



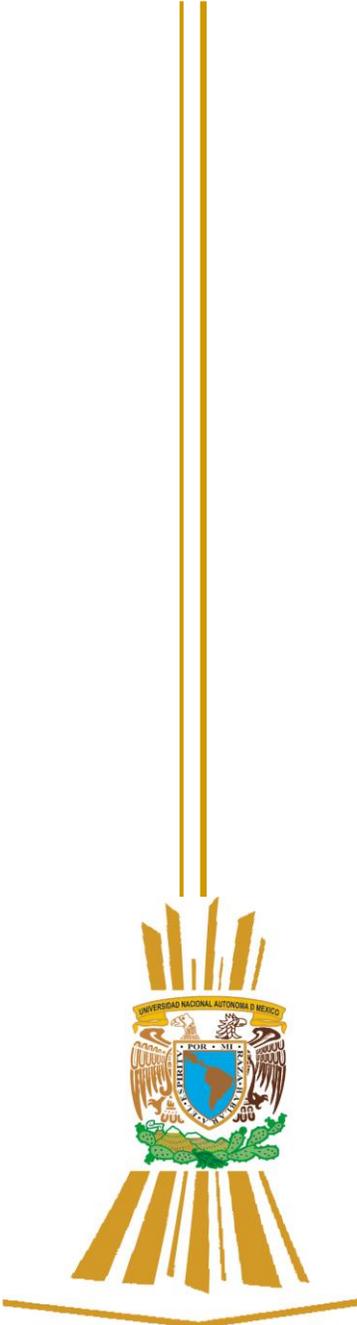
UNIVERSIDAD AUTÓNOMA DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES
ARAGÓN

“AUTOMATIZACIÓN DE UNA LÍNEA DE EMPAQUE
(LLENADO DE PASTILLAS)”

T E S I S
QUE PARA OBTENER EL TÍTULO DE :
INGENIERÍO MECÁNICO ELÉCTRICO
ÁREA: ELÉCTRICA Y ELECTRÓNICA
P R E S E N T A :
JOSÉ GUADALUPE CEDILLO MEJORADA

ASESOR: ING. NARCISO ACEVEDO HERNÁNDEZ





Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

AGRADECIMIENTOS

A nuestra máxima casa de estudios la Universidad Nacional Autónoma de México, por brindarnos las herramientas necesarias para convertirnos en unos profesionistas orgullosos de su país.

A los profesores que nos brindaron sus conocimientos y experiencias.

A DIOS

Por haber estado siempre presente en mis convicciones dándome la fortaleza necesaria para superar los momentos y los retos más difíciles, permitiéndome ver con claridad el camino que he seguido a través de mi vida en la búsqueda de mis ilusiones y metas que gracias a él he logrado alcanzar con éxito.

A MIS PADRES

Que con su amor, protección y sabiduría forjaron en mí las bases para ser una persona responsable, con principios y valores que han sido esenciales en el alcance de mis logros.

A MI ESPOSA E HIJOS

Les agradezco su paciencia, su comprensión y su apoyo para lograr la meta propuesta y que siempre confiaron en mí y que siempre han sido mi motivo de superación.

A MIS HERMANOS Y AMIGOS

Gracias por el cariño y apoyo moral que siempre he recibido de ellos con el cual he logrado culminar mi esfuerzo, nunca encontraré la forma de agradecerles su ayuda incondicional.

A MI ASESOR

Gracias por su apoyo y que de una u otra manera siempre estuvo al tanto del desarrollo de la culminación de este trabajo.

Gracias a todos.

Contenido

AUTOMATIZACIÓN DE UNA LÍNEA DE EMPAQUE (LLENADO DE PASTILLAS)

OBJETIVOS

Entender la configuración física y modular del equipo de sus diferentes entradas y salidas (digitales o analógicas), procedimientos de instalación de equipo, lenguaje de programación de los controladores de Lógicos Programables enfocados a una aplicación práctica en la industria farmacéutica.

JUSTIFICACIÓN

De primera instancia es concluir con mi formación académica, luego es dar a la gente que consulte dicha Tesis una base de consulta y/o información para sus trabajos en la escuela, trabajo, etc.

Introducción

Capítulo 1

	Pág.
Controladores Lógicos Programables (PLC)	1
1.1 Descripción de Controladores Lógicos Programables.....	2
1.2 Sistemas de Controladores Lógicos Programables.....	2
1.3 Ventajas de los Controladores Lógicos Programables.....	4

Capítulo 2

	Dispositivos del Sistema SLC 500	5
2.1	Familia SLC 500.....	6
2.2	Configuración y arreglo de equipos.....	7
2.3	Configuración modular de equipo.....	10
2.4	Procedimiento de instalación.....	15
2.5	Protocolos de comunicación.....	18
2.6	Lámparas de Diagnósticos de un SLC 500.....	22

Capítulo 3

	Instrucciones de programación del sistema SLC-500	26
3.1	Instrucciones básicas	27
3.1.1	Instrucciones de Bit.....	27
3.1.2	Descripción general de las instrucciones de bit.....	28
3.1.3	Instrucciones inmediatas I/O.....	31
3.1.4	Enclavamiento de salida (OTL) y desenclavamiento de salida (OTU).....	32
3.1.5	One Shot Rising (OSR).....	33
3.2	Instrucciones de temporizador y contadores	33
3.2.1	Descripción general de las instrucciones de temporización.....	34
3.2.2	Descripción general de contadores hacia arriba y hacia abajo....	39
3.2.3	Contador de alta velocidad (HSC).....	42
3.2.4	Restablecimiento (RES).....	45
3.3	Instrucciones de matemáticas, comparación y de movimiento	46
3.3.1	Descripción general de las Instrucciones matemáticas.....	47
3.3.2	Descripción general de las Instrucciones lógicas de comparación.....	59
3.3.3	Descripción general de las Instrucciones de Movimiento.....	62

3.4	Instrucciones de flujo de programa	64
3.4.1	Salto (JMP) y etiqueta (LBL).....	64
3.4.2	Saltar a subrutina (JSR), subrutina (SBR), y retornar (RET).....	65
3.4.3	Restablecimiento de control maestro (MCR).....	68
3.4.4	Fin temporal (TND).....	69
3.4.5	Suspender (SUS).....	69

Capítulo 4

	Automatización de una línea de Llenado de Pastillas	71
4.1	Planteamiento del control y automatización de línea de Llenado.	72
4.2	Programación para el control y automatización de la línea de empaque	76
4.3	Diagramas de conexión punto a punto.	87
4.4	Diagramas eléctricos.	93

Capítulo 5

	Conclusiones	120
	Bibliografía	122

INTRODUCCION

En la actualidad las empresas debido a la alta competitividad requieren ir mejorando sus sistemas de producción para las exigencias del mercado y por lo cual requieren de métodos que le ayuden a optimizar tiempos, planificar, clasificar sus productos, tener un mínimo de errores que lleven a una pérdida de tiempo. Que representa una pérdida cuantiosa de dinero en caso de tener paro en sus equipos.

Hasta el momento la automatización de maquinas y procesos han permitido mejorar la productividad, han reducido los costos, se obtiene en un menor tiempo, no se requiere de mucha gente en la líneas de producción ni de empaque, se obtiene productos en Stock.

Por eso es que debido al avance de la tecnología se ha venido sustituyendo del control convencional de botoneras, timer, relevadores, contadores como dispositivos individuales y circuitería para realizar un proceso específico a sistemas de control automatizado que cuenta con un sistema de módulos de entradas y salidas para las diferentes variables de señales analógicas o digitales y que mediante un procesador realiza diferentes funciones de control de automatizado que puede ser programado una y otra vez para que realicen una secuencia lógica. Además cuenta internamente con Timer, contadores, expresiones matemáticas, lógicas, etc, que pueden ser usadas para llevar a cabo la programación y ser guardados en la memoria del procesador.

Es por esto que quise escribir un poco acerca de los beneficios de un Sistema Lógico Programable y llevarlo a cabo con un proceso real que es la Automatización de la Llenadora de Pastillas, de una Línea de Empaque.

CONTROLADORES LÓGICOS PROGRAMABLES (PLC)

Cuando hablamos de los controladores lógicos programables (PLC) en la mayoría de los casos se desconoce su uso, para que sirve, en que nos ayuda y en realidad a simple vista no se tiene una aplicación práctica en nuestra vida cotidiana, sin embargo está más ligada a nosotros sin darnos cuenta; por ejemplo: El ensamble de la televisión que casi todos tienen en su casa fue realizado por un Autómata Programable, el de un automóvil, estéreo, refrigeradores, estufas, y casi todos los equipos electrónicos y electrodomésticos; en las grandes industrias es donde mayormente son usados por ejemplo en el llenado de bebidas o gaseosas, en la fabricación empacado de medicinas, etc. Aun en las computadoras que muchos usamos también fueron ensamblados por autómatas.

Sin menospreciar algunas otras aplicaciones o utilidades de los Controladores Lógicos Programables (PLC), en este capítulo se dará una descripción de manera general de cómo está constituido, y algunas ventajas con respecto a los Sistemas de Control tradicionales. Además trataremos de darle una aplicación de una manera práctica y explicar el porqué fue un tema de mi Tesis.

1.1 Descripción de los controladores Programables

Los primeros Controladores Lógicos Programables (PLC's) fueron introducidos alrededor 1970, en las industrias automotivas ya que en sus comienzos fueron requiriendo monitorear y verificar los paros arriba de un mes. Con la ayuda de los PLC's se resolvieron más rápidamente estos inconvenientes además era mucho más rápido las modificaciones en el sistema de control; por lo tanto se les fue adquiriendo más confianza y fueron adquiriendo más fuerza su uso en muy poco tiempo.

Los primeros controladores lógicos programables fueron designados para reemplazar los gabinetes de control convencional basados básicamente en relevadores. Ahora ¿por qué los PLC's pasan a sustituir los sistemas de control convencional?; porque su instalación es más sencillas, ocupa considerablemente menos espacio y energía, tiene LED's de modos de diagnósticos, operación y falla, pero lo más importante son programables. Por lo cual su popularidad creció rápidamente en la Industria de Sistemas de Control y como su captación prospero, la industria requirió mucha más capacidad de entradas y salidas I/O, más memoria y en adición más funciones. En los modernos PLC's ha sido posible incrementar su potencia para casi cualquier sistema de control requerido.

La General Electric Company enuncia la siguiente definición para un Controlador Programable:

“... es una *computadora industrial* que acepta entradas de interruptores y sensores, evalúa estas entradas en acorde con el *programa cargado*, y genera salidas para controlar maquinas y procesos.”

De esta definición nosotros podemos deducir que un PLC es una computadora que puede ser programada para perfeccionar mucho más las operaciones de los sistemas de control; que con los sistemas de control convencional no era posible a menos que los paneles fuesen recableados, o las tarjetas de circuitos impresos deberían ser construidas nuevamente cuando se requiera realizar una modificación en la secuencia del Sistema de Control. Con un PLC la modificación se puede realizar fácilmente conectando un programador y presionando algunas teclas para la modificando del programa. Ahora supongamos que una máquina no es tan requerida, el PLC puede ser removido y rehusado en alguna otra máquina, siendo una computadora industrial el PLC puede operar propiamente en el calor, en un ambiente sucio y en un ambiente de ruido eléctrico que es muy común encontrarlo en muchas de las industrias, sin embargo; es necesario hacer mención que un PLC maneja señales de comunicación que requieren una conducción independiente de las demás señales.

1.2 Sistemas de controladores lógicos programables.

En la figura 1.1 se muestra un diagrama a bloques de un Sistema de Control Lógico Programable. Todos las partes requeridos que componen un PLC están ubicados dentro de línea punteada. El programador es únicamente requerido durante la programación, cuando se tiene alguna falla o se requiere realizar alguna modificación en el programa.

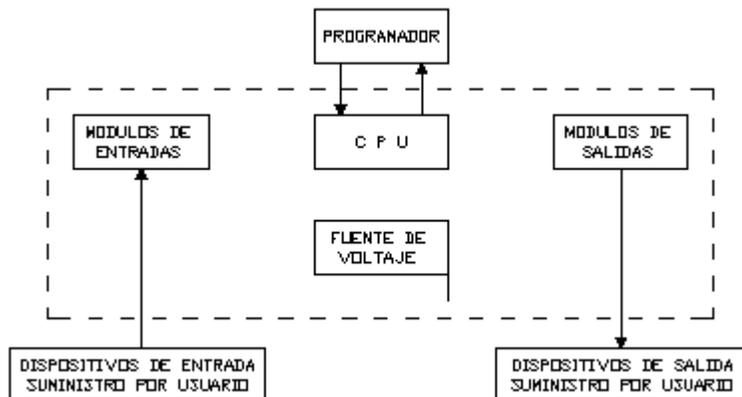
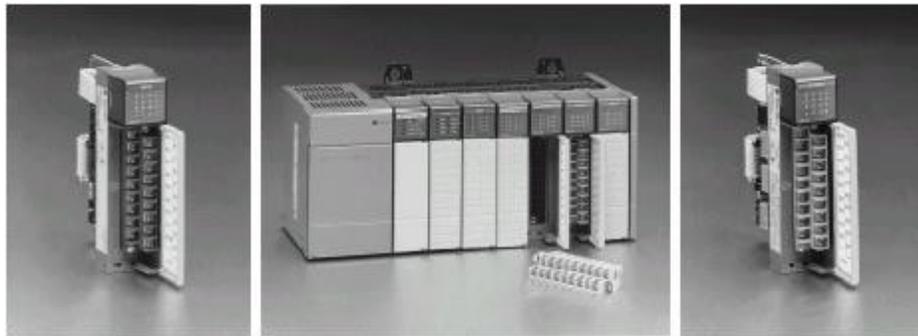


Figura 1.1 Diagrama a bloques de un Sistema de Control Lógico Programable.

La Unidad Central de Procesamiento (CPU) es el cerebro o el corazón del sistema, la cual está compuesta o contiene Acceso de Memoria Aleatoria (RAM) para el almacenamiento de programas y datos, tiene una Memoria de Lectura Únicamente (ROM) la cual sirve para el almacenamiento de las operaciones del sistema y una Unidad Micro procesadora. El CPU monitorea el estado de señales de entrada del sistema, resuelve la lógica del programa y controla las salidas del sistema.

Los Módulos de entrada y salida forman la interface entre CPU y el “mundo exterior” (mediante el uso de señales de entrada o salida de los dispositivos usados). Existen diferentes tipos de entradas o salidas las cuales serán habilitadas dependiendo del uso de la señal o potencia requerida. La figura 1.2 muestra algunos de los módulos que son habilitados:



En la figura 1.2 se muestra a los extremos módulos de entrada/salidas habituales para la conexión de señales externas, al centro del dibujo se observa el PLC (SLC 500) completo con su chasis, fuente de voltaje, procesador y módulos de entrada / salida.

Los módulos de entrada/salida condicionan las señales que entran y las que salen del PLC. Estos módulos convierten las señales de entrada y salida a un Bajo de Voltaje de DC usado por el Bus de datos del CPU y filtra la salida de cualquier pico, interferencia, o cualquier otro ruido eléctrico que pueda ser presentado en la línea. Las señales de salida son físicamente

aisladas desde el CPU mediante el uso de opto-acopladores. (Los módulos de I/O serán discutidos en el capítulo 2).

La fuente de Voltaje suministra el Voltaje de DC requerido para el CPU, el programa y los módulos de I/O. Cabe hacer mención que los módulos requieren de una Fuente de Voltaje Externa.

