

CONTROLADOR LÓGICO PROGRAMABLE. GUÍA DE
PROGRAMACIÓN Y EJEMPLOS

NATALIA ANDREA FLÓREZ SANTANA

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y
TELECOMUNICACIONES
BUCARAMANGA
2020

CONTROLADOR LÓGICO PROGRAMABLE. GUÍA DE
PROGRAMACIÓN Y EJEMPLOS

NATALIA ANDREA FLÓREZ SANTANA

Trabajo de grado para optar al título de Ingeniera Electricista

Director

JULIO AUGUSTO GÉLVEZ FIGUEREDO

Magíster en Potencia Eléctrica

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y
TELECOMUNICACIONES
BUCARAMANGA
2020

AGRADECIMIENTOS

Agradezco a Dios por su bondad y fidelidad, por permitirme culminar esta carrera.

A la Universidad Industrial de Santander por darme la oportunidad de materializar mis sueños.

Al profesor Julio Gélvez y su familia, por la confianza depositada en mí para realizar este proyecto.

Al Ingeniero Rexnik Galeano por creer en mi trabajo para iniciar mi vida profesional en RYCTEL SAS.

DEDICATORIA

Dedico este proyecto a Dios, porque todo en mi vida se trata de Él.

A mi madre que con mucha valentía y amor afrontó lo que Dios le dispuso a lo largo de la vida con el objetivo de apoyarme para cumplir mis metas.

A mi tía Gilma, que con sus sabios consejos y su amor constante me enseñó que todo es posible para el que confía en Dios. Por estar pendiente de todo lo que me sucedía, por animarme a continuar y recordarme que lo importante siempre es terminar.

A mi hermana Paola, por su paciencia en esta época de estudio y por sentirse orgullosa de mis logros.

A mi primo Alex por su disposición de ayudar siempre cada vez que requería de algo.

A mi novio Fernan, por compartir su amor conmigo, por apoyarme para terminar este proyecto y compartir el sueño de seguir de la mano juntos.

A mi iglesia Nuevo Amanecer y a CRU, por permitirme servir y hablar de Jesús a otros.

A mis compañeros, con quienes compartimos tiempo, estudio y amistad.

CONTENIDO

	pág.
INTRODUCCIÓN	16
1. SOFTWARE DE PROGRAMACIÓN TWIDOSUITE	19
1.1. REQUISITOS DEL SISTEMA	19
1.2. CREACIÓN DE UN PROGRAMA	20
1.3. DESCRIBIR PROGRAMA	20
1.4. PROGRAMAR	22
1.4.1. Configurar.	22
1.4.1.1. Configurar el material (Hardware).	22
1.4.1.2. Configurar los datos (Software).	24
1.4.1.3. Configurar el comportamiento.	26
1.4.1.4. Definir las protecciones.	27
1.4.2. Programar.	28
1.4.2.1. Editar programa.	28
1.4.2.2. Definir símbolos.	28
1.4.3. Depurar el programa.	28
1.5. DOCUMENTAR	31
2. FUNCIONES DEL SOFTWARE TWIDO SUITE. EJEMPLOS EXPLICATIVOS	32
2.1. ENUNCIADO DE LOS EJEMPLOS	32
2.1.1. Control de acceso a parqueadero.	32
2.1.1.1. Características del control	32
2.1.1.2. Circuito de potencia y control del ejemplo control de acceso a parqueadero.	34
2.1.2. Control secuencial de tres motores.	35
2.1.2.1. Características del control	35
2.1.2.2. Circuito de potencia y control del ejemplo arranque de tres motores en cascada	37
2.1.3. Empacadora de productos.	38

2.1.3.1.	Características del control	39
2.1.3.2.	Circuito de potencia y control del ejemplo empacadora de productos.	41
2.2.	BLOQUES FUNCIONALES	42
2.2.1.	Contador progresivo/regresivo (%Ci).	42
2.2.2.	Temporizador (%TMi).	43
2.2.2.1.	Solución de los ejemplos empleando la función contador y temporizador.	45
2.2.3.	Registro de bits de desplazamiento (%SBR).	48
2.2.3.1.	Solución de los ejemplos empleando la función registro de desplazamiento.	50
2.2.4.	GRAFCET.	54
2.2.4.1.	Diagrama GRAFCET.	54
2.2.4.2.	Solución de los ejemplos empleando el diagrama GRAFCET.	56
2.2.5.	DRUM (%DRi).	60
2.2.5.1.	Solución de los ejemplos empleando la función DRUM.	61
2.2.6.	Contador de pasos.	64
2.2.6.1.	Solución del ejemplo arranque estrella/triángulo de tres motores que se energizan en cascada M3, M2, M1.	64
2.2.7.	FIFO (%Ri)	69
2.2.7.1.	Solución del ejemplo empacadora de productos.	69
2.2.8.	Palabras de memoria y bloques de operación.	72
2.2.8.1.	Palabras de memoria.	72
2.2.8.2.	Bloques de operación.	72
3.	COMUNICACIÓN MODBUS ENTRE EL PLC TWIDO Y EL VARIADOR DE VELOCIDAD ALTIVAR PARA EL CONTROL DE VELOCIDAD DE UN MOTOR TRIFÁSICO	75
3.1.	EQUIPOS EMPLEADOS PARA LA COMUNICACIÓN	75
3.1.1.	PLC Twido TWDLCAE40DRF.	75
3.1.2.	Motor de inducción trifásico.	75
3.1.3.	Variador de velocidad ALTIVAR 31.	76
3.1.4.	Equipo de cómputo (PC).	76
3.2.	CONEXIÓN ENTRE EL PLC TWIDO, VARIADOR DE VELOCIDAD ALTIVAR 31 Y EQUIPO DE CÓMPUTO	76

3.2.1.	Conexión física entre el PLC Twido y el equipo de cómputo.	76
3.2.2.	Conexión física entre el variador y el PLC Twido:	77
3.2.3.	Conexión física entre el variador de velocidad y el motor de inducción trifásico.	77
3.3.	CONFIGURACIÓN DEL VARIADOR DE VELOCIDAD ALTIVAR 31 . .	78
3.3.1.	Menú de control del motor (drC).	79
3.3.2.	Menú de entradas/salidas I-O.	82
3.3.3.	Menú de control Ctrl.	82
3.4.	CONFIGURACIÓN MODBUS EN EL PLC TWIDO TWDLCAE40DRF .	83
3.5.	PROGRAMACIÓN DE LA COMUNICACIÓN MODBUS	86
3.5.1.	Macros DRIVE.	86
3.5.2.	Macros COM.	88
3.6.	PROGRAMA PARA EL CONTROL DE VELOCIDAD DE UN MOTOR DE INDUCCIÓN EMPLEANDO COMUNICACIÓN MODBUS.	90
4.	CONTROL DE TEMPERATURA DE UN HORNO ELÉCTRICO USANDO LA FUNCIÓN PID DEL SOFTWARE TWIDO	92
4.1.	CONTROL PROPORCIONAL, INTEGRAL Y DERIVATIVO	92
4.2.	ANÁLISIS TEÓRICO	93
4.3.	IMPLEMENTACIÓN DEL SISTEMA DE CONTROL	95
4.3.1.	Configuración del módulo analógico.	95
4.3.2.	Regulación continua PID.	97
4.4.	CONFIGURACIÓN DE LA FUNCIÓN PID	98
4.5.	PROGRAMACIÓN CONTROL PID HORNO	102
4.6.	MONTAJE FÍSICO DEL SISTEMA DE CONTROL	103
4.7.	RESULTADOS OBTENIDOS	104
5.	OBSERVACIONES	107
6.	CONCLUSIONES	109

LISTA DE FIGURAS

	pág.
Figura 1. Ventana de la pestaña <i>Proyecto</i>	20
Figura 2. Ventana de la pestaña <i>describir</i>	21
Figura 3. Configuración de la dirección IP en el PLC	22
Figura 4. Ventana de la pestaña <i>Programar, Configurar</i>	23
Figura 5. Pestaña “Diagrama de cableado para la base TWDLCAE40DRF” . .	23
Figura 6. Ventana de la pestaña <i>Programar, Configurar, Configurar el compor-</i> <i>tamiento</i>	27
Figura 7. Ventana de la pestaña <i>Programar, Configurar, Definir las protecciones</i>	27
Figura 8. Ventana de la pestaña <i>Programar</i> y seleccionar <i>Programar</i>	28
Figura 9. Configuración de la dirección IP en el PC	29
Figura 10. Conexión Ethernet entre el PC y PLC	30
Figura 11. Comunicación establecida entre el PC y el PLC	30
Figura 12. Ventana de la pestaña <i>Documentar</i>	31
Figura 13. Barrera de acceso a parqueadero	32
Figura 14. Diagrama de potencia del ejemplo control de acceso a parqueadero .	34
Figura 15. Diagrama de control del ejemplo control de acceso a parqueadero . .	34
Figura 16. Tres motores en cascada	35
Figura 17. Diagrama de potencia del ejemplo arranque de tres motores en cascada	38
Figura 18. Diagrama de control del ejemplo arranque de tres motores en cascada	38
Figura 19. Empacadora de productos	39
Figura 20. Diagrama de potencia del ejemplo empacadora de productos	42
Figura 21. Diagrama de control del ejemplo empacadora de productos	42
Figura 22. Diagrama de funcionamiento del temporizador TON	44
Figura 23. Diagrama de funcionamiento del temporizador TOFF	44
Figura 24. Diagrama de funcionamiento del temporizador TP)	45
Figura 25. Solución del ejemplo control de acceso a parqueadero empleando la función temporizador y contador	46
Figura 26. Solución del ejemplo arranque de tres motores en cascada usando la función temporizador y contador	47

Figura 27. Diagrama de funcionamiento de la función registro de desplazamiento	49
Figura 28. Modo de operación de la función registro de desplazamiento para el ejemplo control de acceso a parqueadero	49
Figura 29. Modo de operación de la función registro de desplazamiento para el ejemplo arranque de tres motores en cascada	50
Figura 30. Solución del ejemplo control de acceso a parqueadero empleando la función registro de desplazamiento	51
Figura 31. Solución del ejemplo control arranque de tres motores en cascada empleando la función registro de desplazamiento	53
Figura 32. Instrucciones para la programación empleando GRAFCET	54
Figura 33. Diagrama GRAFCET del ejemplo control de acceso a parqueadero .	55
Figura 34. Diagrama GRAFCET del ejemplo control secuencial de tres motores en cascada	55
Figura 35. Diagrama GRAFCET del ejemplo empacadora de productos	56
Figura 36. Solución del ejemplo control de acceso a parqueadero empleando el diagrama GRAFCET	57
Figura 37. Solución del ejemplo arranque de tres motores en cascada empleando el diagrama GRAFCET	58
Figura 38. Posición de pines en el tambor electromecánico (DRUM)	60
Figura 39. Posición de pines en el tambor electromecánico (DRUM)	61
Figura 40. Solución del control de acceso a parqueadero empleando la función conmutador del tambor (DRUM)	62
Figura 41. Solución del ejemplo arranque de tres motores en cascada empleando la función conmutador del tambor (DRUM)	63
Figura 42. Funcionamiento del bloque contador de pasos	64
Figura 43. Conexión estrella/triángulo	64
Figura 44. Solución del arranque de tres motores en cascada empleando la función contador de pasos	66
Figura 45. Funcionamiento del bloque FIFO	69
Figura 46. Solución del ejemplo empacadora de productos empleando la función FIFO	70
Figura 47. Solución del ejemplo empacadora de productos empleando palabras de memoria y bloques de operación	73

Figura 48. PLC TWIDO TWDLCAE40DRF	75
Figura 49. Motor de inducción trifásico	76
Figura 50. Variador de velocidad ALTIVAR 31	76
Figura 51. Cable UTP con terminales RJ45	77
Figura 52. Cable TSXCRJMD25	77
Figura 53. Esquema de conexión variador de velocidad ALTIVAR 31	78
Figura 54. Acceso a los menús ALTIVAR 31	79
Figura 55. Eliminación del filtro del bucle de velocidad	82
Figura 56. Espacio de trabajo gráfico del PLC Twido y el variador de velocidad sin comunicación	83
Figura 57. Configuración del PLC como maestro	84
Figura 58. Configuración del variador ALTIVAR 31 como esclavo	84
Figura 59. Espacio de trabajo gráfico del PLC Twido y el variador de velocidad enlazados	85
Figura 60. Configuración de la macro Drive	87
Figura 61. Configuración de la macro COM	89
Figura 62. Control de velocidad de un motor de inducción empleando las funciones MACROS.	91
Figura 63. Curva de calentamiento del horno eléctrico	94
Figura 64. Selección del módulo de expansión analógico TM2AMM3HT	95
Figura 65. Zona de configuración de las entradas-salidas del módulo analógico	96
Figura 66. Tabla de configuración de entradas - salidas del módulo analógico	97
Figura 67. Configuración de periodo para el modo de exploración	97
Figura 68. Principio de funcionamiento regulación PID	98
Figura 69. Ventana de selección del PID a configurar y utilizar	99
Figura 70. Ventana de configuración del PID, pestaña general	99
Figura 71. Ventana de configuración del PID, pestaña entrada	100
Figura 72. Ventana de configuración del PID, pestaña PID	100
Figura 73. Ventana de configuración del PID, pestaña AT	101
Figura 74. Ventana de configuración del PID, pestaña salida	102
Figura 75. Programación LADDER para el control de temperatura de un horno eléctrico	102

Figura 76. Esquema de conexiones físicas 103
Figura 77. Circuito de amplificación 104
Figura 78. Arranque en frío del horno empleado para el control PID 105
Figura 79. Respuesta del control PID al estabilizarse el sistema 105

LISTA DE TABLAS

	pág.
Tabla 1. Variables de control para el ejemplo control acceso a parqueadero . . .	33
Tabla 2. Variables de control para el ejemplo arranque de tres motores en cascada	36
Tabla 3. Variables de control para el ejemplo arranque estrella/triángulo de tres motores en cascada	37
Tabla 4. Variables de control del ejemplo empacadora de productos	40
Tabla 5. Variables de control del ejemplo control de velocidad de un motor de inducción empleando comunicación modbus	90

RESUMEN

Título: Controlador Lógico Programable. Guía de programación y ejemplos sencillos¹

Autora: Natalia Andrea Flórez Santana²

Palabras claves: Aprendizaje, Automatización, Guía de programación, PLC, Twido.

DESCRIPCIÓN:

El Controlador Lógico Programable es un dispositivo que dispone de herramientas que facilitan la automatización de procesos industriales empleando un equipo digital programable. La Escuela de Ingeniería Eléctrica, Electrónica y Telecomunicaciones (E3T) en la actualidad dispone de cuatro PLC de referencia Twido de Schneider Electric. Estos dispositivos se encuentran en óptimo funcionamiento y actualmente se usan para las prácticas de laboratorio de la asignatura Accionamientos Eléctricos. Dada la extensión en el contenido de esta asignatura, y de las horas prácticas de laboratorio programadas, no se alcanza con el objetivo de cubrir en su totalidad las funciones que dispone este equipo. En este proyecto de grado se profundizó en la explicación de las principales funciones y facilidades que presenta este Controlador Lógico Programable en la automatización de procesos que normalmente se encuentran en la industria. Se realizó un documento escrito como herramienta de aprendizaje que facilite el entendimiento, la programación y el uso del PLC Twido y software TwidoSuite. Este documento permite mejorar el proceso de aprendizaje de quienes buscan ampliar su conocimiento respecto al PLC que se trabaja actualmente en el curso. De esta manera, se contribuye al mejoramiento del aprendizaje de los conceptos que se imparten en la asignatura Accionamientos Eléctricos y Automatización de Procesos.

¹Trabajo de Grado

²Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones. Director: Julio Augusto Gélvez Figueredo. Magíster en Potencia Eléctrica

ABSTRACT

Title: Programmable Logic Controller. Programming guide and simple examples¹

Author: Natalia Andrea Flórez Santana²

Key Words: Learning, Automation, Programming Guide, PLC, Twido.

DESCRIPTION:

The Programmable Logic Controller (PLC) is a device that has tools that facilitate the automation of industrial processes using programmable digital equipment. The School of Electrical, Electronic and Telecommunications Engineering (E3T) currently has four Schneider Electric Reference PLC's. These devices are in optimal performance and are currently used for laboratory practices of the subject Electric Drives. Given the extent in the content of this subject, and the scheduled laboratory practical hours, it is not achieved with the aim of fully covering the functions available to this equipment. In this degree project, it was further elaborated in the explanation of the main functions and facilities presented by this Programmable Logic Controller in the automation of processes normally found in the industry. A written document was made as a learning tool that facilitates the understanding, programming and use of the Twido PLC and TwidoSuite software. This document allows to improve the learning process of those who seek to expand their knowledge regarding the PLC currently being worked on in the course. In this way, it contributes to the improvement of the learning of the concepts taught in the subject Electric Drives and Process Automation.

¹Bachelor Thesis

²Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones. Director: Julio Augusto Gélvez Figueredo. Magíster en Potencia Eléctrica

INTRODUCCIÓN

- Este trabajo de grado pretende ser una ayuda para quien desee profundizar en la programación del PLC Twido empleando el software TwidoSuite, se apoya en los apuntes de clase del profesor Julio Augusto Gélvez Figueredo quien autoriza el uso de los programas que empleaba en la asignatura de accionamientos eléctricos del programa de Ingeniería Eléctrica.

- Se recomienda si se está manejando el software TwidoSuite por primera vez, visualizar el tutorial que se presenta al ejecutar el programa, así mismo, el software permite acceder, en línea, la información correspondiente a la configuración, programación, utilización y conexionado de los autómatas de la familia Twido y Software TwidoSuite (manual de usuario), para consultar dicha información se ingresa pulsando el icono interrogante que aparece en la parte superior derecha del programa o colocando el puntero en un bloque funcional y oprimiendo F1.

- La metodología del trabajo consta de tres ejemplos bases en los cuales se emplean diferentes funciones para ejecutar en el mismo programa, haciendo uso de las mismas entradas y salidas.

Los ejemplos se han titulado:

- Control de acceso a parqueadero
- Arranque de tres motores en cascada
- Empacadora de productos

- En el capítulo 1 se realiza una breve descripción de las principales funciones entre las que se tienen: Contador, temporizador, registro de desplazamiento, conmutador del tambor, contador de pasos, registro FIFO, bloques de operación, además, se explica la secuencia para la creación de un programa, y la forma de *depurar* un programa.

- En el capítulo 2 se presenta el bloque funcional resaltando las principales características de funcionamiento y observaciones al programar y se ejecutan los ejemplos empleando un mismo bloque funcional.

- En el capítulo 3 se presenta la información básica correspondiente a la configuración del PLC Twido y del variador de velocidad altivar 31 para establecer una comunicación Modbus; también se resaltan las macros Drive y macros com como funciones que facilitan la programación y uso del protocolo Modbus, al finalizar el capítulo se presenta el programa diseñado para realizar la comunicación y el control del motor de inducción.
- En el capítulo 4 (último capítulo) se recuerdan las principales características del control PID, se detallan los parámetros para configurar cada uno de los 4 PID disponibles en el software TwidoSuite y se presenta el programa para el control de temperatura de un horno eléctrico.

1. SOFTWARE DE PROGRAMACIÓN TWIDOSUITE

1.1. REQUISITOS DEL SISTEMA

El equipo de cómputo debe cumplir con unos requisitos de hardware mínimos para la ejecución del software de programación TwidoSuite. Como requisitos mínimos del hardware se requiere de un procesador Pentium a 466 Hz, RAM de 128 MB y disco duro de 100 MB.

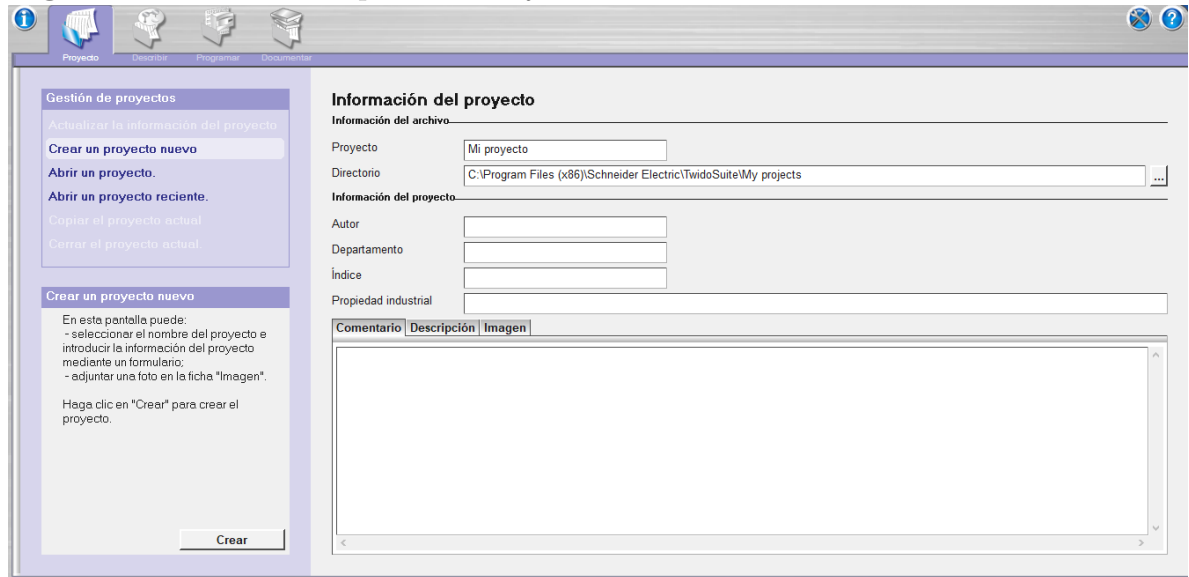
El software de programación TwidoSuite está diseñado para ejecutarse en diferentes sistemas operativos como Windows 2000/XP/Vista o versiones superiores a estas. (TwidoSuiteV2.3, Guía de programación, 2011, p.10). Según sea el sistema operativo se requiere una versión específica, por ejemplo, el sistema operativo Windows 7 requiere la versión TwidoSuite V32.

Se recomienda ver el tutorial que se presenta al abrir el software TwidoSuite si se está ejecutando por primera vez.

Al abrir el software TwidoSuite y seleccionar *Modo programación* se presenta una ventana con cuatro pestañas: *Proyecto*, *Describir*, *Programar* y *Documentar* como se muestra Figura 1.

1.2. CREACIÓN DE UN PROGRAMA

Figura 1: Ventana de la pestaña *Proyecto*



En la pestaña *Proyecto* se activa un menú el cual solicita la siguiente información para la creación de un proyecto nuevo: un nombre que no supere los 36 caracteres alfanuméricos sin espacios y la información de autoría del proyecto que no es indispensable para la creación de un programa, así mismo, se indica la dirección donde quedarán guardados los archivos del proyecto (archivo ejecutable y la carpeta de archivos), esta dirección se puede modificar si lo requiere el programador.

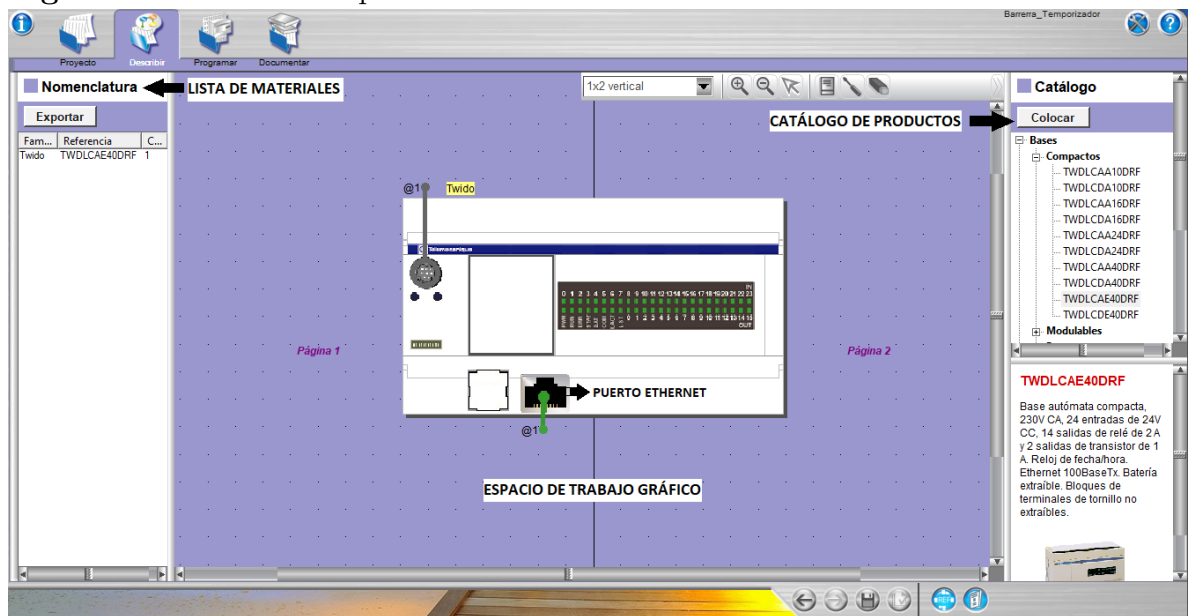
1.3. DESCRIBIR PROGRAMA

En la pestaña *Describir* se presentan tres ventanas (Figura 2):

- Catálogo de productos: Contiene productos disponibles como *bases, módulos de ampliación, módulos de ampliación serie, adaptador serie, cartucho RTC, HMI y elementos de redes.*
- Espacio de trabajo gráfico: Se configura el equipo o PLC a utilizar seleccionado de la ventana *catálogo*, ubicado en la parte superior derecha. Se selecciona el PLC compacto de referencia TWDLCAE40DRF y se desplaza hasta el espacio de trabajo gráfico.

- Lista de materiales (Nomenclatura): Presenta el listado de los equipos seleccionados de la ventana *catálogo* ubicados en el espacio de trabajo gráfico. Para visualizar esta lista, es indispensable agregar los equipos en el espacio de trabajo gráfico y hacer clic sobre el icono “REF” que aparece en la parte inferior derecha.

Figura 2: Ventana de la pestaña *describir*



Según sea la referencia del PLC base (PLC Twido) a configurar y el protocolo de comunicación a emplear, se requerirá o no de módulos de comunicación adicionales; en el caso de la referencia a utilizar TWDLCAE40DRF dispone de los protocolos de comunicación Ethernet (TCP/IP), Modbus y comunicación de conexión remota, esta última permite intercambiar la información entre un número reducido de PLC's mediante comandos que se pueden visualizar en el manual Twido (TwidoSuiteV2.3, Guía de programación, 2011, p.108).

Para acceder a la sección *Redes Ethernet*, se hace doble clic en el puerto Ethernet del PLC (Figura 2).

En la sección *Redes Ethernet, Configuración dirección IP*, opción *Configurada*, se carga la dirección IP que tiene asignado el PLC, por ejemplo la IP del autómata empleado es 192.168.71.90 como se muestra en la Figura 3. Al ingresar la dirección IP, la aplicación

TwidoSuite carga de forma predeterminada la máscara de subred basada en la categoría de la dirección IP, la dirección de pasarela usualmente la proporciona el administrador de red. Ver en la figura 3 que la dirección IP y la dirección de pasarela son iguales a excepción de los últimos dígitos.

Figura 3: Configuración de la dirección IP en el PLC

Configuración

Redes Ethernet

Configuración dirección IP

Desde un servidor Dirección IP: 192 . 168 . 71 . 90

Configurada Máscara de subred: 255 . 255 . 255 . 0

Dirección de la pasarela: 192 . 168 . 71 . 1

IP Marcada

Especificar una dirección IP Marcada Especificar una dirección IP para la conexión marcada: . . .

Tiempo de inactividad

NOTA: El autómata detectará las conexiones TCP activas y pasivas y cerrará las que estén inactivas durante el tiempo aquí indicado. Si el tiempo de inactividad máximo se pone en 0, el autómata no lo detectará.

Introduzca el tiempo de inactividad de la conexión TCP. 10 min(s) Predeterminado

Dispositivos remotos

Índice	Dirección IP esclava	Unit ID	Tiempo de espera de conexión (100ms)
1			
2			

Aceptar Cancelar

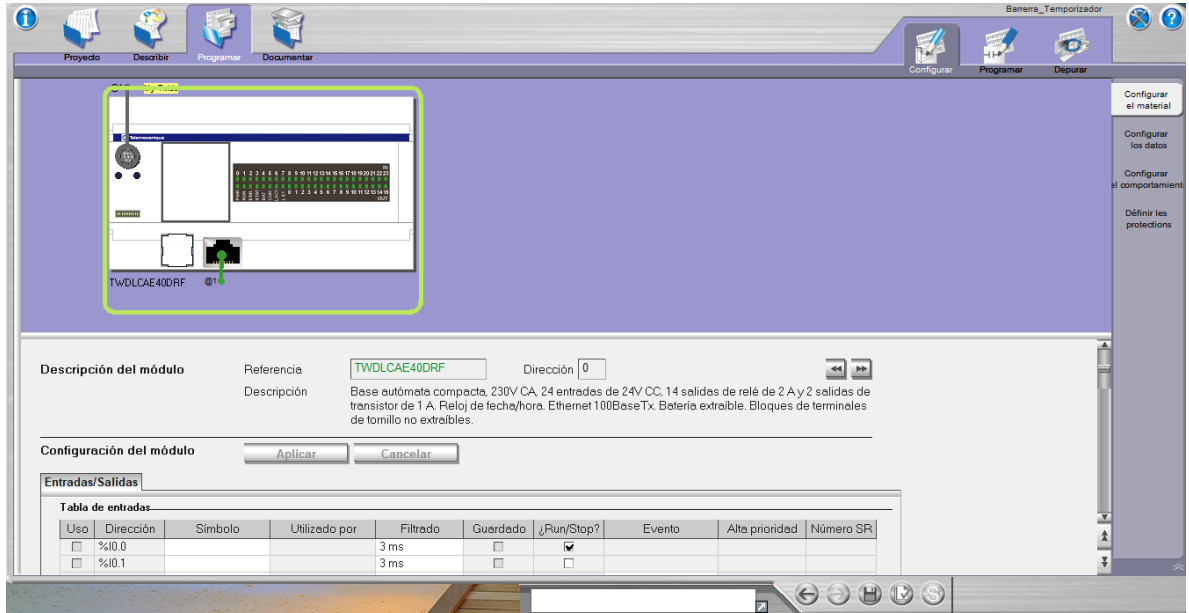
1.4. PROGRAMAR

En la ventana de la pestaña *Programar* se presentan tres secciones: *Configurar*, *Programar*, *Depurar*.

