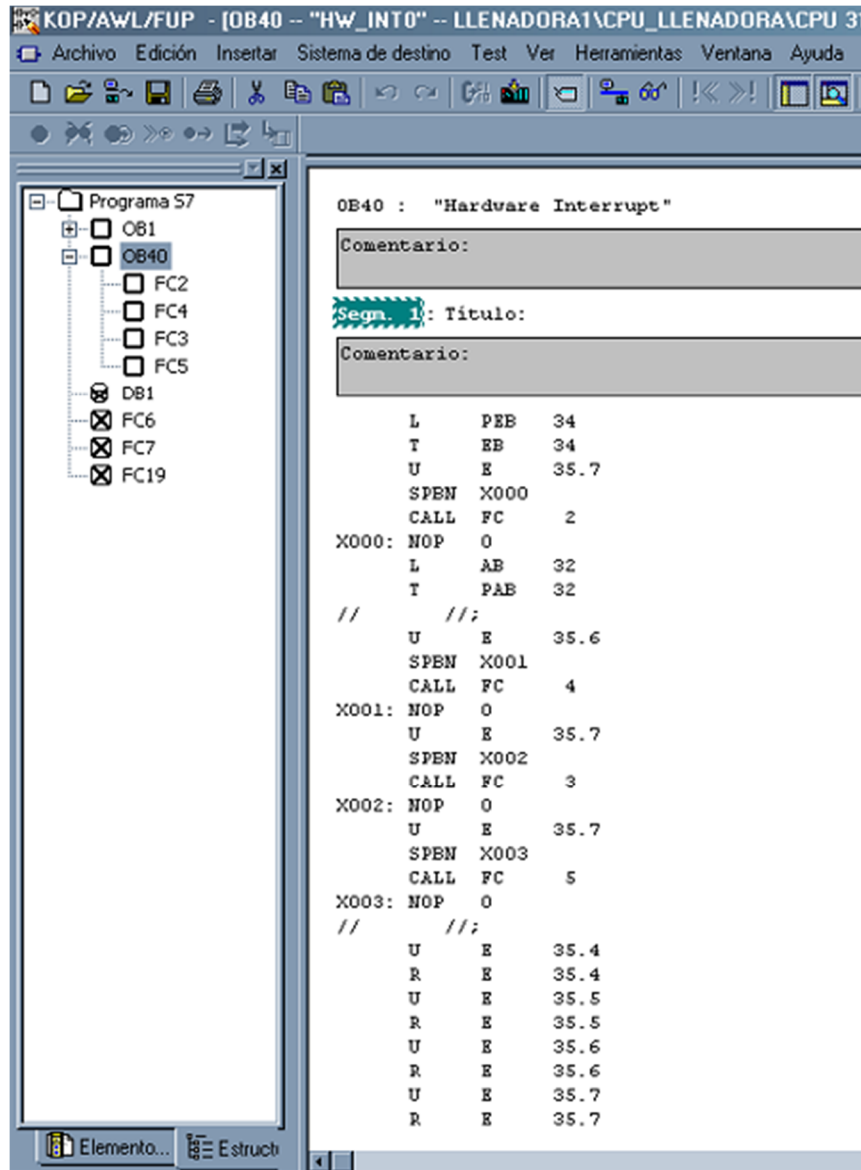


```
// ORGANIZATION_BLOCK OB 3
// *** Error in Line 1533 (OB 3): The program in the OB must be formulated differently
//VAR_TEMP
// loc0: ARRAY [0..19] of BYTE;
// END_VAR
// BEGIN
// NETWORK
//
// L PIB 34;
// T IB 34;
// A I 35.7;
// JCN X000;
// CALL FC 2;
// X000: NOP 0;
// L QB 32;
// T PQB 32;
//
//
// A I 35.6;
// JCN X001;
// CALL FC 4;
// X001: NOP 0;
// A I 35.7;
// JCN X002;
// CALL FC 3;
// X002: NOP 0;
// A I 35.7;
// JCN X003;
// CALL FC 5;
// X003: NOP 0;
//
//
// A I 35.4;
// R I 35.4;
// A I 35.5;
// R I 35.5;
// A I 35.6;
```

Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- F. Pegar y acondicionar el código del OB 3 al OB 40 que realiza la función equivalente, se debe cambiar las direcciones de entradas y salidas digitales en la figura siguiente, esto se debe de realizar pues ahora tiene que adecuarlo al programa S7.

