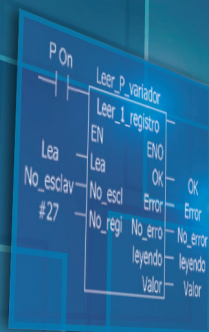


MANIOBRAS EN AUTÓMATAS PROGRAMABLES

DE ACUERDO CON
LA NORMA IEC-61131-3

Juan Pérez Cruz
Manuel Pineda Sánchez
Rubén Puche Panadero
José Roger Folch
Ángel Sapena Bañó



EDITORIAL
UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Juan Pérez Cruz

Manuel Pineda Sánchez

Rubén Puche Panadero

Jose Roger Folch

Angel Sapena Bañó

**Maniobras en autómatas programables
de acuerdo
con la Norma IEC-61131-3**

EDITORIAL

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Los contenidos de esta publicación han sido evaluados por el Comité Editorial UPV mediante el sistema *doble ciego*

Para referenciar esta publicación utilice la siguiente cita: PÉREZ CRUZ, J.,[et al.](2013) *Maniobras en autómatas programables de acuerdo con la norma IEC 6113..* Valencia : Universitat Politècnica

Primera edición, 2013

©Juan Pérez Cruz
Manuel Pineda Sánchez
Rubén Puche Panadero
José Roger Folch
Ángel Sapena Bañó,

© de la presente edición: Editorial Universitat Politècnica de València

Distribución: pedidos@editorial.upv.es /
Tel. 96 387 70 12/ www.editorial.upv.es / Ref. 444

Imprime: By print percom S.L.

ISBN: 978 84 8363 967 2
Impreso bajo demanda

Queda prohibida la reproducción, la distribución, la comercialización, la transformación y, en general, cualquier otra forma de explotación, por cualquier procedimiento, de la totalidad o de cualquier parte de esta obra sin autorización expresa y por escrito de los autores.

Impreso en España

Índice

Índice de Figuras	9
Prólogo	17
1. Objetivo	19
2. Introducción	21
2.1. Tipos de Unidades Organizativas del Programa (POU)	23
2.2. Componentes de las Unidades Organizativas del Programa (POU)	25
3. Introducción a las tecnologías de programación enumeradas en la Norma IEC61131-3	27
3.1. Creación de un ejemplo básico de POU (bloque de función).	29
3.1.1. Generación de un bloque de función con el CX-Programmer. ...	29
3.1.2. Generación de un bloque de función con CODESYS v3.3.	42
3.2. Aplicación en ejemplos prácticos de la función “fun_and”.	54
4. Funciones de tiempo.....	55
4.1. Función temporizador de tipo “TON”	56
4.1.1. Aplicación de la función “TON”.....	58
4.2. Función temporizador de pulso de entrada (PT).....	59
4.2.1. Función temporizador tipo (PT) con CX-Programmer.	60
4.2.2. Aplicación de la función “PT”.	62
4.3. Función temporizador de tipo “TOF”	63
4.3.1. Temporizador (TOF) con CX-Programmer y lenguaje de contactos o ladder.....	64
4.3.2. Función temporizador tipo “TOF” con CX-Programmer y lenguaje ST.....	66
4.3.3. Aplicación de la función “TOF”.	70

5. Funciones de conteaje.	73
5.1. Contadores en Codesys	73
5.1.1. Función contador creciente o “CTU”	73
5.1.2. Función contador decreciente o “CTD”	74
5.1.3. Función contador reversible o “CTUD”	76
5.2. Contadores según la Norma IEC 61131-3	78
5.2.1. Creación de un contador según norma de tipo ascendente o “CTU”	78
5.2.2. Creación de un contador según la norma, de tipo descendente o “CTD”	81
5.2.3. Creación de un contador según norma, de tipo reversible o “CTUD”	84
6. Bloques funcionales o programas con maniobras industriales simples.....	89
6.1. Pulsador de paro y marcha	90
6.1.1. Objetivo	90
6.1.2. Planteamiento	91
6.1.3. Cronograma	91
6.1.4. Resolución del bloque funcional con la tarea paro-marcha.....	91
6.1.5. Aplicaciones prácticas de la función “Marcha_Paro”	93
6.2. Pulsador de paro y marcha y protección mediante un relé térmico.....	94
6.2.1. Objetivo	94
6.2.2. Planteamiento	94
6.2.3. Cronograma	95
6.2.4. Diseño del bloque funcional con la tarea paro-marcha y relé térmico	95
6.3. Bloque funcional de un arrancador estrella-triángulo.	97
6.3.1. Objetivo	97
6.3.2. Planteamiento	98