1.3. Ventajas de los controladores lógicos.

Podemos deducir que encontraremos muchas más ventajas en el uso de un PLC, que de algún otro sistema de control con las mismas características. Sin embargo cabe mencionar que si un *sistema de control* es demasiado pequeño tendríamos que analizar algunas características como podrían ser su costo. A continuación enunciaremos algunas de las ventajas del uso de PLC

- **ES PROGRAMABLE.** Comenzando con que un PLC es programable es un simple y gran beneficio. Recargable para una modificación día a día del control requerido para el proceso, de ser necesario, lo cual puede realizarse fácilmente sin tener que modificar el cableado de campo.
- **ES FLEXIBLE.** Luego de que los PLC's sean usados, para diferentes máquinas es posible que requieran diferentes controladores. Por lo que un solo PLC puede controlar varias máquinas y todas al mismo tiempo si fuese necesario.
- **CONFIABLE.** Comenzando con que es un dispositivo de estado sólido, un PLC es más confiable que un relevador mecánico o eléctrico y que los relevadores de tiempo.
- **REUSABLE.** Si una máquina no es tan requerida, el PLC puede ser usado en otra máquina.
- **MANTENIMIENTO.** Los PLC poseen modos de diagnósticos en adición a alguna falla o problema. Los módulos de entrada / salidas son fácilmente removidos y reemplazados.
- **ECONÓMICOS.** Un PLC con numerosos relevadores, timers, contadores, secuenciometros y algunas otras funciones su costo es mucho menor que de algunos dispositivos similares.
- **TAMAÑO PEQUEÑO.** Un PLC requiere de mucho menos espacio que otro tipo de sistema de control.

Capítulo

2

DISPOSITIVOS DE UN SISTEMA SLC 500

No importa que tan sencillo o complicado pueda ser la programación de un PLC, se requiere conocer lo fundamental en los arreglos del mismo, tenemos que tener una idea clara de cómo está constituido, como está formada la Familia de un SLC 500, cual es su configuración, cual es su arreglo modular. Todo esto es de suma importancia ¿por qué? Supongamos que se quiere comercializar el uso del PLC a una Empresa, como lo lograremos si nosotros no conocemos ni lo más mínimo en el equipo, como su fuente de poder ya sea alterna o directa, los tipos de chasis, las combinaciones en tarjetas de entrada y salidas, si existen módulos para aplicaciones especiales y aun más importante cuáles son sus limitaciones, por lo que es de suma importancia explicar cómo está constituida esta familia. Aunado a esto describiremos en forma breve como interpretar las lámparas de diagnósticos de un SLC 500, cuando está operando correctamente, cuando se presenta alguna falla en el procesador, etc. Así mismo, como la forma correcta de instalar un procesador, cuales son los cuidados se deben tener en cuenta y finalmente hablaremos un poco acerca de los protocolos de comunicación Protocolo RIO, Protocolo Data High Plus (DH+), etc.

2.1. Familia SLC 500

La familia de controladores programables SLC 500 después de la flexibilidad y la potencia de su programado son uno de los más pequeños y económico de la familia de los PLC's. La familia consiste de los siguientes controladores:

	ARREGLO DE EQUIPO	EQUIPO MODULAR			
	SLC-500	SLC-5/01	SLC-5/02	SLC-5/03	SLC-5/04
CPU	SLC-500	SLC-5/01	SLC-5/02	SLC-5/03	SLC-5/04
MEMORIA BASE	1K	1 ó 4K	4K	12K	12, 60K
PUNTOS I/O	20,30 ó 40 72 Max. w/exp	256	480	960	960
EEPROM	1 ó 4K	1 ó 4K	1 ó 4K	32 ó 64K	32 ó 64K
AUMENTAR INSTRUCCIONES	No	No	Sí	Sí	Sí
PROGRAMACIÓN EN LÍNEA	No	No	No	Sí	Sí
PUNTO DE CONEXIÓN EN DH+	No	No	No	No	Sí



Figura 2.1. Módulos Procesadores (CPU), SLC 5/02, SLC 5/03 y SLC 5/04

En la figura 2.1 se puede observar los diferentes procesadores de la familia SLC que además se requiere de algunos otros componentes que integran esta familia y que se enuncia a continuación:

- Chasis
- Fuente de Voltaje
- Módulos I/O
- Software de programación.

2.2. Configuración y arreglo de equipo.

Un PLC está constituido por diferentes partes las cuales tienen una finalidad determinada, en la figura 2.2 se enuncian cada de ellas para que nos familiaricemos y aprendamos como está conformado:

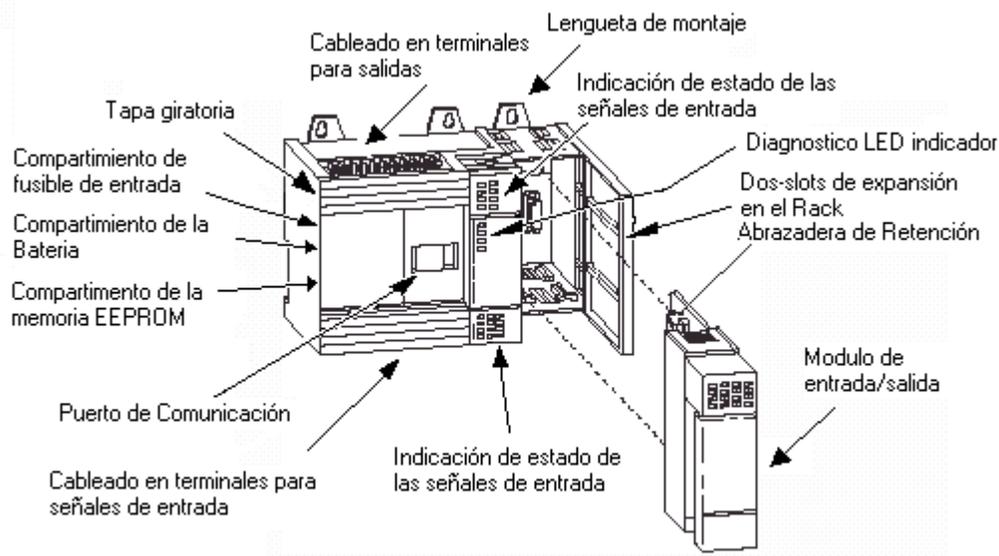


Figura 2.2 Partes que conforman el PLC (SLC-500)

El arreglo del equipo del SLC 500 incluye un Procesador Central (CPU) con una capacidad de unas 1000 instrucciones, una Fuente de Voltaje integrada, y un arreglo numérico de puntos de conexión en Entradas / salidas en un paquete completo.

Esta versión está habilitada en 20, 30 y 40 puntos de Entradas / Salidas, además todas estas unidades aceptan dos Slots de expansión en el Rack, como se muestra en la figura 2.3.

Cada dispositivo tiene un LED indicador para mostrar el estado de cada una de las Entradas/Salidas.

La fuente de Voltaje interna del SLC 500 para el respaldo del Procesador puede operar en 120/240 volts AC o 24 volts DC.

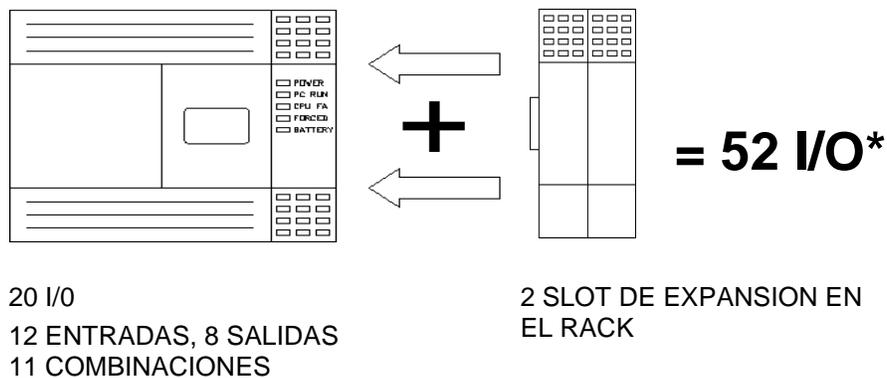
Además tiene cinco LED's de diagnóstico en el frente de cada Procesador para indicar:

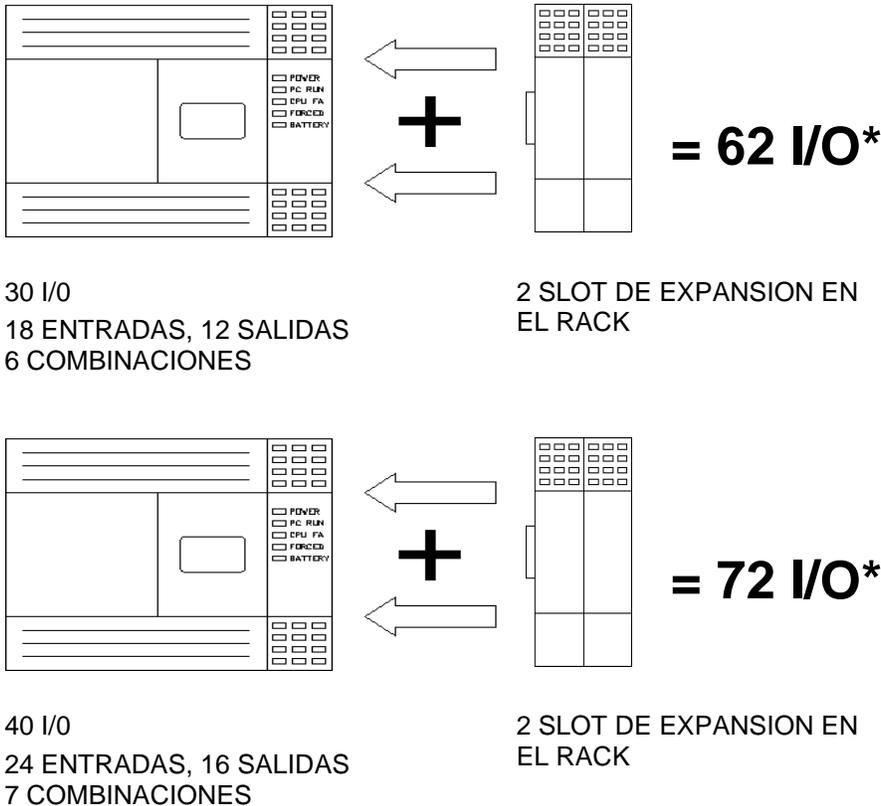
- POWER
- PC RUN
- CPU FAULT
- FORCED I/O
- BATTERY LOW

El Contador de Alta Velocidad opcional. El arreglo de la unidad que posee 24 volts de DC en el circuito de entrada es equipado con un Contador de Alta Velocidad. Usted posee la opción (colocar un clip o un puente) de usar la primera entrada como un Contador de Alta Velocidad (8Khz máximo). La instrucción para el Contador de Alta Velocidad es una instrucción designada para tomar ventaja de su especial característica.

Habilitación del uso de la fuente. En los arreglos de equipos que requieren de acometida de Voltaje en AC, poseen 24 volts de DC, 200mA de corriente para disponibles para su uso. Este voltaje de DC puede ser usado convenientemente para conectar la entrada de un sensor. El SLC 500 es fácilmente montado sobre su panel en un riel DIN con una adición de un adaptador en el riel DIN.

Las fuentes de voltaje son protegidas por sobrecarga y es caso de ser necesario se reemplaza el fusible interno dentro SLC 500.





* El número máximo es de 16 salidas a relevador en el Rack de Expansión.

Figura. 2.3. SLC 500 con 20, 30 y 40 puntos de conexión de I/O y dos SLOT de expansión en el Rack.

Como se pudo observar en la figura 2.3 estos son los tres tamaños diferentes de I/O que proveen 24 líneas de potencia diferentes de I/O a configurar.

- (11) 20 I/O - 12 Entradas y 8 Salidas.
- (6) 30 I/O - 18 Entradas y 12 Salidas.
- (7) 40 I/O - 24 Entradas y 16 Salidas.

En todos los estilos de arreglos en equipos aceptan dos Slots opcionales en el Rack de expansión que provee 32 puntos adicionales de Entradas/Salidas.

Especificaciones de arreglo de equipo

Configuración 20 I/O

Numero de catalogo	Entrada	Salida	Tipo de entrada	Tipo de salida	Suministro de voltaje	Contador de Alta Velocidad	Uso de potencia
1747-L20A			120 VAC	Relevador	120/240 VAC	No	-
1747-L20B				Triac	120/240 VAC	No	-
1747-L20C				Relevador	120/240 VAC	Sí	24 V-200mA

1747-L20D	12	8	24 VDC Sinking	Triaca	120/240 VAC	Sí	24V-200mA
1747-L20E				Transistor Source	120/240 VAC	Si	24V-200mA
1747-L20F				Relevador	24 VDC	Sí	-
1747-L20G				Transistor Source	24 VDC	Sí	-
1747-L20L			24 VCD Sourcing	Transistor Sink	120/240 VAC	Sí	24V-200mA
1747-L20N				Transistor Sink	24 VDC	Sí	-
1747-L20P			240 VAC	Triac	120/240 VAC	No	-
1747-L20R				Relevador	120/240 VAC	No	-

Configuración 30 I/O

Numero de catalogo	Entrada	Salida	Tipo de entrada	Tipo de salida	Suministro de voltaje	Contador de Alta Velocidad	Uso de potencia
1747-L30A	18	12	120 VAC	Relevador	120/240 VAC	No	-
1747-L30B				Triac	120/240 VAC	No	-
1747-L30C			24 VDC Sinking	Relevador	120/240 VAC	Sí	24 V-200mA
1747-L30D				Triac	120/240 VAC	Sí	24V-200mA
1747-L30L			24 VCD	Transistor Sink	120/240 VAC	Sí	24V-200mA
1747-L40P			240 VAC	Triac	120/240 VAC	No	-

Configuración 40 I/O

Numero de catalogo	Entrada	Salida	Tipo de entrada	Tipo de salida	Suministro de voltaje	Contador de Alta Velocidad	Uso de potencia
1747-L40A	24	16	120 VAC	Relevador	120/240 VAC	No	-
1747-L40B				Triac	120/240 VAC	No	-
1747-L40C			24 VDC Sinking	Relevador	120/240 VAC	Sí	24 V-200mA
1747-L40D				Triac	120/240 VAC	Sí	24V-200mA
1747-L40E				Transistor Source	120/240 VAC	Sí	24V-200mA
1747-L40F			24 VCD	Relevador	24 VDC	Sí	-
1747-L40L				Transistor Sink	120/240 VAC	Sí	24V-200mA
1747-L40P				240 VAC	Triac	120/240 VAC	No

2.3. Configuración Modular del Equipo

Para muchas aplicaciones se requiere de *flexibilidad*, y los equipos modulares ofrecen una gran variedad de opciones en Entradas/Salidas. Usted puede optimizar su sistema de control mediante la selección de sus componentes si conoce y especifica de acuerdo a sus requerimientos. El sistema modular es habilitado en versiones de 4, 7, 10 y 13 slots. El

sistema modular puede ser configurado con un máximo de tres Racks (30 slots en total) a un máximo de 256 puntos de Entradas/Salidas.

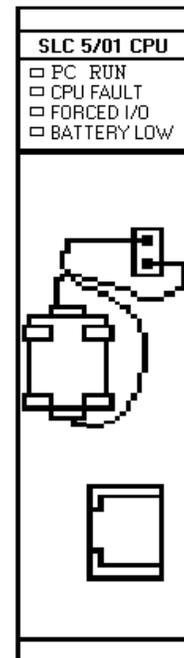
Una gran variedad de I/O voltajes para puntos individuales son habilitados para auxiliar y crear una justa y apta aplicación.

Procesador SLC 5/01

Características de equipo
(1747-L511 y 1747-L514)

El procesador SLC 5/01 provee:

- Memoria de programa opcional de 1K o 4K en palabras (4K o 16 K de datos de palabra respectivamente).
- Direccionamiento de arriba de 256 I/O locales.
- Respuesta de comunicación DH485 punto a punto.
- Para su programación se usa una Terminal Hand-Held (HHT) o un software de programación.
- Inscrito en UL, aprobado CSA, cumple con todos los mandatos CE cuando el producto o paquete es marcado.



Procesador SLC 5/02

Características de equipo
(1747-L524 series B y series C)

El procesador SLC 5/02 presenta un aumento en las instrucciones, incrementando la posibilidad de diagnóstico, y expande la su capacidad de comunicación. El SLC 5/02 posee las siguientes características:

- Memoria de programa de 4K (16 K de datos de palabra).
- Máximo direccionamiento de 480 I/O locales, expandible vía remota I/O o mediante Device Net.
- Máximo direccionamiento de 4000 entradas remotas y 4000 salidas remotas.
- Respuesta de comunicación de DH485 punto a punto.
- Para su programación se usa una Terminal Hand-Held (HHT) o un software de programación.
- Inscrito en UL, aprobado CSA, cumple con todos los mandatos CE cuando el producto o paquete es marcado.