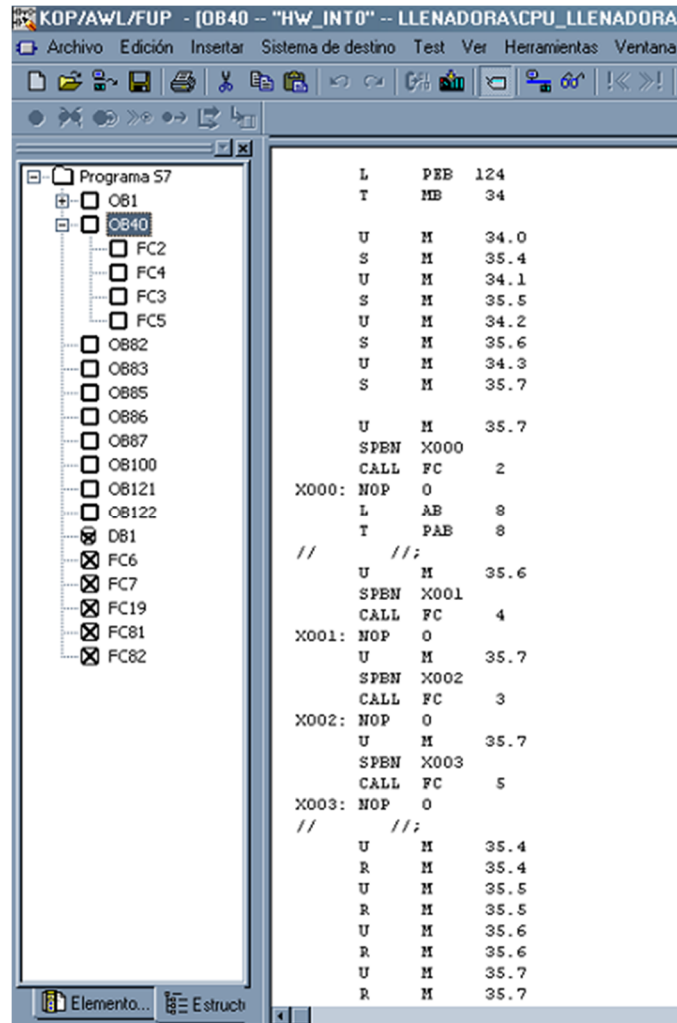
Figura 143. Pegar código del OB 3 al OB 40



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- G. Direcciones cambiados del OB 40 como muestra la figura siguiente, por ejemplo se cambió el PEB 34 al PEB 124 que es la dirección del DI 24/DO 16 que muestra la figura 140.