6.3.3. Resolución del bloque funcional del arrancador estrella triángulo, en lenguaje de contactos (LD)	98
6.3.4. Resolución del bloque funcional del arrancador estrella triángulo, en lenguaje de texto estructurado (ST)	101
6.3.5. Función del arrancador estrella triángulo, con el CX-Programmer	103
6.3.6. Aplicaciones de la función “Arrancador_estrella_triángulo.	105
6.4. Bloque funcional de control de mantenimiento del nivel de un depósito	106
6.4.1. Objetivo.....	106
6.4.2. Planteamiento	106
6.4.3. Resolución del Bloque funcional de la aplicación “Depósito” .	106
6.4.4. Aplicaciones de la función “Depósito”	107
6.5. Bloque funcional de control de mantenimiento del nivel de un depósito (2).	109
6.5.1. Objetivo.....	109
6.5.2. Planteamiento.....	109
6.5.3. Resolución del Bloque funcional de la aplicación “Deposito_tiempo”.....	109
6.5.4. Aplicaciones de la función “Depósito_tiempo”.	111
6.6. Bloque funcional de control de mantenimiento del nivel de un depósito (3).	112
6.6.1. Objetivo.....	112
6.6.2. Planteamiento.....	112
6.6.3. Resolución del Bloque funcional (FB) de la aplicación “Deposito_tiempo_avería”.....	113
6.7. Bloque funcional de control de mantenimiento del nivel de un depósito (4).	115
6.7.1. Objetivo.....	115
6.7.2. Planteamiento.....	115
6.7.3. Resolución del Bloque funcional (FB) de la aplicación “Deposito_2sensores”.....	116

6.8. Bloque funcional de control de mantenimiento del nivel de un depósito (5)	119
6.8.1. Objetivo	119
6.8.2. Planteamiento	119
6.8.3. Resolución del FB de la aplicación “Control_abastecimiento_de_2_cosumos”	120
6.8.4. Aplicaciones de las FB denominadas depósito.....	123
6.9. Bloque funcional de control de achique de agua en un local	124
6.9.1. Objetivo	124
6.9.2. Planteamiento	124
6.9.3. Resolución del FB de la aplicación “inundación_garaje”	125
6.10. Bloque funcional de control de achique de agua en un local 2	129
6.10.1. Objetivo.....	129
6.10.2. Planteamiento.....	130
6.10.3. Resolución del FB de la aplicación “achique_galeria_de_servicio”.....	130
6.10.4. Otras aplicaciones de este bloque de función.....	134
7. Bloques funcionales dedicados al control de aparcamientos.....	135
7.1. Función para el control básico de un parking, con un único sensor.....	135
7.1.1. Objetivo	135
7.1.2. Planteamiento	136
7.1.3. Resolución del FB de la aplicación “parking_sensor_vigilante”	136
7.2. Función para control básico de un aparcamiento (parking 2), con dos sensores.....	140
7.2.1. Objetivo	140
7.2.2. Planteamiento	141
7.2.3. Resolución del FB de la aplicación “parking_2_sensores”	141