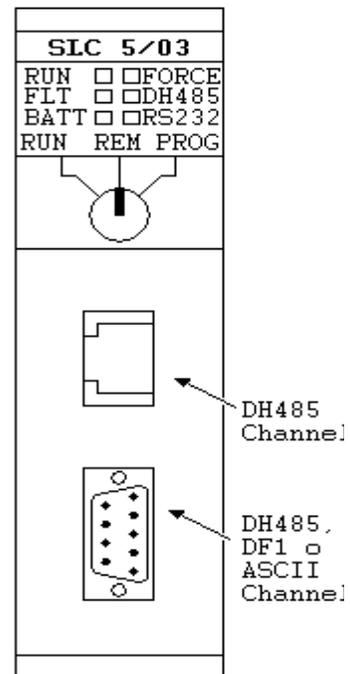


Procesador SLC 5/03

Características de equipo
(1747-L532)

El procesador SLC 5/03 ofrece las siguientes características:

- Memoria de programa de 12K de palabras y 4K de datos de palabra adicionales.
- Direccionamiento de 960 I/O locales, expandible vía Remota las I/O o vía Device Net.
- Máximo direccionamiento de 4000 entradas remotas y 4000 salidas remotas.
- Respuesta de comunicación de DH485 punto a punto.
- Comunicación RS-232/DF1 o protocolo DH485.
- Comunicación ASCII (Procesador SLC 5/03 OS302 o mayor).
- Protocolo DF1 Half-duplex Master (Procesador SLC 5/03 OS302 o mayor).
- Reloj de tiempo real.
- Programación en línea.
- Direccionamiento indirecto (Procesador SLC 5/03 OS302 o mayor).
- Capacidad Matemática expandible (Procesador SLC 5/03 OS302 o mayor).



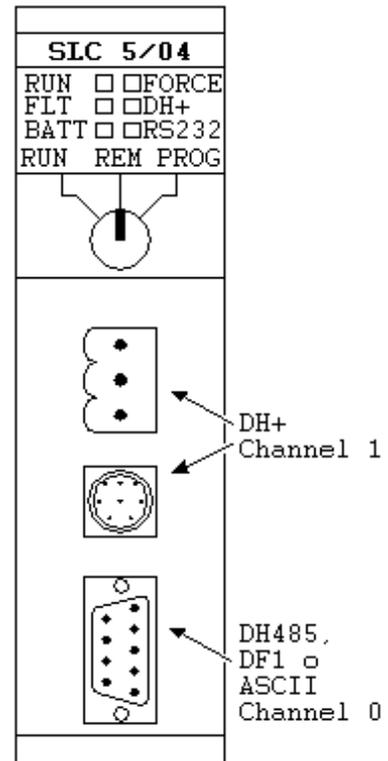
- Adición de instrucciones que calculan, intercambian, y escalan con parámetros (Procesador SLC 5/03 OS302 o mayor).
- Lista de Multi-puntos (Procesador SLC 5/03 OS302 o mayor).
- Inscrito en UL, aprobado CSA, cumple con todas las normas o mandatos de CE cuando el producto o paquete es marcado.

Procesador SLC 5/04
Características de equipo
(1747-L541, 1747-L542 o 1747-L543)

El procesador SLC 5/04 ofrece las siguientes características:

- Memoria de programa de 12K, 28K o 60K de palabras y adicionales 4K de datos de palabra.
- Direccionamiento de 960 I/O locales, expandible vía Remota las I/O o vía Device Net.
- Máximo direccionamiento de 4000 entradas remotas y 4000 salidas remotas.
- Comunicación DH+.
- Comunicación RS-232/DF1 o protocolo DH485 o ASCII.
- Protocolo DF1 Half-duplex Master (Procesador SLC 5/04 OS401 o mayor).
- Reloj de tiempo real.
- Coprocesador matemático.
- Programación en línea.
- Direccionamiento indirecto (Procesador SLC 5/04 OS401 o mayor).
- Capacidad de expansión matemática.
- Adición de instrucciones que calculan, intercambian, y escalan con parámetros (Procesador SLC 5/04 OS401

- o mayor).
- Lista de Multi-puntos (Procesador SLC 5/04 OS401 o mayor).
- Inscrito en UL, aprobado CSA, cumple con todas las normas o mandatos de CE cuando el producto o paquete es marcado.



Rack

El Rack es la casa del CPU y los módulos de I/O. La fuente de voltaje va montada a la izquierda al lado izquierdo del Rack. Todos los componentes son fácilmente deslizables dentro del Rack ya que tienen una guía a lo ancho del mismo. No se requiere de herramientas para insertar y mover el procesador SLC 5/0 o módulo de I/O.

DESCRIPCIÓN	NÚMERO DE CATALOGO
Rack de 4 slots	1746-A4
Rack de 7 slots	1746-A7
Rack de 10slots	1746-A10
Rack de 13slots	1746-A13

En la figura 2.4 se muestra los diferentes tipos de Rack que existen para el SLC 500 hacia adelante, de la marca Allen Bradley:

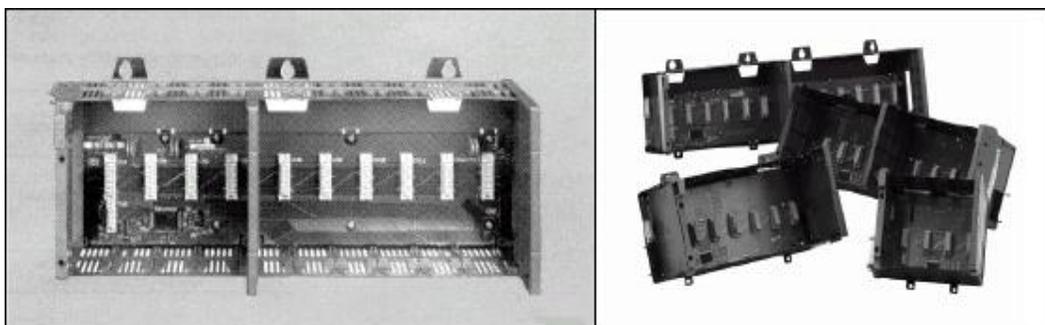


Figura 2.4 Racks, marca Allen Bradley

Fuentes de voltaje.

Las fuentes de voltaje suministran la potencia para el CPU y cada uno de los Slots de I/O. Todas las fuentes de voltaje están protegidas por un fusible, el cual puede ser reemplazado por uno similar en caso de corto circuito, en la figura 2.5 se muestran las diferentes tipos de Fuente de Voltaje con las que cuenta la familia del SLC.

IMPORTANTE:

Cada Rack o sistema modular requiere de una fuente de voltaje.

Para una fuente de AC el voltaje puede ser seleccionado en 120/240 volts esto se realiza mediante un puente (jumper). Simplemente se mueve el puente de lugar de acuerdo al voltaje seleccionado.

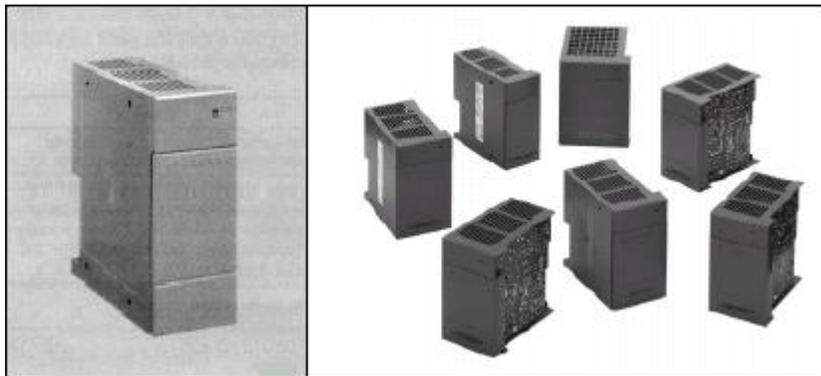


Figura. 2.5 Fuentes de voltaje usados en los procesadores SLC 5/01y posteriores.

IMPORTANTE:

La fuente de voltaje no ocupa un espacio o Slot en el Rack: Este simplemente se monta al lado izquierdo del Rack con dos tornillos.

La fuente de voltaje posee un LED que se ilumina cuando está funcionando correctamente, en la figura 2.6 se muestran algunos típicos de las fuentes, su forma de conexión y protección por fusible etc.

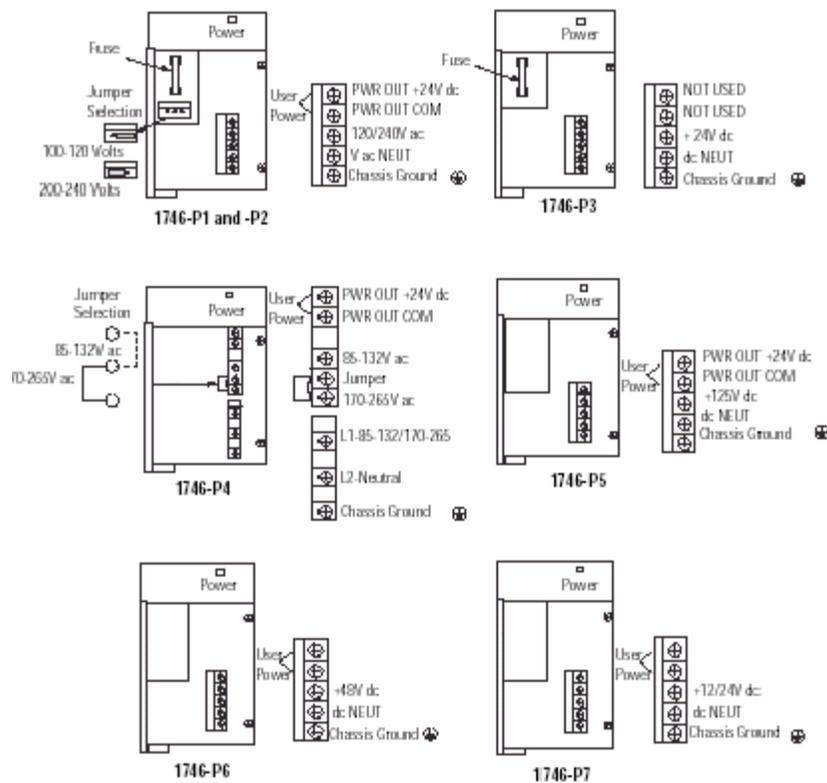


Figura. 2.6 Fuentes de voltaje y conexionado.

2.4. Procedimientos de instalación.

Se puede montar la unidad tipo hardware compacto directamente al panel posterior del envoltorio utilizando las lengüetas de montaje y tornillos #10 y #12. El par máximo requerido es 3.4 N-m (30 lbs-pulg).

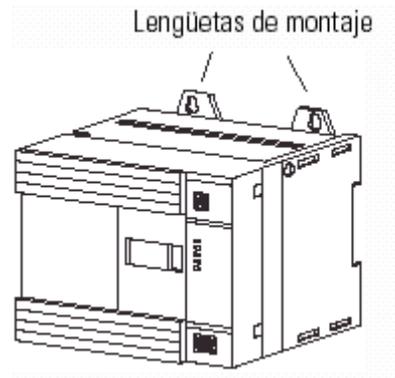
ATENCIÓN!

Tener cuidado con las partículas metálicas cuando se taladre los agujeros en el controlador. Las partículas metálicas puedan causar corto circuito en los componentes electrónicos del controlador y dañarlo.

Los capacitores en los módulos de entrada tienen una carga almacenada que puede causar un choque no letal. Evitar montar el controlador en una posición en la que el personal de instalación o servicio podrían estar en peligro como resultado de una reacción inesperada.

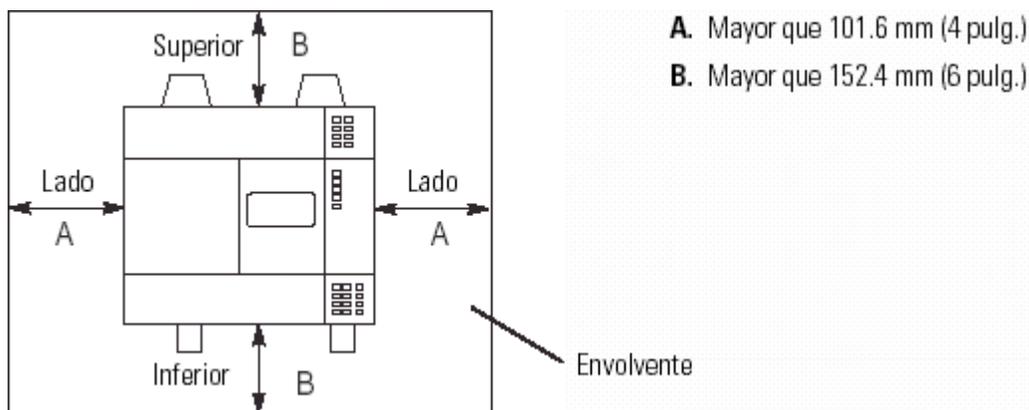
Para instalar el controlador usando los tornillos de montaje:

1. Colocar el controlador en la superficie de montaje. (Asegurar que los espacios para el controlador sean apropiados. Ver la siguiente sección para obtener más información.)
2. Marcar las posiciones de las lengüetas de montaje y taladrar los agujeros necesarios
3. Montar el controlador.



Espacios del controlador.

Seguir la orientación del controlador y los espacios mínimos mostrados a continuación para permitir el enfriamiento por convección dentro del envoltorio. El aire dentro del envoltorio debe mantenerse dentro de los límites de 0°C a +60°C (+32°F a +140°F).



Procedimiento de instalación de la batería

1. Abrir la puerta del procesador.
2. Retirar el puente del conector de la batería. Guardar el puente en un lugar seguro para un posible uso futuro sin la batería.
3. Insertar una batería nueva o de repuesto en el sujetador asegurándose de que quede sujeta con las abrazaderas de retención.
4. Insertar el conector de la batería dentro del conector. Ver la siguiente figura 2.7.

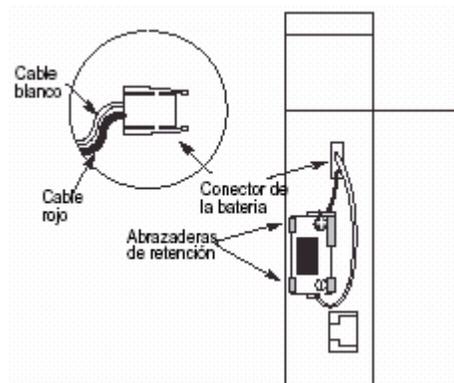


Figura. 2.7 Batería de Respaldo para el procesador.

5. Cerrar la puerta del procesador.

IMPORTANTE:

Si el procesador tiene una batería— la batería es una opción para el procesador SLC 5/01 (1747-L511) — asegurar que esté conectada antes de instalar el procesador dentro del chasis. Esto proporciona memoria de seguridad (backup) para el procesador en caso de que falle el suministro de alimentación eléctrica al controlador.

Instalación del procesador

Asegurarse que la alimentación eléctrica del sistema esté desconectada; luego insertar el procesador en la ranura 0 del chasis 1746.

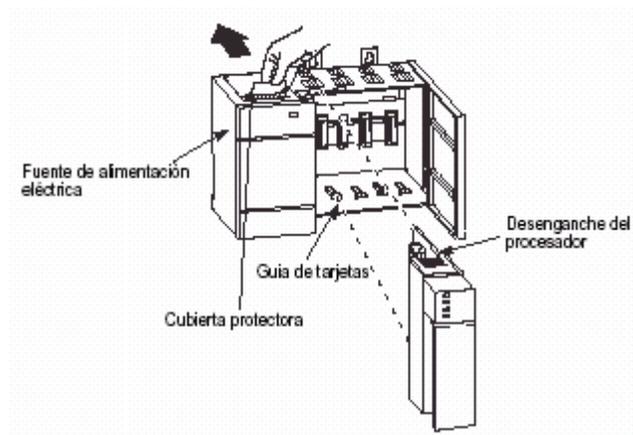


Figura. 2.8 Rack modular para introducir el procesador.