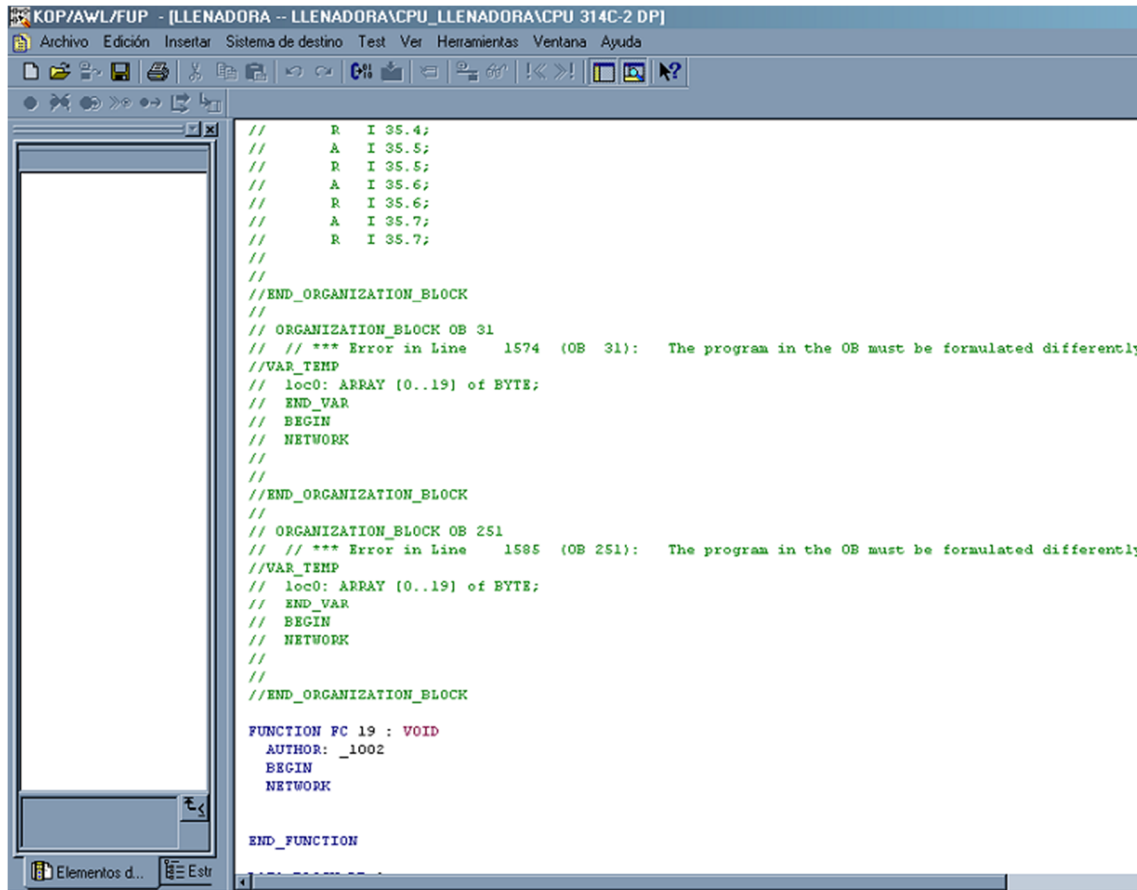
Figura 144. Acondicionamiento del código del OB 40



Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

- H. La figura siguiente muestra el OB 31 que es un redisparo del tiempo del ciclo no tiene código, por lo tanto no se necesita buscar un bloque de organización equivalente para el S7 y el OB 251, que es un algoritmo de regulación PID no es llamado en el programa original, por lo tanto no se necesita buscar un bloque de organización equivalente para el S7.

Figura 145. El OB 31 y OB 251 dentro del archivo fuente



```
//      R   I 35.4;
//      A   I 35.5;
//      R   I 35.5;
//      A   I 35.6;
//      R   I 35.6;
//      A   I 35.7;
//      R   I 35.7;
//
//
//END_ORGANIZATION_BLOCK
//
// ORGANIZATION_BLOCK OB 31
// // *** Error in Line 1574 (OB 31): The program in the OB must be formulated differently
//VAR_TEMP
// loc0: ARRAY [0..19] of BYTE;
// END_VAR
// BEGIN
// NETWORK
//
//
//END_ORGANIZATION_BLOCK
//
// ORGANIZATION_BLOCK OB 251
// // *** Error in Line 1585 (OB 251): The program in the OB must be formulated differently
//VAR_TEMP
// loc0: ARRAY [0..19] of BYTE;
// END_VAR
// BEGIN
// NETWORK
//
//
//END_ORGANIZATION_BLOCK

FUNCTION FC 19 : VOID
AUTHOR: _1002
BEGIN
NETWORK

END_FUNCTION
```

Fuente: elaboración propia, con base en el programa Administrador SIMATIC S7.