1.4.1. Configurar. En esta sección se presentan cuatro pestañas: *Configurar el material*, *Configurar los datos*, *Configurar el comportamiento* y *Definir las protecciones*

1.4.1.1. Configurar el material (Hardware). Al seleccionar la pestaña *Programar*, *Configurar* se visualiza la pestaña *Configurar el material* ubicada en la parte lateral derecha como se muestra en la figura 4.

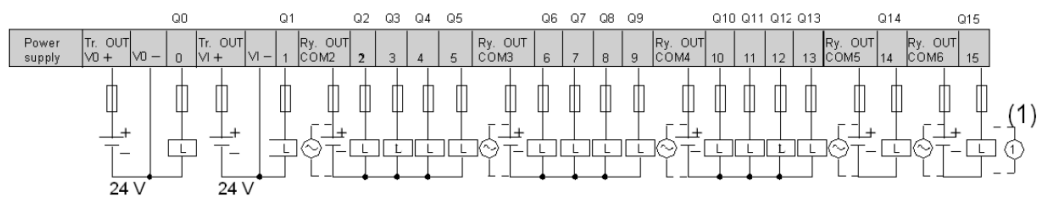
Figura 4: Ventana de la pestaña *Programar*, Configurar



En esta sección se configuran los recursos de hardware que son las entradas y salidas digitales (binarias) del autómata. Se asigna $\%I_{i,j}$ para las entradas y $\%Q_{i,j}$ para las salidas, donde el subíndice i representa la dirección del módulo (0 a 7 módulos de extensión) y el subíndice j representa la posición de la entrada ó salida digital en el PLC.

En la configuración del módulo de entradas y salidas se asigna una dirección para la entrada Run/Stop, para este caso la dirección seleccionada es $\%I0.0$ en el correspondiente módulo. La entrada Run/Stop habilita la ejecución del programa que es diferente al arranque ó parada del proceso.

Figura 5: Diagrama de cableado para la base TWDLCAE40DRF



Fuente: SCHNEIDER ELECTRIC, Telemecanique. Autómatas programables TWIDO. Bases compactas y modulares, guía de hardware, 2011, pág. 114.

Las salidas Q0 Y Q1 son salidas a transistor para la referencia empleada, ver figura 5.

También existen entradas y salidas de tipo analógicas, en el caso de la familia Twido se requiere adicionar un módulo de ampliación de entradas/salidas analógicas al PLC base.

1.4.1.2. Configurar los datos (Software). Hace referencia a los elementos de función configurables como son los objetos simples, bloques de función, objetos de E/S y objetos avanzados.

- **Objetos simples:** En esta sección se presentan las constantes y objetos de memoria.
 - Bits de memoria (%M): Son bits de un espacio que se utilizan para guardar valores intermedios durante la ejecución del proceso, admite 128 bits para las referencias de los autómatas: TWDLCA10DRF, TWDLCA16DRF; y 256 bits para las demás referencias.
 - Bits de sistema (%S): Proporcionan información del control y el funcionamiento del PLC. Contiene un número máximo de 128 bits para su uso, algunos bits pueden ser leídos y escritos (modificados) desde el programa, por ejemplo, cuando el bit de sistema %S22 se pone en 1 (sólo en la ejecución del programa), los pasos activos de la función GRAFCET se ponen en 0.
 - Palabras de memoria (%MW): Almacenan valores en la memoria de datos durante la ejecución del programa. Contiene un número máximo de 3000 palabras de memoria para su uso que pueden ser leídas y escritas durante la ejecución del programa.
 - Palabras de sistema (%SW): Son palabras de 16 bits internas que dan información del control del sistema, algunas pueden ser leídas y escritas desde el programa, otras sólo pueden ser leídas y no se pueden modificar.
 - Constantes (%KW): Almacenan constantes ó mensajes alfanuméricos. Sólo se puede modificar desde el software TwidoSuite. Contiene un número máximo de 256 para su uso, pueden ser leídas pero no escritas durante la ejecución del programa.
- **Bloques funcionales:** Son funciones predeterminadas del software que facilitan la programación del PLC.

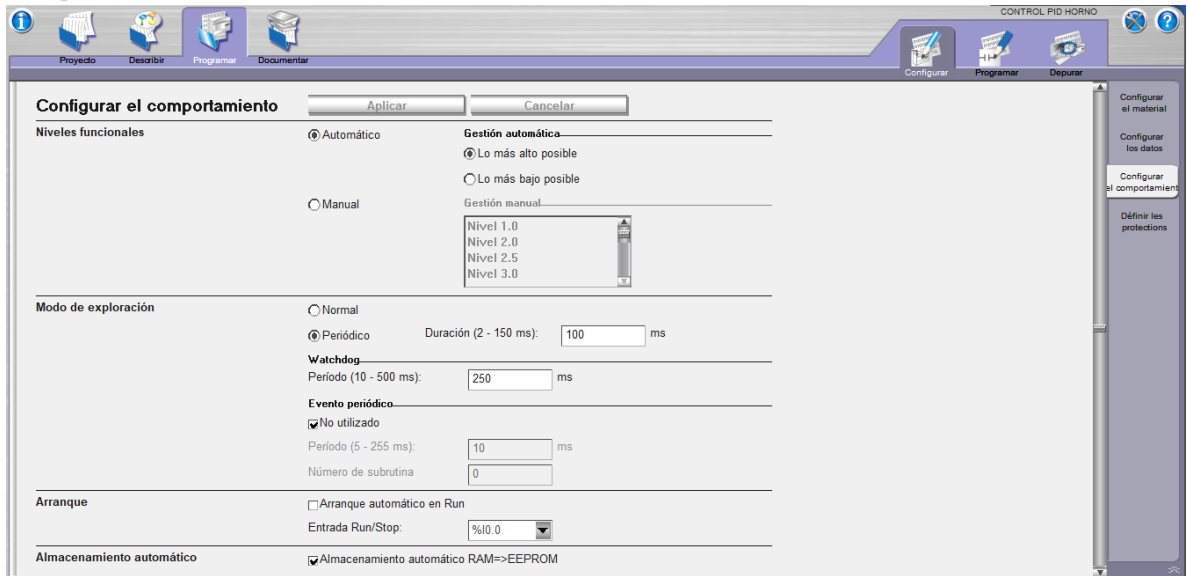
- Temporizadores: (%T_{Mi}): Dispone de 64 (números) en la referencias TWDL-CAA10DRF y TWDLCAA16DRF, y de 128 para los demás autómatas, los temporizadores se puede configurar tipo T0N, TOF ó TP.
 - Contador progresivo o regresivo (%C_i): Realiza la operación de contar de forma progresiva y/o regresiva a partir de un valor preseleccionado. Dispone de 128 contadores.
 - Controlador del conmutador de tambor ó DRUM (%DR): Se disponen de 4 controladores de tambor (0 a 3 controladores compactos), cada controlador tiene una cantidad de 8 pasos.
 - Registros LIFO/FIFO (%R_i): Permite configurar y utilizar 4 registros LIFO/FIFO, esta función puede almacenar hasta 16 palabras de 16 bits en cada bloque.
 - Registros de desplazamiento (%SBR): Dispone de 8 registros de desplazamiento de 16 bits (j=0:15) que pueden ser asignados mediante una instrucción %SBR_{i,j} para su ejecución.
 - Contador de pasos (%SC): Dispone de 8 contadores de pasos. Los bits del contador de pasos tiene un rango de 0 a 255 (j:0:255) que se ejecutan mediante la instrucción %SC_{i,j}.
- **Objetos de E/S:** Son funciones que se ejecutan por medio de las entradas y salidas físicas del PLC en funcionamiento, estas funciones no pueden ser simuladas.
- Contadores rápidos (%FC): El valor preestablecido está entre un rango de 0 - 65535 eventos en modo estándar y entre un rango 0 - 4294967295 de eventos en modo de palabra doble.
 - Generadores de pulsos (%PLS/ %PWM): Se utiliza como generador de señales de ondas cuadradas ó modulación de ancho de pulso, las salidas %Q0.0 y %Q0.1 están destinadas para el uso esta función. En la configuración se puede seleccionar una base de tiempo de: 1s, 10ms, 0.57ms, 0.142ms. No es posible realizar simulación empleando esta función, sólo es posible ver resultados durante la ejecución.
 - Contadores muy rápidos (%VFC_i): Esta función se emplea para las entradas que requieran frecuencias hasta 20kHz y el bloque de función se puede

configurar como: Contador progresivo, regresivo, progresivo/regresivo ó frecuencímetro.

- **Objetos avanzados:** Son bloques de función avanzados que emplean palabras y bits especializados.
 - **Fecha y hora en tiempo real (%SCH):** Son utilizados para controlar las funciones fecha y hora, para ello es necesario la configuración del reloj en tiempo real (día, mes, año, hora) que activará el bit de salida programado. Contiene un máximo de 16 objetos de reloj en tiempo real para su uso.
 - **PID:** Es un mecanismo de control retroalimentado que responde a las variaciones de procesos que requieran regulación permanente frente a perturbaciones externas ó propias del sistema. Contiene un máximo de 14 objetos PID para su uso. Esta función no permite visualizar la solución mediante la simulación, sólo es posible en tiempo real. El periodo de muestreo está en un rango de 0.01s y 100s.
 - **Funciones de macros:** Las macros de comunicación para Twido permiten realizar intercambio de datos con cualquier tipo de esclavo Modbus.
 - Macros Comm:* Grupo de librería que permite enviar solicitudes de escritura o lectura de un bit ó una palabra a un esclavo Modbus.
 - Macros Drive:* Grupo de librería que envía solicitudes de escritura de un bit ó una palabra a unidades externas mediante diferentes protocolos de comunicación, por ejemplo Modbus, CANopen.

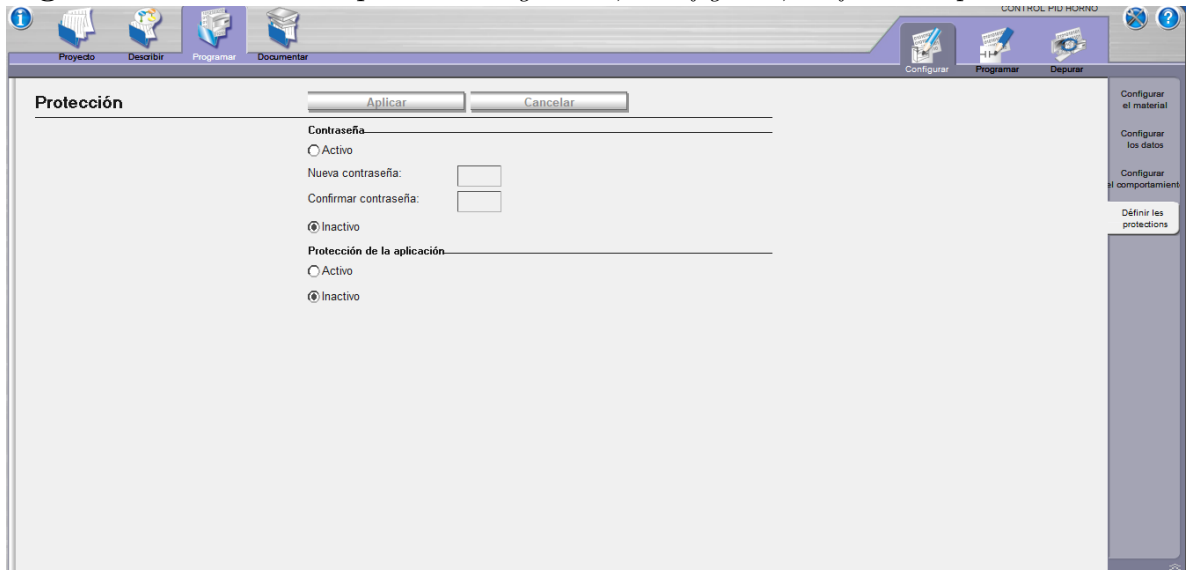
1.4.1.3. Configurar el comportamiento. En esta ventana (Figura 6) también se puede configurar la entrada Run/Stop ó el arranque automático en RUN según lo prefiera el programador. La descripción detallada sobre configuración de los ajustes de esta sección se pueden visualizar en el manual de ayuda en línea del software de programación TwidoSuite V2.3.

Figura 6: Ventana de la pestaña *Programar, Configurar, Configurar el comportamiento*



1.4.1.4. Definir las protecciones. En esta ventana (Figura 7) se puede agregar una contraseña al programa creado, en docencia o formación no se recomienda agregar contraseña ya que, si no se tiene la clave, se restringe el acceso de realizar alguna modificación en el programa.

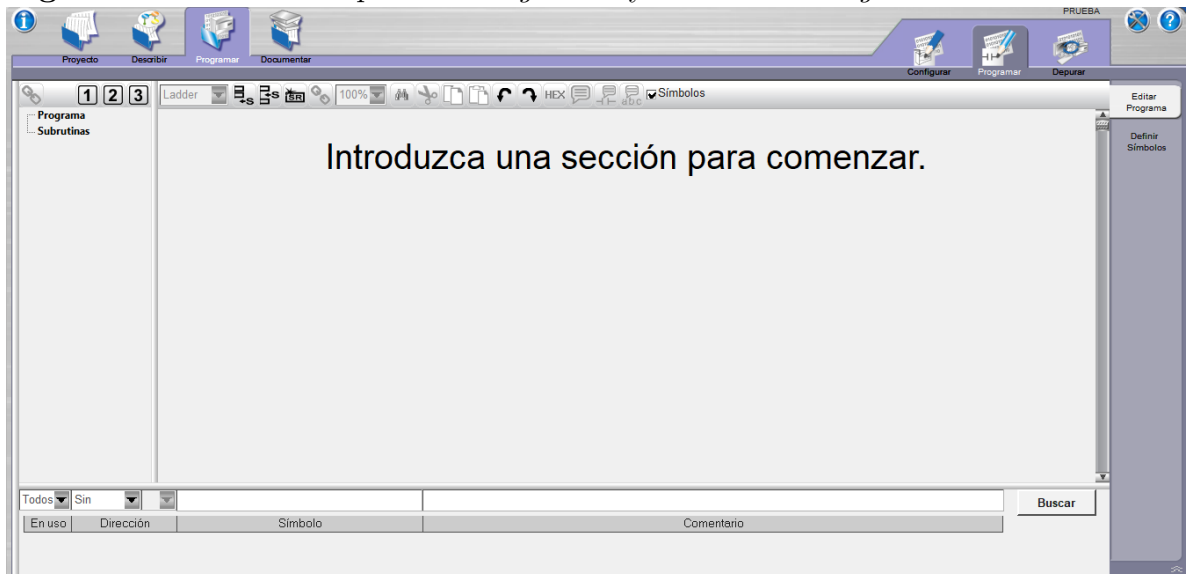
Figura 7: Ventana de la pestaña *Programar, Configurar, Definir las protecciones*



1.4.2. Programar. Al seleccionar la pestaña *Programar* y seleccionar *Programar* se presentan dos pestañas: *Editar programa*, *Definir símbolos*.

1.4.2.1. Editar programa. En esta ventana (Figura 8) se desarrolla la programación principal en la cual se agregan los bloques de funciones y la programación en el lenguaje Ladder Logic ó Lista. Estos lenguajes se pueden conmutar debido a que pueden pasar del modo gráfico a lista o viceversa, siempre y cuándo estos sean compatibles. El lenguaje de programación es modificable durante el proceso de edición.

Figura 8: Ventana de la pestaña *Programar* y seleccionar *Programar*



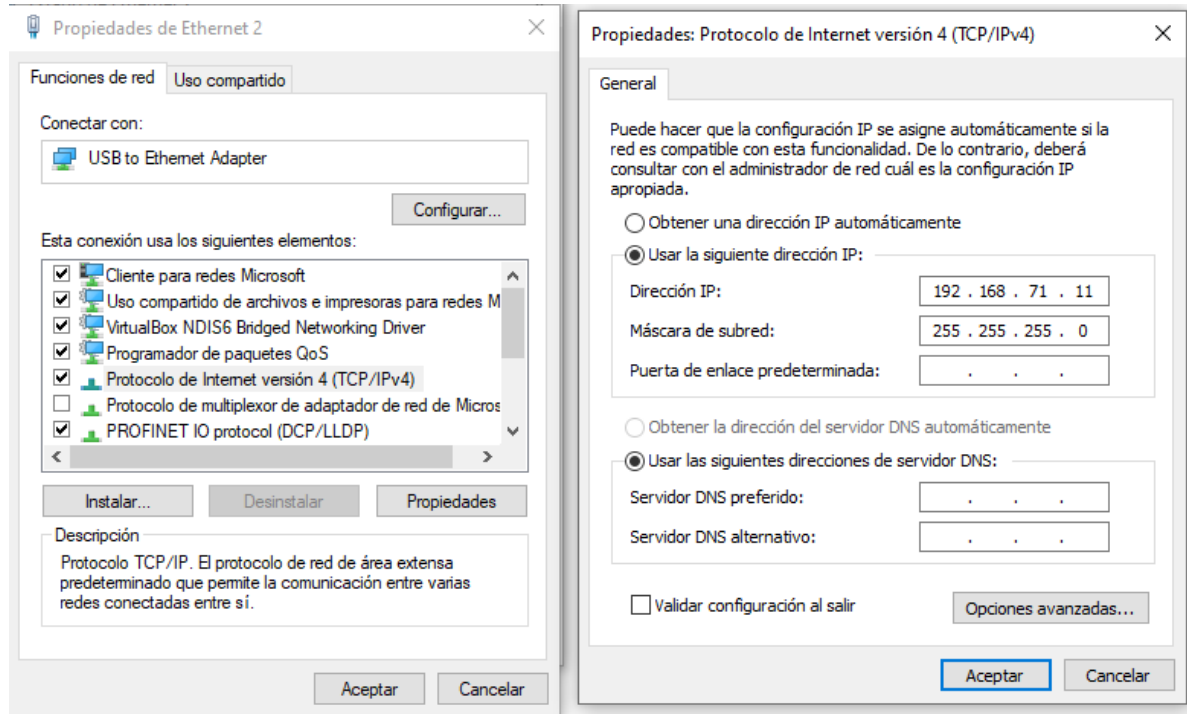
1.4.2.2. Definir símbolos. Las variables declaradas en la programación principal se muestran en la pestaña *Definir símbolos* con el nombre y dirección que han sido asignadas en la programación. Las variables se pueden agregar en primera instancia en esta pestaña y luego emplearlas en la programación.

1.4.3. Depurar el programa. Se emplea para transferir el programa entre el PC y el PLC ó viceversa, para ello se selecciona la pestaña *Programar* y *Depurar*.

En el computador se configuran las propiedades de conexión de red Ethernet, protocolo de internet versión 4 (TCP/IPv4) se ingresa la dirección IP que no es la misma dirección del PLC ya que cada equipo debe tener una única dirección IP asignada, por ello, se

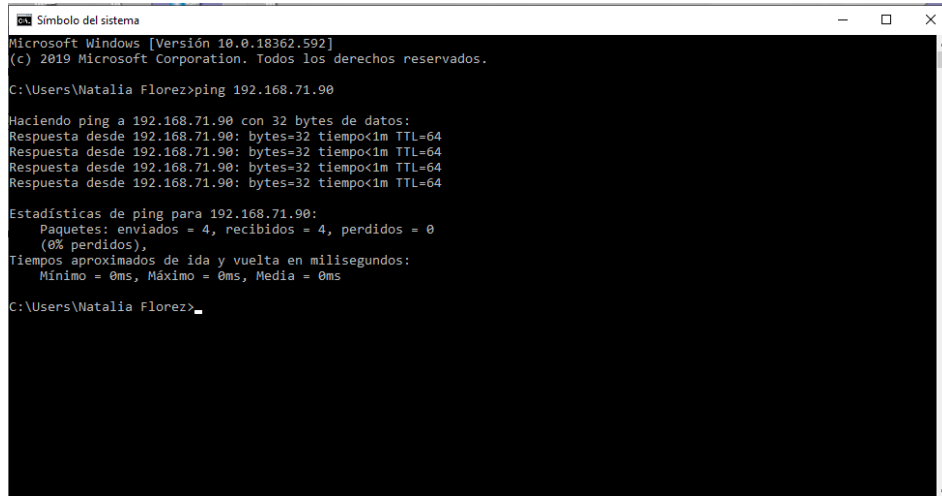
deben variar los últimos dígitos como se muestra en la figura 9.

Figura 9: Configuración de la dirección IP en el PC



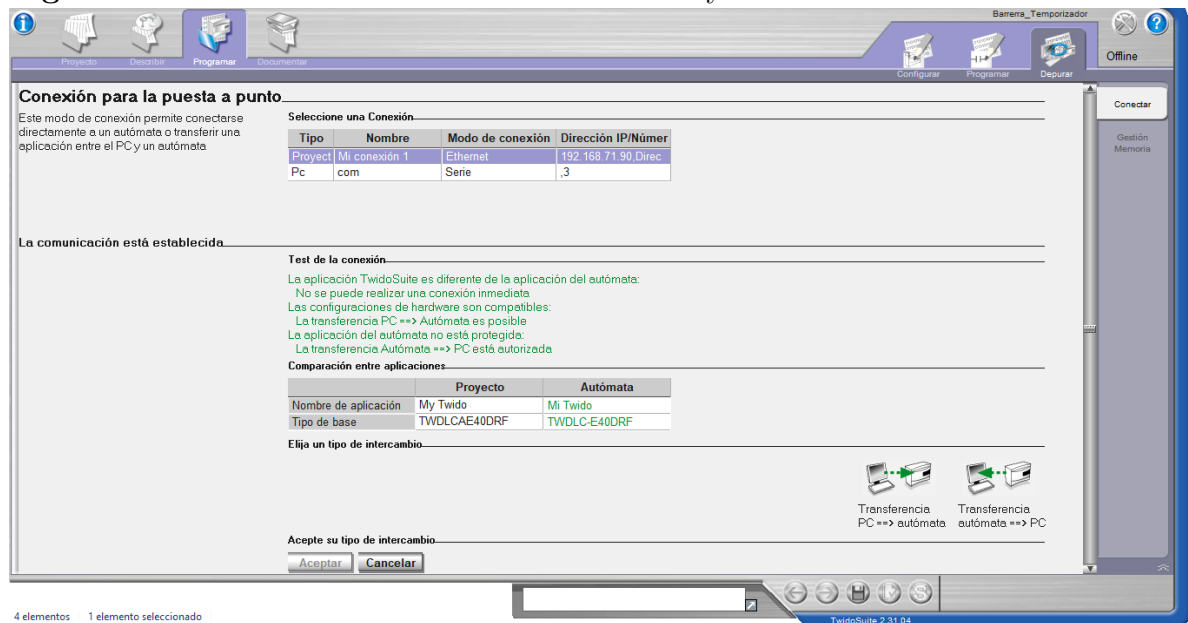
Para verificar que el PC reconoce el puerto Ethernet conectado, se ingresa a la aplicación símbolo del sistema del sistema del PC (cmd.exe) y se ejecuta la siguiente línea de comando “ping 192.168.71.90” que corresponde a la dirección IP del PLC, la respuesta de la solicitud cuando hay comunicación entre el PC y el PLC se muestra como en la figura 10.

Figura 10: Conexión Ethernet entre el PC y PLC



Para verificar la comunicación entre el PC y el PLC, se ingresa a la pestaña *Configurar*, *Depurar* y en la parte inferior derecha se da clic en el símbolo *gestionar conexión* la cual presenta una sección donde se selecciona del modo de conexión (Ethernet) y la dirección IP del PLC, seguido de esto, se da clic en aplicar y aceptar la comunicación, si los pasos anteriores se realizaron de forma correcta, debe aparecer una sección que reconoce la comunicación establecida con el PLC y se visualiza como en la figura 11.

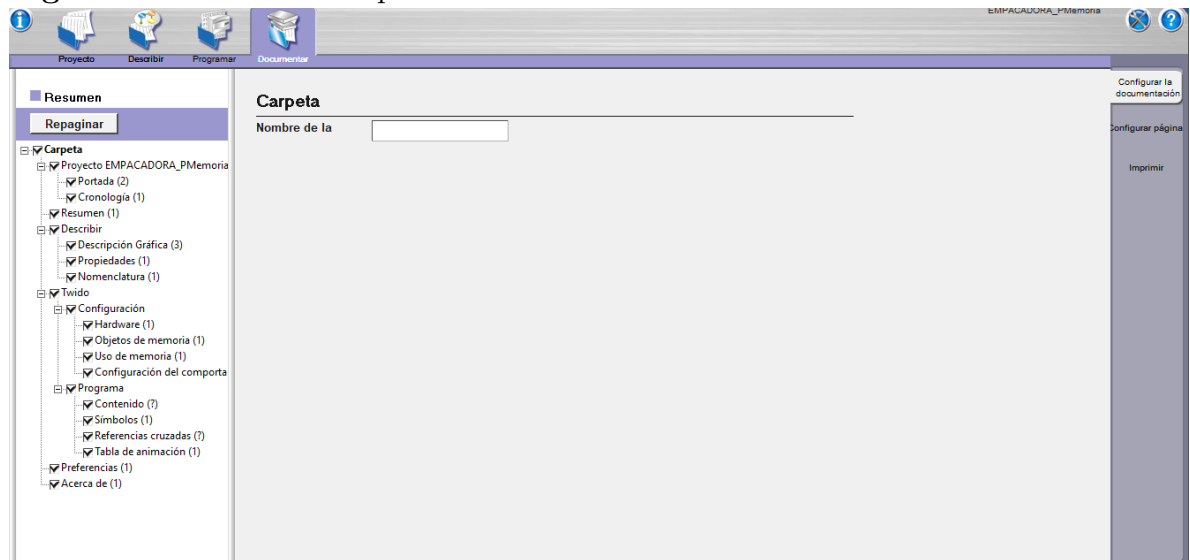
Figura 11: Comunicación establecida entre el PC y el PLC



1.5. DOCUMENTAR

El software Twido hace un resumen de las secciones creadas previamente: *Creación del programa, Descripción, Configuración del hardware, software y del Programa*, este documento puede ser exportado en un Formato de Documento Portátil (PDF, según sus siglas en inglés) según lo requiera el programador (Figura 12).

Figura 12: Ventana de la pestaña *Documentar*

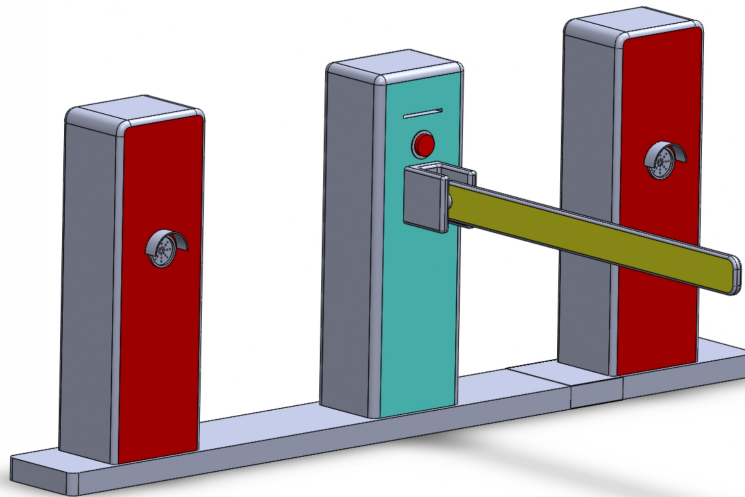


2. FUNCIONES DEL SOFTWARE TWIDO SUITE. EJEMPLOS EXPLICATIVOS

2.1. ENUNCIADO DE LOS EJEMPLOS

2.1.1. Control de acceso a parqueadero. Se desea controlar el acceso a un parqueadero público de tal modo que, el vigilante pulsa un botón para iniciar el proceso el cual hace que la barrera inicie el ascenso hasta alcanzar un determinado tope, y espere un tiempo de 20 seg que permita el ingreso ó salida del vehículos. El número de cupos de parqueo es de 10 (para el ejemplo planteado), por lo tanto debe llevarse el control de puestos libres. Cuando el parqueadero tenga cupo lleno: si el fotodetector sensa la salida del vehículo, la barrera se elevará permitiendo la salida. Si el fotodetector sensa la entrada del vehículo y el cupo está lleno, la barrera no se elevará así se pulse el botón de inicio.

Figura 13: Barrera de acceso a parqueadero



- 2.1.1.1. Características del control**
- Al pulsar la estación de arranque inicia el ascenso de la barrera (proceso de apertura) para el ingreso de los vehículos. (Siempre y cuando hallan cupos disponibles)
 - Al alcanzar la máxima posición, se acciona el final de carrera alto (FCA), desenergizando el motor que acciona la barrera.

- En estado “abierta” cuenta 10 segundos para permitir el ingreso de vehículos.
- Terminado el tiempo (10 seg) y habiendo ingresado el vehículo, inicia el descenso de la barrera (proceso de cierre). La presencia del vehículo es supervisada por un fotodetector que elevará la barrera en caso de existir un obstáculo durante el descenso ó permitirá continuar el proceso descenso, no habiendo detectado ningún obstáculo.
- Contar el número total de vehículos que permanecen en el parqueadero, con la condición que si se alcanza la capacidad máxima de cupos no se permita el ingreso de vehículos.

En la tabla 1 se presentan las variables de control empleadas para programar la solución del presente ejemplo; haciendo uso de las funciones que serán explicadas en las siguiente secciones.