IMPORTANTE:

Los procesadores modulares SLC 500 deben insertarse en la ranura izquierda (ranura 0), como se muestra en la figura 2.8.

Adicionalmente, retirar la cubierta protectora después de instalar el procesador.

2.5. Protocolos de comunicación.

En esta sección comprenderemos las diferencias en los protocolos de comunicación. Existe capacidad para los protocolos siguientes:

- **DH-485**
 Todos los procesadores SLC 500 pueden comunicarse en la red DH-485. Existen varios dispositivos de puente y Gateway para crear un interface del canal SLC 500 DH-485 a otros dispositivos tal como la tarjeta 2760-RB (con el cartucho de protocolo 2760 SFC3), 1770-KF3, 1747-KE y 1785-KA5. Cuando se usea los dispositivos de puente o Gateway, referirse a la documentación de usuario específica para configurar su sistema.
- **DH+**
 El SLC 5/04 tiene capacidad para la comunicación y conectividad DH+ a una red DH+.
- **Full-dúplex DF1 y maestro/esclavo DF1**
 Los procesadores SLC 5/03 y SLC 5/04 y el controlador MicroLogix 1000 tienen capacidad para los protocolos DF1 desde sus conexiones RS-232.
- **ASCII**
 Los procesadores SLC 5/03 OS301, OS302 y SLC 5/04 OS400, OS401 tienen capacidad para el protocolo ASCII definido por el usuario.
- **Uso de las características de transferencia**

Protocolo de comunicación DH-485

La red DH-485 ofrece:

- la interconexión de 32 dispositivos
- capacidad de maestros múltiples
- control de acceso de paso de testigo
- la capacidad de añadir o eliminar nodos sin perturbar la red
- una longitud de red máxima de 1219 m (4,000 pies)

Protocolo de la red DH-485

La sección siguiente describe el protocolo usado para controlar transferencias de mensaje en la red DH-485. El protocolo tiene capacidad para dos clases de dispositivos: iniciadores y contestadores. Todos los iniciadores en la red tienen la oportunidad de iniciar transferencias de mensaje. Se usa un algoritmo de paso de testigo para determinar cuál iniciador tiene el derecho de transmitir.

Rotación del testigo DH-485

Un nodo que retiene el testigo puede enviar paquetes válidos a la red. El parámetro de retención de testigo determina el número de transmisiones (más reintentos) cada vez que el nodo recibe el testigo.

Después que un nodo envía un paquete de mensaje, intenta dar el testigo a su sucesor enviando un paquete de “paso de testigo”. Si no ocurre actividad de red, el iniciador intenta encontrar un sucesor nuevo.

El rango de dirección de nodo para un iniciador es 0-31. El rango de dirección de nodo para todos los contestadores es 1-31. Ha de existir por lo menos un iniciador en la red.

Nota:

La dirección máxima que el iniciador busca antes de ajustar la línea automáticamente a cero es el valor en el parámetro configurable “dirección de nodo máxima”. El valor predeterminado de este parámetro es 31 para todos los iniciadores y contestadores.

Nota:

Los procesadores fijos, SLC 5/01, SLC 5/02, SLC 5/03 y SLC 5/04 no permiten que la dirección de nodo cero se aplique. Si intenta aplicar un cero, la dirección de nodo uno se convierte en la dirección de nodo del procesador. La dirección de nodo de cero es reservada para un dispositivo de programación tal como la terminal portátil (HHT) o computadora personal utilizando software de programación.

Inicialización de la red DH-485

La inicialización de la red comienza cuando un período de inactividad excede el tiempo de un “límite de tiempo sobrepasado de vínculo muerto”. Cuando el “límite de tiempo sobrepasado de vínculo muerto” se excede, generalmente el iniciador con la dirección más baja reclama el testigo.

La construcción de una red comienza cuando el iniciador que reclamó el testigo trata de pasar el testigo al nodo sucesor. Si el intento de pasar el testigo falla, o si el iniciador no tiene un sucesor establecido (por ejemplo, al momento de encendido), comienza una búsqueda lineal de un sucesor a partir del nodo de arriba. Vuelve automáticamente al nodo 0 cuando alcanza el valor de dirección de nodo máximo.

Cuando el iniciador encuentra otro iniciador activo, pasa el testigo a dicho nodo, el cual a su vez repite el proceso hasta que el testigo sea pasado por toda la red al primer nodo. En ese momento, la red estará en el estado de operación normal.

Consideraciones de software

Consideraciones de software incluyen la configuración de la red y los parámetros que se pueden establecer según los requisitos específicos de la red. A continuación aparecen factores de configuración que tienen un efecto importante en el rendimiento de la red:

- el número de nodos en la red
- las direcciones de los nodos
- la velocidad en baudios
- la selección de dirección de nodo máxima
- SLC 5/03 solamente – el factor de retención de testigo
- el número máximo de dispositivos de comunicación.

Protocolo de comunicación de Data Highway Plus

La red Data Highway Plus emplea la comunicación entre dispositivos semejantes con un sistema de paso de testigo para rotar el maestro del vínculo entre un máximo de 64 nodos. Puesto que este método no requiere la encuesta (polling), ayuda a proporcionar un transporte de datos fiable y eficiente. Las características de la red DH+:

- programación remota de los procesadores PLC-2, PLC-3, PLC-5 y SLC 500 en su red
- conexiones directas a los procesadores PLC-5 y terminales de programación industriales
- reconfiguración y expansión fáciles si desea añadir más nodos en el futuro
- una velocidad de comunicación de 57.6 Kbaud

La red DH+ usa límites de tiempo sobrepasados establecidos en fábrica para reinicializar la comunicación de paso de testigo si el testigo se pierde debido a un nodo defectuoso.

Protocolo de comunicación RS-232

Los procesadores SLC 5/03 y SLC 5/04 tienen capacidad para el protocolo de dúplex total DF1 y el protocolo maestro/esclavo de dúplex medio DF1 vía la conexión RS-232 a una computadora principal (usando canal DF1).

Protocolo de full-dúplex DF1

El protocolo full-dúplex DF1 (también conocido como protocolo de punto a punto DF1) se proporciona para aplicaciones en que la comunicación de punto a punto RS-232 es necesaria. Este tipo de protocolo tiene capacidad para transmisiones simultáneas entre dos dispositivos en ambas direcciones. Puede usar el canal 0 como puerto de programación o como puerto de dispositivos semejantes usando la instrucción MSG.

En el modo full-dúplex, el procesador SLC 5/03 ó SLC 5/04 puede enviar y recibir mensajes. Cuando el procesador envía mensajes, lo hace en forma de respuestas incorporadas, las cuales son símbolos transmitidos dentro de un paquete de mensaje.

Cuando el procesador SLC 5/03 ó SLC 5/04 recibe mensajes, sirve como dispositivo final – un dispositivo que detiene la transmisión de paquetes de datos. El procesador no hace caso de las direcciones de destino y fuente recibidas en los paquetes de datos. Sin embargo, el procesador cambia estas direcciones en la respuesta que transmite como respuesta a cualquier paquete de datos de comando que ha recibido.

Ya que los procesadores SLC 5/03 y SLC 5/04 se consideran como “dispositivos finales” (la transmisión del paquete de datos se detiene en el procesador), no se hace caso de las direcciones de destino y fuente en el paquete de datos. Si usa un módem con el canal 0 DF1 en el modo full-dúplex, éste debe tener capacidad de funcionar en el modo de full-dúplex. Típicamente, un módem de marcado se usa para la comunicación por líneas telefónicas.

Protocolo de comunicación ASCII

Los procesadores SLC 5/03 OS301, OS302 y SLC 5/04 OS400, OS401 tienen capacidad para protocolo ASCII definido por el usuario configurando RS-232 (canal 0) para el modo de usuario. En el modo de usuario, todos los datos recibidos se colocan en un búfer. Para obtener acceso a los datos, use las instrucciones ASCII en su programa de escalera. También puede enviar datos de cadena ASCII a la mayoría de los dispositivos añadidos que aceptan el protocolo ASCII.

Nota:

Solamente las instrucciones ASCII se pueden usar cuando el modo de usuario se configura. Si usa una instrucción de mensaje (MSG) que hace referencia a canal 0, ocurrirá un error.

Cómo usar las características de transferencia

Hay tres tipos de transferencia disponibles en los procesadores SLC 5/03 y SLC 5/04. Su operación y bits asociados se describen a continuación.

Transferencia DH+ a DH-485 – (Todos los procesadores SLC 5/04)

Este tipo permite que el SLC 5/04 sirva como puente entre una red DH+ y una red DH-485. Cuando el bit S:34/0 se restablece, los paquetes de comunicación que entran en el canal 0 (configurado para DH-485), los cuales no están destinados para el procesador SLC 5/04, vuelven a ser enviados desde del canal 1 en la red DH+.

Además, los paquetes de comunicación que entran en el canal 1 (DH+), los cuales no están destinados para el procesador SLC 5/04, vuelven a ser enviados desde el canal 0 en la red DH-485. Esta actividad tendrá algún efecto en el tiempo de escán del programa de escalera del procesador SLC 5/04, pero estos efectos no son significativos ya que solamente un paquete de transferencia es encaminado nuevamente durante cada escán.

Transferencia DF1 a DH+ – (Procesadores SLC 5/04 OS401 y posteriores)

Este tipo le permite conectar una computadora al puerto en serie del procesador SLC 5/04 (canal 0 configurada para el full-dúplex DF1) y acceder a cualquier nodo en la red DH+ sin importar la velocidad en baudios de la red DH+. También puede conectar un módem al puerto en serie para discar en cualquier nodo en la red DH+.

Transferencia de E/S remota (Procesadores SLC 5/03 OS302 y SLC 5/04 OS401)

Este tipo permite que el sistema SLC 5/04 sirva como puente entre la red DH+ y la red de E/S remota compatible con el módulo de E/S remotas 1747-SN. La transferencia se habilita cuando el bit S:34/5 se establece. Esto permite que las computadoras personales en la red DH+ carguen y descarguen aplicaciones a dispositivos tales como Panel View 550, Panel View 900 ó Panel View 1200 y Data Liners en la red de E/S remotas.

Consideraciones cuando la transferencia DF1 a DH+ se habilita

Cómo entrar en línea con un procesador SLC 5/04 usando el full-dúplex DF1.

Si desea entrar en línea usando el full-dúplex DF1, asegúrese que la dirección destino bajo la pantalla de configuración en línea de dúplex total esté establecida al canal 1 de la dirección de nodo DH+ del procesador SLC 5/04 destino. Si la dirección destino no se ha establecido y el procesador SLC 5/04 tiene la característica de transferencia de DF1 a DH+ habilitada, los paquetes de comando del software de programación pueden ir a un procesador diferente del procesador SLC 5/04.

Cómo transmitir un mensaje usando el full-dúplex DF1 hacia un procesador SLC 5/04 con la transferencia DF1 a DH+ habilitada

Si el procesador SLC 5/04 receptor tiene la transferencia habilitada, asegúrese que el parámetro del nodo destino esté establecido a la dirección DH+ canal 1 del procesador SLC 5/04.

Cómo transmitir un mensaje usando el full-dúplex DF1 desde un procesador SLC 5/04 con la transferencia DF1 a DH+ habilitada

Si se usa un procesador con la transferencia DF1 a DH+ habilitada para transmitir mensajes desde el canal 0 (configurado para el full-dúplex DF1), asegúrese que la dirección del nodo DH+ del procesador SLC 5/04 aparezca en la dirección fuente DF1 bajo la pantalla de configuración de modo de sistema de canal 0. Si la dirección no se establece correctamente, las respuestas retornando al procesador SLC 5/04 pueden ser enviadas a otros nodos en la red DH+.

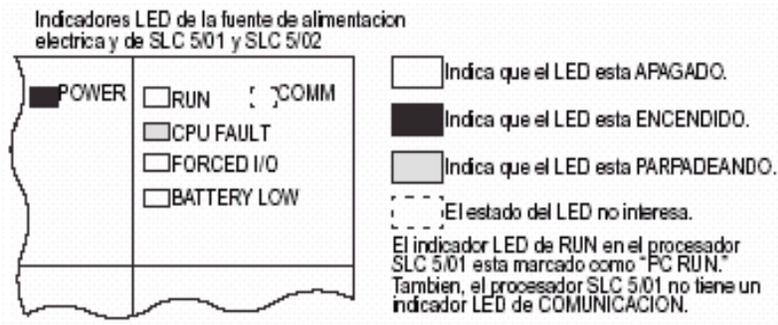
Cómo comunicar desde un procesador SLC 5/04 usando direccionamiento PLC-2

Si se usa un procesador SLC 5/04 con la transferencia DF1 a DH+ habilitada y se trata de transmitir un mensaje desde el canal 0 usando las instrucciones de MENSAJE, no usar el tipo de mensaje 485 CIF. Usar los tipos de mensaje 500CPU o PLC5. Si se intenta usar el tipo de mensaje 485 CIF, el procesador SLC 5/04 que transmite el mensaje no recibirá respuestas del nodo con el que está tratando de comunicarse.

2.6. Lámparas de diagnóstico de un SLC 500

Conectar la alimentación eléctrica al procesador

1. Energizar la fuente de alimentación eléctrica del chasis.
2. Revisar los indicadores LED de la fuente de alimentación eléctrica y del procesador. El indicador LED de power (alimentación eléctrica) en la fuente de alimentación eléctrica debe estar encendido y el indicador LED de fault (fallo) en el procesador debe estar parpadeando.



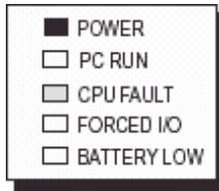
ATENCIÓN!

Alejar al personal del controlador y equipo cuando se aplique alimentación eléctrica. Es necesario que haya una persona lista para operar un interruptor de parada de emergencia en caso que sea necesario desconectar la alimentación eléctrica al equipo del controlador. También, consulte la Parte II de NFPA 70E para obtener pautas adicionales de seguridad relacionadas con las prácticas laborales.

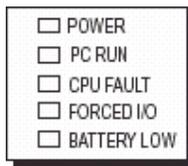
Nunca se debe estar dentro de la maquinaria para activar un interruptor ya que se puede producir un movimiento inesperado y causar lesiones.

Identificación de los mensajes LED del controlador compacto

1. Aplicar alimentación eléctrica al controlador.
2. Comparar los indicadores LED del controlador con los indicadores LED ubicados a la izquierda de cada tabla
3. Leer tabla para identificar la causa probable.
4. Seguir los pasos de acción recomendados para cada causa probable hasta que la causa sea identificada.



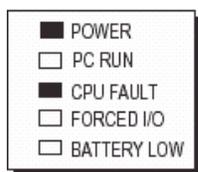
Causa probable	Acción recomendada
Condición de encendido inicial de fábrica del CPU.	<ol style="list-style-type: none"> 1. Si los procedimientos de encendido en el Fixed Hardware Style Controllers Installation and Operation Manual. 2. Cargar el programa de lógica de escalera Para desactivar el LED de CPU Fault
No hay alimentación eléctrica de línea	Verificar el voltaje de línea y las conexiones de los terminales de alimentación eléctrica sean los adecuados.



Causa probable	Acción recomendada
No hay alimentación eléctrica de línea.	Verificar el voltaje de línea y las conexiones de los terminales de alimentación eléctrica sean los adecuados.
Se quemó el fusible de la fuente alimentación Eléctrica.	<ol style="list-style-type: none"> 1. Verificar que las conexiones de la alimentación eléctrica de entrada estén bien. 2. Revisar el fusible de alimentación eléctrica de entrada. Si el fusible se quema otra vez, cambie el controlador compacto.

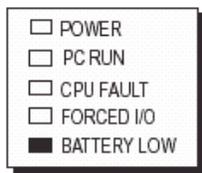
Tipo de error:
La alimentación eléctrica del sistema es inadecuada.

Fuente de alimentación eléctrica Sobrecargada	Este problema puede ocurrir intermitentemente si la fuente de alimentación eléctrica está ligeramente sobrecargada cuando varía la carga de entrada y la temperatura. Si está usando un chasis de 2 ranuras, revise la compatibilidad del módulo para evitar sobrecargar la alimentación eléctrica del backplane.
---	---



Causa probable	Acción recomendada
Error de memoria de la CPU o CPU con fallo	Apagar y volver a encender. Si vuelve a aparecer el indicador LED de CPU FAULT, cambiar el controlador.