## 4.2. Pruebas de migración

Es necesario, antes de la implementación, realizar pruebas en los módulos de entradas y salidas digitales, en la siguiente figura se muestran algunas pruebas realizadas para observar que realmente se accionen estos módulos, con esto se comprueba el funcionamiento correcto de todo el equipo dentro del tablero a instalar.



Figura 146. **Pruebas de migración de entradas y salidas digitales forzando variables**

	Address	Symbol	Display format	Status value	Modify value
1	IB 0		BIN	2#0000_1000	
2	IB 0		HEX	B#16#08	
3	IB 0		DEC	8	
4	I 0.0	"I1.0"	BOOL	false	
5	I 0.3	"I1.3"	BOOL	true	
6					

	Address	Symbol	Display format	Status value
1	Q 6.4	"Q:4.4"	BOOL	true
2				
3	Q 4.4	"Q:3.4"	BOOL	true
4	Q 4.1	"Q:3.1"	BOOL	true
5	Q 4.0		BOOL	true
6				

Fuente: SIEMENS, S. A.

Ahora se debe de conectar el cable Industrial Ethernet a la laptop y al controlador para enviar señales desde nuestra computadora al controlador, como muestra la figura anterior. Se tiene accionado algunas entradas y salidas digitales por medio del forzado de variables, por lo tanto se tiene la siguiente figura que muestra la activación en módulo.

Figura 147. **Pruebas de migración activación de entradas y salidas digitales en módulos**



Fuente: SIEMENS, S. A.

Al activarse las señales en los módulos se debe de verificar que el actuador se active, en este caso sería la activación de cada contactor que alimentara en campo algún equipo eléctrico como muestra la figura siguiente.

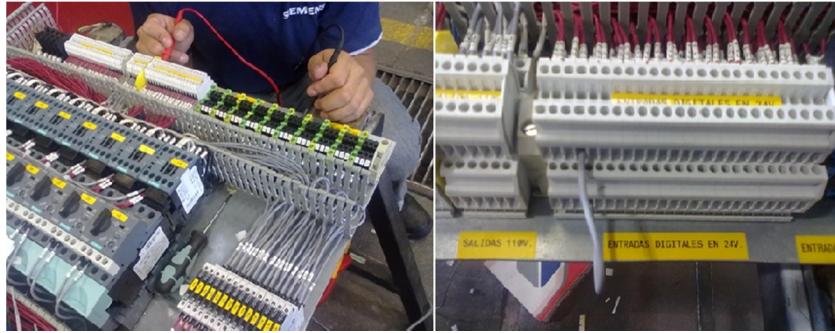
**Figura 148. Pruebas de migración de los actuadores**



Fuente: SIEMENS, S. A.

Para la comprobación de las entradas digitales se envía una señal de 24 V en el bit del respectivo módulo para simular la detección de algún objeto como lo hace un sensor digital en campo, para ello se debe buscar un punto en el tablero con ese nivel de voltaje y simplemente se puentea hacia la entrada digital que se necesita comprobar, como lo muestra la figura siguiente.

Figura 149. **Pruebas de migración de entradas digital**



Fuente: SIEMENS, S. A.

En campo se debe conectar por medio del cable Industria Ethernet el controlador y la laptop para descargar el programa que se convirtió al S7 anteriormente a la CPU del controlador, en las siguientes figuras se observa algunas partes de la implementación de migración realizada, como son el lavado y llenado de los garrafones y se verifica el funcionamiento correcto de todo el ciclo del proceso de producción. También se puede adicionar algunas otras funciones nuevas en el programa durante la migración, si se desea incorporar algún otro equipo a automatizar.