Tabla 1: Variables de control para el ejemplo control acceso a parqueadero

Parámetros			
Dirección	Estado	Símbolo	Descripción
%I0.0	NA	Run/Stop	Estado del autómata.
%I0.2	NA	BP	Estación de arranque que eleva la barrera para el ingreso ó salida de vehículos.
%I0.3	NA	FCA	Final de carrera alto que detiene la barrera en la posición máxima.
%I0.4	NA	FD_IN	Fotodetector a la entrada del parqueadero.
%I0.5	NA	FCB	Final de carrera bajo que detiene la barrera en la posición más baja. Sólo se activará cuando termine el temporizador (10 Seg) y no se detecten obstáculos para el descenso.
%I0.6	NA	FD_OUT	Fotodetector a la salida del parqueadero.
%Q0.1		MOTOR_ARRIBA	Barrera en elevación.
%Q0.2		MOTOR_ABAJO	Barrera en descenso.

2.1.1.2. Circuito de potencia y control del ejemplo control de acceso a parqueadero. Se empleó el programa CadeSIMU (Versión 3.0) para graficar y simular el circuito de potencia y control con lógica cableada (simbología Europea) para los ejemplos presentados.

Figura 14: Diagrama de potencia para el ejemplo control de acceso a parqueadero

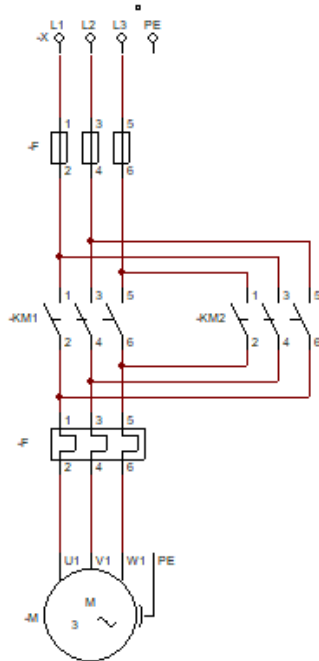
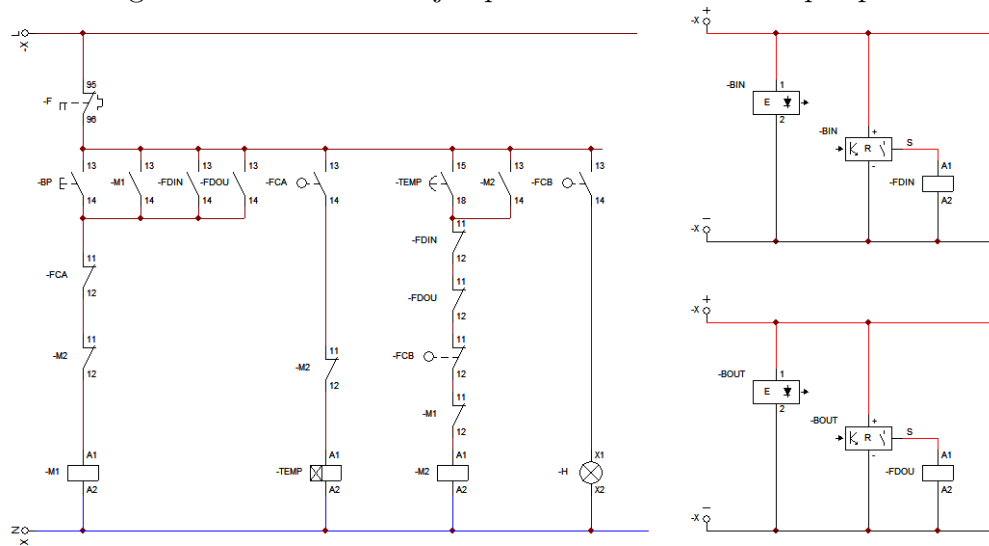
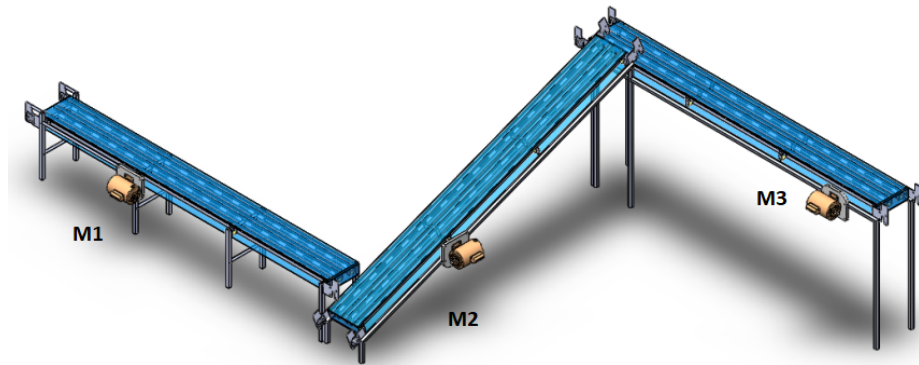


Figura 15: Diagrama de control del ejemplo control de acceso a parqueadero



2.1.2. Control secuencial de tres motores. Se requiere energizar tres motores de inducción arranque directo para controlar tres secciones de una banda transportadora, el “encendido” de estos motores debe seguir la secuencia: Motor 3 (M3), motor 2 (M2), motor 1 (M1) y para el “apagado”: Motor 1 (M1), motor 2 (M2) y motor 3 (M3), respectivamente. Cada motor debe disponer de una protección de sobrecorrientes.

Figura 16: Tres motores en cascada



2.1.2.1. Características del control • Al pulsar la estación de arranque; se energizan los motores en la siguiente secuencia : Motor 3 (M3), Motor 2 (M2) y Motor 1 (M1) a intervalos de 10 Seg.

- Al pulsar la estación de parada se desenergizan los motores en el siguiente orden: Motor 1 (M1), Motor 2 (M2), y Motor 3 (M3), dependiendo de cuales estén energizados, a intervalos de 10 Seg.

- Al existir una falla se acciona cualquiera de las protecciones de sobrecorrientes de los motores, estos deben parar instantáneamente.

Tabla 2: Variables de control para el ejemplo arranque de tres motores en cascada

Parámetros			
<i>Dirección</i>	<i>Estado</i>	<i>Símbolo</i>	<i>Descripción</i>
%I0.0	NA	Run/Stop	Estado del autómata.
%I0.1	NA	OL1	Protección de sobrecorriente para motor M1.
%I0.2	NA	OL2	Protección de sobrecorriente para motor M2.
%I0.3	NA	OL3	Protección de sobrecorriente para motor M3.
%I0.4	NC	PARADA	Botón pulsador para desenergizar los motores.
%I0.5	NA	ARRANQUE	Botón pulsador para la energización de los motores.
%Q0.1		MOTOR_M1	Motor en marcha M1.
%Q0.2		MOTOR_M2	Motor en marcha M2.
%Q0.3		MOTOR_M3	Motor en marcha M3.

Tabla 3: Variables de control para el ejemplo arranque estrella/triángulo de tres motores en cascada

Parámetros			
<i>Dirección</i>	<i>Estado</i>	<i>Símbolo</i>	<i>Descripción</i>
%I0.0	NA	Run/Stop	Estado del autómata.
%I0.1	NA	OL1	Protección de sobrecorriente para motor M1.
%I0.2	NA	OL2	Protección de sobrecorriente para motor M2.
%I0.3	NA	OL3	Protección de sobrecorriente para motor M3.
%I0.4	NC	PARADA	Botón pulsador para desenergizar los motores.
%I0.5	NA	ARRANQUE	Botón pulsador para la energización de los motores.
%Q0.0		MOTOR_M3	Motor M3 conectado a la RED.
%Q0.1		MOTOR_M3Y	Motor M3 conectado en YE.
%Q0.2		MOTOR_M3D	Motor M3 conectado en DELTA.
%Q0.3		MOTOR_M2	Motor M2 conectado a la RED.
%Q0.4		MOTOR_M2	Motor M2 conectado en YE.
%Q0.5		MOTOR_M2	Motor M2 conectado en DELTA.
%Q0.6		MOTOR_M1	Motor M1 conectado a la RED.
%Q0.7		MOTOR_M1	Motor M1 conectado en YE.
%Q0.8		MOTOR_M1	Motor M1 conectado en DELTA.

2.1.2.2. Circuito de potencia y control del ejemplo arranque de tres motores en cascada : Programa creado con el software CadeSIMU (Versión 3.0).

Figura 17: Diagrama de potencia del ejemplo arranque de tres motores en cascada

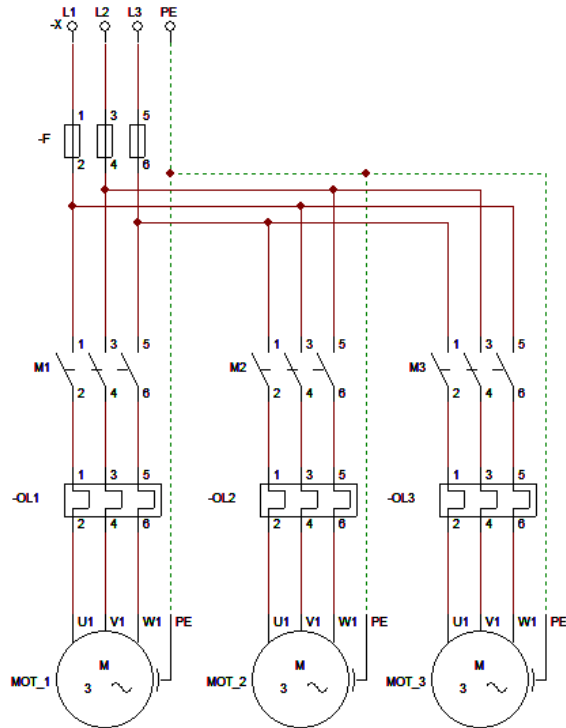
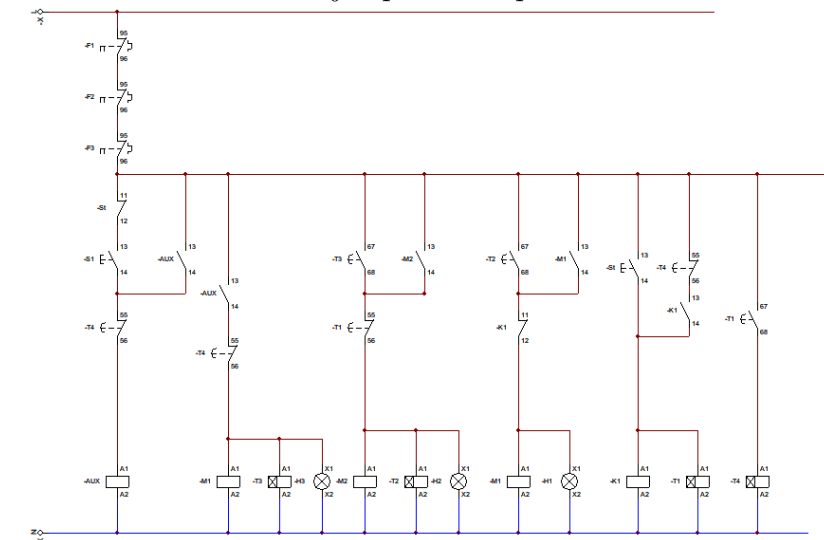


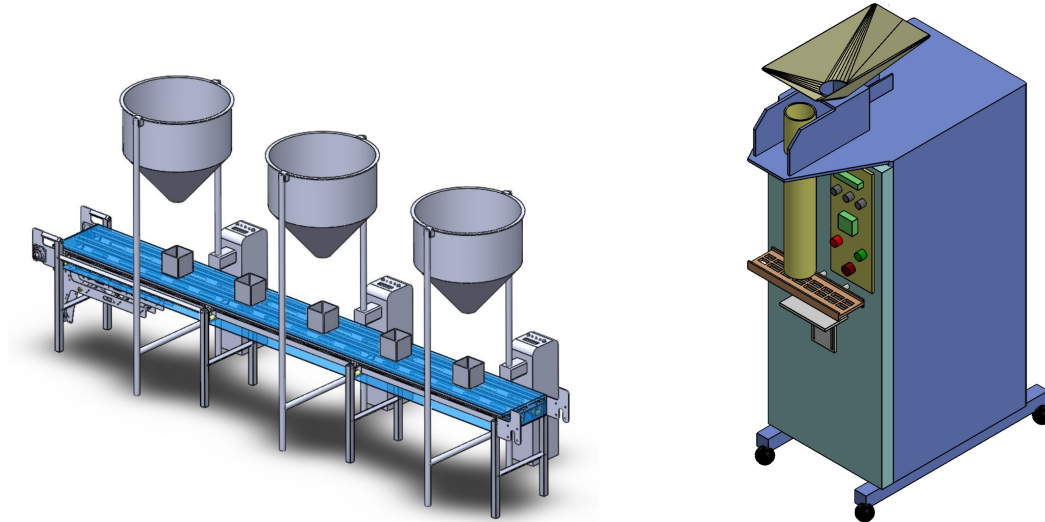
Figura 18: Circuito de control del ejemplo arranque de tres motores en cascada



2.1.3. Empacadora de productos. Se desea automatizar el proceso de una empa-
cadora de productos desde el proceso de la selección y tamaño, hasta el proceso final de

sellado del producto. El llenado del producto se determina por la acción de la gravedad y el censado de un indicador en el empaque que acciona la cortadora y selladora.

Figura 19: Empacadora de productos



2.1.3.1. Características del control • Si se acciona cualquiera de las protecciones de sobrecorrientes en la tolva (electroválvula), banda transportadora y la cortadora selladora, estos deben detenerse inmediatamente.

- Seleccionar el tipo de producto (Arroz, lentejas, arveja).
- Seleccionar el tamaño del producto (500g, 1000g).
- Tener la posibilidad de confirmar o cancelar el producto.
- Al confirmar el pedido, encender la banda transportadora y la electroválvula de la tolva del producto seleccionado.
- Para el empaque; cuando el fotodetector detecte una muesca en el empaque, la cortadora selladora se accionará y la electroválvula de la empacadora cambiará de estado dejando caer el producto dentro del empaque, al dejar de cortar y sellar, la electroválvula volverá a su normalidad.
- Si el alguno de los productos se agota, se activa una alarma que detiene el proceso.
- Llevar la cuenta de los productos empacados.

Tabla 4: Variables de control del ejemplo empacadora de productos

Dirección	Estado	Símbolo	Descripción
%I0.0	NA	Run/Stop	Estado del autómata.
%I0.1	NC	OL1	Protección de sobrecorriente de la banda transportadora.
%I0.2	NC	OL2	Protección de sobrecorriente de la tolva.
%I0.3	NC	OL3	Protección de sobrecorriente de la cortadora-selladora.
%I0.4	NA	ARROZ	Producto para ser empacado.
%I0.5	NA	ARVEJA	Producto para ser empacado.
%I0.6	NA	LENTEJAS	Producto para ser empacado.
%I0.7	NA	T500G	Tamaño del empaque.
%I0.8	NA	T1000G	Tamaño del empaque.
%I0.10	NA	CONFIRMAR	Se aprueba la selección del tamaño y producto.
%I0.11	NC	CANCELAR	Se rechaza la selección del tamaño y producto.
%I0.12	NA	IN_PROCESO	Inicio del proceso de empacado.
%I0.13		PARADA	Parada del proceso de empacado.
%I0.14	NA	B_TRANSPORTA	Energiza la banda transportadora.
%I0.15	NA	FD	Fotodetector que sensa la muesca en el empaque.
%I0.17	NA	PROD_ARROZ	Final de carrera que indica que la tolva contiene producto.
%I0.8	NA	PROD_ARVEJA	Final de carrera que indica que el empaque está puesto en la máquina.
%I0.9	NA	PROD_LENTEJAS	Final de carrera que indica que el empaque está puesto en la máquina.

%Q0.1	SALIDA_P1	Indica que el que el producto seleccionado es de 500g.
%Q0.2	SALIDA_P2	Indica que el que el producto seleccionado es de 1000g.
%Q0.3	CINTA_TRANS	Funcionamiento de la cinta transportadora.
%Q0.4	E.Valv_T1	Accionamiento eléctrico de la electroválvula de la tolva 1
%Q0.5	E.Valv_T2	Accionamiento eléctrico de la electroválvula de la tolva 2
%Q0.6	E.Valv_T3	Accionamiento eléctrico de la electroválvula de la tolva 3
%Q0.7	CORTA_SELLA	Máquina de corte y sellado que se acciona por medio de un sensor.
%Q0.8	E.Valv_EMPACA	Accionamiento eléctrico de la electroválvula ubicada en la parte superior de la empacadora.

2.1.3.2. Circuito de potencia y control del ejemplo empacadora de productos. Programa creado con el software CadeSIMU (Versión 3.0).

Figura 20: Diagrama de potencia del ejemplo empacadora de productos

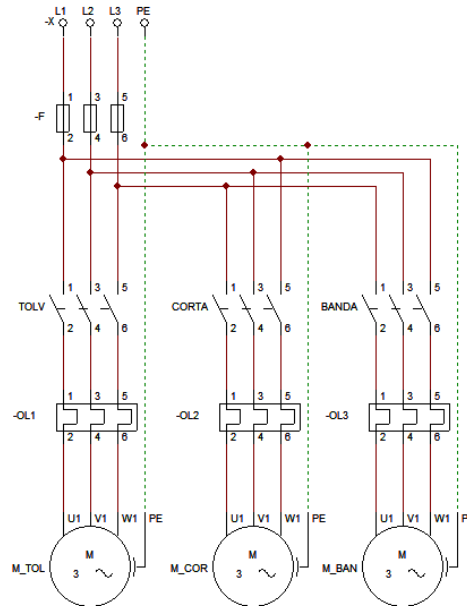
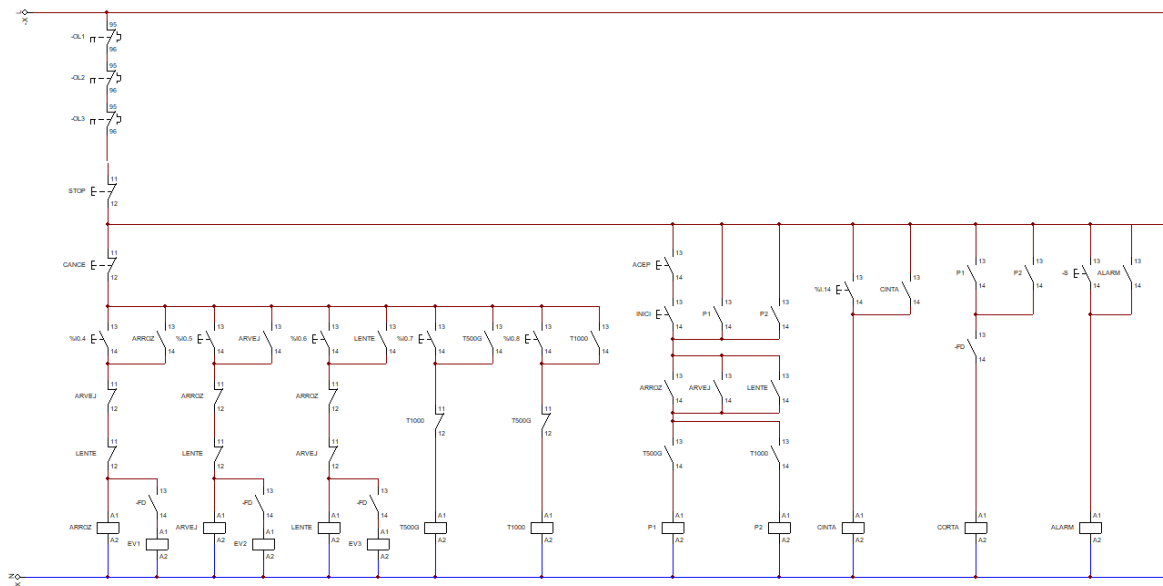


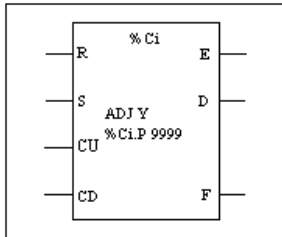
Figura 21: Diagrama de control del ejemplo empacadora de productos



2.2. BLOQUES FUNCIONALES

2.2.1. Contador progresivo/regresivo (%Ci). Este bloque funcional incrementa la cuenta en forma ascendente ó descendente según detecte un flanco positivo al bit

de entrada CU (cuenta ascendente) o CD (cuenta descendente). Se disponen de 128 contadores para su uso.



Ci.V-Palabra que muestra el valor actual

Ci.P-Palabra con la que se configura el valor a contar.

R- *Reset*: Establece el valor actual en cero ($\%Ci.V=0$).

S- *Restablecer entrada (Reset)*: Iguala el valor actual al valor preestablecido. ($\%Ci.V = \%Ci.P$).

CU- *Conteo progresivo de la entrada*: Incrementa en 1 el valor actual ($\%Ci.V$) al recibir un flanco ascendente en el bit CU.

CD-*Conteo regresivo de la entrada*: Disminuye en 1 el valor actual ($\%Ci.V$) al recibir un flanco ascendente en el bit CD.

E-*Conteo regresivo salida de desborde*: Cuando el valor actual es de 0 y recibe un flanco ascendente en el bit CD, el contador pasa de 0 a 9999 y se pone en alto el bit E ($\%Ci.E=1$).

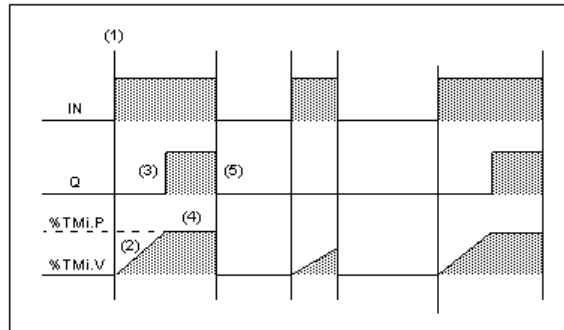
D-*Salida predeterminada alcanzada*: $\%Ci.D$ asigna un valor de 1, cuando el valor actual es igual al predeterminado. No se detiene el conteo.

F-*Conteo progresivo salida de desborde*: Cuando el valor actual es de 9999 y recibe un flanco ascendente en el bit CD, el contador pasa de 9999 a 0 y se pone en alto el bit F ($\%Ci.F=1$).

2.2.2. Temporizador ($\%TMi$). Bloque funcional que cuenta incrementos de tiempo según sea el valor base (10us , 100ms, 1s, 1m). El bloque temporizador se puede configurar de tres tipos diferente ON, OFF Y TP.

TON (*Temporizador diferido a la energización*): Se lanza al llegar un flanco positivo al bit de entrada IN, esta entrada debe permanecer alta hasta que el valor actual alcance el tiempo programado, en ese instante se activará la salida Q y permanecerá en alto hasta que el bit de entrada IN se desactive. Si el temporizador está contando y la entrada IN se desactiva, la salida Q seguirá en bajo.

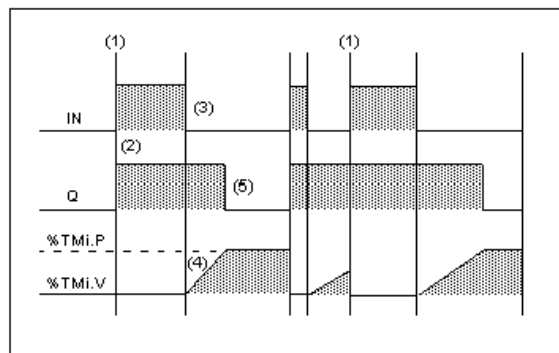
Figura 22: Diagrama de funcionamiento del temporizador TON



Fuente: AUTÓMATAS PROGRAMABLES TWIDO. Bases compactas y modulares, guía de hardware, Schneider Electric, 2011. p.491.

TOF (*Temporizador diferido a la desenergización*): La salida Q se pone alto al activarse la entrada IN y se lanza la cuenta al llegar un flanco negativo al bit de entrada IN. La salida debe permanecer alta hasta que el valor actual alcance el tiempo programado, en ese instante se desactivará la salida Q. Si el temporizador está contando y la entrada IN se activa, la salida Q seguirá en alto.

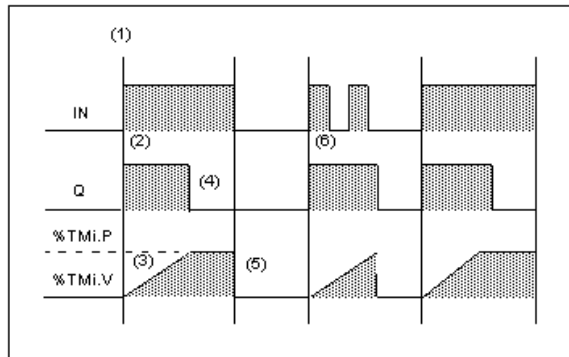
Figura 23: Diagrama de funcionamiento del temporizador TOFF



Fuente: AUTÓMATAS PROGRAMABLES TWIDO. Bases compactas y modulares, guía de hardware, Schneider Electric, 2011. p.490.

TP (*Pulso de temporizador*): El temporizador se lanza al llegar un flanco positivo al bit de entrada IN, en ese instante se activará la salida Q y permanecerá en alto hasta que el tiempo programado alcance el valor actual, en este instante la salida Q será cero. Si el temporizador está contando y la entrada IN se desactiva, la salida Q seguirá en alto.

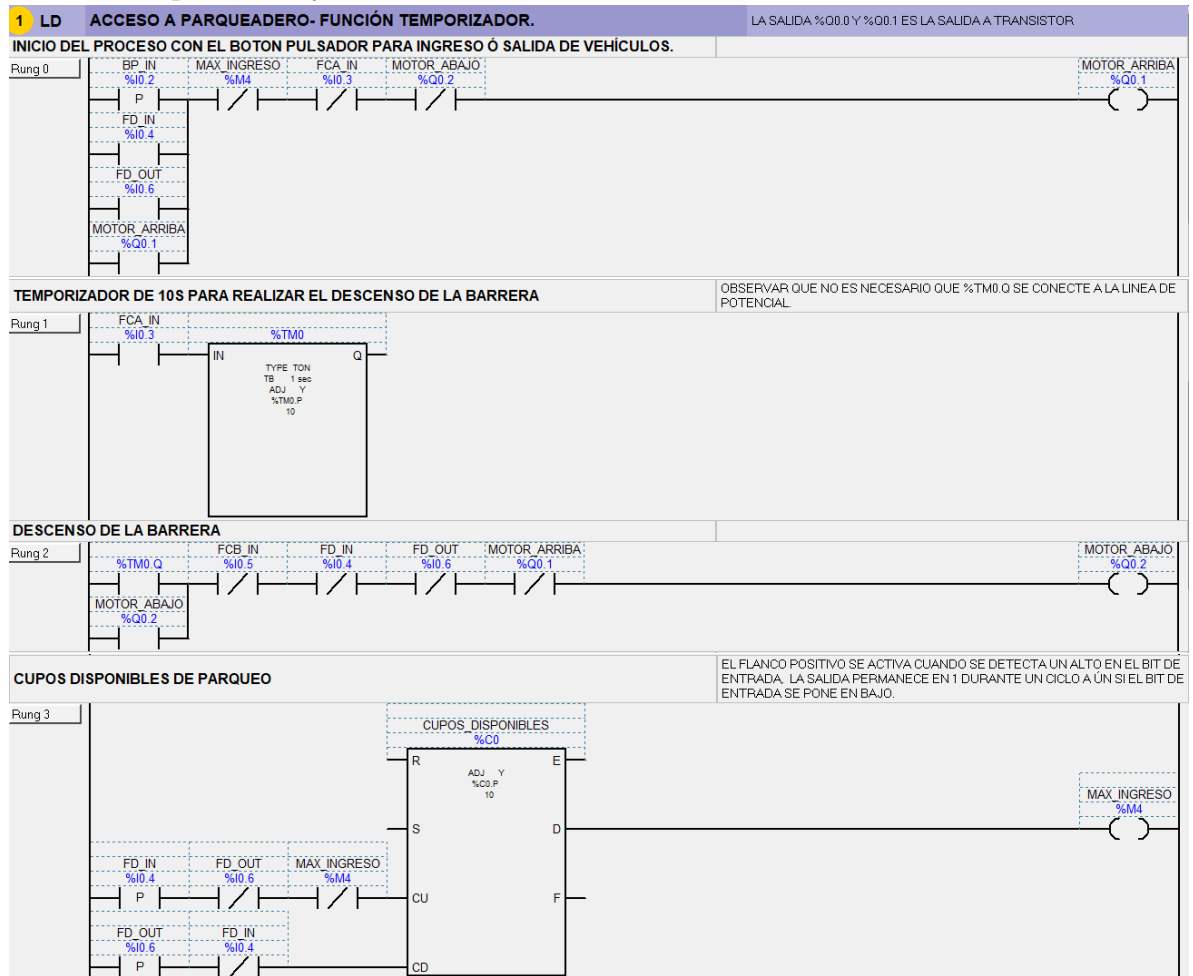
Figura 24: Diagrama de funcionamiento del temporizador TP



Fuente: AUTÓMATAS PROGRAMABLES TWIDO. Bases compactas y modulares, guía de hardware, Schneider Electric, 2011. p.492.