7.3. Función para control de un parking (3), con dos sensores y una barrera	147
7.3.1. Objetivo	147
7.3.2. Planteamiento	147
7.3.3. Resolución del FB de la aplicación “parking_2_sensores_barrera”	148
7.4. Función para control de un parking (4), con dos plantas de aparcamiento	153
7.4.1. Objetivo	153
7.4.2. Planteamiento	154
7.4.3. Resolución del FB de la aplicación “parking_2_plantas”	154
7.5. Otras aplicaciones	160
8. Función para control de una taladradora automática.....	161
8.1. Función para control de una taladradora automática (1), un ciclo	162
8.1.1. Objetivo	162
8.1.2. Planteamiento	162
8.1.3. Resolución del FB de la aplicación “Taladro”	163
8.2. Función para control de una taladradora automática (2), multi-ciclo	168
8.2.1. Objetivo	168
8.2.2. Planteamiento	169
8.2.3. Resolución del FB de la aplicación “Taladro_repetición_ciclos”	170
8.3. Función para taladradora automática (3), multiciclo con alarmas	175
8.3.1. Objetivo	175
8.3.2. Planteamiento	176
8.3.3. Resolución del FB de la aplicación “Taladro_repetición_ciclos_alarma”	176
8.4. Aplicaciones similares	179

9. Reguladores Industriales en los Autómatas Programables.....	181
9.1. Función Regulador Proporcional (P)	183
9.1.1. Objetivo.....	184
9.1.2. Planteamiento.....	185
9.2. Función Integral (I)	187
9.2.1. Objetivo.....	188
9.2.2. Planteamiento y programación.....	188
9.3. Función regulador derivativo (o diferencial) (D).....	191
9.3.1. Objetivo.....	192
9.3.2. Planteamiento y programación.....	192
9.4. Función Proporcional e Integral (PI).....	195
9.4.1. Objetivo.....	195
9.4.2. Planteamiento y programación.....	195
9.5. Función Proporcional y Derivativa (PD)	198
9.5.1. Objetivo.....	198
9.5.2. Planteamiento y programación.....	198
9.6. Función del regulador Proporcional, Integral y Derivativa (PID)	200
9.6.1. Objetivo.....	200
9.6.2. Planteamiento y programación con Codesys.	200
9.6.3. Planteamiento y programación con CX-Programmer.	203
10. Bibliografía	207

Índice de Figuras

Figura 1.-Partes de una POU. _____	25
Figura 2.- Entorno de programación del CX-Programmer. _____	29
Figura 3.- Características de los menús seleccionables. _____	30
Figura 4.- Opciones para insertar un bloque de función. _____	31
Figura 5.- Opciones de renombrar y guardar el bloque de función en un archivo _____	33
Figura 6.- Entorno de programación de un bloque de función. _____	34
Figura 7.- Menú de inserción de nuevas variables. _____	35
Figura 8.- Opciones de la variable. _____	36
Figura 9.- Definición de la variable de entrada “a”. _____	37
Figura 10.- Instrucciones que debe realizar el bloque de función. _____	37
Figura 11.- Aspecto de la ventana de programación con una nueva función. _	38
Figura 12.- Selección de un nuevo bloque de función creado por el usuario. _	38
Figura 13.- Proceso de inserción de un nuevo bloque de función. _____	39
Figura 14.- Aspecto de una función creada dentro de un programa. _____	40
Figura 15.- Proceso de conexión de un bloque funcional. _____	40
Figura 16.- Selección de una variable de entrada/salida. _____	41
Figura 17.- Aspecto de un programa realizado mediante un bloque funcional. 41	
Figura 18.- Aspecto de un programa realizado mediante dos bloques funcionales. _____	42
Figura 19.- Entorno de programación Codesys v3.3. _____	43
Figura 20.- Agregar una POU nueva en Codesys. _____	44
Figura 21.- Designación de una nueva POU, en Codesys. _____	45
Figura 22.- Elección del POU "fun_and" y sus características. _____	46
Figura 23.- Estructura de programación del bloque de función. _____	47
Figura 24.- Bloque de función “fun_and” creado en Codesys. _____	48
Figura 25.- Aspecto del bloque de función generado en Codesys. _____	49

