



Miguel Ángel Ridao Carlini

# INTRODUCCIÓN A LA PROGRAMACIÓN DE AUTÓMATAS PROGRAMABLES USANDO CoDeSys

Página de créditos

Índice

Biografía

Reseña

Editorial Universidad de Sevilla

Escuela Técnica Superior de Ingeniería



# INTRODUCCIÓN A LA PROGRAMACIÓN DE AUTÓMATAS PROGRAMABLES USANDO CoDeSys



Miguel Ángel Ridaó Carlini

# INTRODUCCIÓN A LA PROGRAMACIÓN DE AUTÓMATAS PROGRAMABLES USANDO CoDeSys



SEVILA 2018

Serie: Ingeniería  
Colección: Monografías de la Escuela Técnica Superior de Ingeniería  
de la Universidad de Sevilla

Comité editorial:

José Beltrán Fortes  
(Director de la Editorial Universidad de Sevilla)  
Araceli López Serena  
(Subdirectora)  
Concepción Barrero Rodríguez  
Rafael Fernández Chacón  
María Gracia García Martín  
Ana Ilundáin Larrañeta  
María del Pópulo Pablo-Romero Gil-Delgado  
Manuel Padilla Cruz  
Marta Palenque Sánchez  
José-Leonardo Ruiz Sánchez  
Antonio Tejedor Cabrera

Reservados todos los derechos. Ni la totalidad ni parte de este libro puede reproducirse o transmitirse por ningún procedimiento electrónico o mecánico, incluyendo fotocopia, grabación magnética o cualquier almacenamiento de información y sistema de recuperación, sin permiso escrito de la Editorial Universidad de Sevilla.

Motivos de cubierta:

Colector Fresnel en la Escuela Técnica Superior de Ingeniería de la Universidad de Sevilla.  
Foto: María Robledo

Edición digital de la primera edición impresa de 2016

- © Editorial Universidad de Sevilla 2018  
C/. Porvenir, 27 - 41013 Sevilla.  
Tlfs.: 954 487 447; 954 487 451; Fax: 954 487 443  
Correo electrónico: eus4@us.es  
Web: <<http://www.editorial.us.es>>
- © Escuela Técnica Superior de Ingeniería de Sevilla 2018
- © Miguel Ángel Ridaó Carlini 2018
- ISBNe: 978-84-472-2135-6  
Digitalización y realización interactiva: ed-Libros

*A Carmen, Carlos y Miguel*

*A mis padres*



## Prólogo

---

Los lenguajes de programación de los autómatas programables, tradicionalmente, han sido dependientes del fabricante, cada uno de ellos diseñaba sus propios lenguajes para sus productos. También es cierto que entre los lenguajes de los distintos fabricantes hay similitudes, por ejemplo, la práctica totalidad proponen un lenguaje de contactos y un lenguaje tipo ensamblador, aunque esto no impide que un programador que cambie de *hardware* deba familiarizarse tanto con un nuevo entorno de programación como con una nueva sintaxis en los lenguajes.

Un primer paso hacia la estandarización, no solo de los lenguajes sino de todos los elementos que intervienen en la automatización con autómatas programables, ha sido la aparición del estándar internacional IEC-61131. Esta norma sigue en constante evolución y está siendo adoptada por un gran número de fabricantes, incluidas las empresas líderes del sector, que en algunos casos mantienen sus lenguajes propios (debido al alto grado de utilización dentro del sector) y añaden nuevos productos siguiendo el estándar.

El objetivo de este libro es dar al lector una introducción a la programación de autómatas programables en el estándar IEC-61131 desde un punto de vista práctico. Uno de los entornos de programación que más fielmente siguen la citada norma es CoDeSys, un *software* desarrollado por la compañía 3S-Smart Software Solutions GmbH, empresa independiente de fabricantes de *hardware* y que proporciona un entorno de programación y *drivers* para la conexión con un cada vez más importante número de productos de distintos fabricantes. En este entorno está enfocada la programación de autómatas programables en este libro.