Tipo de error: Falla de la CPU.	Módulo de memoria con fallo	1. Desconectar la alimentación eléctrica y luego retirar el módulo de memoria del controlador. 2. Volver a energizar el controlador. Si un LED de CPU FAULT cambia a parpadeante, cambiar el módulo de memoria parpadeante, cambiar el módulo de memoria existente con un módulo de reemplazo.
	El firmware del procesador está instalado incorrectamente	Si se está actualizando el procesador a un nivel de firmware diferente, asegurarse de que la orientación de la abrazadera del firmware sea igual a las direcciones de la unidad de actualización.



Independientemente de las condiciones de cualquier otro indicador de estado LED, siempre cambiar la batería cuando el LED de BATTERY LOW esté encendido, si desea batería RAM de respaldo

Consultar la siguiente guía para determinar el estado de los indicadores LED:

- Indica que el LED está pagado
- Indica que el LED está encendido
- Indica que el LED está parpadeando

Fallos del procesador SLC

Los fallos del procesador se dividen en los tipos siguientes:

- errores de encendido
- errores de ida a marcha
- errores de marcha
- errores de instrucción del programa de usuario

Errores de encendido

Código de error (hex)	Descripción	Causa probable	Acción recomendada
0001	Error NVRAM.	<ul style="list-style-type: none"> • Ruido, • relámpago, • conexión incorrecta a tierra, falta de supresión de sobretensión en las salidas con cargas inductivas o • fuente de alimentación eléctrica insuficiente. • Pérdida de batería o condensador auxiliar. 	<p>Corregir el problema, volver a cargar el programa y ejecutar.</p> <p>Podemos usar la característica de carga automática con un módulo de memoria para volver a cargar automáticamente el programa y entrar en el modo de marcha.</p>
0002	Límite de tiempo sobrepasado inesperado del control (watchdog) de hardware	<ul style="list-style-type: none"> • Ruido, • relámpago, • conexión incorrecta a tierra, falta de supresión de sobretensión en las salidas con cargas inductivas o • fuente de alimentación eléctrica insuficiente 	<p>Corregir el problema, volver a cargar el programa y ejecutar. Puede usar la característica de carga automática con un módulo de memoria para volver a cargar automáticamente el programa y entrar en el modo de marcha.</p>
0003	Error de memoria del módulo de memoria. Este error también puede ocurrir cuando se entra en el modo de marcha REM.	El módulo de memoria está alterado.	Volver a programar el módulo de memoria. Si el error persiste, reemplazar el módulo de memoria.
0007	Fallo durante la transferencia del módulo de memoria.	El módulo de memoria está alterado.	Volver a programar el módulo de memoria. Si el error persiste, reemplace el módulo de memoria.

Capítulo

3

INSTRUCCIONES DE PROGRAMACIÓN

Siempre es importante saber las instrucciones de programación más, sin embargo, si no se tiene los conocimientos mínimos de control, neumática, hidráulica, mecánica, eléctrica, instrumentación (sin menospreciar otras disciplinas). Además de una cierta intuición en la lógica de operación de los equipos y maquinas industriales difícilmente se podrán aplicar. Aún así es necesario conocerlas para un apoyo de nuestro conocimiento ya que no es importante saber todo de memoria sino saber aplicarlas.

3.1. Instrucciones básicas

Este capítulo contiene información general acerca de las instrucciones generales y explica cómo funcionan en su programa de aplicación. Cada una de estas instrucciones básicas incluye información acerca de:

- cómo aparecen los símbolos de instrucción
- cómo usar la instrucción

3.1.1. Instrucciones de bit

Instrucción		Propósito	Página
Mnemónico	Nombre		
XIC	Examina si es cerrado	Examina un bit para una condición activada	31
XIO	Examina si es abierto	Examina un bit para una condición desactivada.	31
OTE	Conecta la salida.	Activa o desactiva un bit.	31
OTL y OTU	Enclavamiento da salida y des enclavamiento de salida	OTL activa un bit cuando el renglón está ejecutado y este bit retiene su estado cuando el renglón no está ejecutado u ocurre un ciclo de potencia. OTU desactiva un bit cuando el renglón está ejecutado y este bit retiene su estado cuando el renglón no está ejecutado o cuando ocurra un ciclo de alimentación eléctrica.	32
OSR	Un frente ascendente	Ocasiona un evento una sola vez.	33
TON	Temporizador a la conexión	Cuenta los intervalos de la base de tiempo cuando la instrucción es verdadera.	36
TOF	Temporizador a la desconexión	Cuenta los intervalos de la base de tiempo cuando la instrucción es falsa.	37
RTO	Temporizador retentivo	Cuenta los intervalos de la base de tiempo cuando la instrucción es verdadera y retiene el valor acumulador cuando la instrucción se hace falsa o cuando ocurra un ciclo de alimentación eléctrica.	38
CTU	Conteo progresivo	Incrementa el valor acumulador a cada transición de falso a verdadero y retiene el valor acumulador cuando la instrucción se hace falsa o cuando ocurra un ciclo de alimentación eléctrica.	41
CTD	Conteo regresivo	Disminuye el valor acumulado a cada transición de falso a verdadero y retiene el valor acumulador cuando la instrucción se hace falsa o cuando ocurra un ciclo de alimentación eléctrica.	42
HSC	Contador de alta Velocidad	Cuenta los impulsos de alta velocidad de una entrada de alta velocidad de controlador fijo.	43
RES	Restablecimiento	Pone a cero el valor acumulado y los bits de estado de un temporizador o contador. No usar con temporizadores TOF.	46

Acerca de las instrucciones básicas

Las instrucciones, cuando se usan en programas de escalera, representan circuitos de lógica cableados usados para el control de una máquina o equipo. Las instrucciones básicas se dividen en tres grupos: bit, temporizador y contador.

Antes de aprender acerca de las instrucciones en cada uno de estos grupos, es recomendable leer la descripción general que precede a dicho grupo:

- Descripción general de las instrucciones de bit
- Descripción general de las instrucciones de temporizador
- Descripción general de las instrucciones de contador

3.1.2 Descripción general de las instrucciones de bit

Las instrucciones operan en un solo bit de datos. Durante la operación, el procesador se puede establecer o restablecer el bit, según la continuidad lógica de los renglones de escalera, también se puede direccionar un bit tantas veces como requiera el programa.

Nota:

No se recomienda usar la misma dirección con instrucciones de salida múltiples.

Las instrucciones de bit se usan con los archivos de datos siguientes:

Archivos de datos de salida y entrada (archivos O: 0 e I: 1)

Representan salidas y entradas externas. Los bits en archivo 1 se usan para representar las entradas externas. En la mayoría de los casos, una sola palabra de 16 bits en estos archivos corresponden a una ubicación de ranura en su controlador con los números de bit correspondientes a números de terminal de entrada o salida. Los bits de la palabra no usados no están disponibles para su uso.

La siguiente tabla explica el formato de direccionamiento para salidas y entradas. Notar que el formato especifica *e* como el número de ranura y *s* como el número de palabra. Cuando se trabaja con instrucciones de archivo, hacemos referencia al elemento como *e.s* (ranura y palabra) tomados juntos.

Formato	Explicación		
O:e.s/b	O	Salida	
	I	Entrada	
	:	Delimitador de elemento	
	e	Elemento de la ranura (decimal)	Ranura 0, adyacente a la fuente de alimentación eléctrica en el primer chasis, se aplica al módulo de procesador. Las ranuras posteriores son ranuras de E/S, numeradas desde 1 hasta un máximo de 30.
I:e.s/b	.	Delimitador de palabra. Requerido sólo si es necesario un número de palabra según lo indicado a continuación.	
	s	Número de palabra	Requerido si el número de entradas o salidas exceden 16 para la ranura. Rango: 0-255 (el rango acepta "tarjetas especiales" de palabras múltiples)
	/	Delimitador de bit	
	b	Numero de terminal	Entradas: 0-15 Salidas: 0-15

Valores predeterminados: El dispositivo de programación muestra una dirección de una manera más formal. Por ejemplo, cuando asigna la dirección **O:5/0**, el dispositivo de programación la muestra como **O:5.0/0** (archivo de salida, ranura 5, palabra 0, terminal 0).

Archivo de estado (archivo S2:)

No se podrá añadir ni eliminar elementos del archivo de estado, se podrá direccionar varios bits y palabras según lo siguiente:

Formato	Explicación		
S:e/b	S	Archivo de estado	
	:	Delimitador de elemento	
	e	Elemento de la ranura (decimal)	Rangos de 0-15 en un controlador fijo o SLC 5/01, 0-32 en un procesador SLC 5/02, 0-83 en un SLC 5/03 OS300, 0-96 en un SLC 5/03 OS301 y posterior y 5/04 OS400 y 0-164 en un SLC 5/04. Estos son elementos de 1 palabra, 16 bits por cada elemento.
	.	Delimitador del bit.	
	/	Delimitador de bit	
	b	Numero de terminal	Entradas: 0-15 Salidas: 0-15

Archivo de datos de bit (B3:)

El archivo 3 constituye el archivo de bit, usado principalmente para instrucciones de bit (lógica de relé), registros de desplazamiento y secuenciadores. El tamaño máximo del archivo es 256 elementos de 1 palabra, un total de 4096 bits. Puede direccionar los bits especificando el número de elemento (0 a 255) y el número de bit (0 a 15) dentro del elemento. También se podrá direccionar los bits numerándolos secuencialmente, 0 a 4095.

Además, se puede direccionar los elementos de este archivo.

Formato	Explicación		
Bf:e/b	B	Archivo de tipo de bit	
	f	Número de archivo. Número 3 es el archivo predeterminado. Un número de archivo entre 10-255 se puede usar si se requiere almacenamiento adicional.	
	:	Delimitador de elemento	
	e	Número de elementos	Rangos de 0-255. Estos son elementos de 1 palabra. 16 bits por cada elemento.
	/	Delimitador de elemento	
b	Numero de bit	Ubicación del bit dentro del elemento. Rangos de 0-15	

Formato	Explicación	
Bf/b	B	Archivo de tipo de bit
	f	Número de archivo. Número 3 es el archivo predeterminado. Un número de archivo entre 10-255 se puede usar si se requiere almacenamiento adicional.
	:	Delimitador de elemento
	b	Numero de bit

Archivo de datos de control (R6:)

Estas instrucciones usan varios bits de control. Estos son elementos de 3 palabras usados con desplazamiento de bit, FIFO, LIFO, instrucciones de secuenciador e instrucciones ASCII ABL, ACB, AHL, ARD, ARL, AWA y AWT. La palabra 0 es la palabra de estado, la

palabra 1 indica la longitud de datos almacenados y la palabra 2 indica la posición. Esto se muestra en la figura 3.1.

En el elemento de control hay ocho bits de estado y un byte de código de error. Un controlador fijo y un elemento de control SLC 5/01 tienen seis bits. Los bits EU y EM no son usados por el procesador.

Elemento de control.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Pal
EN	EU	DN	EM	ER	UL	IN	FD	Código de error								0
Longitud de arreglo de Bit o archivo (LEN)																1
Indicador de Bit o posición (POS)																2

Figura. 3.1 Elementos de control de Datos de Control.

Bits direccionables

- EN = Habilitación
 - EU = Habilidad de descarga
 - DN = Efectuado
 - EM = Pila vacía
 - ER = Error
 - UL = Descarga (desplazamiento de bit solamente)
 - IN = Inhibición (Este es el bit de marcha (RN bit 9) para instrucciones ASCII)
 - FD = Encontrado (SQC solamente)
- El código de error se muestra en HEX y no es direccionable.

Palabras direccionables

- LEN = Longitud
- POS = Posición

Asignar direcciones de control según lo siguiente:

Formato	Explicación	
Rf:e	R	Archivo de control
	f	Número de archivo. Número 6 es el archivo predeterminado. Se puede usar un número de archivo entre 10-255 se puede usar si se requiere almacenamiento adicional.
	:	Delimitador de elemento
	e	Número de Elemento. Rangos de 0-255. Estos son elementos de 3 palabras. Ver la figura 3.1.

Direccionar los bits y palabras usando el formato **Rf:e.s/b** donde

- Rf:e** ya se explico anteriormente
- :** es el delimitador de palabra
- s** indica el subelemento
- /** es el delimitador de bit
- b** indica el bit

Archivo de datos enteros (N7:)

Usar estas direcciones (al nivel de bit) según se requiera en el programa. Estos son elementos de 1 palabra direccionables al nivel de elemento y bit.

Asignar las direcciones de enteros según lo siguiente:

Formato	Explicación	
Nf:e/b	N	Archivo de enteros
	f	Número de archivo. Número 7 es el archivo predeterminado. Un número de archivo entre 10-255 se puede usar si se requiere almacenamiento adicional.
	:	Delimitador de elemento
	e	Número de Elemento. Rangos de 0-255. Estos son elementos de 1 palabra. 16 bits por cada elemento.

/	Delimitador de bit	
b	Número de bit.	Ubicación del bit dentro del elemento. Rangos de 0-15.

3.1.3 Instrucciones inmediatas I/O

Examine si cerrado (XIC)

---] [---
Instrucción de entrada

Fijo	5/01	5/02	5/03	5/04	Micro
✓	✓	✓	✓	✓	✓

Usar la instrucción XIC en el programa de escalera para determinar si un bit está activado. Cuando la instrucción se ejecuta, si la dirección de bit está activada (1), entonces la instrucción es evaluada como verdadera. Cuando la instrucción se ejecuta, si el bit direccionado está desactivado (0), entonces la instrucción es evaluada como falsa.

Estado De dirección de bit	Instrucción XIC
0	Falsa
1	Verdadera

Ejemplos de dispositivos que se activan o desactivan incluyen:

- un botón pulsador cableado a una entrada (direccionado como I:1/9)
- un temporizador que controla una luz (direccionado como T4:3/DN)

Examine si abierto (XIO)

---] / [---
Instrucción de entrada

Fijo	5/01	5/02	5/03	5/04	Micro
✓	✓	✓	✓	✓	✓

Usar una instrucción XIO en el programa de escalera para determinar si un bit está desactivado. Cuando la instrucción se ejecuta, si el bit direccionado está desactivado (0), entonces la instrucción es evaluada como verdadera. Cuando la instrucción se ejecuta, si el bit direccionado está activado (1), entonces la instrucción es evaluada como falsa.

Estado de dirección de bit	Instrucción XIO
0	Verdadera
1	Falsa

Ejemplos de dispositivos que se activan o desactivan incluyen:

- botón de paro normalmente cerrada (N.C.) cableada a una entrada (I:1/10)
- un temporizador que controla una alarma audible (direccionado como T4:3/TT)

Active la salida (OTE)

--- () ---
Instrucción de salida

Fijo	5/01	5/02	5/03	5/04	Micro
✓	✓	✓	✓	✓	✓

Usar una instrucción OTE en su programa de escalera para activar/desactivar un bit cuando las condiciones de renglón son evaluada como verdaderas/falsas respectivamente.

Las instrucciones OTE se restablecen cuando:

- Entra o regresa al modo de marcha REM, prueba REM o cuando se restaura la alimentación eléctrica.
- El OTE se programa dentro de una zona de restablecimiento de control maestro (MCR) inactiva o falsa.

Nota

Un bit que está establecido dentro de una subrutina usando una instrucción OTE permanece establecido hasta que la subrutina se escanee nuevamente.

3.1.4 Enclavamiento de salida (OTL) y des enclavamiento de salida (OTU)

Enclavamiento de salida (OTL) y Des enclavamiento de salida (OTU)

--- (L) ---

--- (U) ---
Instrucciones de Salida

Flujo	S101	S102	S103	S104	Micro
✓	✓	✓	✓	✓	✓

OTL y OTU son instrucciones de salida de retención. OTL sólo activa un bit, en cambio, OTU sólo desactiva un bit. Estas instrucciones se usan generalmente en parejas, con ambas instrucciones direccionando el mismo bit.

El programa puede examinar un bit controlador por instrucciones OTL y OTU tantas veces como sea necesario.