Figura 26.- Acceso a la programación de una aplicación. _____	50
Figura 27.- Declaración de las variables a utilizar. _____	51
Figura 28.- Ejemplo utilización de un bloque de función. _____	52
Figura 29.- Utilización de un mismo bloque de función en dos líneas de programa. _____	53
Figura 30.- Descripción del TON y simbología de representación con PT=10s. _____	57
Figura 31.- Descripción del temporizador “TON” del CX-Programmer. _____	58
Figura 32.- Descripción temporizador PT y simbología de representación. _____	60
Figura 33.- Programación del temporizador de impulso, mediante el CX-Programmer. _____	61
Figura 34.- Utilización del temporizador de impulso en una aplicación. _____	62
Figura 35.- Descripción TOF y simbología de representación con PT=10s. _____	63
Figura 36.- Programación del bloque de función TOF, en lenguaje “LD”. _____	65
Figura 37.- Ejemplo de utilización en una aplicación del bloque de función TOF. _____	65
Figura 38.- Programación del bloque de función TON. _____	67
Figura 39.- Programación del bloque de función. _____	68
Figura 40.- Ejemplo de aplicación de ambos bloques de función. _____	70
Figura 41.- Descripción del contador CTU. _____	74
Figura 42.- Descripción del contador CTD. _____	75
Figura 43.- Descripción del contador CTUD. _____	77
Figura 44.- Declaración de variables de la “Contador_ascendente”. _____	79
Figura 45.- Contador de pulsos incremental de acuerdo con la norma. _____	80
Figura 46.- Comparador de las variables CV y PV. _____	80
Figura 47.- Reinicialización del contador. _____	81
Figura 48.- Bloque de función normativo del contador ascendente CTU. _____	81
Figura 49.- Declaración de variables de la función CTD. _____	82
Figura 50.- Establecer el contador a su valor máximo. _____	82
Figura 51.- Tarea de descenso controlado del contador CTD. _____	83

Figura 52.- Tarea de activación de la salida Q. _____	83
Figura 53.- Bloque de función normativo del contador descendente CTD. __	84
Figura 54.- Declaración de variables de la función CTUD. _____	85
Figura 55.- Puesta a cero del contador CTUD. _____	85
Figura 56.- Puesta al valor máximo del contador CTUD. _____	86
Figura 57.- Incremento y decremento del contador reversible CTUD. _____	86
Figura 58.- Activación de las salidas correspondientes del CTUD. _____	87
Figura 59.- Botoneras de marcha paro y de botoneras con botón de paro tipo seta. _____	90
Figura 60.- Cronograma de la función “Marcha_Paro”. _____	91
Figura 61.- Declaración de variables de la función “Marcha_Paro”. _____	92
Figura 62.- Programación de la función “Marcha_Paro”, en lenguaje LD. __	92
Figura 63.- Programación de la función “Marcha_Paro” en lenguaje ST. ____	92
Figura 64.- Aspecto del bloque de función “Marcha_Paro”. _____	93
Figura 65.- Cronograma de la función “Marcha_Paro_térmico”. _____	95
Figura 66.- Declaración de variables de la función “Marcha_Paro_Termico”. 95	
Figura 67.- Implementación de tareas de la función. _____	96
Figura 68.- Aspecto del bloque de función “Marcha_Paro_termico”. _____	96
Figura 69.- Variables de la función “Arrancador_estrella_triangulo”. _____	99
Figura 70.- Puesta en marcha del contactor de línea. _____	99
Figura 71.- Temporizador para llevar a cabo la conmutación. _____	100
Figura 72.- Control de los contactores de estrella y de triángulo. _____	100
Figura 73.- Bloque de función del arrancador estrella-triángulo. _____	101
Figura 74.- Variables para la función “Arrancador_estrella_triangulo_st”. _	102
Figura 75.- Programación estrella triángulo en texto estructurado. _____	102
Figura 76.- Función “arrancador_estrella_triangulo”, con CX-Programmer. 104	
Figura 77.- Aspecto de la función y programa en CX-Programmer. _____	104
Figura 78.- Declaración de variables de la función “Deposito”. _____	106
Figura 79.- Tarea de la función “Deposito”. _____	107