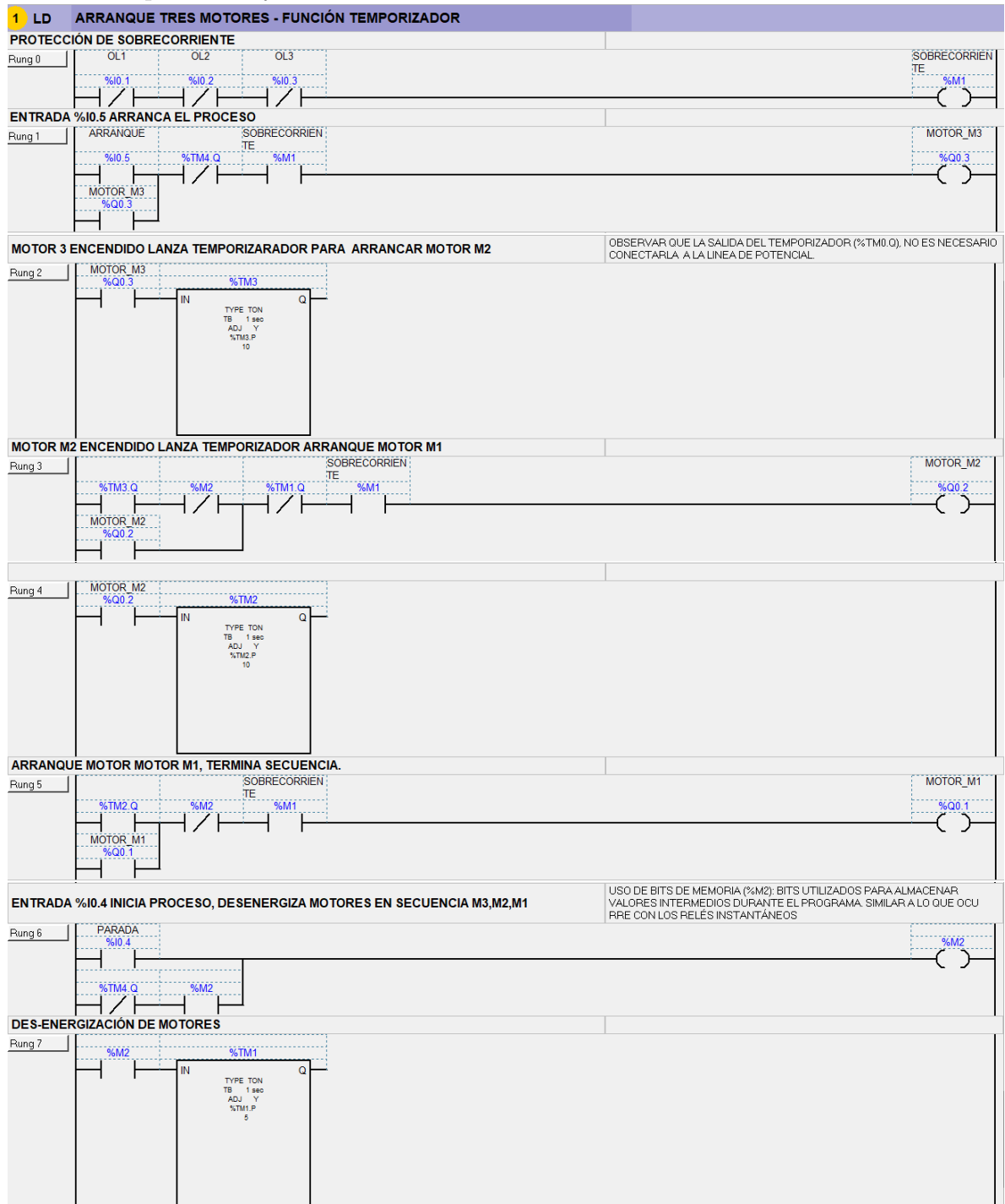
2.2.2.1. Solución de los ejemplos empleando la función contador y temporizador. Para la programación del ejemplo presentado en la figura 25, se empleó el temporizador TON que se activa al llegar un flanco positivo al bit de entrada IN. La salida del temporizador activa %Q0.2 cuando termina el conteo de 10 Seg. El contador es empleado para conocer el número de cupos disponibles en el parqueadero, cuando el contador se iguala al valor preestablecido (3), el bit de salida D se pone alto activando la memoria %M4, que impide la entrada de vehículos al parqueadero.

Figura 25: Solución del ejemplo control de acceso a parqueadero empleando la función temporizador y contador



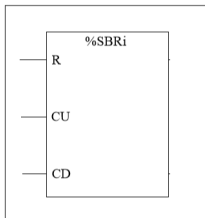
En el ejemplo de la figura 25 se emplean dos tipos de temporizadores, el TON y TOF que son empleado para energizar y desenergizar los motores, respectivamente, también es posible realizar la solución de este ejemplo empleando sólo temporizadores TON, TOF ó TP.

Figura 26: Solución del ejemplo arranque de tres motores en cascada usando la función temporizador y contador





2.2.3. Registro de bits de desplazamiento (%SBR). Es un corrimiento secuencial de bits (pueden ser altos o bajos) que se desplazan a la derecha o izquierda según se aplique un flanco ascendente al bit de entrada CU ó CD.



SBRi.j- *Bits de registro:* El número de registros disponibles son 8, de $i=0$ a 7 y los bits de registro son 16, $j=0$ a 15.

R- *Restablecer:* Un flanco positivo en el bit R “Resetea” los bits del registro, es decir; que el valor de los bits se hacen bajos ó ceros.

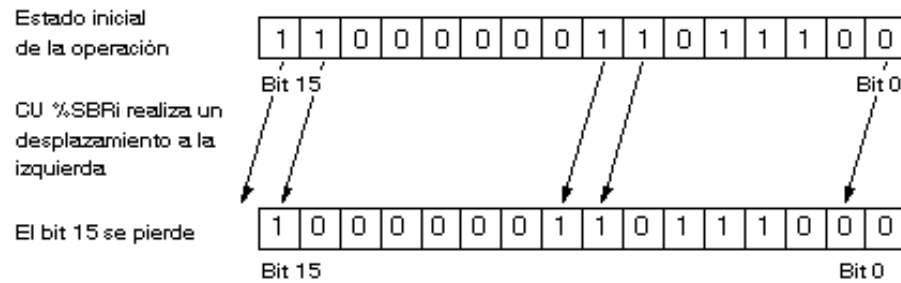
CU- *Desplazamiento de entrada a la izquierda:* Un flanco ascendente en el bit CU desplaza los bits del registro a la izquierda

una posición.

CD- *Desplazamiento de entrada a la derecha:* Un flanco ascendente en el bit CD desplaza los bits del registro a la derecha una posición.

En la figura 27 se presenta un ejemplo del estado inicial de la operación con los bits de datos cargados en las 16 posiciones, al ejecutar el registro bits de desplazamiento se realiza el corrimiento de los datos a la izquierda (CU), el dato cargado en el bit 15 al correr a la izquierda, se perderá. Cada uno de los 16 bits se pueden asignar de forma independiente con la dirección %SBRi.j.

Figura 27: Diagrama de funcionamiento de la función registro de desplazamiento



Fuente: AUTÓMATAS PROGRAMABLES TWIDO. Bases compactas y modulares, guía de hardware, Schneider Electric, 2011. pág. 416.

Figura 28: Modo de operación de la función registro de desplazamiento para el ejemplo control de acceso a parqueadero

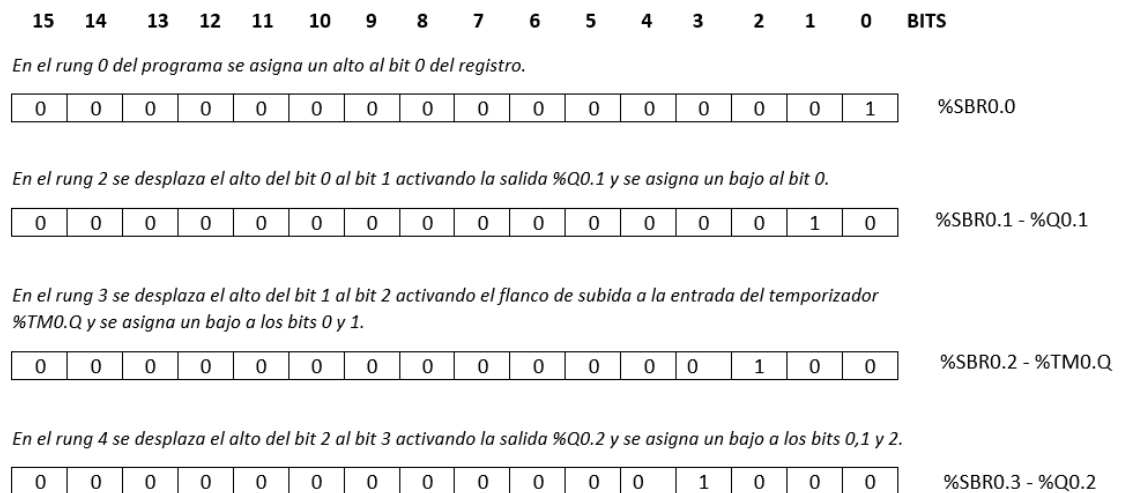
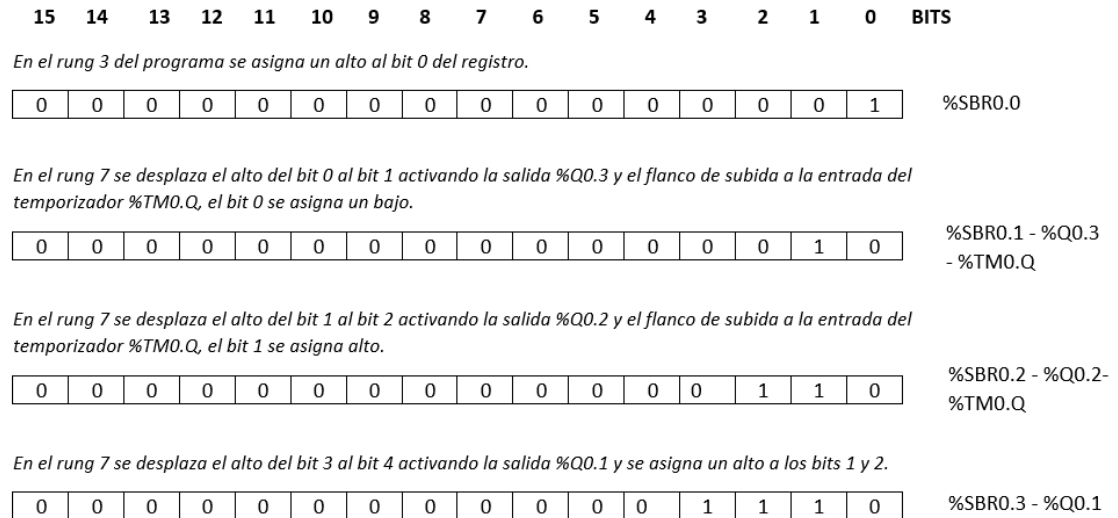
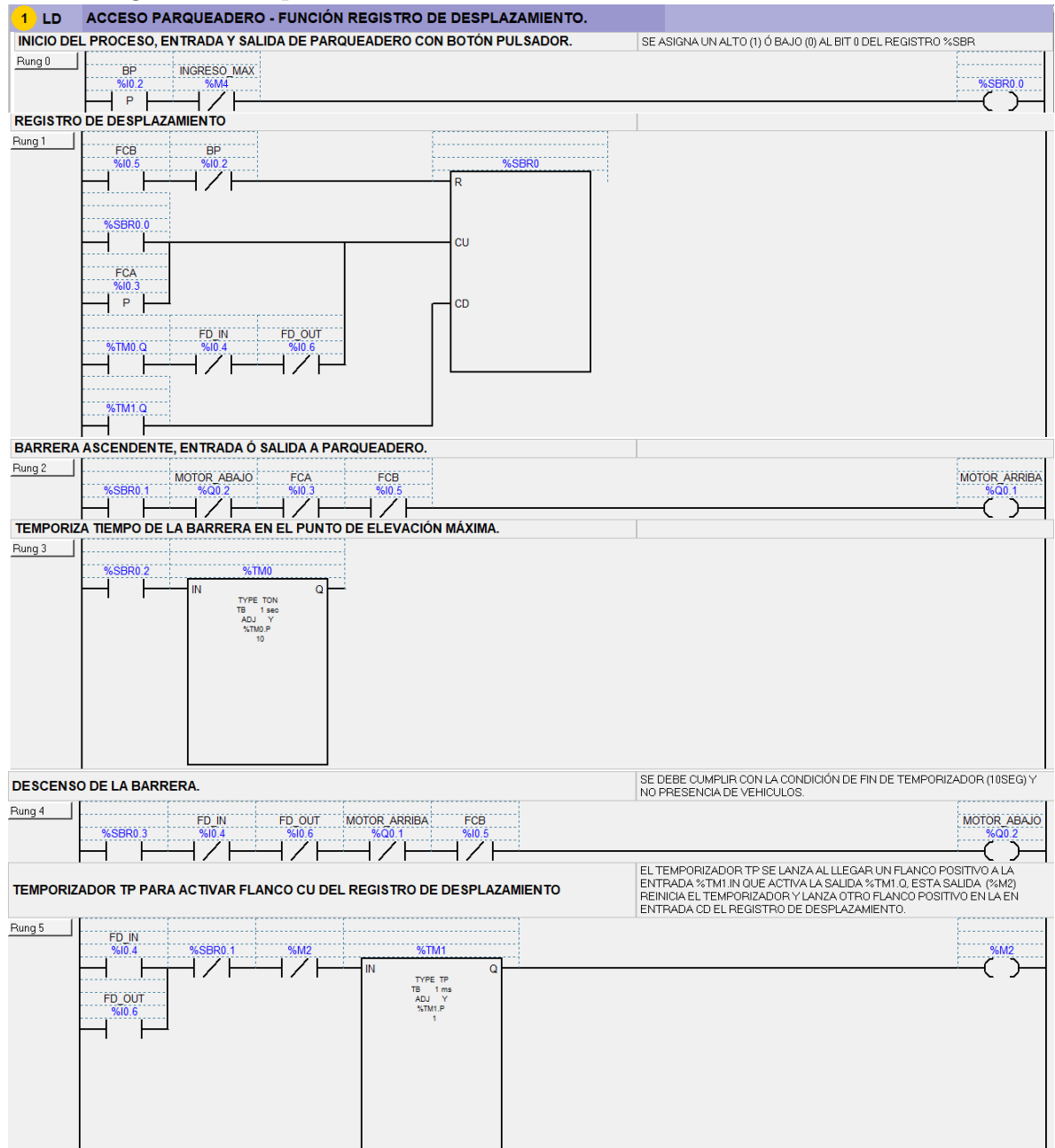


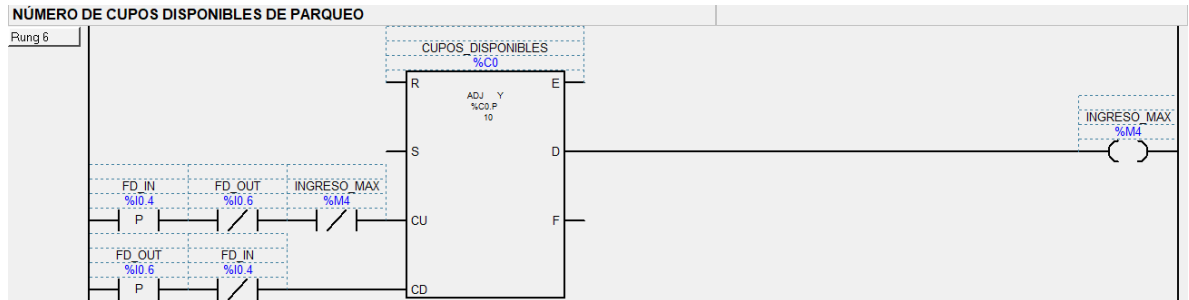
Figura 29: Modo de operación de la función registro de desplazamiento para el ejemplo arranque de tres motores en cascada



2.2.3.1. Solución de los ejemplos empleando la función registro de desplazamiento. En el rung 5 de la figura 30 se presenta la activación del temporizador TP cuando se pulsa la entrada %I0.4 ó %I0.6 que corresponde a los fotodetectores de entrada y salida, el uso del temporizador TP con una base de tiempo de 1ms es empleado para realizar dos corrimientos a la derecha desde el bit 3 al bit 1 en el registro de desplazamiento, el corto tiempo del temporizador se debe a que no se quiere activar el bit 2 y sólo el bit 1 pero la función no permite realizar saltos en los bits.

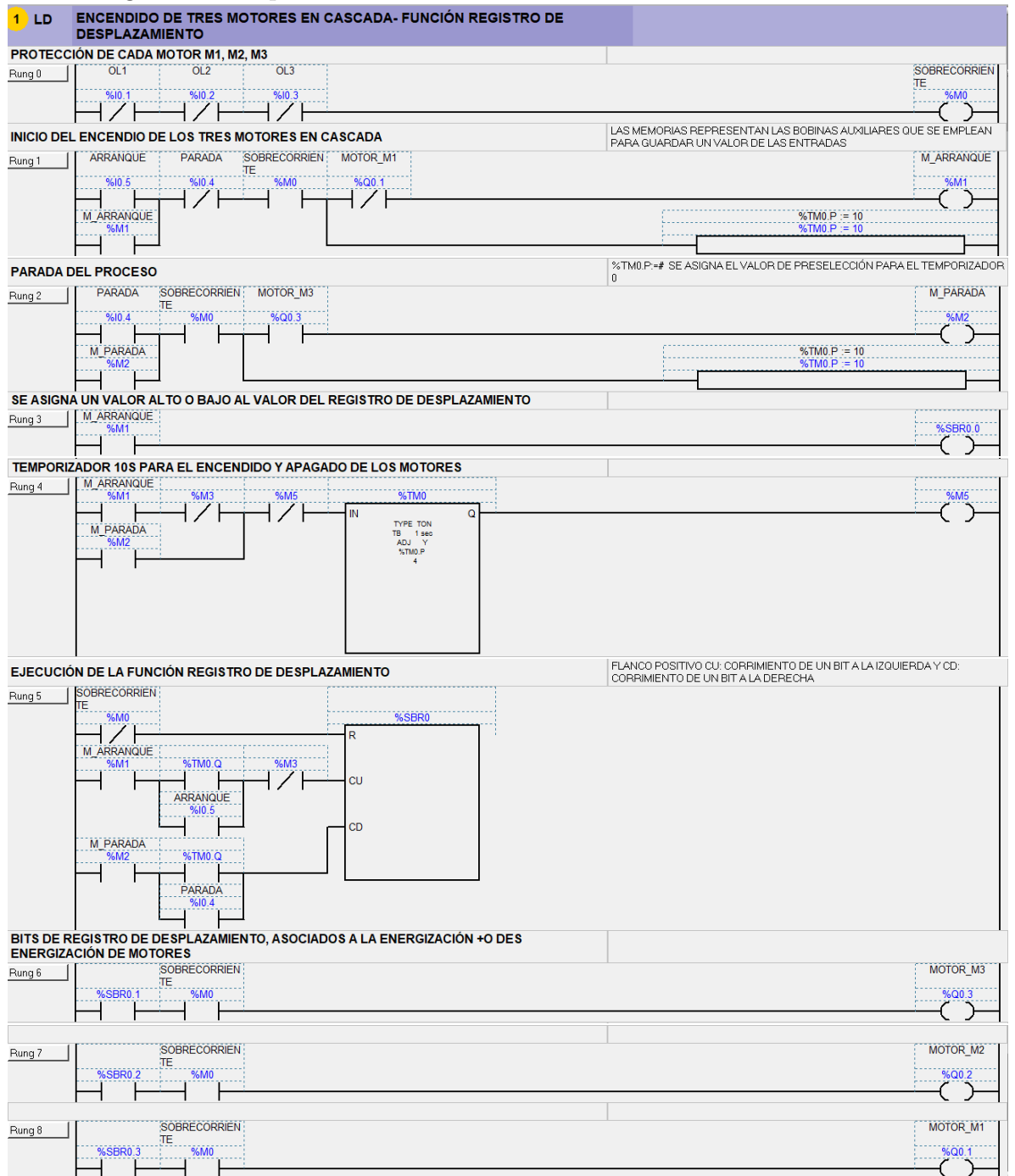
Figura 30: Solución del ejemplo control de acceso a parqueadero empleando la función registro de desplazamiento





Para el ejemplo del arranque de tres motores en cascada se empleó la función registro de desplazamiento, esta función facilita la programación ya que el proceso del encendido y apagado de los motores es secuencial (Figura 29). El modo de operación del registro, desplaza bits a la derecha o izquierda sin realizar saltos, para ello, se emplea la entrada CU para activar el encendido de los motores y CD para el apagado de los motores como se presenta en la figura 31.

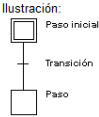



Figura 31: Solución del ejemplo arranque de tres motores en cascada empleando la función registro de desplazamiento



2.2.4. GRAFCET. El diagrama de control con etapas y transiciones es un modelo de representación gráfica que permite descomponer un proceso en estados ó etapas y condiciones (transiciones) para ir de un estado a otro, en cada estado se puede realizar una acción del proceso. El software TwidoSuite no admite la programación en modo gráfico.

Las instrucciones y objetos necesarios para programar un diagrama GRAFCET se presentan en la figura 32.

Figura 32: Instrucciones para la programación empleando GRAFCET

Representación gráfica (1)	Transcripción en el lenguaje de TwidoSuite	Función
Ilustración: 	=* i	Comenzar el paso inicial (2).
	# i	Activar el paso i tras desactivar el paso actual.
	.* i	Iniciar el paso i y validar la transición asociada (2).
	#	Desactivar el paso actual sin activar ningún otro paso.
	#Di	Desactivar el paso i y el paso actual.
	== POST	Iniciar procesamiento posterior y finalizar procesamiento secuencial.
	%Xi	Se puede comprobar y escribir el bit asociado con el paso i (el número máximo de pasos depende del autómata).
 Xi	LD %Xi, LDN %Xi AND %Xi, ANDN %Xi, OR %Xi, ORN %Xi XOR %Xi, XORN %Xi	Comprobar actividad del paso i.
 Xi	S %Xi	Activar el paso i.
 Xi	R %Xi	Desactivar el paso i.

Fuente: CONTROLADORES PROGRAMABLES TWIDO. Guía de referencia de software: Telemecanique, Schneider Electric, 2011, pág. 367.

2.2.4.1. Diagrama GRAFCET. El diagrama GRAFCET constan de tres etapas:

1. Pre-proceso: No es obligatorio. Prepara variables a utilizar en el proceso.
2. Proceso: Describe mediante acciones y transiciones el proceso.
3. Post-proceso: Desarrolla las acciones a realizar en cada etapa.

Figura 33: Diagrama GRAFCET del ejemplo control de acceso a parqueadero

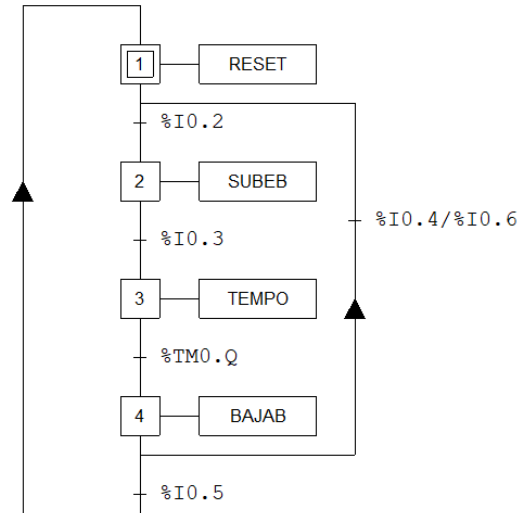


Figura 34: Diagrama GRAFCET del ejemplo control secuencial de tres motores en cascada

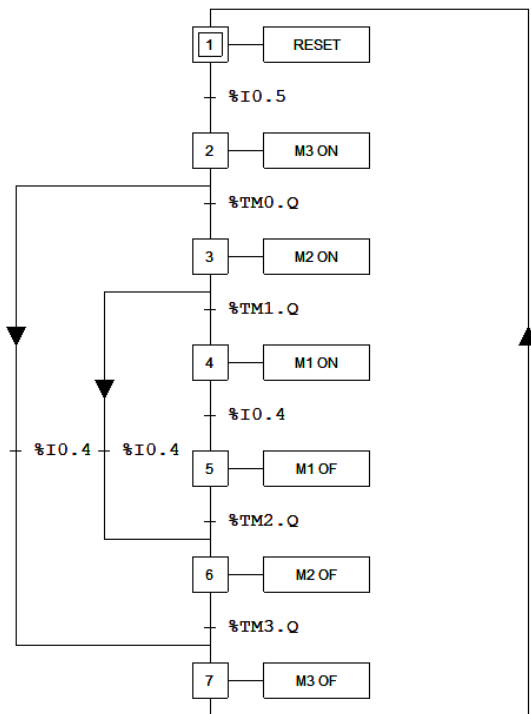
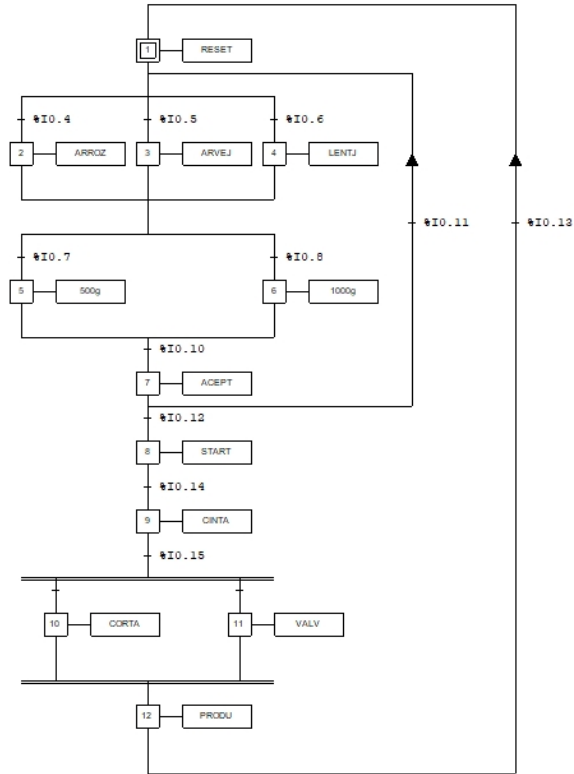
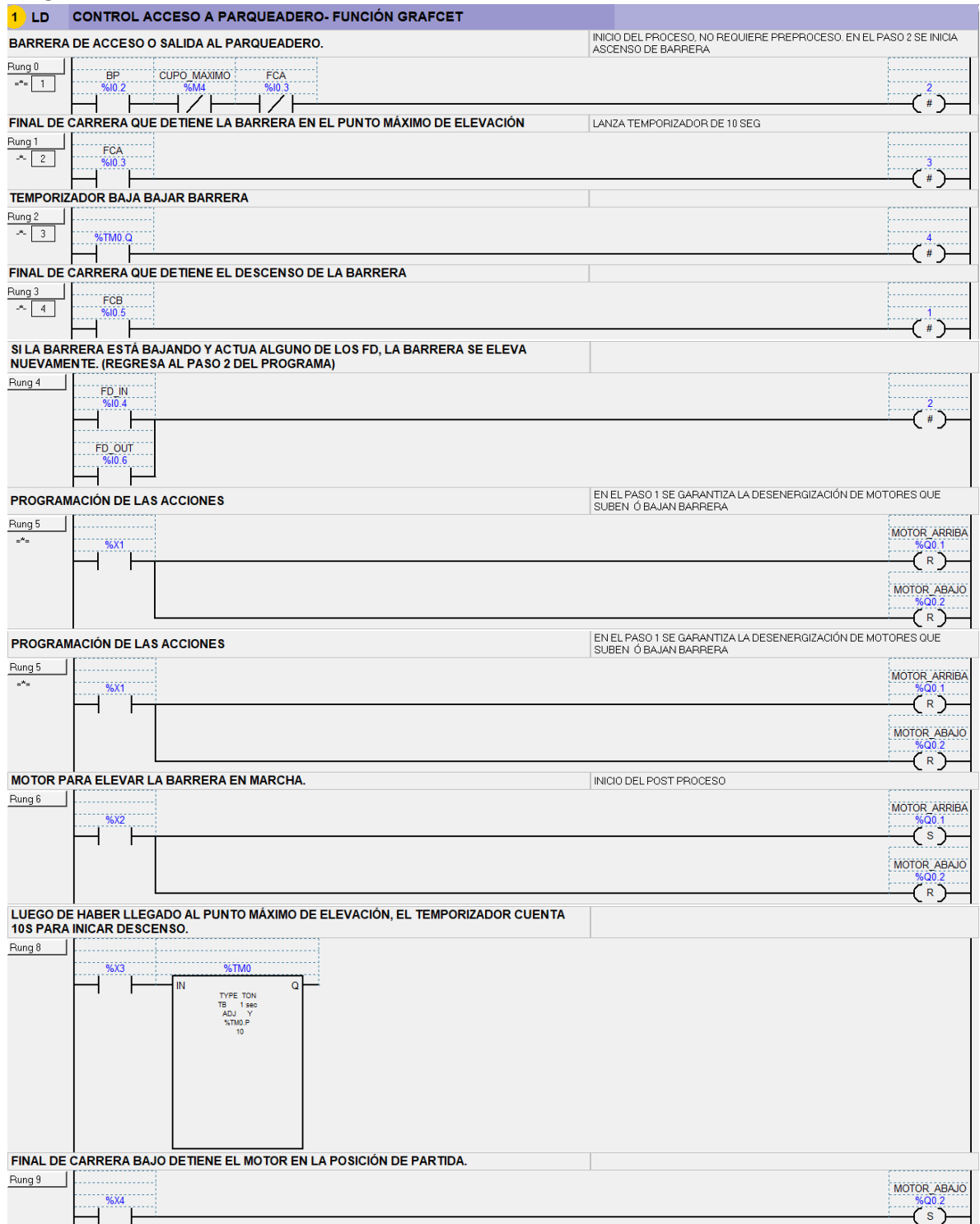


Figura 35: Diagrama GRAFCET del ejemplo empaedora de productos

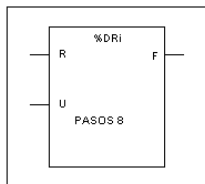
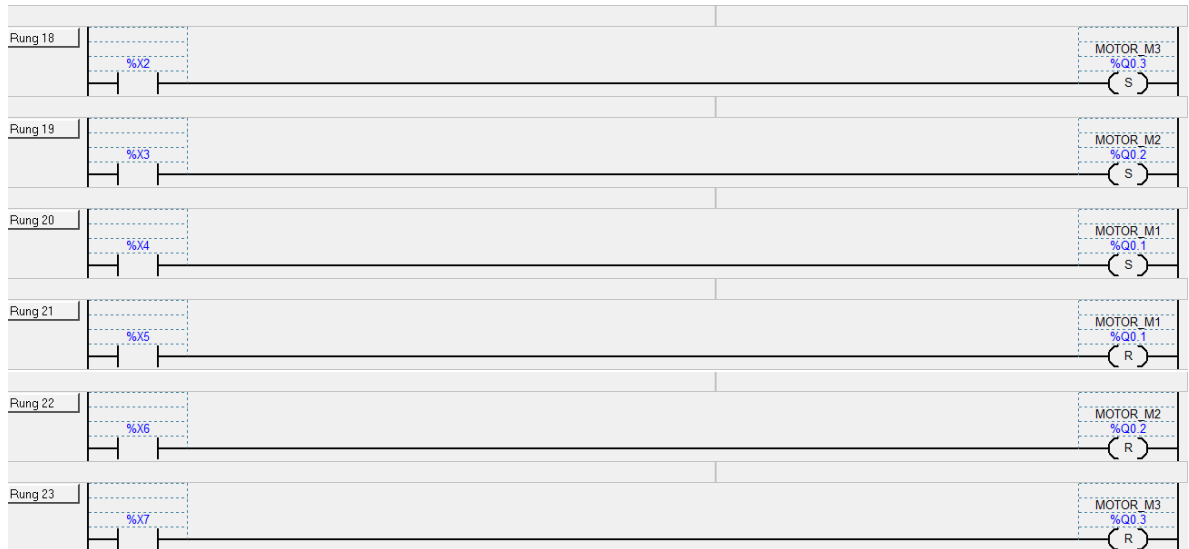


2.2.4.2. Solución de los ejemplos empleando el diagrama GRAFCET. El Preproceso (que consiste en preparar algunas condiciones o variables a utilizar en el proceso) no siempre es necesario programarlo, como en el presente ejemplo.