Figura 150. **El tablero antes de la implementación de migración**



Fuente: Cervecería CABRO S.A. Xela.

Figura 151. **El tablero después de la implementación de migración**



Fuente: Cervecería CABRO, S. A. Xela.

Figura 152. **Implementación de migración lavado de garrafones**



Fuente: Cervecería CABRO, S. A. Xela.

Figura 153. **Implementación de migración llenado de garrafones**



Fuente: Cervecería CABRO, S. A. Xela.

## CONCLUSIONES

1. El software Administrador SIMATIC S7 proporciona la herramienta para poder llevar a cabo la migración de archivos S5 a S7, dentro de las herramientas opcionales se tiene el convertidor de archivos S5 el cual transforma un archivo S5 a S7.
2. En este software se realiza la configuración teórica del hardware y se aprende a programar los tres tipos de lenguaje de programación KOP, FUP Y AWL.
3. Los controladores lógico programable S7-300 y S7-400 actualmente están en el mercado, estos utilizan el software Administrador SIMATIC S7 para su programación de automatización que se necesita de acuerdo a la aplicación requerida.
4. La migración se realiza por medio de controladores lógico programable S7-300 y S7-400, para esto se lleva a cabo varios pasos desde una buena elección de la CPU hasta la implementación del archivo convertido de S5 a S7.
5. Una máquina llenadora de bebidas, está compuesta por sensores y actuadores. Los sensores son las partes que detectan presencia o ausencia de objetos y los actuadores realizan el llenado de los objetos. Para que el controlador lógico programable realice las funciones del ciclo de llenado de objetos, se descarga a la CPU el programa convertido a S7 por medio de la comunicación industrial Ethernet.

6. Se realizan pruebas en el tablero nuevo a instalar para comprobar el correcto funcionamiento de todo el equipo eléctrico y el controlador lógico programable.
  
7. En la implementación se comprueba todo el ciclo de llenado de objetos, se corrige parte de la programación para cumplir con el ciclo y se agregan nuevas líneas de programación requeridas para una futura adición de equipos eléctricos.

## RECOMENDACIONES

1. Existen tres tipos de lenguaje de programación KOP, FUP y AWL en el Administrador SIMATIC S7, se debe aprender a programar en el lenguaje de programación AWL. Esto se debe porque al realizar la migración de archivos de S5 a S7, el convertidor utiliza este lenguaje y con esto se facilita la interpretación del programa en dado caso que se necesite realizar una modificación en ella.
2. Actualizar los conocimientos a las nuevas tecnologías de controladores lógico programable, porque algunos llegan al final de su ciclo de vida, por lo tanto se debe migrar a las nuevas tecnologías para seguir con el proceso de la máquina llenadora de bebidas.





## BIBLIOGRAFÍA

1. MANDADO PÉREZ, Enrique. *Autómatas Programables*. Lenza, Manuel (prol.). Buenos Aires: Cengage Learning Argentina, 2008. 463 p.
2. Servicios de modernización y optimización. *Soporte de migración SIMATIC S5-S7* [en línea]. <<http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&siteid=cseus&aktprim=1&extranet=standard&viewreg=WW&objid=16612444&treeLang=es>>. [Consulta: 30 de mayo de 2013].
3. *SIMATIC S7* [CD-ROM]. SIEMENS S.A. SITRAING, 22 de Agosto de 2006 [ref. de 15 diciembre de 2012].
4. Sensores y Actuadores Industriales. Actuadores Industriales [en línea]. [http://tv.uvigo.es/uploads/material/video/1709/ISAD\\_tema7\\_2.pdf](http://tv.uvigo.es/uploads/material/video/1709/ISAD_tema7_2.pdf). [Consulta: 25 de marzo de 2013].