Este libro está dirigido a las personas que, sin experiencia previa en Automatización, quieren tomar un primer contacto con la programación de autómatas programables. En cualquier caso no se trata de un libro de texto que abarque la automatización industrial, sino que debe ser usado como un complemento enfocado en la programación de autómatas dentro de un curso básico de automatización. En concreto, en la Escuela Superior de Ingeniería de Sevilla, se está utilizando en las asignaturas de

introducción a la automatización para los alumnos de ingeniería en paralelo con las clases teóricas y prácticas de laboratorio.

Una de las facilidades que ofrece CoDeSys es la posibilidad de ejecutar programas en modo simulación con un entorno gráfico que permite la simulación de procesos. Esto hace que para utilizar y programar los ejemplos de este libro no se requiera de un autómata programable real, sino que se proporciona en los distintos ejemplos el proceso simulado, con indicación de las entradas y salidas. Aunque esto suponga una facilidad para el aprendizaje, no quiero dejar de insistir en la necesidad de complementar el aprendizaje siguiendo este libro con la utilización por parte del lector de dispositivos reales.

Para el seguimiento del libro no es imprescindible aunque sí recomendable tener ciertos conocimientos sobre automatización: sensores y actuadores, modo de funcionamiento de un autómata, etc. En cualquier caso, los conocimientos básicos necesarios para el seguimiento del libro han sido resumidos en el Capítulo 1. Así mismo, también se le presuponen al lector unos mínimos conocimientos de programación convencional, por ejemplo en lenguaje C o similar.

Entiendo que la única forma de aprender a programar consiste en programar por uno mismo, y en esta línea está diseñado este libro. Junto al libro se proporcionan 75 ejemplos y ejercicios ejecutables en CoDeSys, algunos resueltos y otros propuestos. La filosofía de trabajo de este libro consiste en su estudio junto a un ordenador en el que se pueda instalar CoDeSys (en la web del desarrollador <<https://www.codesys.com>> en la sección *Download* se puede descargar la versión 2.3 gratuita que funciona en modo simulación), ejecutar los ejemplos y resolver los ejercicios a medida que se avanza en el contenido del libro. De hecho, en muy pocos casos he incluido el código de los programas dentro del libro, sino que se hace referencia a los ejemplos de CoDeSys que acompañan a este libro y que se pueden descargar de la web <<http://disa.us.es/LibroCodesys/Ejercicios.rar>>.

Como se ha indicado anteriormente, el libro comienza con una breve introducción a conceptos básicos de automatización necesarios para el seguimiento del libro. Los Capítulos 2 y 3 se dedican respectivamente a una introducción a la norma IEC-61131 y al entorno de programación en CoDeSys, con objeto que desde el primer momento el lector pueda realizar ejercicios básicos de programación. Los Capítulos 4 y 5 están dedicados a los aspectos básicos generales de la norma: las variables, tipos de datos y las unidades de programación, denominadas POU.

Los Capítulos 6, 7, 9, 10 y 11 se dedican a los cinco lenguajes que define la norma IEC-61131-3, incluyendo tanto la sintaxis como la utilización de los editores en el entorno CoDeSys. Entre estos capítulos se intercala el Capítulo 8 dedicado a dos de los elementos más habitualmente utilizados en cualquier programa de control: los contadores y los temporizadores. La idea de incluirlo entre los capítulos de los lenguajes de programación viene dada, en primer lugar, porque para entender su uso es necesario conocer algún lenguaje de programación previamente. En segundo lugar, el poder utilizar contadores y temporizadores da una mayor riqueza a los ejemplos propuestos para explicar el resto de los lenguajes de programación.

Todos los ejemplos que se proponen hasta el Capítulo 11 están dedicados al control de automatismos lógicos, es decir, las variables que se controlan son de tipo booleano. Sin embargo, hoy en día un autómata programable se entiende como un dispositivo capaz de tratar y controlar también señales de tipo analógico. En este sentido, el Capítulo 12 es una introducción a la utilización de este tipo de señales, incluyendo el diseño de algunos controladores básicos.

Finalmente, el capítulo 13 propone una serie de problemas de una dificultad un tanto superior a los ejercicios propuestos en los capítulos anteriores.