Figura 36: Solución del ejemplo control de acceso a parqueadero empleando el diagrama GRAFCET







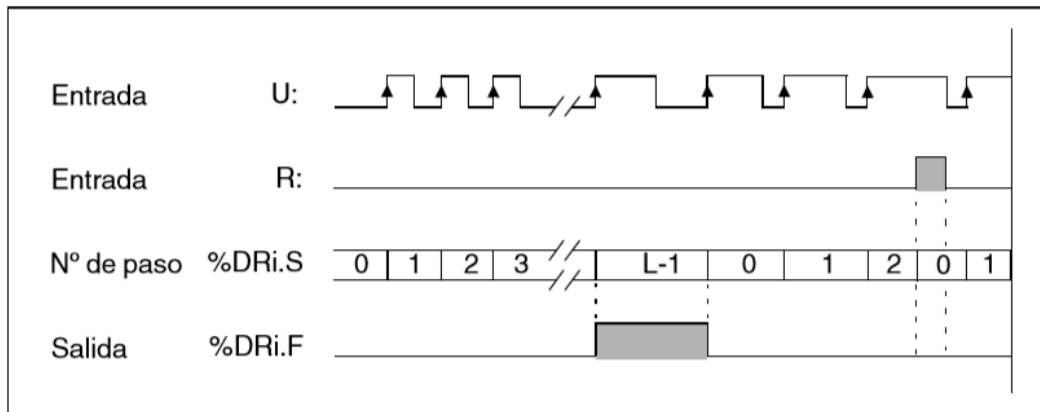
2.2.5. DRUM (%DRi) Este bloque funcional opera bajo el principio de un tambor electromecánico (DRUM), su funcionamiento está asociado al cambio de posición de acuerdo a eventos externos. Las muescas altas y bajas en el tambor electromecánico corresponden a un alto (1) ó un bajo (0) en el DRUM programado del PLC, respectivamente.

Figura 38: Posición de pines en el tambor electromecánico (DRUM)



Al realizar doble clic en el bloque %DR, se abre una ventana con una matriz de datos organizada en ocho pasos (0 a 7) y distribuidos en columnas numeradas de (0 a 16) enumeradas de 0 a F. Se asigna a la lista de bits de control una salida configurada ó a una palabra de memoria. En el transcurso del paso actual, los bits de control adquieren los estados binarios definidos para este paso como se muestra en la figura 39.

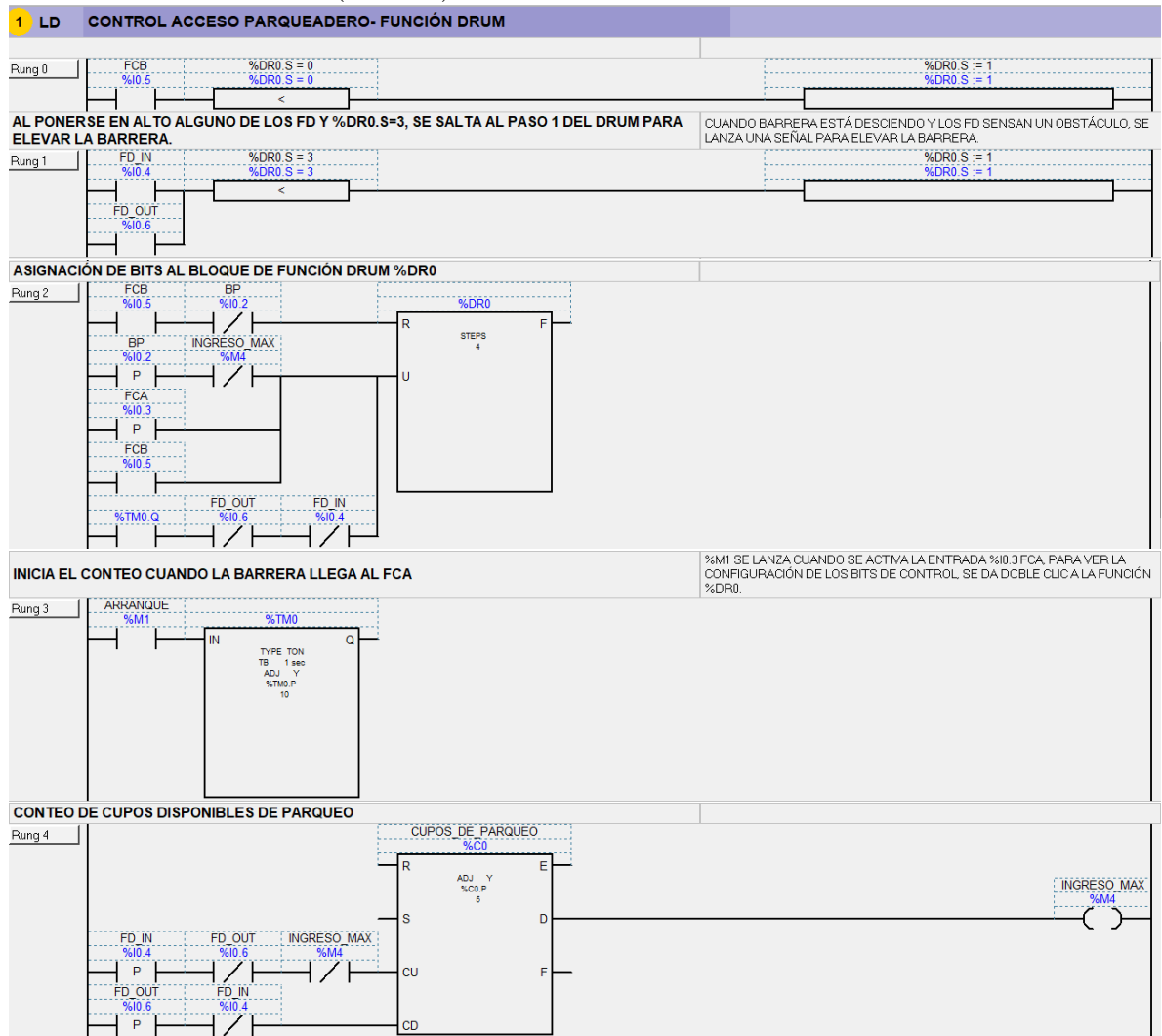
Figura 39: Posición de pines en el tambor electromecánico (DRUM)



Fuente: AUTÓMATAS PROGRAMABLES TWIDO. Bases compactas y modulares, guía de hardware: Schneider Electric, 2011, pág. 471.

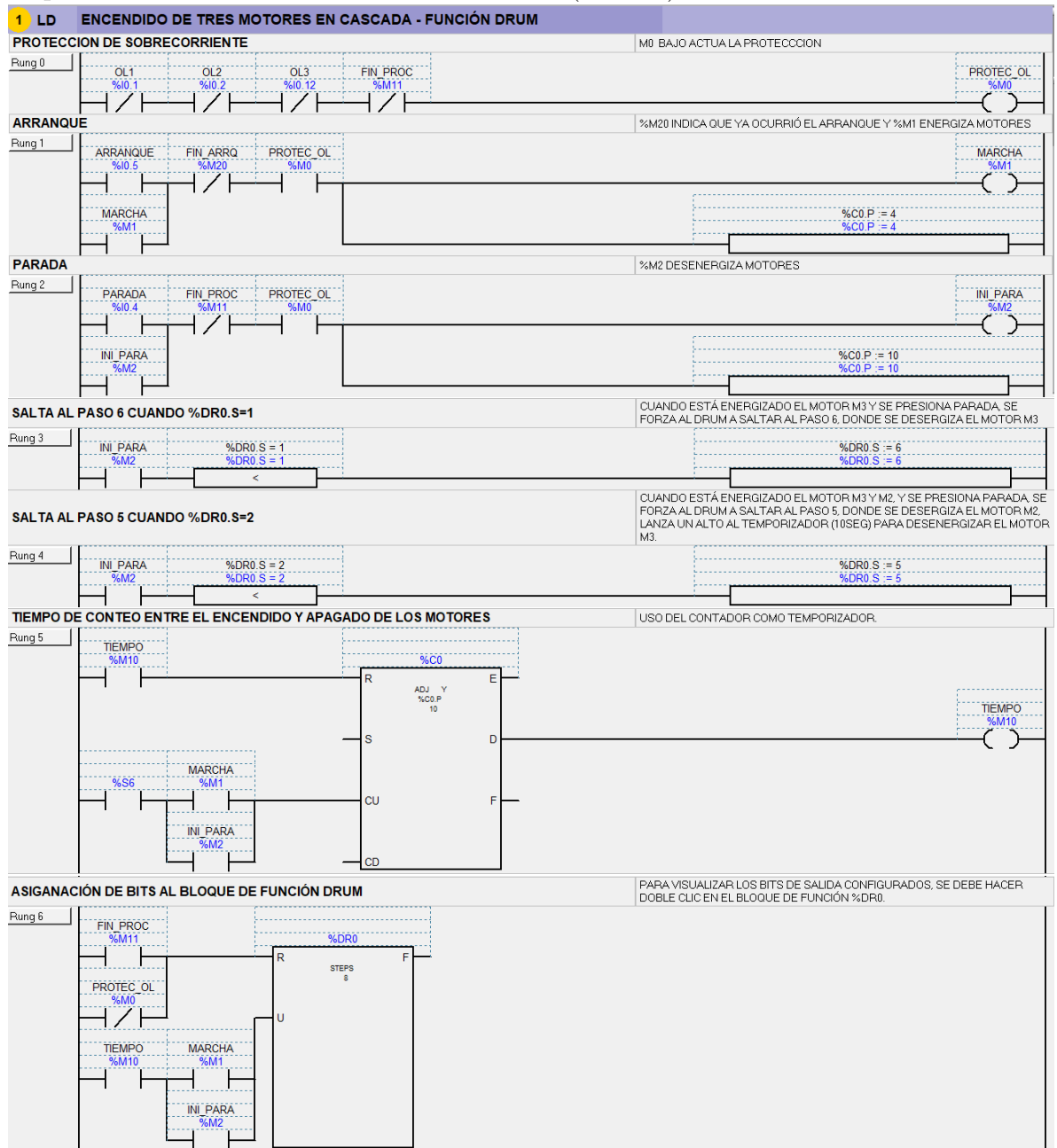
2.2.5.1. Solución de los ejemplos empleando la función DRUM. Para la solución del ejemplo propuesto, se emplean los bloques de comparación y de operación para realizar saltos de pasos en el bloque de función DRUM %DR. Además, este bloque permite visualizar el número del paso actual durante la simulación y ejecución del programa.

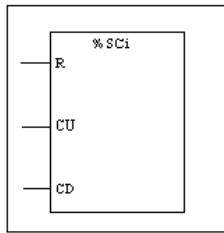
Figura 40: Solución del control de acceso a parqueadero empleando la función conmutador del tambor (DRUM)



En la Figura 41 se emplea un contador como temporizador empleando la palabra del sistema %S6 que funciona como un reloj interno con una base de tiempo de 1Seg.

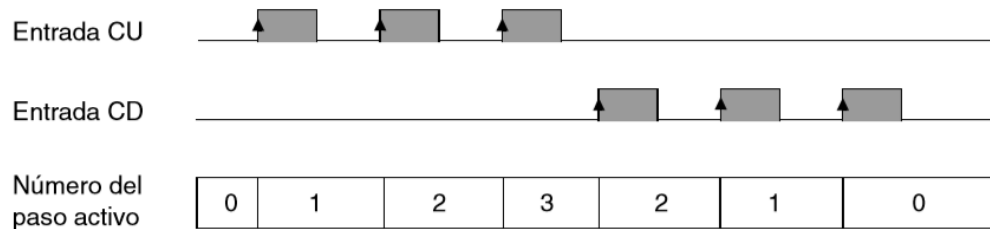
Figura 41: Solución del ejemplo control secuencial de tres motores en cascada empleando la función conmutador del tambor (DRUM)





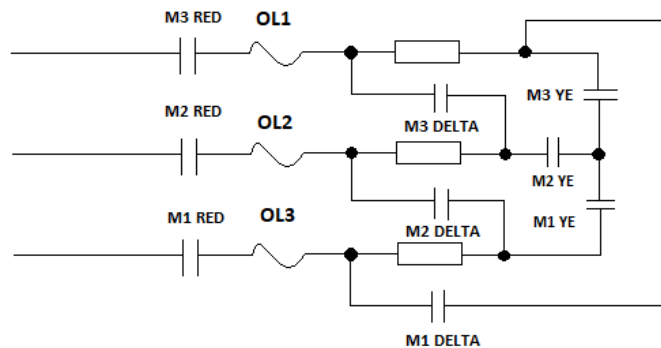
2.2.6. Contador de pasos. El bloque de función del contador de pasos (%SCi) permite realizar una serie de pasos que pueden tomar un alto (1) o bajo (0) y pueden ser asignados a acciones. El incremento o decremento del paso depende de si se incrementa ó se activa un flanco positivo al bit de entrada CU ó CD. Es importante recordar que cada vez que un paso esté activo, el bit asociado (bit de contador de pasos %SCi.j) se establece en 1. El contador de pasos puede controlar los bits de salida (%Qi.j), los bits internos (%Mi) pero sólo se puede activar un paso de un contador de pasos cada vez.

Figura 42: Funcionamiento del bloque contador de pasos



Fuente: SCHNEIDER ELECTRIC, Telemecanique. Autómatas programables TWIDO. Bases compactas y modulares, guía de hardware, 2011, pág. 418.

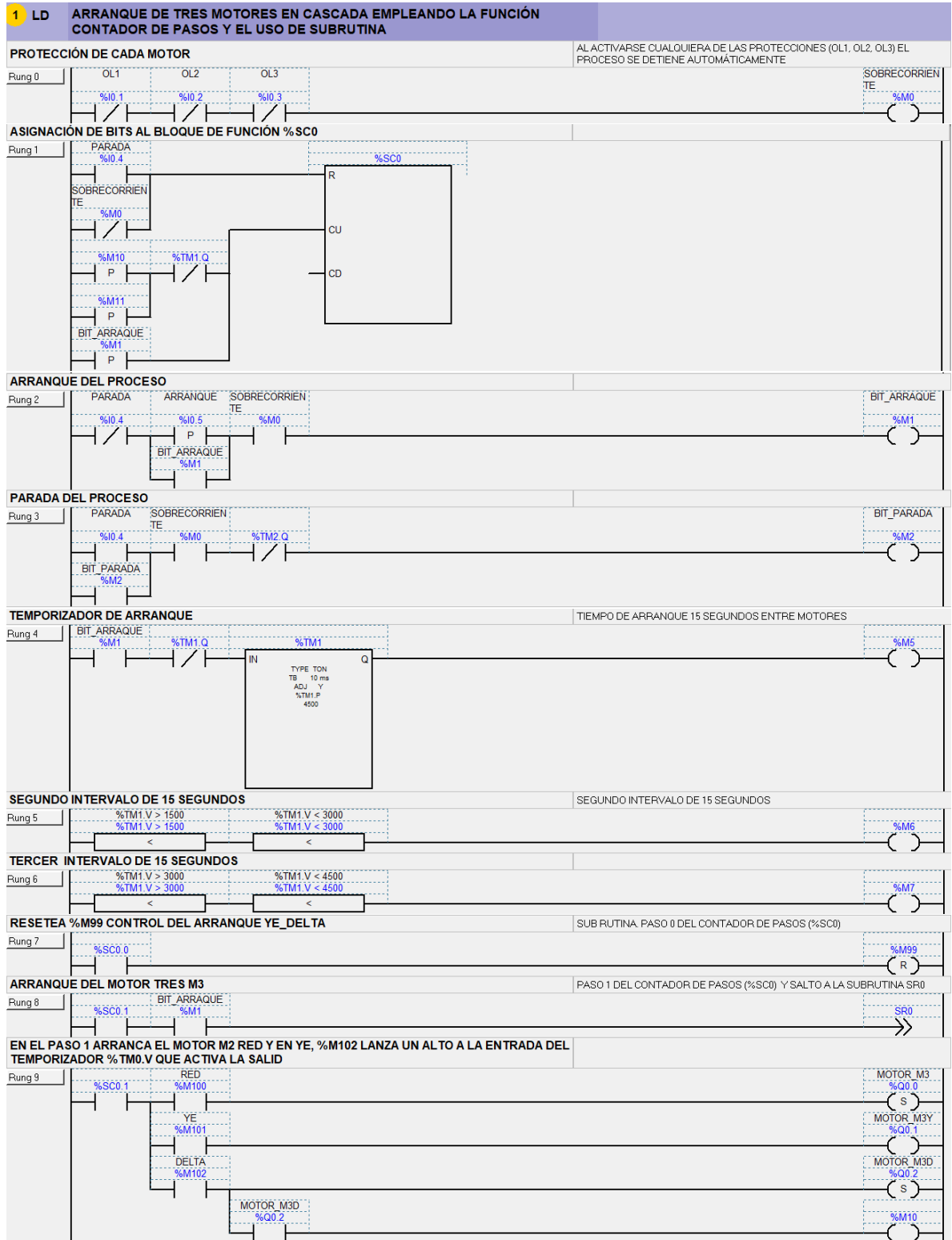
Figura 43: Dibujo explicativo del ejemplo arranque estrella/triángulo de tres motores

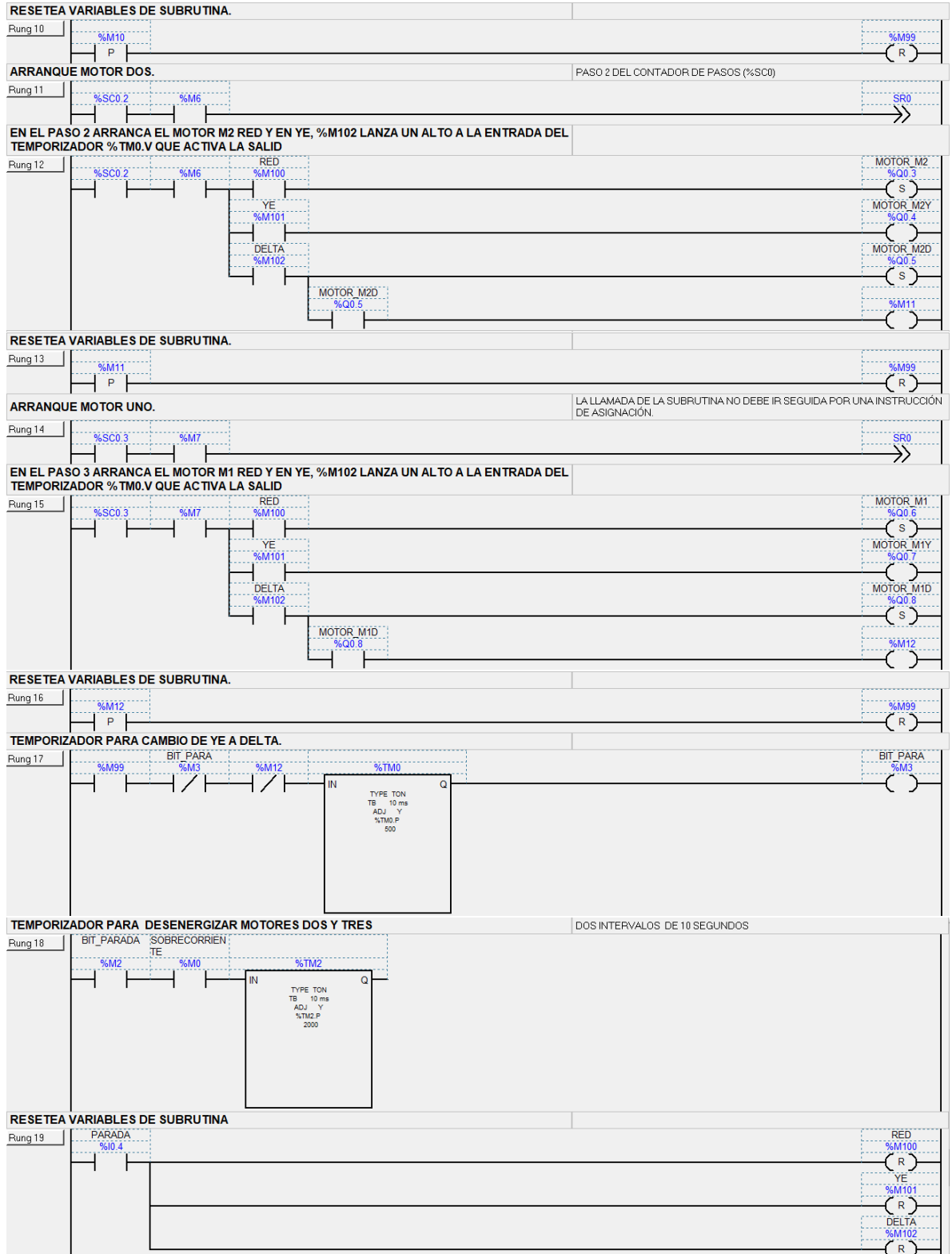


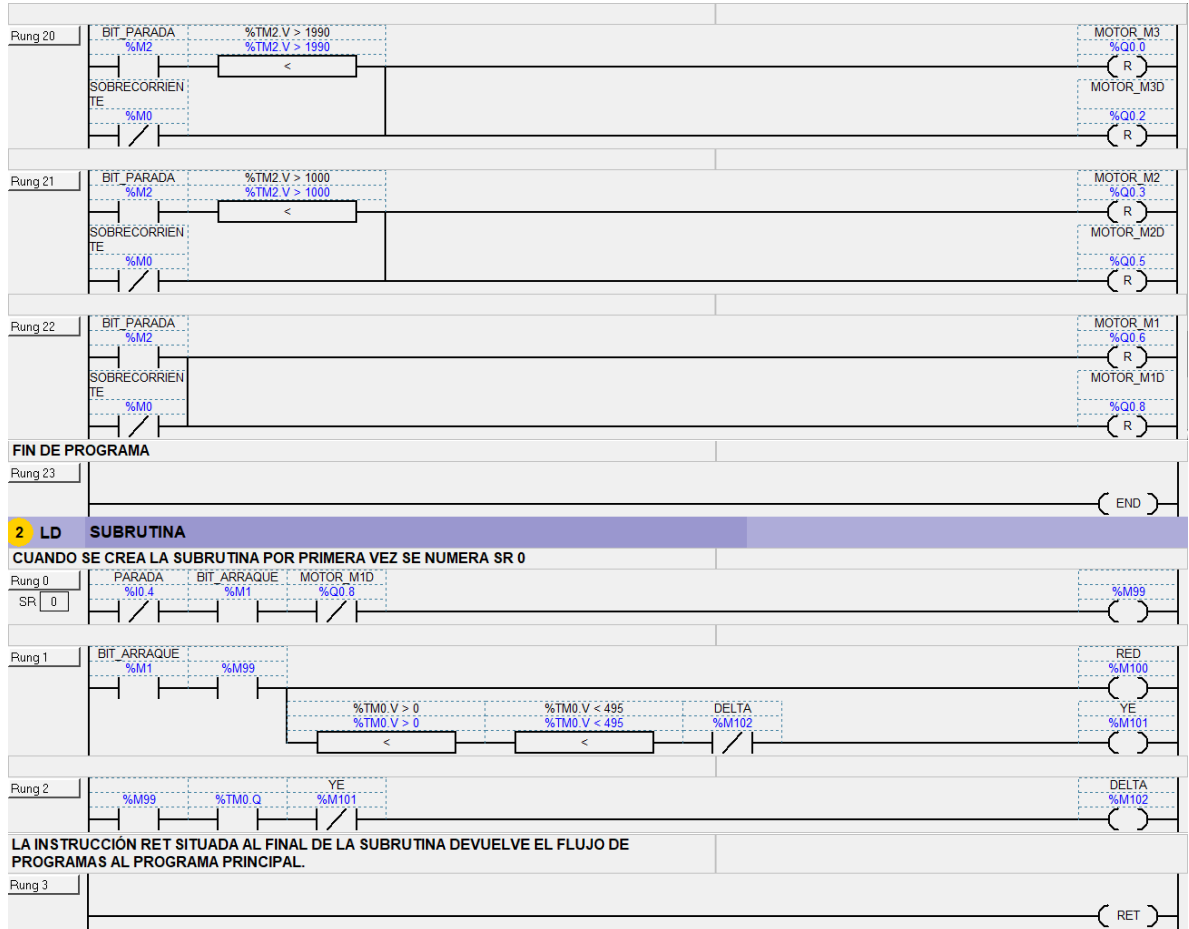
2.2.6.1. Solución del ejemplo arranque estrella/triángulo de tres motores que se energizan en cascada M3, M2, M1. El uso de subrutinas es empleado para ejecutar una serie de pasos dentro del programa principal. Para el ejemplo aplicado, se

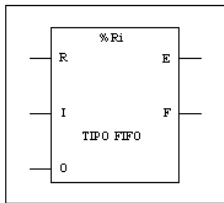
programó el temporizador de Red-YE a RED-DELTA en la subrutina, la eficiencia es que sólo se programa una vez el proceso y el programa puede ingresar a la subrutina ejecutando la acción cuando el programador lo considere.

Figura 44: Solución del arranque de tres motores en cascada empleando la función contador de pasos



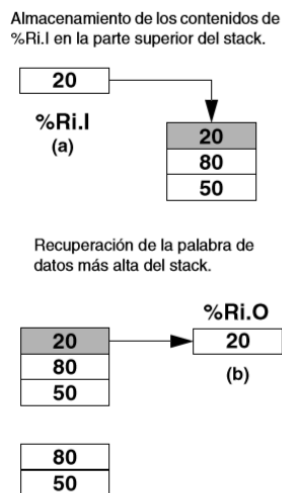






2.2.7. FIFO (%Ri) Al recibir un flanco ascendente al bit de entrada I, se guarda la palabra que está denota como $\%Ri.I := \text{“Valor a guardar”}$ en la parte superior de la cola, al recibir un flanco ascendente en el registro $\%Ri.O$ es sacado de la parte inferior de la cola de datos desplazando una posición.