## ANEXOS

### A. Tabla de referencia cruzada de CPU S5 a S7

S5 CPU	S5 Max Memory	Initial S7 CPU to consider	Working Memory of S7 CPU
<b>Low End PLCs</b>			
6ES5 090-8MA01	4KB	6ES7 312-5BE03-0AB0	32KB
6ES5 095-8MA03	16KB	6ES7 313 5BF03-0AB0	64KB
6ES5 095-8MB02	16KB	6ES7 313-6CF03-0AB0	64KB
6ES5 095-8MC01	16KB	6ES7 313-6BF03-0AB0	64KB
6ES5 095-8MD01	16KB	6ES7 313-6CF03-0AB0	64KB
6ES5 095-8ME01	16KB	6ES7 313-6CF03-0AB0	64KB
6ES5 095-8FB01	16KB	6ES7 315-6FF01-0AB0 (*safety PLC, not redundant)	128KB
6ES5 100-8MA02	2KB	6ES7 312-1AE13-0AB0	32KB
6ES5 102-8MA02	4KB	6ES7 312-1AE13-0AB0	32KB
6ES5 103-8MA03	20KB	6ES7 314-6CG03-0AB0	96KB
<b>Intermediate PLCs</b>			
6ES5 941-7UA11		6ES7 315-2AG10-0AB0	128KB
6ES5 941-7UB11	18KB	6ES7 315-2AG10-0AB0	128KB
6ES5 942-7UA11		6ES7 315-2AG10-0AB0	128KB
6ES5 942-7UB11	42KB	6ES7 315-2AG10-0AB0	128KB
6ES5 942-7UF15	32KB	6ES7 315-6FF01-0AB0	128KB
6ES5 942-7UH11	32KB	6ES7 414-4HJ04-0AB0	768KB
6ES5 943-7UA11		6ES7 315-2AG10-0AB0	128KB
6ES5 943-7UB11	48KB	6ES7 315-2AG10-0AB0	128KB
6ES5 943-7UA21		6ES7 315-2AG10-0AB0	128KB
6ES5 943-7UB21	48KB	6ES7 315-2AG10-0AB0	128KB
6ES5 944-7UA11		6ES7 315-2AG10-0AB0	128KB
6ES5 944-7UB11	96KB	6ES7 315-2AG10-0AB0	128KB
6ES5 944-7UA21		6ES7 315-2AG10-0AB0	128KB
6ES5 944-7UB21	96KB	6ES7 315-2AG10-0AB0	128KB
6ES5 945-7UA13	256KB	6ES7 317-2AJ10-0AB0	512KB
6ES5 945-7UA23	384KB	6ES7 317-2AJ10-0AB0	512KB
<b>High End PLCs</b>			
6ES5 921-3UA11		6ES7 412-2XJ05-0AB0	512KB
6ES5 922-3UA11	86KB	6ES7 412-2XJ05-0AB0	512KB
6ES5 928-3UA21	110KB	6ES7 414-2XJ05-0AB0	1MB
6ES5 928-3UB21	110KB	6ES7 414-2XJ05-0AB0	1MB
6ES5 946-*****		6ES7 416-2XN05-0AB0	5.6MB
6ES5 947-*****		INCLUDED IN CPU 416 ABOVE	
6ES5 948-3UA11	640KB	6ES7 416-2XN05-0AB0	5.6MB
6ES5 948-3UA21	1664KB	6ES7 417-4XT05-0AB0	30MB
6ES5 948-3UR12	640KB	6ES7 414-4HJ04-0AB0	1.4MB
6ES5 948-3UR22	1664KB	6ES7 417-4HL04-0AB0	20MB

Fuente: SIEMENS, S. A.