Figura 80.- Aspecto del bloque de función “Depósito”.	107
Figura 81.- Declaración de variables de la nueva función “Deposito_tiempo”.	110
Figura 82.- Tiempo de retardo para reiniciar la marcha de la bomba.	110
Figura 83.- Aspecto del bloque de función.	110
Figura 84.- Esquema de un depósito, con dispositivo de alarma	113
Figura 85.- Declaración de variables de la función “Deposito_tiempo_averia”.	113
Figura 86.- Maniobra de control de la bomba	114
Figura 87.- Activación señal de alarma.	114
Figura 88.- Aspecto del bloque de función “Deposito_tiempo_averia”.	114
Figura 89.- Esquema de un tipo de depósito a automatizar.	116
Figura 90.- Declaración de variables de la función “Deposito_2sensores”.	117
Figura 91.- Tarea de control de la bomba	117
Figura 92.- Activación de la señal de alarma.	118
Figura 93.- Aspecto del bloque de función “deposito_2sensores”.	118
Figura 94.- Esquema del funcionamiento del depósito.	120
Figura 95.- Declaración de variables de “Control_abastecimiento_2_consumos”.	120
Figura 96.- Puesta en marcha de la bomba.	121
Figura 97.- Control de la válvula de la salida B.	121
Figura 98.- Activación de la alarma.	122
Figura 99.- Aspecto bloque de función “Control_abastecimiento_2_consumos”.	123
Figura 100.- Detalle del sistema a controlar.	125
Figura 101.- Declaración de variables de la función “inundación_garaje”.	126
Figura 102.- Puesta en marcha de la bomba.	126
Figura 103.- Indicación de presencia de agua en el garaje.	127
Figura 104.- Indicación de presencia de agua en el garaje.	127
Figura 105.- Activación alarma por posibles averías.	128

Figura 106.- Aspecto del bloque de función “inundación_garaje”. _____	128
Figura 107.- Detalle del sistema a controlar. _____	129
Figura 108.- Declaración de variables de “achique_galeria_servicio”. _____	130
Figura 109.- Programación en ST de “achique_galerias_servicio”. _____	131
Figura 110.- Declaración de variables de “achique_galerias_servicio”. _____	131
Figura 111.- Conexión/desconexión de la bomba 1. _____	132
Figura 112.- Temporizador. _____	132
Figura 113.- Conexión/desconexión de la bomba 2. _____	133
Figura 114.- Aspecto del bloque de función “achique_galerias_servicio”. _____	133
Figura 115.- Esquema del parking a automatizar. _____	135
Figura 116.- Declaración de variables del “Parking_sensor_vigilante”. _____	137
Figura 117.- Programación del contador de control del parking. _____	138
Figura 118.- Activación piloto y/o cartel de entrada libre. _____	138
Figura 119.- Aspecto del bloque de función “Parking_sensor_vigilante”. _____	139
Figura 120.- Esquema del parking a automatizar, modelo 2 _____	140
Figura 121.- Declaración de variables de “Parking_2_sensores”. _____	141
Figura 122.- Cronograma relativo al movimiento del vehículo entre sensores”. _____	142
Figura 123.- Detección de entrada de un coche al parking _____	143
Figura 124.- Detección salida de un coche. _____	143
Figura 125.- Contador/descontador de vehículos. _____	144
Figura 126.- Señal de acceso libre al parking. _____	145
Figura 127.- Aspecto del bloque de función “Parking_2_sensores” _____	145
Figura 128.- Ubicación de la barrera. _____	147
Figura 129.- Declaración de variables de “Parking_2_sensores_barrera”. _____	149
Figura 130.- Detección entrada de coche. _____	149
Figura 131.- Salida de un coche del parking. _____	150
Figura 132.- Control de barreras. _____	151
Figura 133.- Aspecto del bloque de función “Parkign_2_sensores_barrera” _____	152