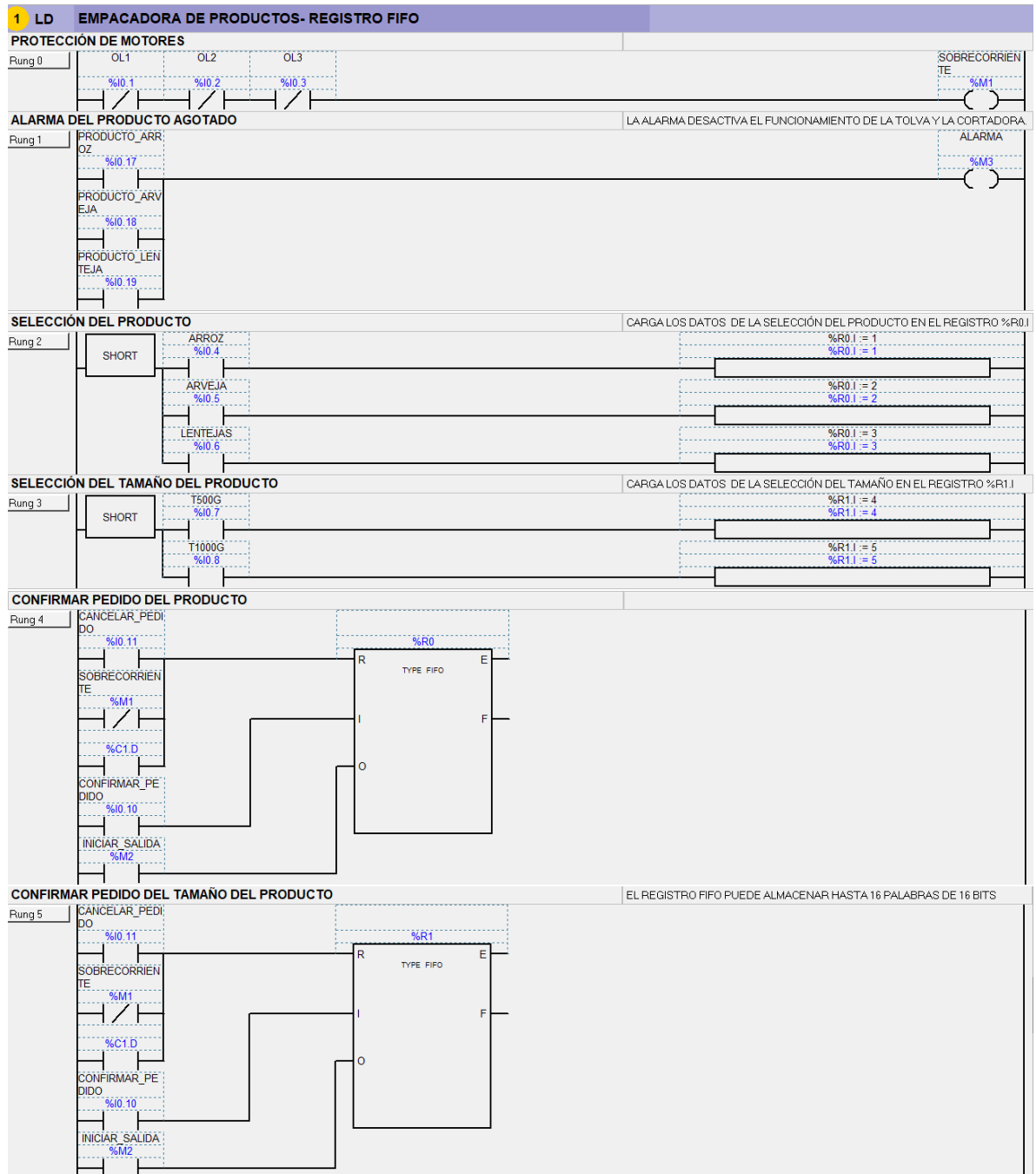
Figura 45: Funcionamiento del bloque FIFO

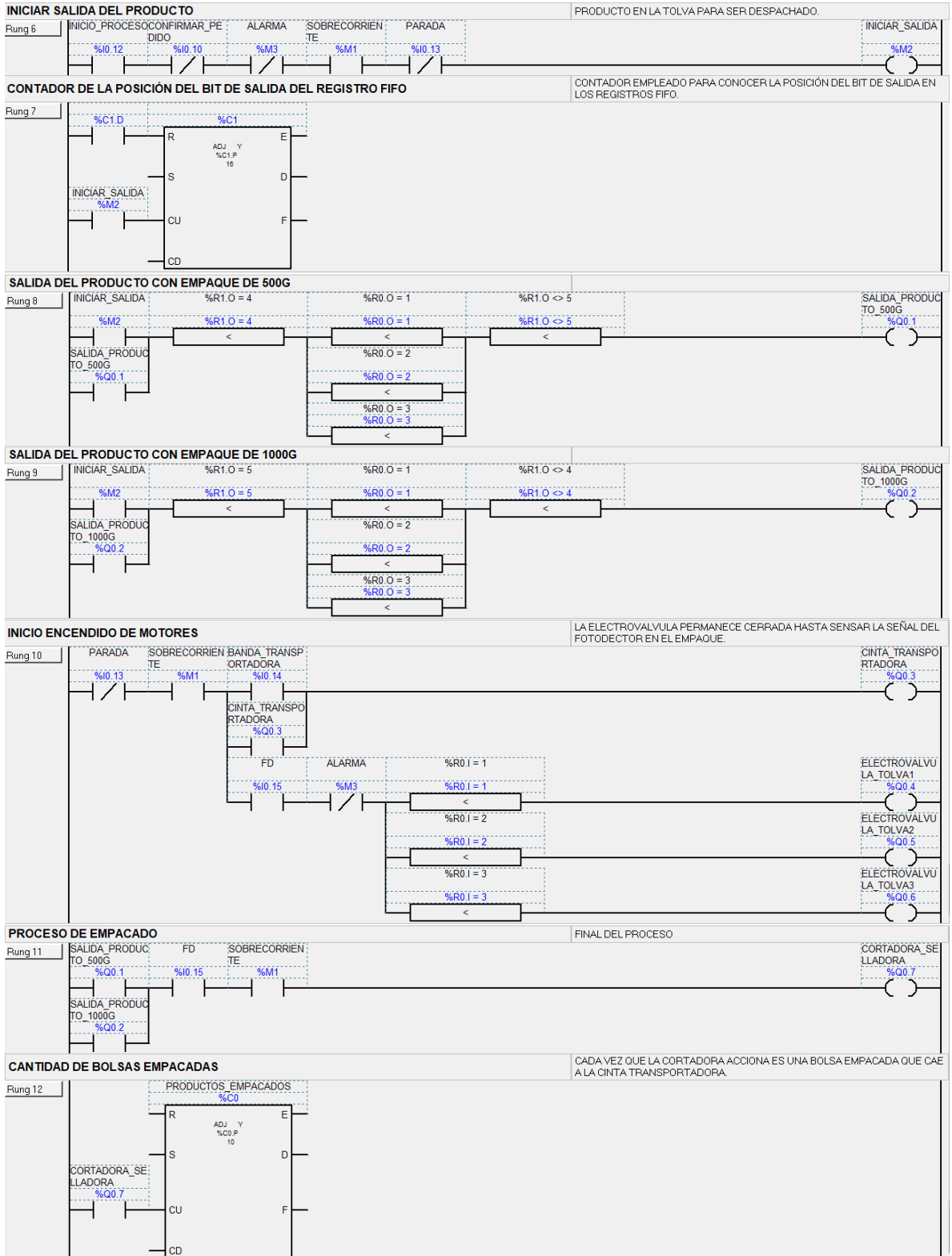


Fuente: AUTÓMATAS PROGRAMABLES TWIDO. Bases compactas y modulares, guía de hardware: Schneider Electric, 2011, pág. 457.

2.2.7.1. Solución del ejemplo empacadora de productos. Esta función guarda palabras de memoria hasta 16 bits, se pueden tomar varios pedidos antes de iniciar el proceso y con sólo ejecutar el programa una vez se pueden obtener varios productos empacados.

Figura 46: Solución del ejemplo empacadora de productos empleando la función FIFO





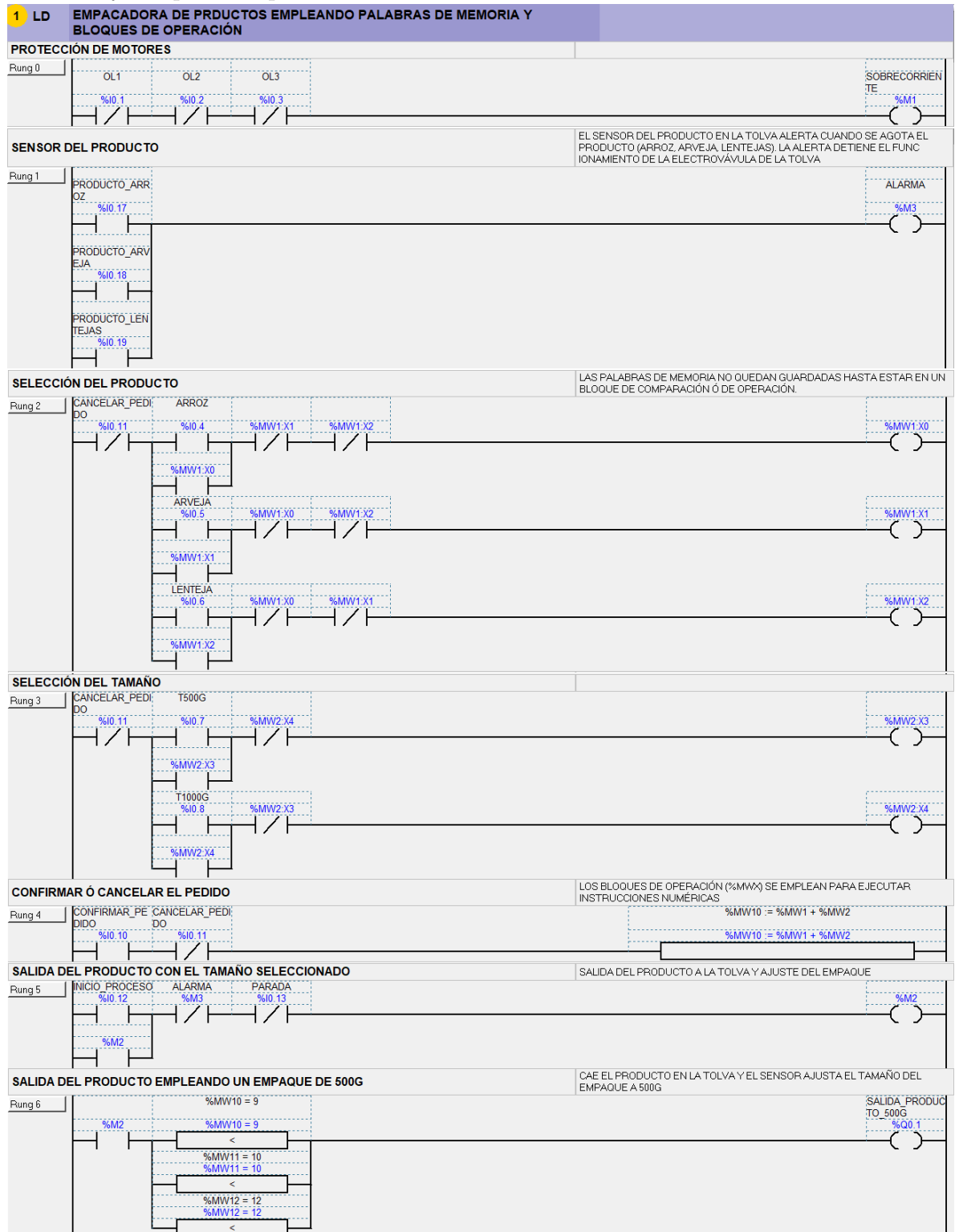
2.2.8. Palabras de memoria y bloques de operación.

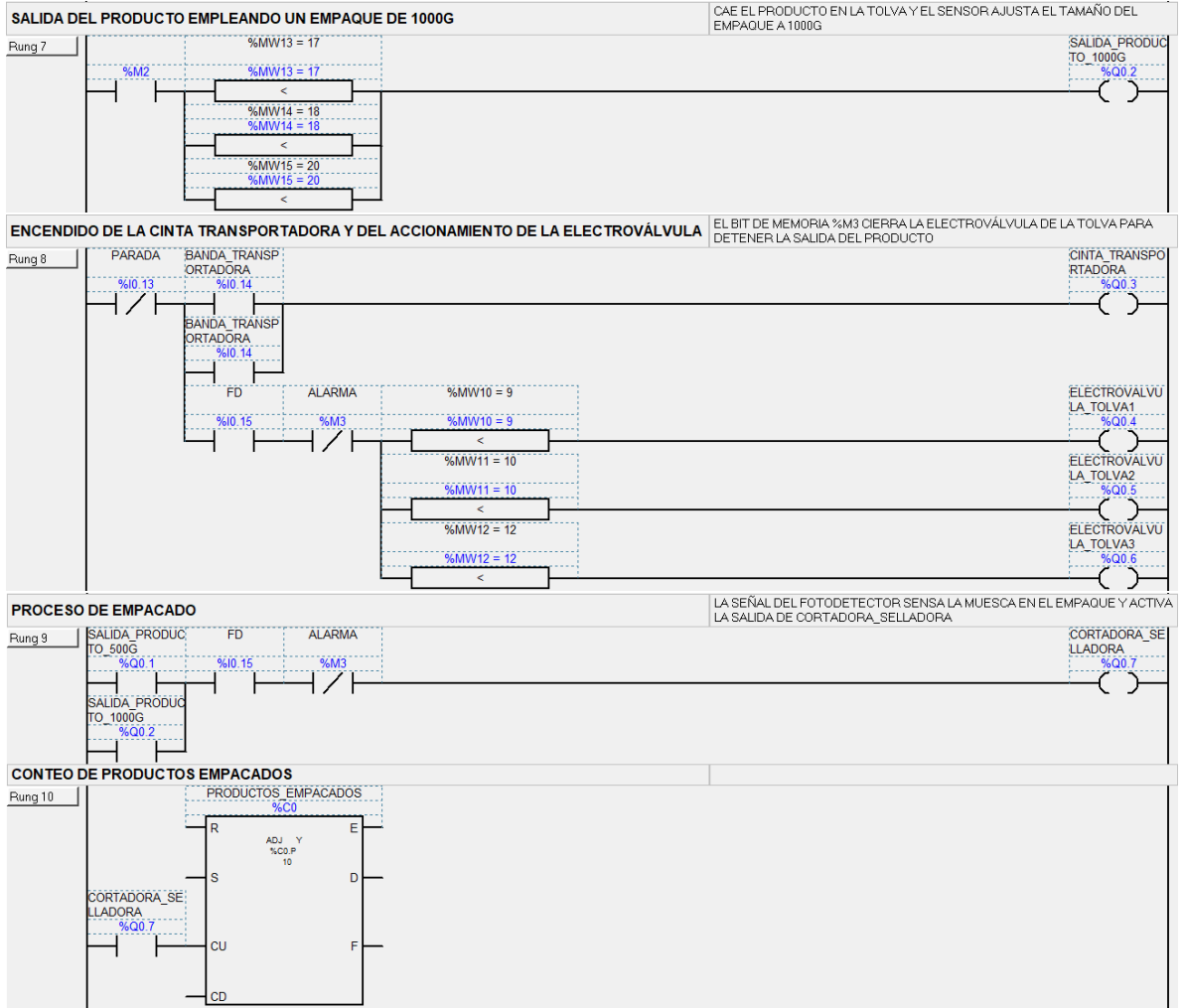
2.2.8.1. Palabras de memoria. Son palabras de 16 bits de memoria que se emplean para almacenar variables requeridas por el programador, estas palabras se pueden leer y escribir durante la ejecución del programa.

2.2.8.2. Bloques de operación. El software permite realizar operaciones lógicas y aritméticas empleando las palabras de memoria y palabras de sistema.

Las palabras de memoria son empleadas para simplificar la programación del programa, se le puede asignar a una misma entrada la misma palabra con un bit diferente, esto ahorra la memoria de datos en el PLC cuando se requiere de una programación extensa.

Figura 47: Solución del ejemplo empaedora de productos empleando palabras de memoria y bloques de operación





3. COMUNICACIÓN MODBUS ENTRE EL PLC TWIDO Y EL VARIADOR DE VELOCIDAD ALTIVAR PARA EL CONTROL DE VELOCIDAD DE UN MOTOR TRIFÁSICO

Modbus es un protocolo de comunicación maestro/esclavo, el maestro envía solicitudes que realiza el esclavo y este responde de forma individual. Un maestro puede enviar, leer ó escribir datos en los registros de cada esclavo, eso se debe a que cada esclavo tiene una dirección única asignada. Este protocolo es muy usado en la industria con los equipos Controladores Lógicos Programables más conocidos como PLC's.

3.1. EQUIPOS EMPLEADOS PARA LA COMUNICACIÓN

3.1.1. PLC Twido TWDLCAE40DRF. Se seleccionó el PLC Twido de referencia TWDLCAE40DRF de marca Schneider Electric. Este autómata tiene un puerto ethernet, un puerto terminal EIA RS485 integrado con conector miniDIN hembra de 8 pines y una fuente de alimentación interna propia de 24Vdc. La tensión de red nominal que admite este PLC es de 100 a 240 Vca.

Figura 48: PLC Twido TWDLCAE40DRF



Fuente: Product data sheet, TWDLCAE40DRF, Schneider electric

3.1.2. Motor de inducción trifásico. Motor de inducción trifásico, tensión nominal 127V/220V (delta/Ye), potencia de 1.5kW y velocidad nominal de 1200 rpm. Los bornes del motor conectados en Y se conectan a los bornes U1,V1,W1 del variador de velocidad, equipo disponible en el laboratorio de máquinas eléctricas de la escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones (E3T).

Figura 49: Motor de inducción trifásico



3.1.3. Variador de velocidad ALTIVAR 31. Variador de velocidad trifásico ALTI-VAR 31 de la marca schneider electric de 0.5HP a 240 Vac. Este equipo cuenta con los protocolos de comunicación industrial modbus y CANopen, para el control de velocidad del motor trifásico se empleó el protocolo de comunicación modbus. Equipo disponible en el almacén de la escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones (E3T). El motor trabaja en vacío con el propósito de no sobrecargar el variador.

Figura 50: Variador de velocidad ALTIVAR 31



Fuente: SCHNEIDER ELECTRIC, Telemecanique. Guía de instalación, variador ALTIVAR 31.

3.1.4. Equipo de cómputo (PC). Se empleó un computador con un sistema operativo windows 10, procesador intel core i3 de 7th generación. En la sección 1.1 se presentan los requisitos mínimos que debe tener el equipo de computo para emplear el software TwidoSuite.

3.2. CONEXIÓN ENTRE EL PLC TWIDO, VARIADOR DE VELOCIDAD ALTIVAR 31 Y EQUIPO DE CÓMPUTO

3.2.1. Conexión física entre el PLC Twido y el equipo de cómputo. La conexión para la comunicación modbus se realizó entre el puerto Ethernet con conector

RJ45 del PLC Twido y el puerto Ethernet con conector RJ45 del PC utilizando un cable UTP. Si el computador no tiene el puerto RJ45 es posible realizar la conexión con un adaptador de Ethernet a USB.

Figura 51: Cable UTP con terminales RJ45



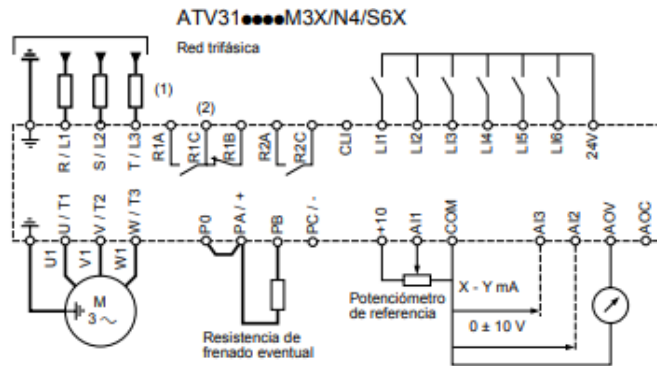
3.2.2. Conexión física entre el variador y el PLC Twido: La conexión física entre el variador y el PLC se realizó a través del puerto Ethernet con conector RJ45 del variador de velocidad al puerto terminal EIA RS485 del PLC Twido, empleando el cable TSXCRJMD25 como se muestra en la figura 52

Figura 52: Cable TSXCRJMD25



3.2.3. Conexión física entre el variador de velocidad y el motor de inducción trifásico. En la figura 53 se presenta el esquema de conexión del variador de velocidad. Se conectaron los bornes U1,V1,W1 a los bornes de conexión del motor de inducción trifásico y se energizó el variador a una tensión de 220V en los bornes R/L1 y S/L2. Los bornes L11, L12, L13, L14, L15, L16 son las entradas lógicas que permiten controlar el variador, se alimentan a una tensión de 24Vdc. Este equipo tiene dos entradas analógicas, la AI2 que es la entrada analógica de tensión y la AI3 que es la entrada analógica de corriente, las salidas de corriente y tensión se muestran como AOC y AOV respectivamente. (Figura 53)

Figura 53: Esquema de conexión variador de velocidad ALTIVAR 31



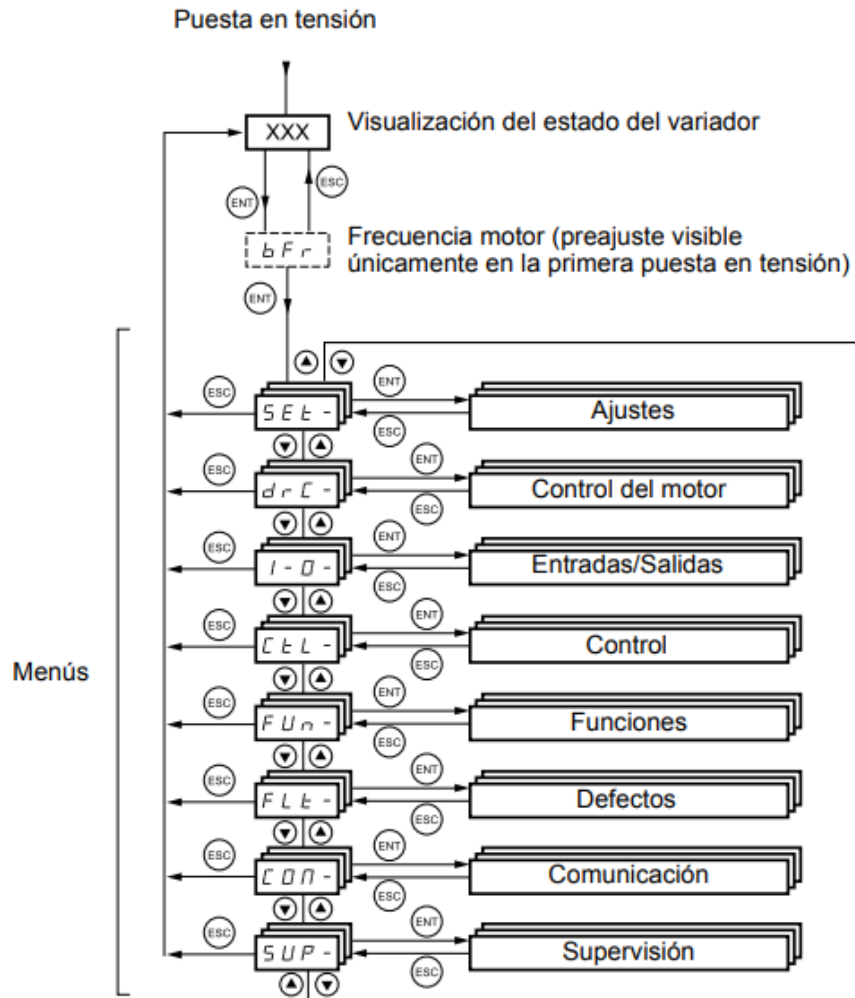
Fuente: SCHNEIDER ELECTRIC, Telemecanique. Guía de instalación ALTIVAR 31, pág. 12.

3.3. CONFIGURACIÓN DEL VARIADOR DE VELOCIDAD ALTIVAR 31

El variador de velocidad es el esclavo configurado para recibir y responder a las solicitudes del maestro PLC Twido a través del protocolo de comunicación modbus.

En el variador de velocidad se visualiza, por medio de una pantalla, 5 ajustes de menú (ver figura 54) de los cuales se explicarán: el menú de control (CtL), el menú de control del motor (drC) y el menú de entradas y salidas (I-0), que son necesarios configurar para la comunicación modbus en el variador.

Figura 54: Acceso a los menús ALTIVAR 31



Fuente: SCHNEIDER ELECTRIC, Telemecanique. Guía de programación ALTIVAR 31, pág. 12.

A continuación se explicarán sólo los comandos modificados de los menús del ALTIVAR 31 para la realización de la comunicación Modbus con el PLC. En esta sección sólo se presentarán los items configurados; las demás opciones se pueden ver en el manual de programación ALTIVAR 31, Telemecanique.

3.3.1. Menú de control del motor (drC). • bFR- Frecuencia estándar del motor

El variador de velocidad está configurado a una frecuencia de fábrica de 50 Hz por lo tanto se ajusta a 60 Hz.

50 Hz: IEC

60 Hz: NEMA

- **UnS- Tensión nominal del motor**

Se configura la tensión nominal que aparece en la placa de características del motor, para el caso del motor trifásico empleado, la tensión nominal es de 220V en conexión YE.

- **Frs- Frecuencia nominal del motor que aparece en la placa de características**

Se configura la frecuencia nominal que aparece en la placa de características del motor, para el caso del motor trifásico empleado, la frecuencia es de 60 Hz.

Si se a configurado inicialmente el parámetro bFr a una frecuencia 60Hz, se ajusta este parámetro (Frs) a 60 Hz, el ajuste de fábrica es de 50Hz.

- **nCr- Corriente nominal del motor, se obtiene de la placa de datos nominales del motor**

Corriente nominal del motor $I_n = 6.94$ A para conexión Y.

- **nSP- Velocidad nominal del motor que aparece en la placa de características**

Si la placa de características no indica la velocidad nominal, sino la velocidad de sincronismo, y el deslizamiento en Hz o en %, la velocidad nominal debe calcularse de la siguiente forma:

$$Velocidad\ nominal = \frac{Velocidad\ de\ sincronismo \times 100 - deslizamiento\ en\ \%}{100}$$

La velocidad nominal del motor según la placa de características es de 1200 RPM.

- **UFt- Elección del tipo de ley tensión/frecuencia, que indica la variación de velocidad en el motor.**

nLd: Ahorro energético, para aplicaciones de par variable sin necesidades dinámicas importantes.

L: Par constante para motores en paralelo o motores especiales.

P: Par variable: aplicaciones de bombas y ventiladores.

n: Control vectorial del flujo sin captador para aplicaciones de par constante. (SCHNEIDER ELECTRIC, Telemecanique. Guía de programación ALTIVAR 31, pág. 24.)

Se selecciona la opción nLd, ya que el motor trabajará en vacío y se modificarán los niveles de tensión y frecuencia para el ejemplo propuesto.

- **nrd- Frecuencia de corte aleatoria**

La modulación de frecuencia aleatoria evita los posibles ruidos de resonancia que pueden producirse con una frecuencia fija. (SCHNEIDER ELECTRIC, Telemecanique. Guía de programación ALTIVAR 31, pág. 25.)

YES: Frecuencia con modulación aleatoria. NO: Frecuencia fija.

- **nrd- Frecuencia de corte**

Se ajusta para reducir el ruido del motor.

Si la frecuencia se ajusta a más de 4 kHz, en caso de sobrecalentamiento el variador disminuirá automáticamente la frecuencia de corte, y la restablecerá cuando la temperatura vuelva a ser normal. (SCHNEIDER ELECTRIC, Telemecanique. Guía de programación ALTIVAR 31, pág. 25)

- **tFr- Frecuencia máxima de salida**

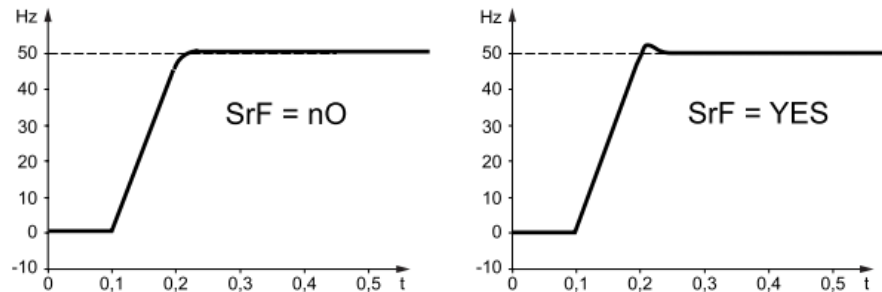
60 Hz por defecto. El rango de ajuste de la frecuencia varía de 10 a 500Hz.

Si se a configurado inicialmente el parámetro bFr a una frecuencia 60Hz, se ajusta este parámetro (Frs) a 60 Hz, el ajuste de fábrica es de 50Hz.

- **SrF- Eliminación del filtro del bucle por velocidad**

El fitro del bucle es un tiempo de respuesta reducido para eliminar un posible aumento de la consigna establecida. NO: El filtro del bucle de velocidad permanece activo. YES: El fitro del bucle de velocidad se elimina.

Figura 55: Eliminación del filtro del bucle de velocidad



Fuente: SCHNEIDER ELECTRIC, Telemecanique. Guía de programación ALTIVAR 31, pág. 25.

- **SCS- Grabación de la configuración**

Str1: Realiza una grabación de la configuración en curso en la memoria EEPROM (Memoria interna del variador). NO: Función no activa.

3.3.2. Menú de entradas/salidas I-O. • tCC- Control 2 hilos/3 hilos

LOC: Control local (RUN/STOP/RESET del variador) 2C: Control 2 hilos 3C: Control 3 hilos Si es el control es a 2C ó 3C, requerirá el uso de las entradas físicas LI1,LIx ó LI1,L2I,LIx, respectivamente.

3.3.3. Menú de control CtrlL. • LAC- Nivel de acceso a las funciones

L1: Acceso a las funciones estándar. En este nivel se permite sobre todo cambiar información con el ALTIVAR 28

L2: Acceso a las funciones avanzadas del menú FUn.

L3: Acceso a las funciones avanzadas y gestión de los canales por configuración. (SCHNEIDER ELECTRIC, Telemecanique. Guía de programación ALTIVAR 31, pág. 38.)

- **Fr1- Configuración consigna 1**

NDB: Consigna por Modbus. En este ítem también aparece la consigna por CANopen ó terminal remoto. Se selecciona de acuerdo al tipo de comunicación empleada.

AI1: Entrada analógica AI1

AI2: Entrada analógica AI2

AI3: Entrada analógica AI3

AIP: Potenciómetro (sólo en ALTIVAR 31)

• **rFC- Conmutación de consigna**

El parámetro rFC permite elegir el canal Fr1 o Fr2 o bien configurar una entrada lógica o un bit de la palabra de control para conmutar a distancia uno a otro. (Guía de programación ALTIVAR 31, pag 39.)

Fr1: Consigna = consigna 1

Fr2: Consigna = consigna 2

LI1: Entrada lógica LI1

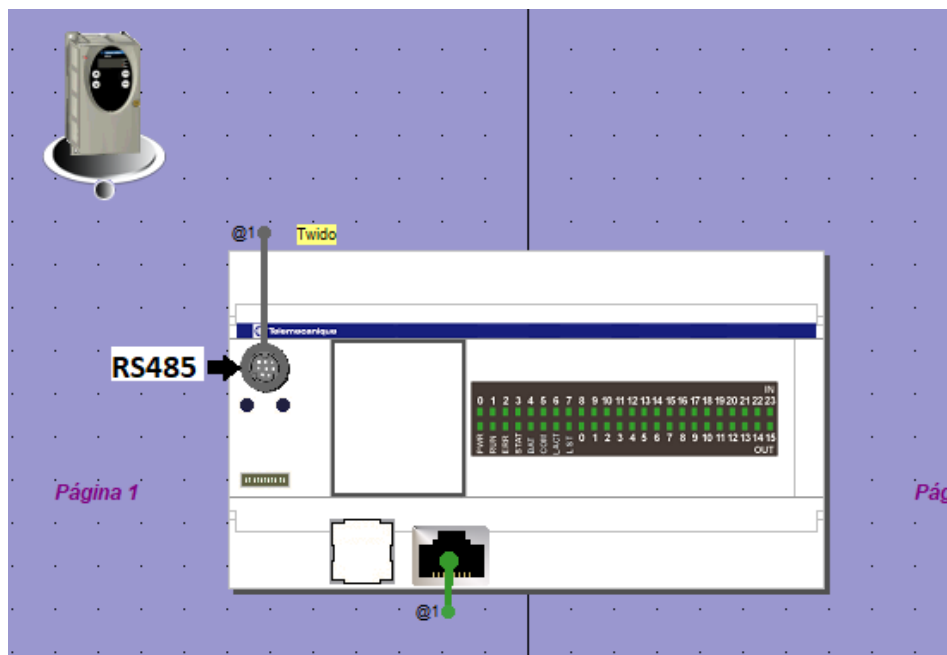
LI2: Entrada lógica LI2

C111: Bit 11 de la palabra de control Modbus

3.4. CONFIGURACIÓN MODBUS EN EL PLC TWIDO TWDLCAE40DRF

En la pestaña *Programar* del software TwidoSuite se agrega el PLC de referencia TWDLCAE40DRF al espacio de trabajo gráfico, así mismo en la pestaña catálogo: *Elementos de redes*, *Elementos Modbus*, se busca la referencia *ATV 31*, que corresponde al variador de velocidad ALTIVAR 31 como se muestra en la figura 56.