*Miguel Ángel Ridao Carlini*  
miguelridao@us.es



# Índice

---

<i>Prólogo</i>	IX
<i>Índice</i>	XIII
<i>Índice de Figuras</i>	XXI
<i>Índice de Tablas</i>	XXV
<b>1 Autómatas Programables</b>	<b>1</b>
1.1 Automatización y Control	1
1.2 Sensores y actuadores	3
1.2.1 Sensores y otras entradas al sistema de control	4
1.2.2 Actuadores y otras salidas del sistema de control	6
1.3 Implementación de Automatismos Lógicos	7
1.4 Definición de autómatas programables	8
1.5 Estructura externa de un autómata programable	9
1.6 Estructura interna de un autómata programable	10
1.6.1 Memoria	11
1.6.2 Módulos de Entrada/Salida	13
1.6.3 Entornos de programación	14
1.7 Funcionamiento de un autómata programable	15
1.7.1 Ciclo del autómata programable	15
1.7.2 Modos de funcionamiento del autómata programable	17
<b>2 Introducción a la Norma IEC-61131</b>	<b>21</b>
2.1 Elementos básicos en la norma IEC-61131-3	22
2.2 Program Organization Units (POU)	23
2.3 Variables	24
2.4 Lenguajes de programación	25



<b>3</b>	<b>Primeros pasos con CoDeSys</b>	<b>27</b>
3.1	Algunos conceptos básicos en CoDeSys	27
3.2	Descarga del software CoDeSys	28
3.3	Ejecución de CoDeSys	28
3.4	Pantalla principal	30
3.5	Primer programa en lógica de contactos	31
3.6	Ejemplo 2. Puesta en marcha y parada de un motor	34
3.7	Ejemplo 3. Cambio del sentido de giro de un motor	37
3.8	Ejemplo 4. Control de un cilindro neumático (I)	39
3.9	Ejemplo 5. Control de un cilindro neumático (II)	41
3.10	Ejemplo 6. Control del nivel de un depósito	42
<b>4</b>	<b>Tipos de Datos y Variables</b>	<b>43</b>
4.1	Tipos de datos	43
4.1.1	Tipos de datos elementales	44
4.1.2	Representación de datos	44
4.1.3	Tipos de datos derivados	45
4.1.4	Tipos de datos genéricos	46
4.2.	Variables	47
4.2.1	Atributos	48
4.2.2	Variables de Entrada/Salida	48
4.2.3	Valores iniciales	50
4.2.4	Matrices y estructuras	50
4.3	Definición de tipos de datos derivados en CoDeSys	52
4.4	Declaración de variables en CoDeSys	53
<b>5</b>	<b>Unidades de Organización de Programas (POU)</b>	<b>55</b>
5.1	Tipos de POU's	55
5.2	Bloques funcionales	57
5.2.1	Estructura de datos asociada a un bloque funcional	59
5.2.2	Llamada al bloque funcional	59
5.2.3	Restricciones en el uso de bloques funcionales	61
5.2.4	Monitorización de bloques funcionales en CoDeSys	61
5.3	Funciones	62
5.3.1	Diferencia en el funcionamiento entre funciones y bloques funcionales	63
5.3.2	Restricciones en el uso de funciones	64
5.4	Programas	64
5.5	Creación de POU's en CoDeSys	64
5.6	Funciones y bloques funcionales estándar	65
5.6.1	Algunas funciones estándar	65
5.6.2	Algunos bloques funcionales estándar	69
5.7	Ejercicios	69
5.7.1	Bloque funcional para Marcha-Parada de un motor	70
5.7.2	Bloque funcional para contar flancos de subida	70
5.7.3	Bloque funcional de detección de flancos	71