Figura 134.- Aspecto del esquema de funcionamiento del parking de 2 plantas. _____	153
Figura 135.- Declaración de variables de “Parking_2 plantas”. _____	155
Figura 136.- Control de entrada al parking. _____	156
Figura 137.-.- Control salida planta inferior. _____	156
Figura 138.- Control salida planta superior. _____	157
Figura 139.- Contador de vehículos de la planta inferior. _____	157
Figura 140.- Contador de vehículos del parking superior. _____	158
Figura 141.- Indicación de que se debe dirigir al parking de la planta superior. _____	158
Figura 142.- Indicación de activación de parking completo. _____	159
Figura 143.- Acceso libre al parking. _____	159
Figura 144.- Aspecto del bloque de función “Parking_2 plantas. _____	159
Figura 145.- Esquema básico de una máquina taladradora. _____	163
Figura 146.- Declaración de variables de la función “Taladro” _____	164
Figura 147.- Accionar el movimiento giratorio el taladro. _____	165
Figura 148.- Control de la bajada del taladro. _____	165
Figura 149.- Tiempo de taladro bajo. _____	166
Figura 150.- Movimiento de subida. _____	166
Figura 151.- Aspecto del bloque de función “Taladro”. _____	167
Figura 152.- Detalle de la máquina a controlar. _____	169
Figura 153.- Declaración de variables de “Taladro_repeticion_ciclos”. ____	170
Figura 154.- Puesta en marcha y paro de los diferentes ciclos de trabajo. ____	171
Figura 155.- Control de marcha del giro del taladro.. _____	171
Figura 156.- Control del descenso del taladro. _____	172
Figura 157.-Tiempo de espera con la taladradora bajo. _____	172
Figura 158.- Maniobra de ascenso. _____	173
Figura 159.- Contador de ciclos. _____	173
Figura 160.- Aspecto del bloque de función “Taladro_repeticion_ciclos”. __	174

Figura 161.- Detalle de la máquina a controlar. _____	175
Figura 162.- Declaración de variables “Taladro_repeticion_ciclo_alarma”. _____	176
Figura 163.- Control tiempo de subida y bajada del taladro. _____	177
Figura 164.- Activación de la señal de alarma. _____	177
Figura 165.- Aspecto del bloque de función “Taladros_repeticion_ciclos_alarma”.. _____	178
Figura 166.- Bucle básico de control en lazo abierto de control. _____	181
Figura 167.- Bucle básico de control en lazo cerrado de control. _____	182
Figura 168.- Bucle básico de control en lazo cerrado de control. _____	183
Figura 169.- Posibles repuesta de un regulador proporcional. _____	184
Figura 170.- Declaración de variables de función “Proporcional”. _____	185
Figura 171.- Programación de la terea de la función “Proporcional”. _____	185
Figura 172.- Esquema de la función “Proporcional”. _____	186
Figura 173.- Posibles repuesta de un regulador integral. _____	188
Figura 174.- Variables de un regulador “Integral”. _____	189
Figura 175.- Programa de un regulador de tipo “Integral”. _____	189
Figura 176.- Esquema de la función “Integral”. _____	190
Figura 177.- Posibles repuesta de un regulador derivativo. _____	192
Figura 178.- Programa en CX_Programer del regulador derivativo. _____	193
Figura 179.- Función “D” e implementación en CX-Programmer _____	194
Figura 180.- Variables de la función “PI” e implementación en Codesys. _____	196
Figura 181.- Función “PI” e implementación en Codesys. _____	196
Figura 182.- Función “PI” estructura externa. _____	197
Figura 183.- Variables de la “PD”. _____	198
Figura 184.- Programa de las tareas de la función “PD”. _____	199
Figura 185.- Estructura de la función “PD”. _____	199
Figura 186.- Variables de la función “PID” en Codesys. _____	201
Figura 187.- Variables de la función “PID” en Codesys. _____	202
Figura 188.- Aspecto de la función “PID” en Codesys. _____	202

Figura 189.- Diseño de la función “PID” en CX-Programmer.	_____	205
Figura 190.- Diseño de una aplicación de la función “PID” en CX-Programmer.	_____	206

Prólogo.

La automatización industrial comprende toda la actividad empresarial del control de una planta o proceso; desde el simple hecho de una maniobra como pudiera ser un estrella/triangulo hasta la automatización integral de toda una planta o proceso, dónde puede no ser necesaria la intervención del ser humano.