Figura 56: Espacio de trabajo gráfico del PLC Twido y el variador de velocidad sin comunicación



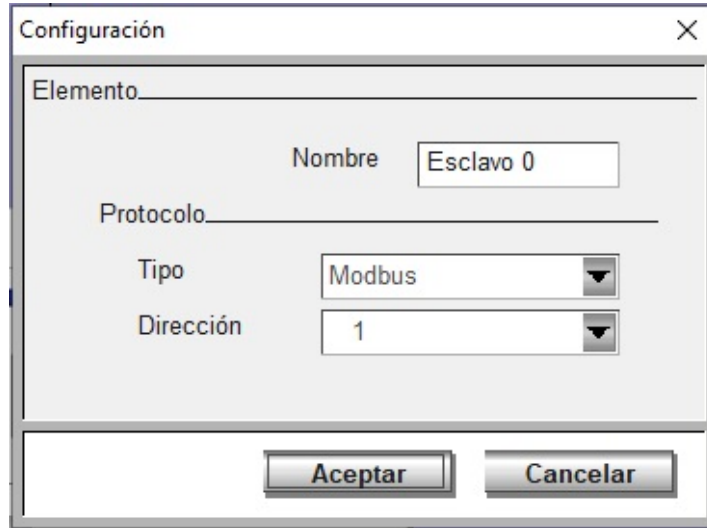
Para configurar como maestro al PLC, se hace doble clic sobre el puerto RS485 (Figura 56), seguido se abre una ventana de configuración del puerto 1 como se muestra en la figura 57. Se selecciona el tipo de protocolo que para el ejemplo a desarrollar es Modbus aunque se puede escoger entre *Conexión remota* ó *ASCII* según lo requiera el programador, en la dirección se selecciona la opción *master* y aceptar a la ventana, de esta forma el PLC queda configurado como el maestro en la comunicación.

Figura 57: Configuración del PLC como maestro



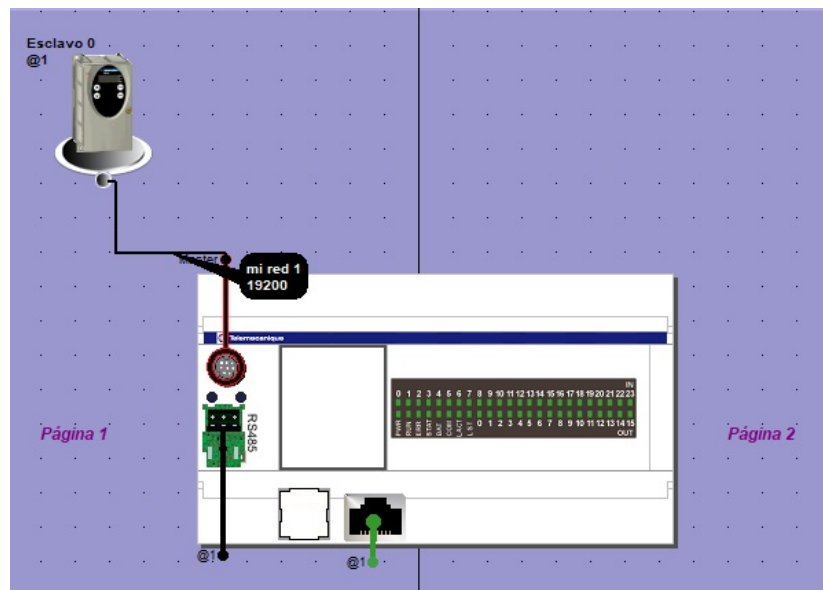
Para configurar como esclavo al ALTIVAR 31, se hace doble clic sobre el icono del variador, seguido se abre una ventana de *configuración* del elemento como se muestra en la figura 58. Se le asigna un nombre, se selecciona el tipo de protocolo Modbus que es el único disponible en este equipo, se selecciona un número de dirección al esclavo, recordar que si existen más esclavos el número de la dirección debe ser única para cada uno, aceptar a la ventana y de esta forma el variador ALTIVAR 31 queda configurado como el esclavo en la comunicación.

Figura 58: Configuración del variador ALTIVAR 31 como esclavo



Al realizar la configuración esclavo/maestro se realiza una conexión virtual uniendo el variador ALTIVAR 31, con el puerto RS485 del PLC, al estar correctamente enlazadas se muestra el nombre de la red y el índice de la red como se aprecia en la figura 59.

Figura 59: Espacio de trabajo gráfico del PLC Twido y el variador de velocidad enlazados



3.5. PROGRAMACIÓN DE LA COMUNICACIÓN MODBUS

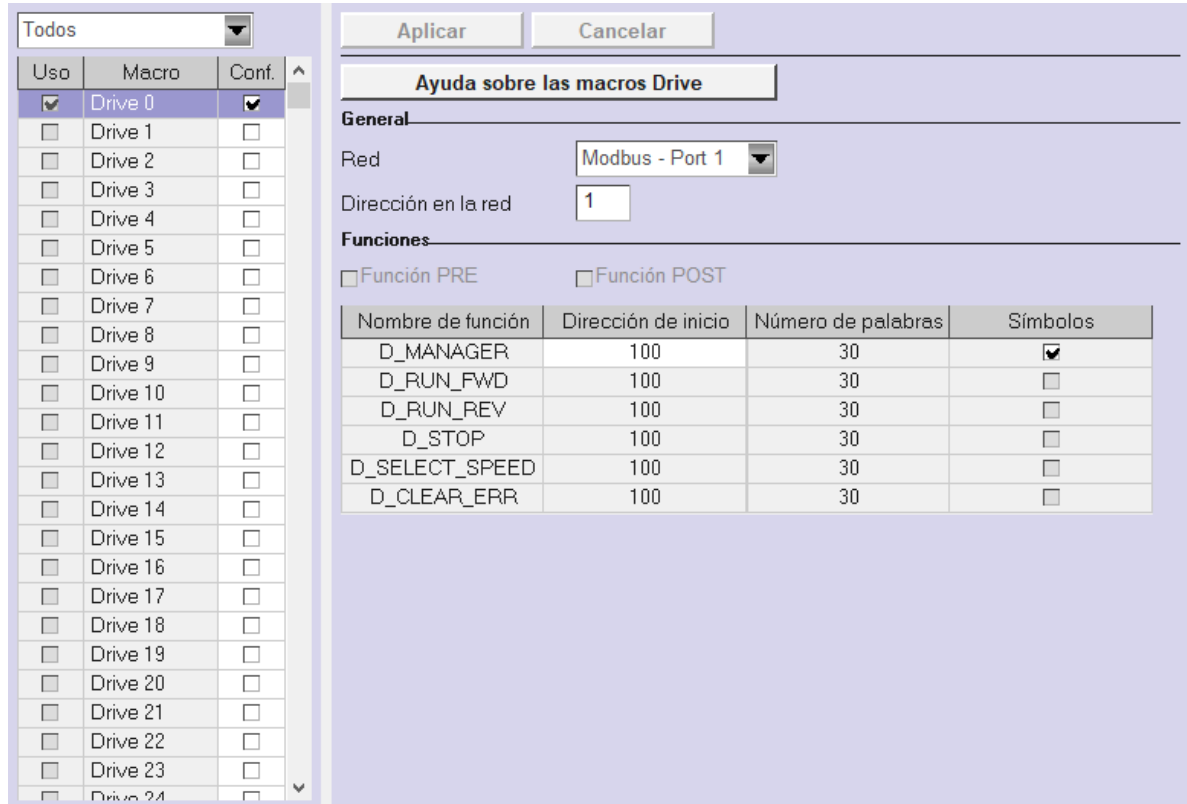
Las macros están diseñadas para facilitar la programación de la comunicación. Su función es estructurar el lenguaje, lo que facilita la lectura y escritura de los programas. La gestión y la creación de la tramas de comunicación se realizan internamente en las macros.

Se emplearon las macros de comunicación para dar solución al control de velocidad de un motor de inducción trifásico.

3.5.1. Macros DRIVE. Las macros DRIVE permiten realizar solicitudes de escritura mediante las funciones de macros a un esclavo y no realiza las solicitudes de lectura de datos.

En la pestaña *Programar, configurar datos, objetos avanzados, Macros Drive*, se selecciona la macro en la casilla *Conf.*, seguido, se activa una ventana como se visualiza en la figura 60, en la cual se debe configurar la red y dirección de la red. Importante verificar que la dirección de la red sea la misma dirección del esclavo a controlar.

Figura 60: Configuración de la macro Drive



Para emplear las funciones macros Drive es obligatorio seleccionar el símbolo de la función `D_MANAGER` (única habilitada), ya que esta función es la encargada de habilitar las demás funciones macros.

En la casilla símbolos se habilita la función `D_MANAGER`, es importante no emplear la misma dirección de inicio de la macro en una palabra del programa ya que se pueden sobre escribir las variables.

La macro Drive 0 emplea 30 bits de palabras de memoria a partir de la dirección 0 a la 29, es decir que si la dirección de inicio es de 100, el rango para asignar a las funciones es de `%MW100` a `%MW129`.

Al aceptar y salir de la ventana se habilitan diferentes funciones macro Drive como son:

- `D_MANAGER`: Esta función habilita la macro, si no está activa no se podrá hacer uso de las demás funciones.
- `D_RUN_FWD`: Activa el variador en sentido un sentido de giro.

- D_RUN_REV: Activa el variador invirtiendo el sentido de giro.
- D_STOP: Detiene el variador.
- D_SELECT_SPEED: Modifica el valor de velocidad escribiendo una nueva velocidad deseada.
- D_CLEAR_ERR: Regresa al último error detectado por el variador empleado.
- D_SETPOINT_MODE_0: Modo del punto de ajuste variador.
- D_SETPOINT_0: Punto de ajuste variador.

3.5.2. Macros COM. Las macros COM permiten realizar solicitudes de escritura y lectura de bit ó palabras a cualquier esclavo.

En la pestaña *Programar, configurar datos, objetos avanzados, Macros COM*, se selecciona la macro presentada en la figura 60, en la cual se debe configurar la red y dirección de la red. Importante verificar que la dirección de la red sea la misma dirección del esclavo a controlar.

Las funciones de esta macro están habilitadas para ser empleadas de forma independiente, por ello, en la casilla *Símbolos* se selecciona la función a emplear, el número de palabras y la dirección de inicio que corresponde a una dirección que no haya sido asignada en el programa ya que se pueden sobre escribir las variables.

Al aceptar y salir de la ventana se habilitan diferentes funciones macro com como son:

Figura 61: Configuración de la macro COM

The screenshot displays the configuration window for a COM macro. On the left, a list of macros from Com 0 to Com 24 is shown, with 'Com 0' selected. The right side of the window is titled 'Ayuda sobre las macros Com' and contains the following configuration options:

- General:**
 - Red: Modbus - Port 1 (dropdown)
 - Dirección en la red: 1 (text field)
- Funciones:**
 - Función PRE
 - Función POST

Below the function options is a table with the following data:

Nombre de función	Dirección de inicio	Número de palabras	Símbolos
C_RD1B	0	8	<input type="checkbox"/>
C_RD1W	0	8	<input type="checkbox"/>
C_WR1B	0	8	<input type="checkbox"/>
C_WR1W	0	8	<input type="checkbox"/>
C_RDNW	120	N+7	<input checked="" type="checkbox"/>
C_WRNW	140	N+9	<input checked="" type="checkbox"/>

- C_RD1B: Esta función permite leer 1 bit a través de la red Modbus.
- C_RD1W: Esta función permite leer 1 palabra a través de la red Modbus.
- C_WR1B: Esta función permite escribir 1 bit a través de la red Modbus.
- C_WR1W: Esta función permite escribir 1 palabra a través de la red Modbus.
- C_RDNW: Esta función permite leer N palabras a través de la red Modbus. Donde N= número de palabras a escribir.
- C_WRNW: Esta función permite escribir N palabras a través de la red Modbus. Donde N= número de palabras a escribir.
- C_RDNW_ADDR1: Se debe iniciar con este símbolo con la primera dirección de palabra que va a leerse.

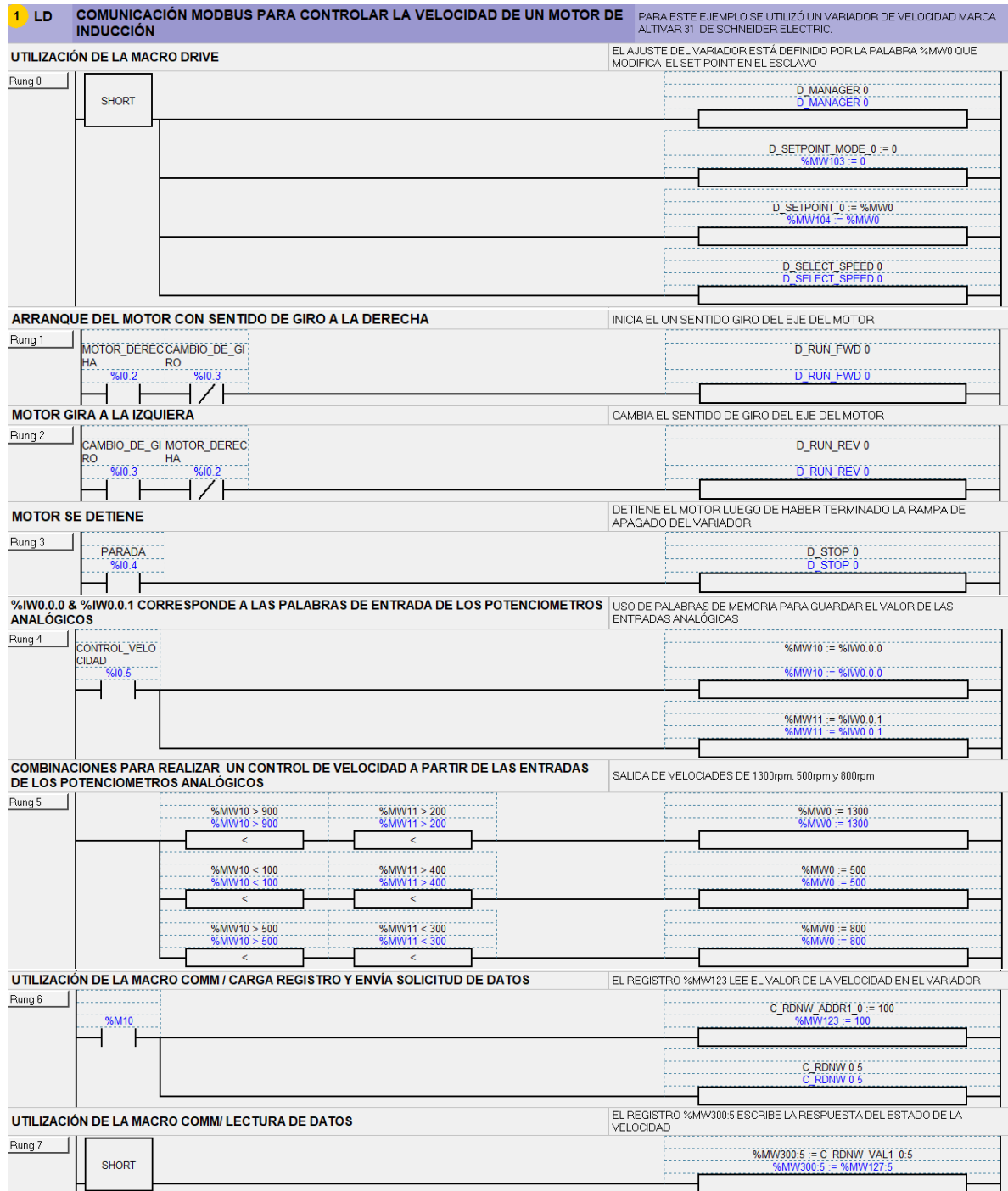
- C_RDNW_ADDR1: Solicita al esclavo leer N cantidad de palabras. onde N= número de palabras a escribir.
- C_WRNW_VAL1: Carga el valor de la palabra leída.

3.6. PROGRAMA PARA EL CONTROL DE VELOCIDAD DE UN MOTOR DE INDUCCIÓN EMPLEANDO COMUNICACIÓN MODBUS.

Tabla 5: Variables de control del ejemplo control de velocidad de un motor de inducción empleando comunicación modbus

Parámetros			
Dirección	Estado	Símbolo	Descripción
%I0.2	NA	MOTOR_DEREC	Se energiza el motor con sentido de giro a la derecha
%I0.3	NA	CAMBIO_GIRO	El motor se detiene para realizar el cambio de sentido
%I0.4	NA	PARADA	Detiene el motor
%I0.5	NA	CONTROL_VEL	Lectura de las entradas analógicas

Figura 62: Control de velocidad de un motor de inducción empleando las funciones MACROS



4. CONTROL DE TEMPERATURA DE UN HORNO ELÉCTRICO USANDO LA FUNCIÓN PID DEL SOFTWARE TWIDO

Los controladores PID son dispositivos físicos (en el PLC serán bloques funcionales que se programan al igual que los dispositivos físicos) de realimentación que aplican acciones correctoras para ajustar el comportamiento de un proceso de acuerdo al error calculado entre un valor medido y un valor que se espera obtener (Valor de referencia ó set point).

En este capítulo se implementará el control PID sobre un horno eléctrico perteneciente a la escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones, en este estudio se considerarán aspectos como la rapidez de respuesta ante perturbaciones que puedan modificar las variables y el error que se presenta para controlar la estabilidad del proceso.

Para realizar el control se empleó el PLC TWDLCAE40DRF de la marca Schneider electric, para la programación y configuración empleando el software TwidoSuite.

4.1. CONTROL PROPORCIONAL, INTEGRAL Y DERIVATIVO

Acción de control proporcional: Es un amplificador con una ganancia ajustable, si la ganancia del K_p del controlador es muy grande, puede generar inestabilidad del sistema. Su importancia implica en que siempre debe existir una acción de control que sea proporcional al error. La relación entre la salida del controlador $u(t)$ y la señal de error $e(t)$ es:

$$u(t) = K_p \times e(t)$$

Un controlador proporcional puede controlar cualquier planta estable, pero su desempeño es limitado y el error en régimen permanente puede eliminarse si se incluye una acción de control integral.

Acción de control integral: La acción de control integral se denomina control de reajuste (reset). La rapidez de cambio en la respuesta del controlador $u(t)$ es proporcional

al error $e(t)$, es decir:

$$\frac{du(t)}{dt} = Ki \times e(t)u(t) = Ki \int_0^t e(\tau)d\tau$$

La acción de control integral ayuda a disminuir el error en estado estable, pero presenta dos desventajas que son:

-Mal funcionamiento si existen señales offset ya que pueden saturar el sistema de control. -Se pueden presentar transitorios elevados.

Acción de control derivativa: Es conocida como “anticipativa” o “predictiva”, ya que con esta es posible determinar cuál es la tendencia de cambio del error. Mejora la rapidez de la respuesta ya que reduce en gran medida el régimen transitorio. -No se debe implementar en sistemas en los que hayan señales de ruido, ya que amplificará también estas señales.

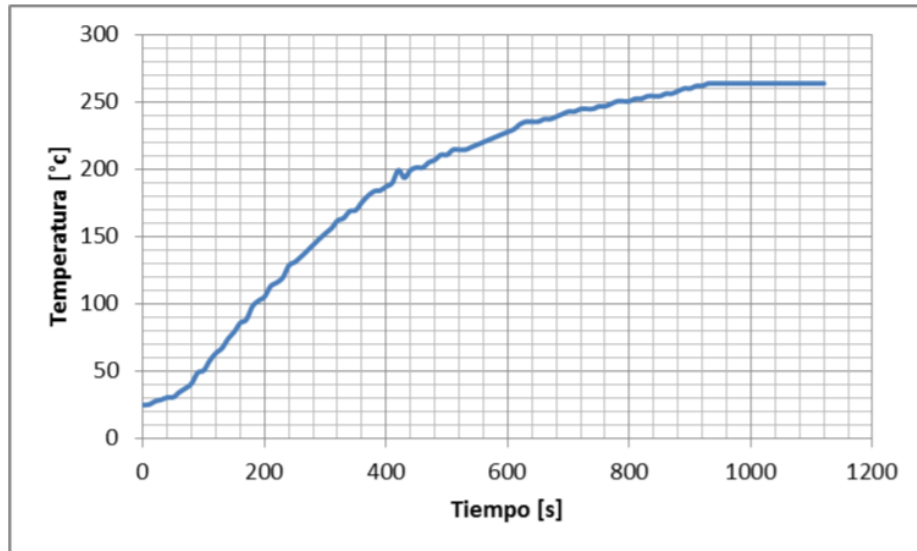
La función que relaciona la salida $u(t)$ y la señal de error $e(t)$ es:

$$u(t) = Kd \times \frac{de(t)}{dt}$$

4.2. ANÁLISIS TEÓRICO

El modelo de la planta se obtuvo a través de la curva de calentamiento del horno (ver figura 62) del proyecto de grado Bitar, Villamizar y Yerena (2013). Esta la obtuvieron al energizar un horno a una tensión de 120 Vac y registrando las temperaturas en función del tiempo.

Figura 63: Curva de calentamiento del horno eléctrico



Fuente: BITAR RODRIGUEZ, José Carlos; VILLAMIZAR CONTRERAS, Javier Orlando; YERENA BAYONA, Gustavo Adolfo. Comparación de las características y comportamiento de un controlador PID industrial con la función PID de un PLC (Controlador Lógico Programable). Bucaramanga: Universidad Industrial de Santander, 2013, pág. 68.

La función de transferencia y la obtención de parámetros de control del horno fueron tomados del proyecto de grado Bitar, Villamizar y Yerena (2013)

Función de transferencia del horno eléctrico

$$H(s) = \frac{0,006561}{s + \frac{1}{335}}$$

Parámetros obtenidos para el control del horno

$$Ti = 335$$

$$Kp = Kc = 2$$

$$Ki = 0,00597$$

4.3. IMPLEMENTACIÓN DEL SISTEMA DE CONTROL

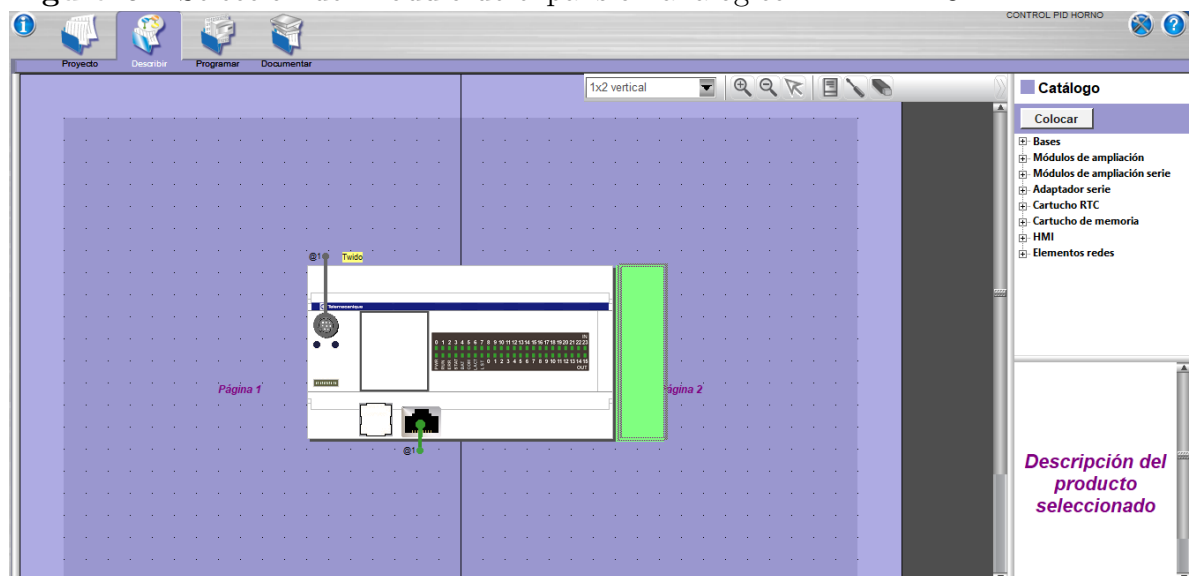
Empleando el modelo del trabajo de grado cuyos autores son Bitar, Villamizar y Yarena (2013), que representa el comportamiento de la planta que se desea controlar y de la obtención de los parámetros del controlador a usar, se realiza la configuración del PLC en el software TwidoSuite, el montaje físico del sistema y por ultimo la recolección de los resultados obtenidos presentados al final de este capítulo.

4.3.1. Configuración del módulo analógico. Algunos autómatas Twido permiten ampliar entradas y salidas agregando módulos de ampliación al PLC base según lo requiera la aplicación. Para este caso se utiliza un módulo E/S analógico para el censado de la variable a medir con el fin de realizar el control PID.

Para configurar el módulo analógico después de crear el proyecto y seleccionar el autómata TWDLCAE40DRF, se selecciona el módulo de ampliación requerido para la aplicación, encontrándose en la pestaña *Catálogo*, *Módulos de aplicación*, *Ampliaciones analógicas*, se selecciona el módulo TM2AMM3HT, este cuenta con las E/S necesarias para a la aplicación.

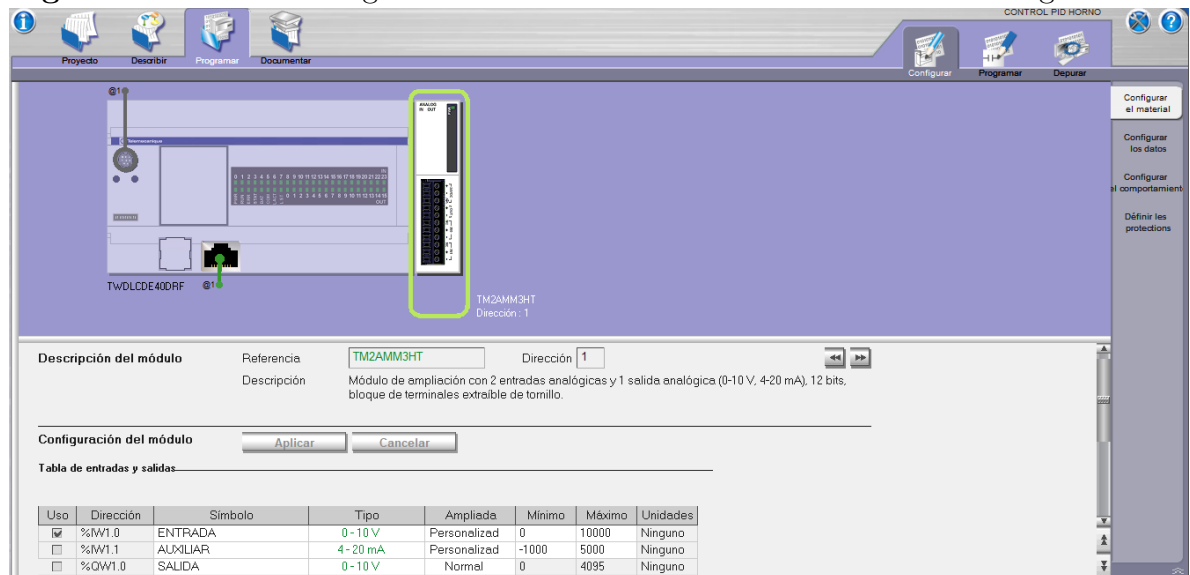
Para agregar el módulo al PLC compacto, se selecciona y se arrastra a la posición como se muestra en la figura 63.

Figura 64: Selección del módulo de expansión analógico TM2AMM3HT



Se hace doble clic sobre el módulo analógico para abrir la ventana de configuración del módulo.

Figura 65: Zona de configuración de las entradas-salidas del módulo analógico



En la tabla de entradas y salidas se muestran las siguientes columnas que presentan las siguientes funciones:

Dirección: Indica las entradas y salidas del módulo analógico, esta opción no es configurable.

Símbolo: Asigna un nombre a la variable si el programador lo requiere.

Tipo: Define el tipo de entrada-salida, esta opción depende del módulo utilizado, específicamente el módulo empleado cuenta con los modos de tensión 0-10 Vdc y de corriente 4-20 mA.

Ampliada: Permite definir los valores mínimos y máximos en función de la opción escogida, donde la función normal ajusta un rango preestablecido y la opción personalizada permite especificar un mínimo y máximo entre -32768 y 32767.