Bajo las condiciones de error irrecuperable, las salidas físicas se desactivan. Una vez corregidas las condiciones de error, el controlador reanuda la operación usando el valor de la tabla de datos de la operación.

Cómo usar OTL

Cuando se asigna una dirección a la instrucción OTL que corresponde a la dirección de una salida física, el dispositivo de salida cableado a este terminal de tornillo está activado cuando el bit está establecido (activado o habilitado).

Cuando las condiciones de renglón se convierten en falsas (después de ser verdaderas), el bit permanece establecido y el dispositivo de salida correspondiente permanece activado.

Una vez habilitada, la instrucción de enclavamiento indica al controlador que active el bit direccionado. Desde ese momento en adelante, el bit permanece activado, pese a la condición del renglón, hasta que el bit esté desactivado (típicamente por una instrucción OTU en otro renglón).

Cómo usar OTU

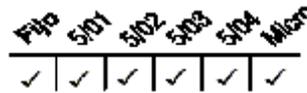
Cuando asigna una dirección a la instrucción OTU que corresponde a la dirección de una salida física, el dispositivo de salida cableado a este terminal de tornillo está desactivado cuando el bit está restablecido (desactivado o inhabilitado).

La instrucción de desenclavamiento indica al controlador que desactive el bit direccionado. Desde ese momento en adelante, el bit permanece desactivado, pese a la condición del renglón, hasta que esté activado (típicamente por una instrucción OTL en otro renglón).

3.1.5 One-Shot Rising (OSR)

One-Shot Rising (OSR)

---[OSR]---
Instrucción de entrada



La instrucción OSR es una instrucción de entrada de retención que ocasiona un evento durante una sola vez. Usar la instrucción OSR cuando un evento debe comenzar basado en el cambio de estado del renglón de falso a verdadero.

Cuando las condiciones de renglón precedentes de la instrucción OSR van de falsas a verdaderas, la instrucción OSR será verdadera durante un escán. Después de completarse un escán, la instrucción OSR se hace falsa, aun cuando las condiciones de renglón precedentes permanecen verdaderas. La instrucción OSR sólo volverá a hacerse verdadera si las condiciones de renglón precedentes efectúan una transición de falso a verdadero.

El controlador le permite usar una instrucción OSR por cada salida en un renglón.

Cómo introducir parámetros

La dirección asignada a la instrucción OSR *no* es la dirección de ONE-SHOT mencionada en el programa, ni indica el estado de la instrucción OSR. Esta dirección permite que la instrucción OSR *recuerde* su estado de renglón anterior.

Usar una dirección de bit desde el archivo de datos del bit o enteros. El bit direccionado está establecido (1) durante un escán cuando las condiciones de renglón precedentes de la instrucción OSR son verdaderas (aun cuando la instrucción OSR se hace falsa); el bit está restablecido (0) cuando las condiciones del renglón precedentes de la instrucción OSR se hacen falsas.

Nota

La dirección de bit que se usa para esta instrucción debe ser única. No usar en otros lugares del programa.

No usar una dirección de entrada o salida para programar el parámetro de dirección de la instrucción OSR.

3.2. Instrucciones de temporizadores y contadores

Como se menciona anteriormente las instrucciones de Bit se dividen en tres grupos: bits, temporizadores y contadores. En esta sección se dará una descripción general acerca de estos dos últimos.

Instrucción		Propósito	Página
Mnemónico	Nombre		
TON	Temporizador a la conexión	Cuenta los intervalos de la base de tiempo cuando la instrucción es verdadera.	36
TOF	Temporizador a la desconexión	Cuenta los intervalos de la base de tiempo cuando la instrucción es falsa.	37
RTO	Temporizador retentivo	Cuenta los intervalos de la base de tiempo cuando la instrucción es verdadera y retiene el valor acumulador cuando la instrucción se hace falsa o cuando ocurra un ciclo de alimentación eléctrica.	38
CTU	Conteo progresivo	Incrementa el valor acumulador a cada transición de falso a verdadero y retiene el valor acumulador cuando la instrucción se hace falsa o cuando ocurra un ciclo de alimentación eléctrica.	40
CTD	Conteo regresivo	Conteo regresivo Disminuye el valor acumulado a cada transición de falso a verdadero y retiene el valor acumulador cuando la instrucción se hace falsa o cuando ocurra un ciclo de alimentación eléctrica.	41
HSC	Contador de alta Velocidad	Cuenta los impulsos de alta velocidad de una entrada de alta velocidad de controlador fijo.	42
RES	Restablecimiento	Pone a cero el valor acumulador y los bits de estado de un temporizador o contador. No usar con temporizadores TOF.	45

3.2.1 Descripción general de las instrucciones de temporizador

Cada dirección de temporizador se compone de un elemento de 3 palabras. Palabra 0 es la palabra de control, palabra 1 almacena el valor preseleccionado y palabra 2 almacena el valor acumulado.

Elemento de control.

	15	14	13	
Pal. 0	EN	TT	DN	Uso interno
Pal. 1	Valor preseleccionado			
Pal. 2	Valor de acumulador			

Bits direccionables

EN = Bit 15 Habilidad
 TT = Bit 14 Temporización del tempor.
 DN = Bit 13 Efectuado

Palabras direccionables

PRE = Valor preseleccionado
 ACC = Valor acumulado

Los bits etiquetados como "uso interno" no son direccionables.

Cómo introducir parámetros

Valor del acumulador (.ACC)

Este es el tiempo transcurrido desde el último restablecimiento del temporizador. Cuando está habilitado, el temporizador lo actualiza constantemente.

Valor preseleccionado (.PRE)

Especifica el valor que el temporizador debe alcanzar antes de que el controlador establezca el bit de efectuado. Cuando el valor acumulado sea igual o mayor que el valor preseleccionado, el bit de efectuado estará establecido. Puede usar este bit para controlar un dispositivo de salida.

Los valores preseleccionados y acumulados para temporizadores tienen un rango desde 0 hasta +32,767. Si el valor preseleccionado o acumulador de temporizador es un número negativo, ocurre un error de tiempo de ejecución.

Base de tiempo

La base de tiempo determina la duración de cada intervalo de base de tiempo. Para los procesadores fijos y SLC 5/02, la base de tiempo ha sido establecida a 0.01 segundo. Para los procesadores SLC 5/02, SLC 5/03, SLC 5/04 y los controladores MicroLogix 1000, la base de tiempo es seleccionable como 0.01 (10 ms) segundo ó 1.0 segundo.

Precisión del temporizador

La precisión del temporizador se refiere al tiempo transcurrido entre el momento en que una instrucción de temporizador está habilitada y el momento en que el intervalo temporizado se ha completado. La inexactitud causada por el escán de programa puede ser mayor que la base de tiempo del temporizador. También debe considerar el tiempo necesario para activar el dispositivo de salida.

La precisión de temporización es ± 0.01 a $+0$ segundos, con un escán de programa de hasta 2.5 segundos. El temporizador de 1 segundo mantiene la precisión con un escán de programa de hasta 1.5 segundos. Si un programa puede exceder 1.5 ó 2.5 segundos, repita el renglón de instrucción del temporizador para que el renglón sea escaneado dentro de estos límites.

Nota:

La temporización podría resultar inexacta si las instrucciones de salto (JMP), etiqueta (LBL), salto a subrutina (JSR) o subrutina (SBR) saltan el renglón que contiene una instrucción de temporizador mientras que el temporizador esté temporizando. Si la duración de salto es menor de 2.5 segundos, no se pierde ningún tiempo; si la duración de salto excede 2.5 segundos, ocurre un error de temporización no detectable. Cuando se usen subrutinas, es necesario que un temporizador esté ejecutado a un mínimo de cada 2.5 segundos para evitar un error de temporización.

Estructura de direccionamiento

Direcciones bits y palabras usando el formato **Tf:e/s/b**

Explicación.		
T	Archivo de temporizador	
f	Número de archivo. Para los procesadores SLC 500, el número predeterminado es 4. Se puede usar un número entre 10-255 para almacenamiento adicional.	
:	Delimitador de elemento	
e	Número de elemento	Estos son elementos de 3 palabras. Para los procesadores SLC 500 el rango es 0-255.
.	Elemento de palabras	
s	Subelemento	
/	Delimitador del bit	
b	Bit	

Temporizador a la conexión (TON)

ML1000	F10	S101	S102	S103	S104
✓	✓	✓	✓	✓	✓



Usar la instrucción TON para activar o desactivar una salida después de que el temporizador haya estado activado durante un intervalo de tiempo preseleccionado. La instrucción TON comienza a contar los intervalos de la base de tiempo cuando las condiciones de renglón se hacen verdaderas. Con tal que las condiciones de renglón permanezcan verdaderas, el temporizador ajusta su valor acumulado (ACC) durante cada evaluación hasta alcanzar el valor predeterminado (PRE). Cuando las condiciones de renglón se hacen falsas, el valor acumulado se reinicializa sin importar si el temporizador ha sobrepasado el límite de tiempo.

Uso de los bits de estado

Este bit	Se establece cuando	Y permanece establecido hasta ocurrir uno de los siguientes eventos
Bit de efectuado del temporizador DN (bit 13)	El valor acumulado es igual o mayor que el valor preseleccionado	las condiciones de renglón se hacen falsas
Bit de temporización del temporizador TT (bit 14)	las condiciones de renglón son verdaderas y el valor acumulado es menor que el valor preseleccionado	las condiciones de renglón se hacen falsas o cuando el bit de efectuado esté establecido
Bit de habilitación del temporizador EN (bit 15)	las condiciones de renglón son verdaderas	las condiciones de renglón se hacen falsas

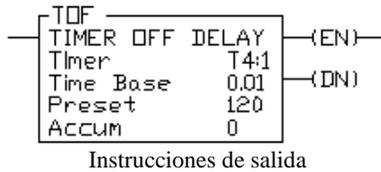
Cuando el procesador cambia del modo de marcha REM, prueba REM al modo de programa REM o la alimentación eléctrica se pierde durante la temporización de la instrucción, pero no ha alcanzado su valor preseleccionado, ocurre lo siguiente:

- El bit de habilitación del temporizador (EN) permanece establecido.
- El bit de temporización del temporizador (TT) permanece establecido.
- El valor acumulado (ACC) permanece sin cambio.

Puede ocurrir lo siguiente al regresar al modo de marcha REM o prueba REM:

Condición	Resultado
Si el renglón es verdadero:	El bit EN permanece establecido. El bit TT permanece establecido El valor ACC está puesto a cero.
Si el renglón es falso:	El bit EN está restablecido. El bit TT está restablecido. El valor ACC está puesto a cero.

Temporizador a la desconexión (TOF)



Usar la instrucción TOF para activar o desactivar una salida después de que su renglón ha estado desactivado durante un intervalo de tiempo preseleccionado. La instrucción TOF comienza a contar los intervalos de la base de tiempo cuando el renglón efectúa una transición de verdadero a falso. Con tal que las condiciones permanezcan falsas, el temporizador incrementa su valor acumulado (ACC) durante cada escán hasta alcanzar el valor preseleccionado (PRE). El valor acumulado se restablecerá cuando las condiciones de renglón se hagan verdaderas, sin importar si el tiempo en el temporizador se ha agotado

Uso de los bits de estado

Este bit	Se establece cuando	Y permanece establecido hasta ocurrir uno de los siguientes eventos
Bit de efectuado del temporizador DN (bit 13)	las condiciones de renglón son verdaderas	las condiciones de renglón se hacen falsas y el valor acumulado es mayor o igual que el valor preestablecido
Bit de temporización de temporizador TT (bit 14)	las condiciones de renglón se hacen falsas y el valor acumulado es mayor o igual que el valor preseleccionado	las condiciones de renglón se hacen verdaderas o cuando el bit de efectuado se restablece
Bit de habilitación del temporizador EN (bit 15)	las condiciones de renglón son verdaderas	las condiciones de renglón se hacen falsas

Cuando la operación del procesador cambia del modo de marcha REM, prueba REM al modo de programa REM o cuando se pierde la alimentación eléctrica durante la temporización de una instrucción de retardo con temporizador desactivado, pero no ha alcanzado su valor preseleccionado, ocurre lo siguiente:

- El bit de habilitación del temporizador (EN) permanece establecido.
- El bit de temporización del temporizador (TT) permanece establecido.
- El bit de efectuado del temporizador (DN) permanece establecido.
- El valor acumulado (ACC) permanece sin cambio.

Se puede ocurrir lo siguiente al regresar al modo de marcha REM o prueba REM:

Condición	Resultado
Si el renglón es verdadero:	El bit TT se restablece. El bit DN permanece establecido. El bit EN se establece. El valor ACC se restablece
Si el renglón es falso:	El bit TT se restablece. El bit DN se restablece. El bit EN se restablece. El valor ACC se establece igual que el valor preseleccionado.

La instrucción de restablecimiento (RES) no se puede usar con las instrucciones TOF porque RES siempre pone a cero los bits de estado así como el valor acumulado.

Nota:

El TOF temporiza dentro de una pareja MCR inactiva.

Temporizador retentivo (RTO)

ML1000	Fijo	S101	S102	S103	S104
✓	✓	✓	✓	✓	✓



Usar la instrucción RTO para activar o desactivar una salida después que el temporizador haya estado desactivado durante un intervalo de tiempo preseleccionado. La instrucción RTO es una instrucción de retención que comienza a contar los intervalos de base de tiempo cuando las condiciones de renglón se hacen verdaderas.

La instrucción RTO retiene su valor acumulado cuando ocurre cualquiera de los eventos siguientes:

- Las condiciones de renglón se hacen falsas.
- Cambiar la operación del procesador del modo de marcha REM o prueba REM al modo de programa REM.
- Se corta la alimentación eléctrica del procesador (siempre que se mantenga una batería auxiliar).
- Ocurre un fallo.

Cuando regresa el procesador al modo de marcha REM, prueba REM y/o las condiciones de renglón se hacen verdaderas, la temporización continúa desde el valor acumulado retenido. Los temporizadores retentivos miden el período acumulativo durante el cual las condiciones de renglón son verdaderas mediante la retención de su valor acumulado.

Uso de los bits de estado

Este bit	Se establece cuando	Y permanece establecido hasta ocurrir uno de los siguientes eventos
Bit de efectuado del temporizador DN (bit 13)	el valor acumulado es igual o mayor que el valor preseleccionado	la instrucción RES apropiada se habilita
Bit de temporización del temporizador TT (bit 14)	las condiciones de renglón son verdaderas y el valor acumulado es menor que el valor preseleccionado	las condiciones de renglón se hacen falsas o cuando se establece el bit de efectuado
Bit de habilitación del temporizador EN (bit 15)	las condiciones de renglón son verdaderas	las condiciones de renglón se hacen falsas

Nota:

Para restablecer el valor acumulado del temporizador retentivo y los bits de estado después de que el renglón RTO se hace falso, debe programar una instrucción de restablecimiento (RES) con la misma dirección en otro renglón.

Cuando el procesador cambia del modo de marcha REM, prueba REM al modo de programa REM o fallo REM, o cuando se pierde la alimentación eléctrica durante la temporización del temporizador, pero todavía sin alcanzar el valor preseleccionado, ocurre lo siguiente:

- El bit de habilitación (EN) del temporizador permanece establecido.
- El bit de temporización (TT) del temporizador permanece establecido.
- El valor acumulado (ACC) permanece sin cambio.

Puede ocurrir lo siguiente al regresar al modo de marcha REM o prueba REM o cuando se restaura la alimentación eléctrica:

Condición	Resultado
Si el renglón es verdadero:	El bit TT permanece establecido. El bit EN permanece establecido. El valor ACC permanece sin cambio y vuelve a incrementar.
Si el renglón es falso:	El bit TT se restablece. El bit DN permanece en su último estado. El bit EN se restablece. El valor ACC permanece en su último estado.

3.2.2 Descripción general de los contadores hacia arriba y hacia abajo

Elementos del archivo de datos del contador

Cada dirección de contador se compone de un elemento de archivo de datos de 3 palabras. Palabra 0 es la palabra de control y contiene los bits de estado de la instrucción. Palabra 1 es el valor preseleccionado. Palabra 2 es el valor acumulado.