Las automatizaciones se basan en el uso de los autómatas programables como controladores tanto de los subprocesos que componen el proceso, de los propios procesos o incluso como supervisores de los mismos; es por eso que cada día se hace más necesario el conocimiento de programación de autómatas programables por parte de sus usuarios.

El presente libro pretende dar una visión de los nuevos lenguajes de programación y nuevas metodologías, así como de las ventajas que presentan frente a métodos tradicionales.

El libro está orientado a la resolución de ejemplos de programación de complejidad creciente dentro del ámbito industrial. Los problemas que se plantean se resuelven utilizando los lenguajes de programación de dos fabricantes distintos, que cumplen con la Norma IEC 61131-3. Ello permite al lector además de familiarizarse con los equipos de esos fabricantes, entender con facilidad los programas basados en la norma citada y sus entornos de programación. También se proponen una serie de ejemplos para que los lectores puedan practicar por su cuenta.

Por último, indicar que con el presente libro no se pretende enseñar a programar (se supone que el lector ya tiene conocimientos de algún lenguaje de programación de autómatas) si no a orientar al lector en una filosofía de cómo abordar los diferentes problemas técnicos más comunes en la industria.

Finalmente, resaltar que la relación de los autores se han ordenado alfabéticamente, pues el libro es el resultado de la colaboración de todos ellos.

1. Objetivo.

El principal objetivo que se plantea con la edición de este libro es acercar al lector a las nuevas reglas y/o normas que se tienen que adoptar desde el punto de vista del diseño y la realización de programas de automatización, para autómatas programables, de tal manera que el lector pueda adquirir la experiencia necesaria para la implantación de las Normas (IEC-61131-3) que en la actualidad rigen para las maniobras realizadas con esos equipos.

Con la implantación de esta nueva Norma (IEC-61131-3) se pretende dar un nuevo avance en la realización de los programas en las automatizaciones industriales mediante autómatas programables. En la actualidad, en el mundo industrial, hay diferentes fabricantes de autómatas. Además cada fabricante ha desarrollado a lo largo del tiempo una variedad de equipos en función de las necesidades que plantean la infinidad de aplicaciones industriales que surgen. En paralelo, cada fabricante ha desarrollado un entorno de programación, con diferentes programas, incluso para sus propios productos.

El resultado de lo anterior es que una misma maniobra ejecutada en equipos distintos o con lenguajes diferentes necesita ser programada de una forma diferente, lo que exige que los programadores necesiten una preparación específica. Por tanto, ante tanta diversidad, se ha desarrollado una Norma específica (IEC-61131-3), mediante la cual se pretende unificar, en lo posible, la forma y técnicas de programación de autómatas programables, independientemente del fabricante y tipo de autómata programable.

Desde los primeros años de la década del 1990-2000, se puede obtener información sobre la evolución de la citada norma, pero aún hoy en día, gran parte de las reglas contenidas en esta normativa, no son cumplidas por la gran mayoría de los equipos, al menos al 100%. Ahora bien, hay que reconocer que los equipos más recientes, ya empiezan a incorporar de alguna manera gran parte de las reglas que recoge dicha norma. Además de la norma citada, que trata de homogeneizar los lenguajes y forma de programación (es decir para el Software). Hay (o han aparecido) otras normas similares aplicadas a los equipos (hardware).

El análisis pormenorizado de esta norma no es el objetivo de este texto, pues existe bibliografía dedicada para tal fin, como por ejemplo, el libro “IEC 61131-3: Programming Industrial Automation Systems”. (Véase las referencias en bibliografía).

También hay que destacar que el estudio, diseño, planteamiento y resolución de maniobras básicas con autómatas programables, tampoco está entre los objetivos de este libro, el lector puede recurrir a la bibliografía, entre otros textos al libro “***Automatización de maniobras industriales mediante autómatas programables***” de los autores Juan Pérez Cruz y Manuel Pineda Sánchez.

Para seguir leyendo haga click aquí