Figura 66: Tabla de configuración de entradas - salidas del módulo analógico

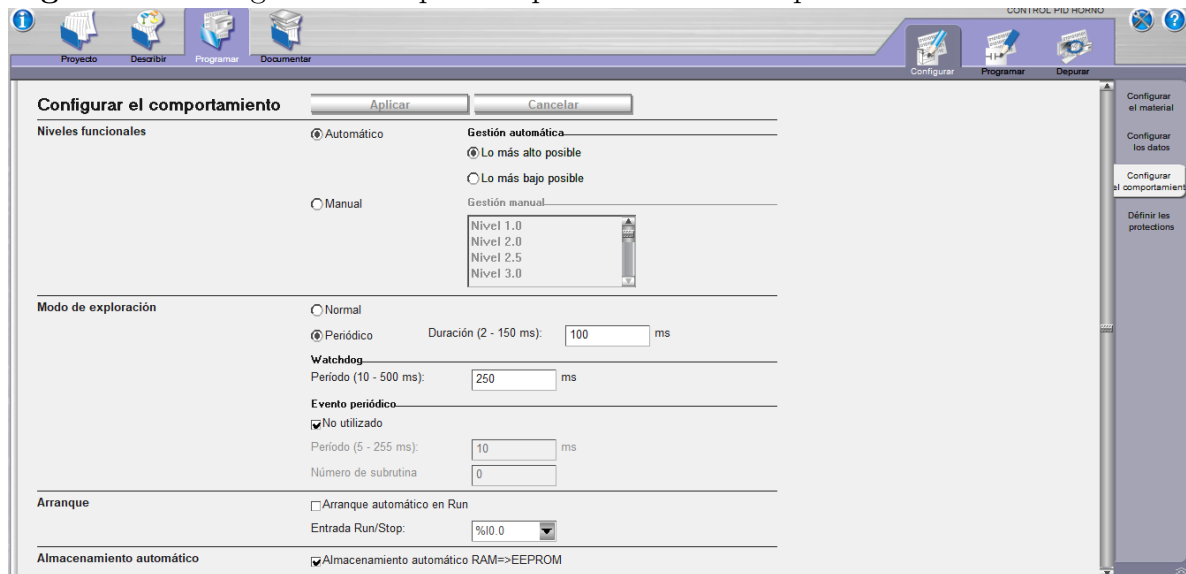
Uso	Dirección	Símbolo	Tipo	Ampliada	Mínimo	Máximo	Unidades
<input checked="" type="checkbox"/>	%IW1.0	ENTRADA	0 - 10 V	Personalizad	0	10000	Ninguno
<input type="checkbox"/>	%IW1.1	AUXILIAR	4 - 20 mA	Personalizad	-1000	5000	Ninguno
<input type="checkbox"/>	%QW1.0	SALIDA	0 - 10 V	Normal	0	4095	Ninguno

La actualización de las entradas y salidas del módulo analógico se realiza de manera sincronizada con la exploración del autómatas en modo RUN.

Los pasos de configuración se presentan a continuación:

En la pestaña *Programar, configurar, configurar el comportamiento*, se activa la opción *periódico* asignando un valor comprendido entre [2 a 150 ms]. Como el proceso que se va a controlar es un proceso térmico cuenta con un tiempo de respuesta lento, que es suficiente para asignar un tiempo de exploración de 100 ms.

Figura 67: Configuración de periodo para el modo de exploración



4.3.2. Regulación continua PID. La herramienta TwidoSuite cuenta con la opción de implementar algoritmos de regulación continua PID para diversos procesos industriales. Esta función recibe una señal analógica de entrada con la medida de la variable que se desea regular, el autómatas Twido realiza los cálculos haciendo uso del algoritmo PID.

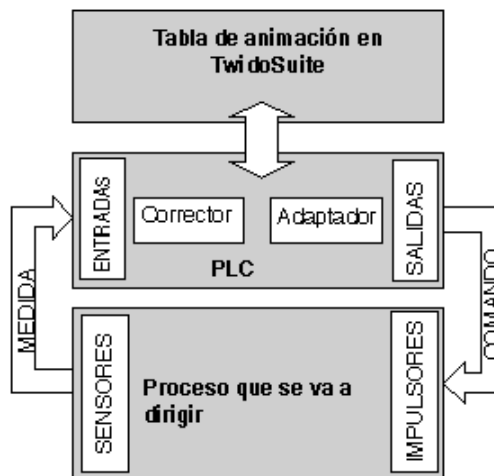
El proceso de bucle de regulación se realiza en tres fases distintas

- La adquisición de datos:
 - Medidas provenientes de los sensores del proceso (analógicos, codificadores).
 - Consignas provenientes, generalmente, de variables internas del autómatas o de datos procedentes de una tabla de animación TwidoSuite.
- Ejecución del algoritmo de regulación PID El envío de los comandos adaptados a las características de los impulsores que van a controlarse a través de las salidas binarias (PWM) o analógicas.

El algoritmo PID elabora la señal de comando a partir de:

- La medida muestreada por el módulo de entrada.
- El valor de la consigna fijada por el operador o por el programa.

Figura 68: Principio de funcionamiento regulación PID

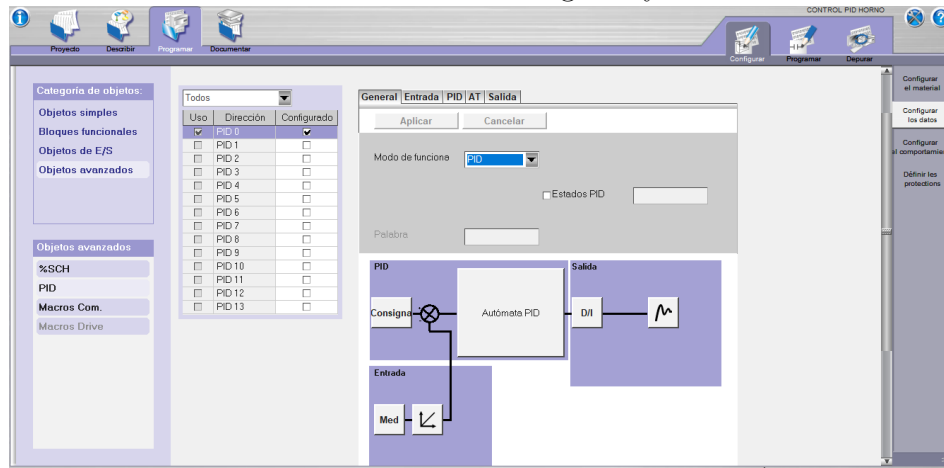


SCHNEIDER ELECTRIC, Telemecanique. Autómatas programables Twido. Bases compactas y modulares, guía de hardware, 2011, pág. 531.

4.4. CONFIGURACIÓN DE LA FUNCIÓN PID

Una vez definidas las posibles entradas y salidas analógicas a utilizar, se procede a configurar el regulador PID, esto se realiza en la pestaña *Programar, configurar, configurar los datos, objetos avanzados, PID*.

Figura 69: Ventana de selección del PID a configurar y utilizar



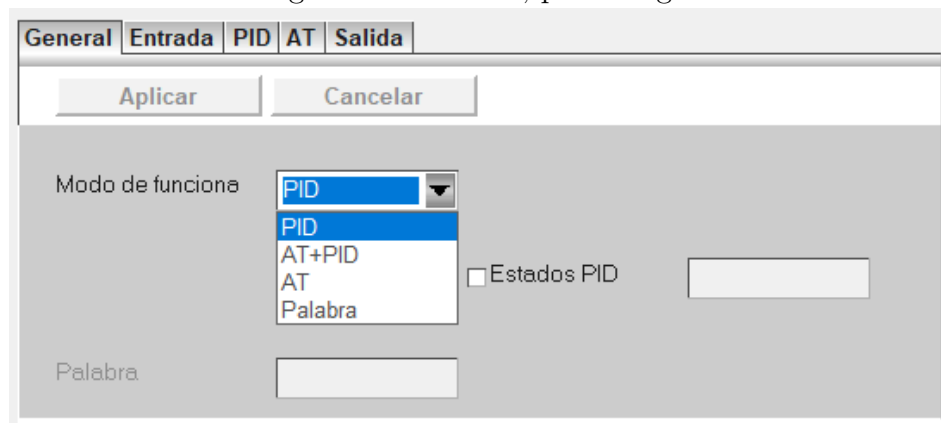
Fuente: Software TwidoSuite

Se cuenta con 14 PID configurables, y para utilizar alguno se debe activar la casilla *Configurado*. Una vez activado el PID se habilitan las opciones en las pestañas para ser configuradas.

En la pestaña *general* se pueden elegir los siguientes modos de funcionamiento: PID, AT+PID, AT, PALABRA

En este caso se habilita el modo PID como se muestra en la figura 62

Figura 70: Ventana de configuración del PID, pestaña general



En la pestaña *entrada* se introduce la entrada analógica seleccionada del módulo analógico la cual es utilizada como medida en la aplicación. En la casilla *conversión* habilita la conversión de la variable del proceso que ha sido especificada como entrada del PID en la casilla *medida*

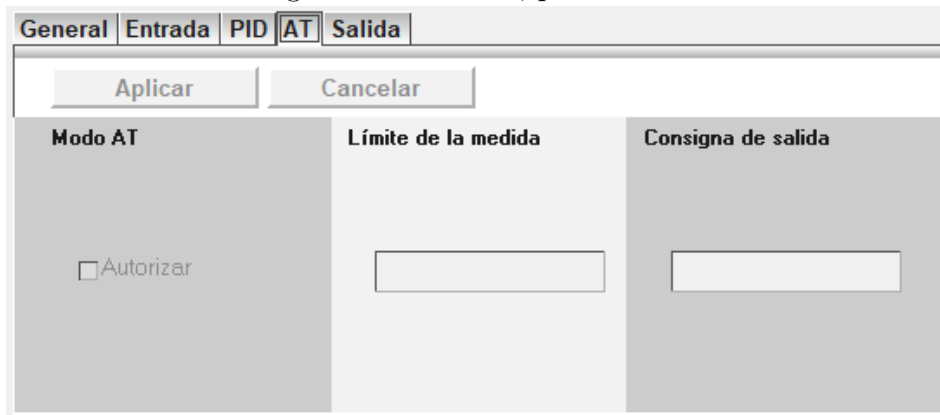
Figura 71: Ventana de configuración del PID, pestaña entrada

En la pestaña PID se establece el valor de la consigna del PID asignando un valor fijo, o por medio de la aplicación mediante una palabra de memoria o por una entrada analógica. Para el caso de estudio, valor de la consigna se dejará con un valor fijo de 34°C. Los parámetros del control son los obtenidos en el análisis teórico del proyecto de grado Yarena y Villamizar (2013).

Figura 72: Ventana de configuración del PID, pestaña PID

La pestaña *AT* sólo se puede configurar si se ha seleccionado el modo de funcionamiento *AT* o *AT+PID*.

Figura 73: Ventana de configuración del PID, pestaña AT



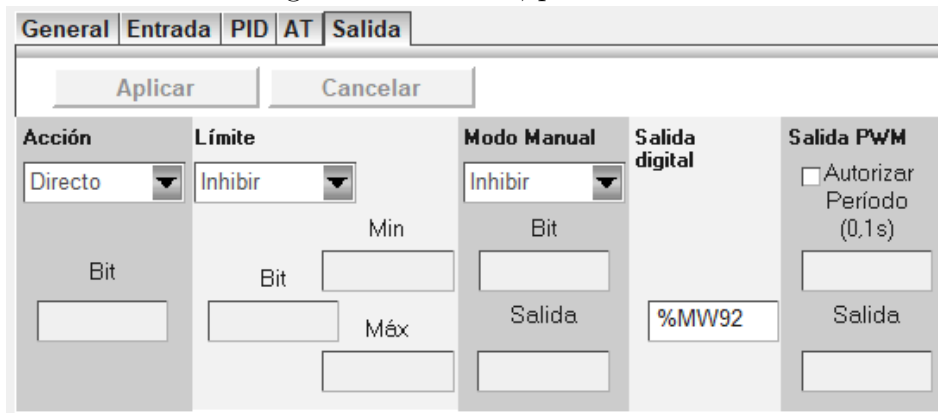
En la pestaña *salida*, se encuentra la casilla *acción*, en ella se puede seleccionar las siguientes opciones:

- Directo: Cuando la variación (Consigna-medida) aumenta, la salida del autómata aumenta.
- Inverso: Cuando la variación (Consigna-medida) disminuye, la salida del autómata aumenta.
- Bit: Se emplea para modificar el funcionamiento de la acción. Se configura un bit interno o una entrada, de modo que si el bit es 0, la acción es inversa y si es 1 la acción es directa.

La casilla *Limite* se utiliza para limitar la salida del PID. Se puede seleccionar entre las siguientes opciones: Inhibir, autorizar, bit.

La casilla *salida digital* indica la salida del regulador PID, se selecciona como una palabra de memoria o la salida del módulo analógico.

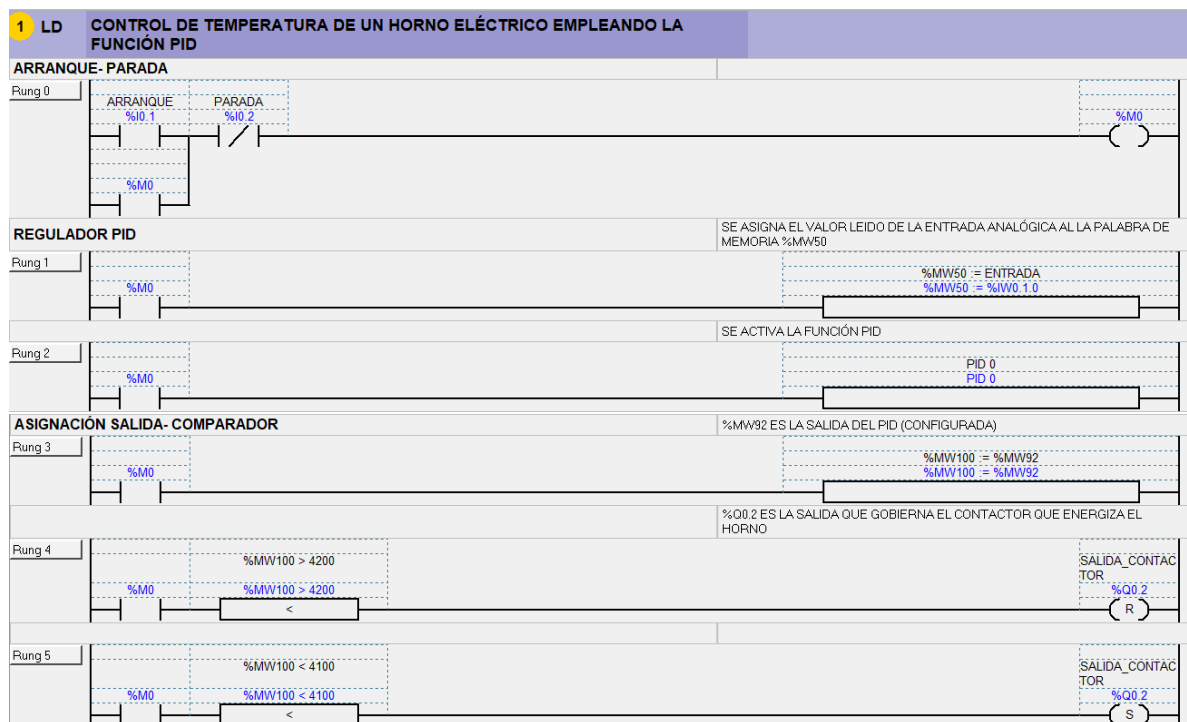
Figura 74: Ventana de configuración del PID, pestaña salida



4.5. PROGRAMACIÓN CONTROL PID HORNO

En la figura 75 se muestra la programación en el software TwidoSuite para el control de temperatura de un horno eléctrico a través del control PID.

Figura 75: Programación LADDER para el control de temperatura de un horno eléctrico

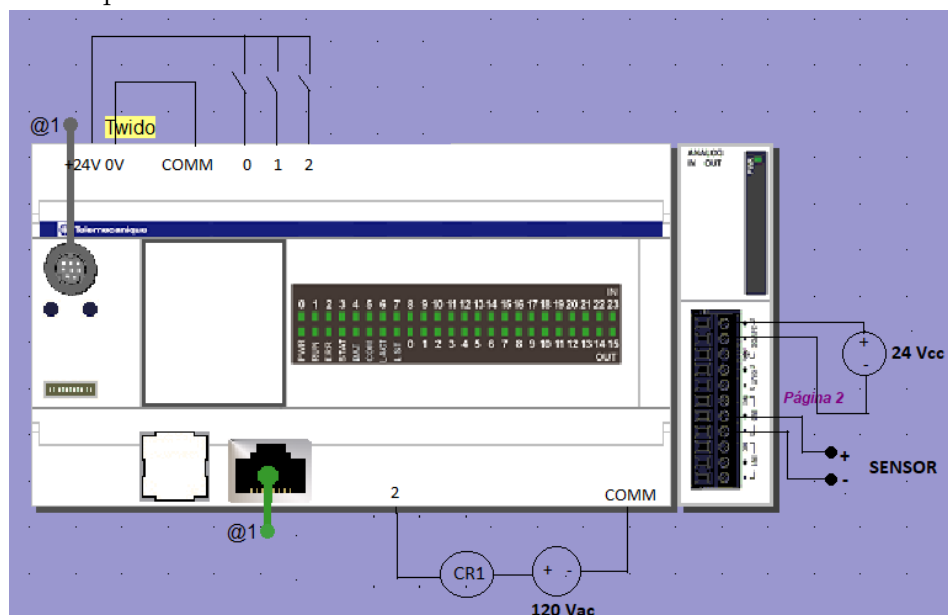


4.6. MONTAJE FÍSICO DEL SISTEMA DE CONTROL

El esquema de conexión se presenta en la siguiente figura donde se muestran las entradas y salidas digitales del PLC y la conexión del módulo analógico.

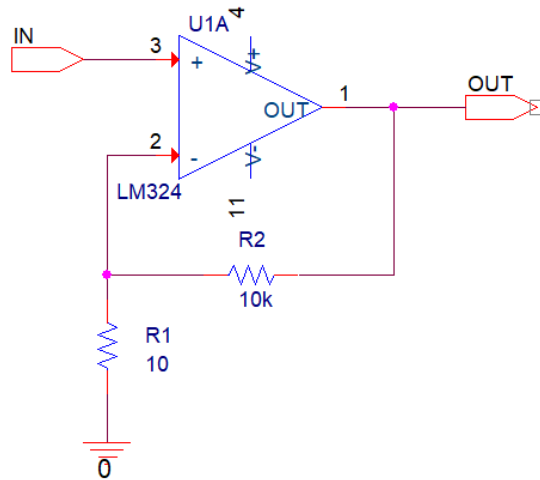
Como se presenta en la figura 76, se usó como elemento de conmutación un contactor auxiliar. El control sobre el horno consiste en la energización ó desenergización de la bobina de contactor auxiliar, dependiendo del valor leído de la termocupla.

Figura 76: Esquema de conexiones físicas



La variación de tensión de la termocupla es del rango de los mili-volts, por este motivo se realizó un circuito amplificador de señal (Figura 77) usando un amplificador operacional LM324 el cual debe contar con un buen desempeño offset.

Figura 77: Circuito de amplificación.



La ganancia está dado por la siguiente ecuación:

$$G = \frac{R2}{R1} + 1$$

$$R2 = 10k \Omega \text{ y } R1 = 10 \Omega$$

Obteniendo una ganancia de 1001 [V/V]

4.7. RESULTADOS OBTENIDOS

Luego de haber realizado la configuración y el montaje, se verificó el funcionamiento PID, comprobando las respuestas del sistema al aumento ó disminución de temperatura. En las siguientes imágenes se presenta los resultados obtenidos.

Figura 78: Arranque en frío del horno empleado para el control PID

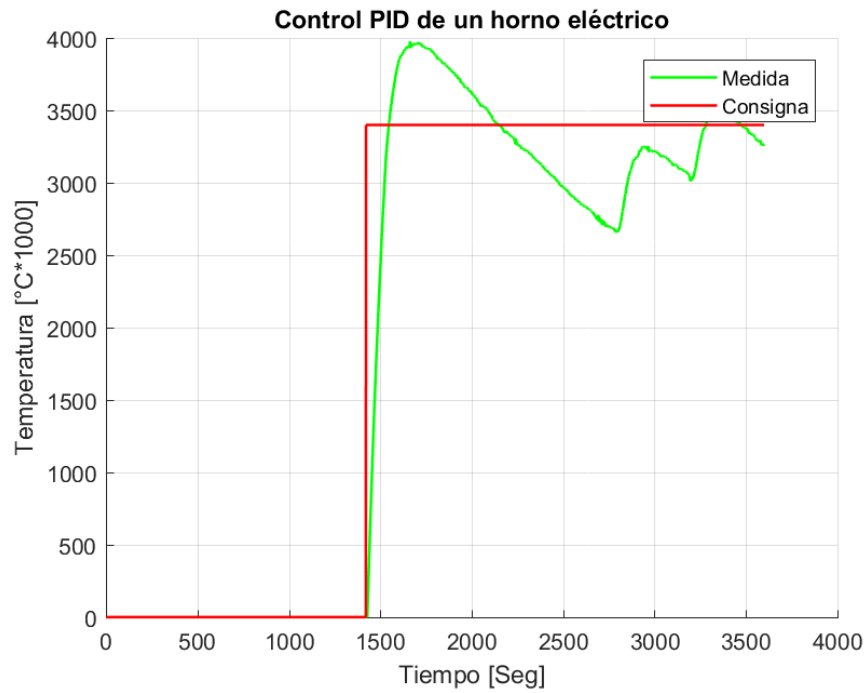
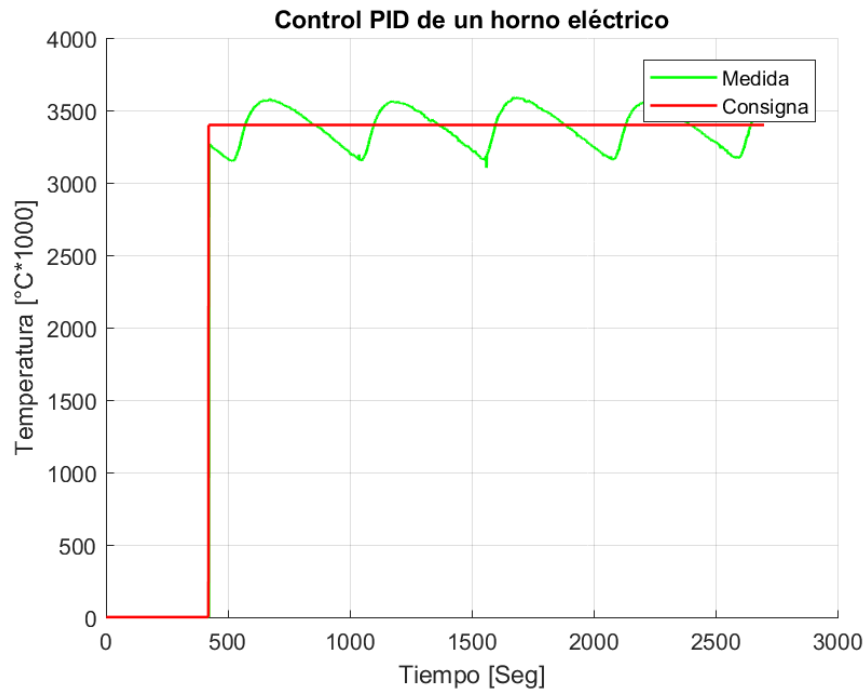


Figura 79: Respuesta del control PID al estabilizarse el sistema



En la figura 78 el horno no se había energizado, al accionarse el contactor auxiliar, el horno inicia su proceso de calentamiento hasta que su temperatura es cercana al set point (34°C), la segunda figura (Figura 79) fue tomada para percibir que el control PID actuaba correctamente al aumento ó disminución de temperatura por un tiempo prolongado.

Los trazos sólo son posibles visualizarlos cuando el programa está en ejecución, para este caso, se ingresó a la configuración de la función PID y en la barra lateral derecha, opción *crear un nuevo trazo*, se visualizó en tiempo real las salidas del sistema. Las gráficas se exportaron en un archivo .csv, y con estos datos se crearon las gráficas en Matlab.

5. OBSERVACIONES

- Si se presentan dificultades o dudas y sugerencias, respecto a las características de un bloque funcional, se recomienda en primera instancia acudir al material de apoyo en línea del software TwidoSuite, antes de buscar manuales, a razón que se dispone de mucha información en internet.
- Al conectar el controlador para ejecutar un proceso (o simularlo con elementos conectados a las entradas y salidas), se recomienda verificar el esquema de conexión de cada equipo, para el caso del PLC las entradas digitales tienen su propia fuente de alimentación de 24Vdc y para el variador de velocidad ALTIVAR 31 se recomienda identificar las entradas y salidas ya que un error frecuente es que se conecte la fuente a las salidas del equipo.
- Al transferir un programa desde el PC al PLC se debe tener especial cuidado pues cada PLC disponible en el almacén de la escuela E3T, tiene configurada una dirección IP de la red clase C (192.168.71.90) la cual se debe conservar; si no se configura correctamente dicha dirección en el programa a transferir se corre el riesgo de cambiar o de borrar la dirección IP, por esta razón es necesario seguir el siguiente protocolo:
 1. Realizar la conexión física del PC al PLC Twido empleando un cable UTP con terminales RJ45.
 2. Verificar la configuración de la dirección IP del PLC a través del puerto Ethernet como se explicó en la sección 1.4.3.
 3. Verificar en las propiedades de conexión de red Ethernet del PC, la configuración de la dirección IP sea diferente a la del PLC pero de la misma red.
 4. Verificar la existencia de la conexión a la red (del PC y PLC). Ejecutando la instrucción *ping “dirección IP”* en *símbolo del sistema* del equipo (Instrucción: cmd.exe).
 5. Antes de *Depurar ó transferir* el programa, verificar que la dirección IP configurada en el programa, sea la misma dirección que tiene el PLC.
 6. Realizar la transferencia (depuración) del programa.
 7. Cuando se transfiere un programa del PLC al PC, no se presentan los títulos ni los comentarios de las secciones “rungs”.

- Se sugiere la simulación empleando una tarjeta de adquisición de datos (TAD) de National Instruments mediante el protocolo Opcserver y Labview, conectando (cableando) las entradas como ejercicio y visualización de Labview como software de control programable desde un equipo (PC) genérico.
- Se sugiere que la escuela programe un curso o seminario en redes de comunicación industrial.

6. CONCLUSIONES

- La automatización de un proceso se puede solucionar empleando diferentes bloques funcionales del PLC, la elección depende de la comodidad del programador; es importante resaltar que hay programas que requieren más líneas de código que otros, según sean las funciones utilizadas en la solución.
- Se presentan 3 procesos a automatizar los cuales se solucionan empleando diferentes funciones, validando la versatilidad, y la facilidad de programación de este software respecto a otras marcas comerciales, es importante resaltar que este software es de uso libre.
- La forma más sencilla para implementar el protocolo de comunicación Modbus es el uso de las funciones macros; ya que estas no requieren un conocimiento profundo referente a la estructura de la trama del protocolo Modbus.
- Se presenta un programa para el control discreto de velocidad de un motor de inducción (1.5kW) disponible en el laboratorio de máquinas eléctricas de la escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones, (E3T). Se debe prestar principal atención en la utilización y uso de los diferentes macros y variables (bit y palabra de control), ya que si son utilizados sin analizar su funcionamiento y su ubicación en el programa, puede tener resultados no deseados. Para el caso del ejemplo presentado en la figura 62, del capítulo 3, se visualiza que el rung 1 debe tener un enclavamiento eléctrico para activar la salida *Motor_Derecha* y *Motor_Cambio de giro*, ya que si se dejan las dos entradas sin enclavamiento eléctrico y se activan simultáneamente, el eje del motor intentará moverse el eje de un lado al otro sin lograr alguna estabilidad de velocidad.
- No se pudo implementar el control de velocidad de un motor de inducción empleando el PID porque el tiempo mínimo de muestreo del PLC es de 0.1 segundos, este tiempo comparado con el τ mecánico del motor es muy grande, en el trabajo de grado Lopez. J y Sanchez. D. Motores de inducción, visualización de la corriente de arranque. Bucaramanga, 2017. p. 54. “el motor tarda aproximadamente dos ciclos en estabilizarse (32 milisegundos)”, que corresponde a 5 τ s mecánicos, este tiempo es mucho menor que los 100ms de tiempo de muestre del PLC.

- Se programó el control de temperatura y se validó con las gráficas mostradas en las figuras 78 y 79 el funcionamiento PID donde se observa la respuesta del control cercana al set point. En la primera gráfica presenta un arranque en frío donde se observa que hay un incremento ó pico de temperatura y luego se estabiliza a través de un punto, la estabilización requiere un tiempo ya que el elemento de control es un contactor y se está oscilando entre un rango amplio de 33°C y 37°C, si este rango se hace mas pequeño menor será el rizado, pero se debe tener presente que se tiene un control electromecánico.

BIBLIOGRAFÍA

ARTEAGA GONZÁLEZ, Alwin Serguey; RUEDA RUEDA, Wilfred, URIBE VEGA, Sergio Giovany. Controlador Lógico Programable TSXnano módulos de entrenamiento y prácticas de laboratorio. Bucaramanga: Universidad Industrial de Santander, 2001.

BALCELLS, Josep; ROMERAL, José Luis. Autómatas programables. España: Marcombo Boixareu, 1997.

BITAR RODRIGUEZ, José Carlos; VILLAMIZAR CONTRERAS, Javier Orlando; YERENA BAYONA, Gustavo Adolfo. Comparación de las características y comportamiento de un controlador PID industrial con la función PID de un PLC (Controlador Lógico Programable). Bucaramanga: Universidad Industrial de Santander, 2013.

GÉLVEZ FIGUEREDO, Julio Augusto. Accionamientos Eléctricos: Control electromecánico. Bucaramanga: Universidad Industrial de Santander, 1994.

RAMIREZ RONDÓN, Oscar Mauricio; SANCHEZ ARENAS, José Ignacio. Redes de comunicación industrial, prácticas de laboratorio. Bucaramanga: Universidad Industrial de Santander, 2011.

SATURNINO, Soria Trello. Sistemas Automáticos Industriales de Eventos Discretos. México D.F: Alfaomega, 2013.

SCHNEIDER ELECTRIC, Telemecanique. Controladores programables Twido, 2013.

SCHNEIDER ELECTRIC, Telemecanique. Guía de programación, 2011.