La palabra de control para las instrucciones de contador incluye seis bits de estado, según lo indicado a continuación:

	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Pal. 0	CU	CD	DN	OV	UN	UA										Uso interno
Pal. 1	Valor preseleccionado															
Pal. 2	Valor acumulado															

Bits direccionables

CU = Habilitación de conteo prog.
CD = Habilitación de conteo reg.
DN = Bit de efectuado
OV = Bit de overflow
UN = Bit de underflow
UA = Actualización del valor acumulado
(HSC en el controlador fijo solamente)

Palabras direccionables

PRE = Valor preseleccionado
ACC = Valor acumulado

Los bits etiquetados como "uso interno" no son direccionables.

Cómo introducir parámetros

Valor acumulado (.ACC)

Es el número de transiciones de falso a verdadero que han ocurrido desde el último restablecimiento del contador.

Valor preseleccionado (PRE)

Especifica el valor que el contador debe alcanzar antes que el controlador establezca el bit de efectuado. Cuando el valor del acumulador se hace igual o mayor que el valor preseleccionado, se establece el bit de estado efectuado. Se puede usar este bit para controlar un dispositivo de salida.

Los valores preseleccionados y acumulados para los contadores oscilan entre -32,768 hasta +32,767 y se almacenan como enteros con signos. Los valores negativos se almacenan en forma de complemento de dos.

Estructura de direccionamiento

Direcciones de contador usando el formato Cf:e.s/b

	Explicación	
C	Contador	
f	Número de archivo. Para los procesadores SLC 500, el valor predeterminado es 5. Un número de archivo entre 10-255 se puede usar para obtener almacenamiento adicional.	
:	Delimitador de elemento	
e	Número de elemento	Estos son elementos de 3 palabras. Para los procesadores SLC 500 el rango es 0-255.
.	Elemento de palabra	
s	Subelemento	
/	Delimitador de bit	
b	Bit	

Asigne direcciones de contador usando el formato Cf:e.s/b

Cómo funcionan los contadores

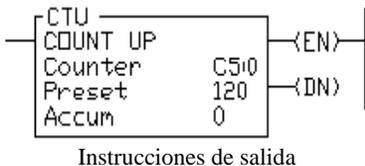
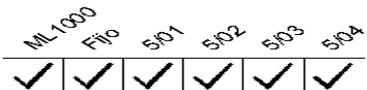
La figura 3.2 muestra cómo funciona un contador. El valor del contador debe permanecer dentro del rango de ±32768 a +32767. Si el valor de conteo excede +32767 ó desciende a menos de ±32768, se establece un bit de overflow (OV) o underflow (UN) de estado del contador.

Un contador se puede poner a cero usando la instrucción de restablecimiento (RES).



Figura. 3.2 Rango de los Contadores para que no se establezca un bit de error.

Conteo progresivo (CTU)



El CTU es una instrucción que cuenta las transiciones de renglón de falso a verdadero. Las transiciones de renglón pueden ser provocadas por eventos ocurriendo en el programa (de la lógica interna o dispositivos de campo externos) tales como piezas que pasan por un detector o que activan un interruptor de límite.

Cuando las condiciones de renglón para una instrucción CTU efectúan una transición de falso a verdadero, el valor acumulado se incrementa en uno, siempre que el renglón que contiene la instrucción CTU se evalúa entre estas transiciones. La capacidad del contador para detectar transiciones de falso a verdadero depende de la velocidad (frecuencia) de la señal de entrada.

Nota

La duración activada y desactivada de una señal de entrada no debe ser más rápida que el tiempo de escán 2x (se entiende un ciclo de trabajo de 50%).

El valor acumulado se retiene cuando las condiciones de renglón vuelven a hacerse falsas. El conteo acumulado se retiene hasta que sea puesto a cero por una instrucción de restablecimiento (RES) que tenga la misma dirección que el contador.

Uso de los bits de estado

Este bit	Se establece cuando	Y permanece establecido hasta ocurrir uno de los siguientes eventos
Bit de overflow de conteo progresivo OV (bit 12)	el valor acumulado cambia a -32,768 (desde +32,767) y continúa contando desde ese punto	se ejecuta una instrucción RES con la misma dirección que la instrucción CTU o bien el conteo se reduce a un valor menor o igual que +32,767 con una instrucción CTD
Bit de efectuado DN (bit 13)	el valor acumulado es igual o mayor que el valor preseleccionado	el valor acumulado se hace menor que el valor preseleccionado
Bit de habilitación de conteo progresivo CU (bit 15)	las condiciones de renglón son verdaderas	las condiciones de renglón se hacen falsas O BIEN se habilita una instrucción RES con la misma dirección que la instrucción CTU

El valor acumulado se retiene después que la instrucción CTU se hace falsa, o cuando la alimentación eléctrica se corta y luego se restaura al controlador. Además, el estado activado o desactivado de los bits de contador efectuado, overflow y underflow es retentivo. El valor acumulado y los bits de control se restablecen cuando se habilita la instrucción RES correcta. Los bits CU siempre se establecen antes de introducir los modos de marcha REM o prueba REM.

Conteo regresivo (CTD)

CTD	
COUNT DOWN	
Counter	C51
Preset	120
ACCUM	0

Instrucciones de salida

ML-1000	F10	5101	5102	5103	5104
✓	✓	✓	✓	✓	✓

El CTD es una instrucción que cuenta las transiciones de renglón de falso a verdadero. Las transiciones de renglón pueden ser causadas por eventos que ocurren en el programa, tales como piezas pasando por un detector o accionando un final de carrera.

Cuando las condiciones de renglón para una instrucción CTD han efectuado una transición de falso a verdadero, el valor acumulado disminuye en un conteo, siempre que el renglón que

contiene la instrucción CTD se evalúa entre estas transiciones. Los conteos acumulados se retienen cuando las condiciones de renglón se hacen falsas nuevamente. El conteo acumulado se retiene hasta que sea puesto a cero por una instrucción de restablecimiento (RES) que tiene la misma dirección que el contador restablecido.

Uso de los bits de estado

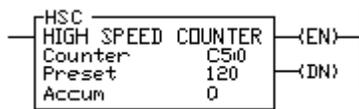
Este bit	Se establece cuando	Y permanece establecido hasta ocurrir uno de los siguientes eventos
Bit de underflow de conteo regresivo UN (bit 11)	el valor acumulado cambia a -32,768 (desde +32,767) y continúa contando regresivamente desde ese punto	una instrucción RES con la misma dirección que la instrucción CTD se ejecuta o bien el conteo es incrementado menor o igual que +32,767 con una instrucción CTU
Bit de efectuado DN (bit 13)	el valor acumulado es igual o mayor que el valor preseleccionado	el valor acumulado se hace menor que el valor preseleccionado
Bit de habilitación de conteo regresivo CD (bit 14)	las condiciones de renglón son verdaderas	las condiciones de renglón se hacen falsas o bien se habilita una instrucción RES con la misma dirección que la instrucción CTD

El valor acumulado se retiene después de que la instrucción CTD se hace falsa, o cuando la alimentación eléctrica al controlador se corta y luego se restaura. Además, el estado activado o desactivado de los bits de contador efectuado, overflow y underflow es retentivo. El valor acumulado y los bits de control se restablecen cuando se habilita la instrucción RES correcta. Los bits CTD siempre se establecen antes de introducir los modos de marcha REM o prueba REM.

3.2.3 Contador de alta velocidad (HSC)

Contador de alta velocidad (HSC)

ML1000	F110	S101	S102	S103	S104
✓	✓	✓	✓	✓	✓



Instrucciones de salida

El contador de alta velocidad constituye una variación del contador CTU. La instrucción HSC se habilita cuando la lógica de renglón es verdadera y se inhabilita cuando la lógica de renglón es falsa.

Nota

La instrucción HSC cuenta transiciones que ocurren en el terminal de entrada I:0/0. La instrucción HSC no cuenta las transiciones de renglón. Habilita o inhabilita el renglón HSC para habilitar o inhabilitar el conteo de transiciones que ocurren en la terminal de entrada I:0/0. Es recomendable colocar la instrucción HSC en un renglón incondicional. No colocar la instrucción XIC con la dirección I:0/0 en serie con la instrucción HSC ya que los conteos se perderán.

El HSC es un contador CTU especial para uso con los procesadores SLC fijos y SLC 5/01 de 24 VCD. Los bits de estado y valores acumulados del HSC no son retentivos.

Nota

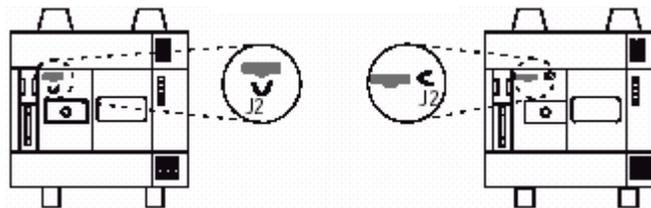
Esta instrucción proporciona el conteo de alta velocidad para los controladores de E/S fijos con entradas de 24 VCC. Se permite una sola instrucción HSC por cada controlador. Para usar la instrucción, debe cortar el puente como se indica a continuación. Se recomienda un cable blindado para reducir el ruido a la entrada.

Operación del contador de alta velocidad

Para la operación del contador de alta velocidad, hay que realizar los pasos siguientes:

1. Desconectar la alimentación eléctrica del controlador fijo.
2. Quitar el envoltente del SLC 500.
3. Localizar y cortar el cable del puente J2. No quitar completamente, pero asegúrese que los extremos del cable del puente cortado no hagan contacto entre sí.

El puente del contador de alta velocidad se ubica debajo del conector de la batería o bien a la derecha del conector de la batería.



4. Volver a colocar la cubierta.

Nota

Ahora la entrada I:0/0 funciona en modo de alta velocidad. La dirección del bit de habilitación del contador de alta velocidad es C5:0/CU. Cuando las condiciones de renglón son verdaderas, se establece C5:0/CU y se cuentan las transiciones que ocurren en la entrada I:0/0.

Para comenzar el conteo de alta velocidad, cargar un valor preseleccionado en C5:0.PRE y habilitar el renglón de contador. Para contar un valor preseleccionado, realizar uno de los pasos siguientes:

- Cambiar al modo de marcha REM o prueba REM de otro modo.
- Encender el procesador en modo de marcha REM.
- Restablecer el HSC usando la instrucción RES.

La recarga automática ocurre cuando el HSC por sí mismo establece el bit DN a la interrupción.

Cada transición de entrada que ocurre en entrada I:0/0 causa que el HSC acumulado se incremente. Cuando el valor acumulado es igual que el valor preseleccionado, se establece el bit de efectuado (C5:0/DN), el valor acumulado se pone a cero y el valor preseleccionado (C5:0.PRE) se carga en el HSC como preparación para la próxima transición de alta velocidad en la entrada I:0/0.

El programa de escalera debe consultar el bit de efectuado (C5:0/DN) para determinar el estado del HSC. Una vez que el bit de efectuado haya sido detectado como establecido, el programa de escalera debe poner a cero el bit C5:0/DN (usando la instrucción OTU de des enclavamiento) antes de que el HSC acumulado vuelva a alcanzar el valor preseleccionado; en caso contrario, el bit de overflow (C5:0/OV) se establecerá.

El HSC es diferente que los contadores CTU y CTD. El CTU y CTD son contadores de software. El HSC es un contador de hardware y funciona asincrónicamente al escán del programa de escalera. El valor acumulado HSC (C5:0.ACC) normalmente se actualiza cada vez que el renglón HSC es evaluado en el programa de escalera. Esto significa que el valor del acumulador de hardware HSC se transfiere al acumulador de software HSC. Usar solamente la instrucción OTE para transferir este valor. La instrucción HSC pone a cero inmediatamente el bit C5:0/UA a continuación de la actualización acumulada.

Muchos conteos HSC pueden ocurrir entre las evaluaciones HSC, los cuales provocarían la inexactitud del bit C5:0.ACC cuando éste sea usado en un programa de escalera. Para permitir un valor acumulado HSC exacto, el bit de acumulado de actualización (C5:0/UA) causa que C5:0.ACC sea actualizado inmediatamente al estado del acumulador de hardware cuando se establezca.

Usar la instrucción RES para restablecer el contador de alta velocidad en dirección C5:0. La instrucción HSC pone a cero el bit de estado, el acumulador y carga el valor preseleccionado durante:

- el encendido
- entrada en el modo de marcha REM
- un restablecimiento

Elementos de datos del contador de alta velocidad

La dirección C5:0 es el elemento de 3 palabras del contador HSC.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Pal. 0	CU	CD	DN	OV	UN	UA										No usado
Pal. 1	Valor preseleccionado															
Pal. 2	Valor de acumulador															

CU = Bit de habilitación de conteo progresivo
CD = Bit de habilitación de conteo regresivo
DN = Bit de efectuado
OV = Bit de overflow
UN = Bit de underflow
UA = Actualización de acumulador (HSC solamente)

- La palabra 0 contiene los bits de estado siguientes de la instrucción HSC:
 - El bit 10 (UA) actualiza la palabra de acumulador del HSC para reflejar el estado inmediato del HSC cuando es verdadero.
 - El bit 12 (OV) indica la ocurrencia de un overflow de HSC.
 - El bit 13 (DN) indica si el valor preseleccionado de HSC ha sido alcanzado.
 - El bit 15 (CU) muestra el estado de habilitación / inhabilitación de la instrucción HSC.

- La palabra 1 contiene el valor preseleccionado que se carga en el HSC cuando se ejecuta la instrucción RES, o cuando se establece el bit de efectuado o cuando se efectúa el encendido inicial.
- La palabra 2 contiene el valor del acumulador HSC. Esta palabra es actualizada cada vez que la instrucción HSC es evaluada y cuando el bit del acumulador de actualización es establecido usando una instrucción OTE. Este acumulador es de sólo lectura. Cualquier valor escrito en el acumulador resulta sobrescrito por el contador de alta velocidad durante la evaluación de instrucción, restablecimiento o introducción del modo de marcha REM.

3.2.4 Restablecimiento (RES)



--- (RES) ---
Instrucción de salida

Usar una instrucción RES para restablecer un temporizador o contador. Cuando se habilita la instrucción RES, restablece la instrucción de retardo del temporizador a la conexión (TON), temporizador retentivo (RTO), conteo progresivo (CTU) o conteo regresivo (CTD) con la misma dirección que la instrucción RES.

Usando una instrucción RES para un:	El procesador restablece el:
Temporizador (No use una instrucción RES con TOF.)	valor ACC a 0 bit DN bit TT bit EN
Contador	valor ACC a 0 bit OV bit UNt bit DN bit CU bit CD
Control	valor POS a 0 bit EN bit EUt bit DN bit EM bit ER bit UL IN y FD van al último estado

Nota:

Cuando restablece un contador, si la instrucción RES está habilitada y el renglón de contador está habilitado, se pone a cero el bit CU o CD.

Si el valor preseleccionado del contador es negativo, la instrucción RES establece el valor acumulado a cero. Esto, a su vez, causa que el bit de efectuado sea establecida por una instrucción de conteo regresivo o conteo progresivo.

ADVERTENCIA:

Ya que la instrucción RES restablece el valor acumulado y los bits de efectuado, temporización y habilitados, *no* use la instrucción RES para restablecer una dirección de temporizador usada en una instrucción TOF. En caso contrario, puede ocurrir la operación inesperada de la máquina o lesiones al personal.