<b>6 Programación en LD</b>	<b>73</b>
6.1 Contactos y bobinas	74
6.1.1 Ejemplos	76
6.2 Bloques funcionales	77
6.2.1 Ejemplos	77
6.3 Entrada EN en funciones y bloques funcionales	78
6.3.1 Ejemplos	79
6.4 Saltos	81
6.4.1 Ejemplo	81
6.5 Editor de LD en CoDeSys	82
6.5.1 Añadir nuevas redes	83
6.5.2 Edición de contactos y bobinas	83
6.5.3 Edición de bloques funcionales	84
6.5.4 Edición de saltos en CoDeSys	85
6.6 Ejemplo	85
6.7 Ejercicios	87
6.7.1 Ejercicio 1	87
6.7.2 Ejercicio 2	87
6.7.3 Ejercicio 3	89
<b>7 Programación en IL</b>	<b>91</b>
7.1 Operadores básicos y llamadas a funciones	93
7.2 Uso de funciones y bloques funcionales	95
7.2.1 Llamada a funciones	95
7.2.2 Llamada a bloques funcionales	97
7.3 Saltos	99
7.4 Ejercicios	100
7.4.1 Ejercicio 1	100
7.4.2 Ejercicio 2	100
7.4.3 Ejercicio 3	101
<b>8 Temporizadores y Contadores</b>	<b>103</b>
8.1 Temporizadores	103
8.1.1 Temporizador de conexión (TON)	104
8.1.2 Temporizador de Tiempo Fijo (TP)	105
8.1.3 Temporizador de desconexión (TOF)	105
8.1.4 Uso de los temporizadores	106
8.2 Contadores	107
8.2.1 Contador de incremento (CTU)	107
8.2.2 Contador de decremento (CTD)	108
8.2.3 Contador de incremento-decremento (CTUD)	109
8.2.4 Uso de los contadores	110
8.3 Ejercicios	110
8.3.1 Ejercicio 1	110

8.3.2	Ejercicio 2	111
8.3.3	Ejercicio 3	111
8.3.4	Ejercicio 4	111
8.3.5	Ejercicio 5	111
8.3.6	Ejercicio 6	113
8.3.7	Ejercicio 7	113
8.3.8	Ejercicio 8	115
<b>9</b>	<b>Programación en ST</b>	<b>117</b>
9.1	Instrucciones en ST	117
9.1.1	Asignación	118
9.1.2	Sentencia IF	118
9.1.3	Sentencia CASE	118
9.1.4	Sentencia FOR	118
9.1.5	Sentencia WHILE	119
9.1.6	Sentencia REPEAT	119
9.1.7	Sentencia EXIT	119
9.2	Uso de funciones y bloques funcionales	120
9.2.1	Llamadas a funciones	120
9.2.2	Llamada a bloques funcionales	120
9.2.3	Sentencia RETURN	121
9.3	Ejercicios	122
9.3.1	Ejercicio 1	122
9.3.2	Ejercicio 2	123
9.3.3	Ejercicio 3	123
<b>10</b>	<b>Programación en FBD</b>	<b>125</b>
10.1	Cómo trabajar con bloques en CoDeSys	125
10.1.1	Posición del cursor gráfico	126
10.1.2	Comandos en FBD	127
10.2	Lenguaje CFC	130
10.3	Ejercicios	130
10.3.1	Ejercicio 1	130
10.3.2	Ejercicio 2	132
10.3.3	Ejercicio 3	133
<b>11</b>	<b>Programación en SFC</b>	<b>137</b>
11.1	Introducción a GRAFCET	138
11.1.1	Etapas	138
11.1.2	Transiciones	139
11.1.3	Arcos orientados	140
11.1.4	Reglas de evolución	140
11.2	Estructuras básicas en GRAFCET	143
11.2.1	Secuencia única	143
11.2.2	Secuencia simultánea	144

11.2.3	Selección de secuencia	145
11.2.4	Salto de etapa	146
11.2.5	Repetición de secuencia	147
11.3	Acciones en GRAFCET	147
11.4	GRAFCET en la norma IEC-61131-3	148
11.4.1	Etapas	148
11.4.2	Transiciones	149
11.4.3	Acciones	150
11.5	GRAFCET en CoDeSys	156
11.5.1	Edición de un GRAFCET	156
11.5.2	Comandos básicos para edición del GRAFCET	157
11.5.3	Edición de transiciones	160
11.5.4	Edición de acciones	161
11.6	Ejercicios	162
11.6.1	Ejercicio 1	162
11.6.2	Ejercicio 2	163
11.6.3	Ejercicio 3	163
11.6.4	Ejercicio 4	164
11.6.5	Ejercicio 5	164
11.6.6	Ejercicio 6	164
11.6.7	Ejercicio 7	165
11.6.8	Ejercicio 8	165
11.6.9	Ejercicio 9	166
11.6.10	Ejercicio 10	167
11.6.11	Ejercicio 11	168
11.6.12	Ejercicio 12	168
<b>12</b>	<b>Usando Señales Analógicas</b>	<b>171</b>
12.1	Señales analógicas	171
12.1.1	Entradas analógicas: del proceso al autómata	172
12.1.2	Salidas analógicas: del autómata al proceso	173
12.2	Tratamiento de señales analógicas en el PLC	174
12.2.1	Cambio de tipo de variable	174
12.2.2	Escalado de señales analógicas	175
12.2.3	Ejemplo	176
12.3	Control de sistemas continuos en un PLC	178
12.3.1	Controlador P	179
12.3.2	Controlador PI	180
12.3.3	Controlador PID	185
12.4	Bloques funcionales para el control de procesos	186
12.4.1	Generador de salida PWM	186
12.4.2	Control de una válvula motorizada	187
12.5	Ejercicios	190