**B. Tabla de referencia cursada de S5 115U a ET200M I/O**

<b>S5 Product</b>	<b>Description</b>	<b>S7 - 300 / ET200M Product</b>	<b>Comment</b>
6ES5 420-7LA11	32 DI 24V DC non-float	6ES7 321-1BL00-0AA0	
6ES5 430-7LA12	32 DI 24V DC Group of 8	6ES7 321-1BH02-0AA0	
6ES5 431-7LA11	16 DI 24/48 V DC Group of 4	6ES7 321-1CH00-0AA0	
6ES5 432-7LA11	16 DI 48/60 V Group of 4	6AG1 321-1CH20-2AA0	
6ES5 435-7LA11	16 DI 115 V AC Group of 1	6ES7 321-1FF10-0AA0	8 Point module
6ES5 435-7LB11	16 DI 115 V AC Group of 2	6ES7 321-1FF01-0AA0	8 Point module
6ES5 435-7LC11	8 DI 115 V AC Group of 1	6ES7 321-1FF10-0AA0	
6ES5 436-7LA11	16 DI 230 V AC Group of 4	6ES7 321-1FH00-0AA0	
6ES5 436-7LB11	16 DI 230V AC Group of 2	6ES7 321-1FF01-0AA0	8 Point module
6ES5 436-7LC11	8 DI 230V AC Group of 1	6ES7 321-1FF10-0AA0	
6ES5 441-7LA12	32 DQ 24V DC 0.5 a non-float	6ES7 322-1BL00-0AA0	
6ES5 451-7LA11	32 DQ 24V DC 0.5 amp	6ES7 322-1BL00-0AA0	
6ES5 451-7LA21	32 DQ 24V DC 0.5 amp	6ES7 322-1BL00-0AA0	
6ES5 453-7LA11	16 DQ 24/60V DC 2 amp		
6ES5 454-7LA12	16 DQ 24V DC 2 amp	6ES7 322-1BF01-0AA0	8 Point module
6ES5 454-7LB11	8 DQ 24V DC 2 amp	6ES7 322-1BF01-0AA0	
6ES5 455-7LA11	16 DQ 48/115V DC	6AG1 322-1CF00-2AA0	8 Point module
6ES5 456-7LA11	16 DQ 115/230V AC	6ES7 322-1FF01-0AA0	8 Point module
6ES5 456-7LB11	8 DQ 115/230V AC	6ES7 322-1FF01-0AA0	
6ES5 457-7LA11	32 DQ 24V DC 100ma	6ES7 322-1BL00-0AA0	
6ES5 458-7LA11	16 DQ Relay 30V DC	6ES7 322-1HH01-0AA0	
6ES5 458-7LB11	8 DQ Relay 250V AC	6ES7 322-1HF01-0AA0	
6ES5 458-7LC11	16 DQ Relay 230V AC	6ES7 322-1HH01-0AA0	
6ES5 482-7LA11	16 DI Source, 16 DQ Source	6ES7 321-1BH50-0AA0	Source In no Out
6ES5 482-7LF11	16 DI Sink, 16 DQ Source		
6ES5 482-7LF21	16 DI Source, 16 DQ Sink		
6ES5 482-7LF31	8DI, 8 DQ Source or Sink	6ES7 321-1BH50-0AA0	Source In no Out
6ES5 485-7LA11	24 DI, 16 DQ 24V DC	6ES7 323-1BL00-0AA0	16 Point In 16 Point Out
6ES5 460-7LA13	8 Channel AI Float	6ES7 331-7KF02-0AB0	
6ES5 465-7LA13	16 Channel AI Non Float	6ES7 331-1KF01-0AB0	8 Channel
6ES5 470-7LA13	8 Channel AQ +-10V 0-20ma	6ES7 332-5HF00-0AB0	
6ES5 470-7LB13	8 Channel AQ +-10V	6ES7 332-5HF00-0AB0	
6ES5 470-7LA13	8 Channel AQ 1-5V 4-20ma	6ES7 332-5HF00-0AB0	

Fuente: SIEMENS, S. A.