12.5.1	Ejercicio 1	190
12.5.2	Ejercicio 2	192
12.5.3	Ejercicio 3	192
<b>13</b>	<b>Problemas</b>	<b>195</b>
13.1	Cruce con tranvía	195
13.2	Cruce completo con tranvía	196
13.3	Automatización con modos de marcha y parada	197
13.3.1	Proceso a controlar	197
13.3.2	Modos de funcionamiento	199
13.3.3	Desarrollo del ejercicio	201
13.3.4	Lista de señales	203
13.4	Control de un ascensor	204
13.4.1	Lista de señales	204
13.5	Control de dos ascensores simples	206
13.5.1	Lista de señales	207
13.6	Control de dos ascensores completos	208
13.6.1	Lista de señales	209
13.7	Control de una mezcladora	209
13.7.1	Lista de señales	212
<b>Apéndice A.</b>	<b>Configuración de CoDeSys</b>	<b>213</b>
A.1	Gestión de librerías	213
A.2	Configuración de tareas	214
<b>Apéndice B.</b>	<b>Visualizaciones en CoDeSys</b>	<b>217</b>
B.1	Editor de Visualizaciones	217
B.2	Configuración básica de elementos	218
B.2.1	Text	220
B.2.2	Text variables	220
B.2.3	Line width	221
B.2.4	Color	221
B.2.5	Colorvariables	222
B.2.6	Motion absolute	222
B.2.7	Motion relative	222
B.2.8	Variables	222
B.2.9	Input	223
B.2.10	Text for tooltip	223
B.3	Configuración avanzada de elementos	223
B.3.1	Table	223
B.3.2	Scrollbar	224

B.3.3	Indicador de aguja	224
B.3.4	Diagrama de barras	224
B.3.5	Inclusión de archivos gráficos	224
B.3.6	Graficas de variables frente al tiempo	225

<i>Referencias</i>	227
--------------------	-----

<i>Índice de Conceptos</i>	229
----------------------------	-----



**MIGUEL ÁNGEL RIDAO CARLINI** es Ingeniero Industrial y Doctor Ingeniero Industrial por la Universidad de Sevilla el año 1995. Profesor en la Escuela Técnica Superior de Ingeniería de Sevilla, tiene una amplia experiencia en la docencia de asignaturas relacionadas con el control de sistemas, tales como “Automatización Industrial”, “Robótica y Automatización” y “Fundamentos de Control”. Asimismo ha participado en numerosos proyectos con empresas en el campo de la automatización.



Los autómatas programables juegan un cada vez más importante papel en la automatización de procesos en la industria. La norma IEC 61131 ha supuesto un importante paso en la estandarización de este tipo de dispositivos y muy especialmente en lo referente a los lenguajes de programación en los que desarrollar los programas de control.

El objetivo de este libro es proporcionar al lector una introducción a la programación de autómatas programables siguiendo el estándar de la norma IEC-61131-3, utilizando para ello el software de desarrollo de aplicaciones para autómatas programables CodeSys. La obra está dirigida a personas que, sin experiencia previa en automatización, quieran tener un primer contacto con la programación de autómatas programables. Con un enfoque práctico, a medida que el lector progresa en el libro podrá ir desarrollando sus propios programas de control y comprobando su funcionamiento sobre sistemas simulados gracias a los más de 70 ejercicios y ejemplos ejecutables en CoDeSys que pueden ser descargados desde la página web del libro.

<<http://disa.us.es/disa/index.php/es/ensenanza-universitaria/publicaciones-docentes>>