### C. Tabla de referencia cruzada de S5 135U y 155U a ET200M I/O

S5 Product	Description	S7 - 300 / ET200M Product	Comment
6ES5 420-7LA11	32 DI 24V DC non-float	6ES7 321-1BL00-0AA0	
6ES5 430-7LA12	32 DI 24V DC Group of 8	6ES7 321-1BH02-0AA0	
6ES5 431-7LA11	16 DI 24/48 V DC Group of 4	6ES7 321-1CH00-0AA0	
6ES5 432-7LA11	16 DI 48/60 V Group of 4	6AG1 321-1CH20-2AA0	
6ES5 435-7LA11	16 DI 115 V AC Group of 1	6ES7 321-1FF10-0AA0	8 Point module
6ES5 435-7LB11	16 DI 115 V AC Group of 2	6ES7 321-1FF01-0AA0	8 Point module
6ES5 435-7LC11	8 DI 115 V AC Group of 1	6ES7 321-1FF10-0AA0	
6ES5 436-7LA11	16 DI 230 V AC Group of 4	6ES7 321-1FH00-0AA0	
6ES5 436-7LB11	16 DI 230V AC Group of 2	6ES7 321-1FF01-0AA0	8 Point module
6ES5 436-7LC11	8 DI 230V AC Group of 1	6ES7 321-1FF10-0AA0	
6ES5 441-7LA12	32 DQ 24V DC 0.5 a non-float	6ES7 322-1BL00-0AA0	
6ES5 451-7LA11	32 DQ 24V DC 0.5 amp	6ES7 322-1BL00-0AA0	
6ES5 451-7LA21	32 DQ 24V DC 0.5 amp	6ES7 322-1BL00-0AA0	
6ES5 453-7LA11	16 DQ 24/60V DC 2 amp		
6ES5 454-7LA12	16 DQ 24V DC 2 amp	6ES7 322-1BF01-0AA0	8 Point module
6ES5 454-7LB11	8 DQ 24V DC 2 amp	6ES7 322-1BF01-0AA0	
6ES5 455-7LA11	16 DQ 48/115V DC	6AG1 322-1CF00-2AA0	8 Point module
6ES5 456-7LA11	16 DQ 115/230V AC	6ES7 322-1FF01-0AA0	8 Point module
6ES5 456-7LB11	8 DQ 115/230V AC	6ES7 322-1FF01-0AA0	
6ES5 457-7LA11	32 DQ 24V DC 100ma	6ES7 322-1BL00-0AA0	
6ES5 458-7LA11	16 DQ Relay 30V DC	6ES7 322-1HH01-0AA0	
6ES5 458-7LB11	8 DQ Relay 250V AC	6ES7 322-1HF01-0AA0	
6ES5 458-7LC11	16 DQ Relay 230V AC	6ES7 322-1HH01-0AA0	
6ES5 482-7LA11	16 DI Source, 16 DQ Source	6ES7 321-1BH50-0AA0	Source In no Out
6ES5 482-7LF11	16 DI Sink, 16 DQ Source		
6ES5 482-7LF21	16 DI Source, 16 DQ Sink		
6ES5 482-7LF31	8DI, 8 DQ Source or Sink	6ES7 321-1BH50-0AA0	Source In no Out
6ES5 485-7LA11	24 DI, 16 DQ 24V DC	6ES7 323-1BL00-0AA0	16 Point In 16 Point Out
6ES5 460-7LA13	8 Channel AI Float	6ES7 331-7KF02-0AB0	
6ES5 465-7LA13	16 Channel AI Non Float	6ES7 331-1KF01-0AB0	8 Channel
6ES5 470-7LA13	8 Channel AQ +-10V 0-20ma	6ES7 332-5HF00-0AB0	
6ES5 470-7LB13	8 Channel AQ +-10V	6ES7 332-5HF00-0AB0	
6ES5 470-7LA13	8 Channel AQ 1-5V 4-20ma	6ES7 332-5HF00-0AB0	

Fuente: SIEMENS, S. A.