1.3. Instrucciones matemáticas, comparación y de movimiento.

Esta sección contiene información general acerca de instrucciones matemáticas y explica cómo funcionan en su programa de lógica. Cada una de las instrucciones matemáticas incluye información acerca de:

- cómo aparece el símbolo de instrucción
- cómo usar la instrucción

Instrucciones matemáticas

Instrucción		Propósito	Página
Mnemónico	Nombre		
ADD	Añadir	Añade la fuente A a la fuente B y almacena el resultado en el destino.	48
SUB	Restar	Resta la fuente B de la fuente A y almacena el resultado en el destino.	49
MUL	Multiplicar	Multiplica la fuente A por la fuente B y almacena el resultado en el destino.	49
DIV	Dividir	Divide la fuente A por la fuente B y almacena el resultado en el destino y el registro matemático.	50
DDV	División doble	Divide el contenido del registro matemático por la fuente y almacena el resultado en el destino y el registro matemático.	50
CLR	Borrar	Pone todos los bits de una palabra a cero.	51
SQR	Raíz cuadrada	Calcula la raíz cuadrada de la fuente y coloca el resultado de entero en el destino.	51
SCP	Datos de escala	Multiplica la fuente por una tasa especificada, añade a un valor offset y almacena el resultado en el destino.	52
SCL	Datos de escala	Multiplica la fuente por una tasa especificada, añade a un valor offset y almacena el resultado en el destino.	53
ABS	Absoluto	Calcula el valor absoluto de la fuente y coloca el resultado en el destino.	53
CPT	Calcular	Evalúa una expresión y almacena el resultado en el destino.	54
SWP	Cambiar	Cambia los bytes bajos y altos de un número especificado de palabras en un archivo de bit, entero, ASCII o cadena.	55
ASN	Arco seno	Acepta el arco seno de un número y almacena el resultado (en radianes) en el destino.	55
ACS	Arco coseno	Acepta el arco coseno de un número y almacena el resultado (en radianes) en el destino.	55
ATN	Arco tangente	Acepta el arco tangente de un número y almacena el resultado (en radianes) en el destino.	56
COS	Coseno	Acepta el coseno de un número y almacena el resultado en el destino.	56

LN	Logaritmo natural	Acepta el logaritmo natural del valor en la fuente y lo almacena en el destino.	56
LOG	Logaritmo de base 10	Acepta el logaritmo de la base 10 del valor en la fuente y almacena el resultado en el destino.	57
SIN	Seno	Acepta el seno de un número y almacena el resultado en el destino.	57
TAN	Tangente	Acepta la tangente de un número y almacena el resultado en el destino.	57
XPY	X a la potencia de Y	Eleva un valor a la potencia y almacena el resultado en el destino.	58

Acerca de las instrucciones matemáticas

La mayor parte de las instrucciones toman dos valores de entrada, realizan la función matemática y colocan el resultado en un lugar de memoria asignado.

Por ejemplo, las instrucciones ADD y SUB toman un par de valores de entrada, los añaden o los restan y colocan el resultado en el destino especificado. Si el resultado de la operación excede el valor permitido, un bit de overflow o underflow se establece.

3.3.1. Descripción general de las instrucciones matemáticas

La información general siguiente se aplica a las instrucciones matemáticas.

Cómo introducir parámetros

- **La fuente** es la(s) dirección(es) del(los) valor(es) en que se realiza una operación matemática, lógica o de movimiento. Esto puede ser direcciones de palabra o constantes de programa. Una instrucción que tiene dos operandos de fuente no aceptan constantes de programa en ambos operandos.
- **El destino** es la dirección del resultado de la operación. Los enteros con signo se almacenan de forma complementaria de dos y se aplican a los parámetros de fuente y destino.
Al usar un procesador SLC 5/03 OS301, OS302 ó un procesador SLC 5/04 OS400, OS401, se pueden usar los valores del punto (coma) flotante y de cadena (especificados al nivel de palabra).

Uso de las direcciones de palabra indexadas

Tiene la opción de usar direcciones de palabra indexadas para parámetros de instrucción especificando direcciones de palabra (excepto los procesadores fijos y SLC 5/01).

Uso de las direcciones de palabra indirectas

Tiene la opción de usar direcciones indirectas a nivel de palabra y a nivel de bit para instrucciones especificando direcciones de palabra cuando usa procesadores SLC 5/03 OS302 y SLC 5/04 OS401.

Actualizaciones de los bits de estado aritmético

Los bits de estado aritmético se encuentran en la palabra 0, bits 0–3 en el archivo de estado del controlador. Después de la ejecución de una instrucción, los bits de estado aritmético en el archivo de estado son actualizados:

Con este bit:		El controlador:
S:0/0	Acarreo (C)	se establece si el acarreo es generado; en caso contrario, se pone a cero.
S:0/1	Overflow (V)	indica que el resultado real de una instrucción matemática no se puede colocar en el destino designado.
S:0/2	Cero (Z)	indica un valor 0 después de una instrucción matemática, de movimiento o lógica.
S:0/3	Signo (S)	indica un valor negativo (menor que 0) después de una instrucción matemática, de movimiento o lógica.

Bit de interrupción por overflow, S: 5/0

El bit de error menor (S:5/0) se establece a la detección de un overflow matemático o división entre 0. Si este bit se establece a la ejecución de una instrucción END o una instrucción de fin temporal (TND) o una regeneración de E/S (REF), se establece el código 0020 de error mayor recuperable.

En las aplicaciones donde ocurre un overflow matemático o una división entre 0, se puede evitar un fallo CPU usando la instrucción de des enclavamiento (OTU) con la dirección S:5/0 en su programa. El renglón se debe encontrar entre el punto de overflow y la instrucción END, TND o REF.

Cambios del registro matemático S:13 y S:14

La palabra de estado S:13 contiene la palabra de *mínimo* significado de los valores de 32 bits de las instrucciones MUL y DDV. Contiene el resto para las instrucciones DIV y DDV. También contiene los cuatro primeros dígitos BCD para las instrucciones de conversión desde BCD (FRD) y conversión a BCD (TOD).

La palabra S:14 contiene la palabra de *máximo* significado para los valores de 32 bits de las instrucciones MUL y DDV. Contiene el cociente no redondeado para las instrucciones DIV y DDV. También contiene el dígito más significativo (dígito 5) para las instrucciones TOD y FRD.

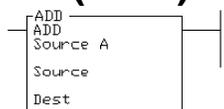
Uso del archivo de datos de punto (coma) flotante (F8:)

Este tipo de archivo es válido para los procesadores SLC 5/03 OS301, OS302 y SLC 5/04 OS400, OS401. Estos son elementos de 2 palabras y direccionables solamente al nivel de elemento.

Asigne las direcciones de punto (coma) flotante de esta manera:

Formato	Explicación
Ff:e	F Archivo de punto (coma) flotante
	F Número del archivo. El número 8 es el archivo predeterminado. Un número de archivo entre 9–255 se puede usar si se requiere almacenamiento adicional
	: Delimitador de elemento
	e Número de Elemento. Tiene un rango de 0–255. Estos son elementos de 2 palabras. Números de 32 bits no extendidos.

Añadir (ADD)



Instrucciones de salida

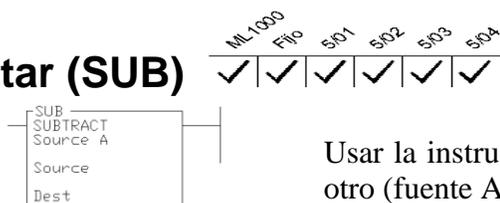


Usar la instrucción ADD para añadir un valor (fuente A) a otro valor (fuente B) y colocar el resultado en el destino.

Actualizaciones de bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	se establece si el acarreo es generado; si no, se restablece (entero). Se pone a cero para el punto (coma) flotante.
Overflow (V)	se establece si overflow es detectado en el destino; en caso contrario, se restablece. Durante overflow, el indicador de error menor también se establece. Para el punto (coma) flotante, el valor de overflow se coloca en el destino. Para un entero, el valor $-32,768$ ó $32,767$ se coloca en el destino. Excepción: si está usando un procesador SLC 5/02, SLC 5/03 ó SLC 5/04 y tiene S:2/14 (bit de selección de overflow matemático) establecido, entonces el overflow sin signo y truncado permanece en el destino.
Cero (Z)	se establece si el resultado es cero; en caso contrario, se restablece.
Signo (S)	se establece si el resultado negativo; en caso contrario, se restablece.

Restar (SUB)



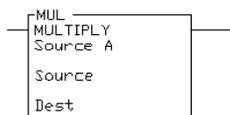
Instrucciones de salida

Usar la instrucción SUB para restar un valor (fuente B) del otro (fuente A) y colocar el resultado en el destino.

Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	se establece si el acarreo es generado; en caso contrario, se restablece (entero). Se pone a cero para el punto (coma) flotante.
Overflow (V)	se establece si es underflow; en caso contrario, se restablece. Durante underflow, el indicador de error menor también se establece. Para el punto (coma) flotante, el valor de overflow se coloca en el destino. Para un entero, el valor $-32,768$ ó $32,767$ se coloca en el destino. Excepción: si está usando un procesador SLC 5/02, SLC 5/03 ó SLC 5/04 y tiene S:2/14 (bit de selección de overflow matemático) establecido, entonces el overflow sin signo y truncado permanece en el destino.
Cero (Z)	se establece si el resultado es cero; en caso contrario, se restablece.
Signo (S)	se establece si el resultado negativo; en caso contrario, se restablece

Multiplicar (MUL)



Instrucciones de salida

Usar la instrucción MUL para multiplicar un valor (fuente A) por el otro (fuente B) y colocar el resultado en el destino.

Actualizaciones de los bits de estado aritmético

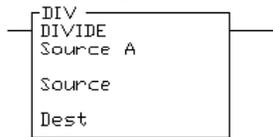
Con este bit:	El procesador:
Acarreo (C)	siempre se restablece.
Overflow (V)	se establece si el overflow se detecta en el destino; en caso contrario, se restablece. Durante el overflow, el indicador de error menor también se establece. El valor $-32,768$ ó $32,767$ se coloca en el destino. Excepción: si usa un procesador SLC 5/02, SLC 5/03 ó SLC 5/04 y tiene S:2/14 (bit de selección de overflow matemático) establecido, el overflow sin signo y truncado permanece en el destino. Para los destinos de punto (coma) flotante, el resultado de overflow permanece en el destino.
Cero (Z)	se establece si el resultado es cero; en caso contrario, se restablece.
Signo (S)	se establece si el resultado negativo; en caso contrario, se restablece

Cambios del registro matemático, S:13 y S:14

Entero – Contiene el resultado con signo de 32 bits de la operación de multiplicación. Este resultado es válido durante el overflow.

Punto (coma) flotante – El registro matemático no se cambia.

Dividir (DIV)



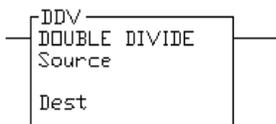
Instrucciones de entrada

Usar la instrucción DIV para dividir un valor (fuente A) entre otro (fuente B). El cociente redondeado se coloca a su vez en el destino. Si el residuo es 0.5 ó mayor, el redondear toma lugar en el destino. El cociente no redondeado se almacena en la palabra más significativa del registro matemático. El resto se coloca en la palabra menos significativa del registro matemático.

Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	Siempre se restablece.
Overflow (V)	Se establece si la división entre cero u overflow se detecta en el destino; en caso contrario, se restablece. Durante el overflow, el indicador de error menor también se establece. El valor 32,767 se coloca en el destino. Excepción: si usa un procesador SLC 5/02, SLC 5/03 ó SLC 5/04 y tiene S:2/14 (bit de selección de overflow matemático) establecido, el overflow sin signo y truncado permanece en el destino. Para los destinos de punto (coma) flotante, el resultado de overflow permanece en el destino.
Cero (Z)	se establece si el resultado es cero; si no, se restablece; no definido si overflow está establecido.
Signo (S)	se establece si el resultado es negativo; si no, se restablece; no definido si el overflow está establecido, se establece si el resultado es negativo; en caso contrario, se restablece

División doble (DDV)



Instrucciones de salida

El contenido de 32 bits del registro matemático se divide entre el valor de fuente de 16 bits y el cociente redondeado se coloca en el destino. Si el residuo es 0.5 ó mayor, se redondea el destino.

Típicamente esta instrucción sigue una instrucción MUL que crea un resultado de 32 bits.

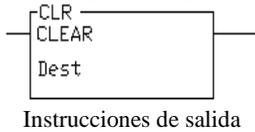
Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	siempre se restablece.
Overflow (V)	se establece si es división en cero o si el resultado es mayor de 32,767 ó menor de -32,768; en caso contrario, se restablece. Durante el overflow, también se establece el indicador de error menor. El valor 32,767 se coloca en el destino.
Cero (Z)	se establece si el resultado es cero; en caso contrario, se restablece
Signo (S)	se establece si el resultado es negativo; en caso contrario, se restablece; no definido si el overflow está establecido.

Cambios del registro matemático, S:13 y S:14

Inicialmente contiene el dividendo de la operación DDV. A la ejecución de instrucción, el cociente no redondeado se coloca en la palabra más significativa del registro matemático. El residuo se coloca en la palabra menos significativa del registro matemático.

Borrar (CLR)

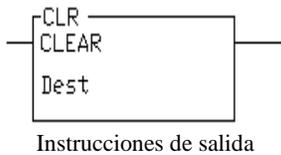


Usar la instrucción CLR para poner a cero el valor de destino de una palabra.

Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	siempre se restablece.
Overflow (V)	siempre se restablece
Cero (Z)	siempre se restablece
Signo (S)	siempre se restablece

Raíz cuadrada (SQR)



Cuando la instrucción es evaluada como verdadera, la raíz cuadrada del valor absoluto de la fuente es calculada y el resultado redondeado se coloca en el destino.

La instrucción calcula la raíz cuadrada de un número negativo sin overflow ni fallos. En las aplicaciones donde el valor de fuente puede ser negativo, use una instrucción de comparación para evaluar el valor de fuente para determinar si el destino puede ser inválido.

Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	es reservado (entero). Para el punto (coma) flotante, siempre está puesto a cero.
Overflow (V)	siempre se restablece
Cero (Z)	se establece cuando el valor de destino es cero.
Signo (S)	siempre se restablece

Cómo escalar con parámetros (SCP)



Instrucciones de salida

Usar la instrucción SCP para producir un valor de salida escalado que tiene una relación lineal entre los valores de entrada y escalados. Esta instrucción tiene capacidad para valores de entero y punto (coma) flotante

Usar la instrucción SCP para producir un valor de salida escalado que tiene una relación lineal entre los valores de entrada y escalados. Esta instrucción tiene capacidad para valores de entero y punto (coma) flotante.

Usar la fórmula siguiente para convertir los datos de entrada analógicos en unidades de ingeniería:

$$y = mx + b$$

Donde:

y = salida escalada

m = pendiente (escala máx. – escala mín.) / (entrada máx. – entrada mín.)

x = valor de entrada

b = offset (intersección y) = escala mín. – (entrada mín. inclinación)

Nota:

La entrada mínima, entrada máxima, escala mínima y escala máxima se usan para determinar los valores de inclinación y offset. El valor de entrada puede salir de los límites de entrada especificados sin requerir la puesta en orden. Por ejemplo, el valor de salida con escala no se encontrará necesariamente fijado entre los valores mínimos y máximos escalados.

Cómo introducir parámetros

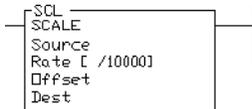
Introducir los parámetros siguientes al programar esta instrucción:

- **El valor de entrada** puede ser una dirección de palabra o una dirección de elementos de datos de punto (coma) flotante.
- **Los valores mínimos y máximos de entrada** determinan el rango de datos que aparece en el parámetro de valor de entrada. El valor puede ser una dirección de palabra, una constante de entero, elemento de datos de punto (coma) flotante o una constante de punto (coma) flotante.
- **Los valores mínimos y máximos escalados** determinan el rango de datos que aparece en el parámetro de salida con escala. El valor puede ser una dirección de palabra, una constante de entero, elemento de datos de punto (coma) flotante o una constante de punto (coma) flotante.
- **El valor de salida escalado** puede ser una dirección de palabra o una dirección de elementos de punto (coma) flotante.

Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	siempre se restablece.
Overflow (V)	se establece si el overflow es generado o si una entrada sin capacidad se detecta; si no, se restablece.
Cero (Z)	Se establece cuando el valor de destino es cero; si no, se restablece.
Signo (S)	Se establece cuando el valor de destino es negativo; si no, se restablece. Siempre se restablece

Escala de datos (SCL)



Instrucciones de salida

Cuando la instrucción es verdadera, el valor en la dirección de fuente se multiplica por el valor del régimen. El resultado redondeado se añade al valor de offset y se coloca en el destino.

Cómo introducir parámetros

El valor para los parámetros siguientes es entre $-32,768$ a $32,767$.

- **La fuente** es una dirección de palabra.
- **El régimen** (o pendiente) es el valor positivo o negativo que se introduce dividido entre 10,000. Puede ser una constante de programa o una dirección de palabra.
- **El offset** puede ser una constante de programa o una dirección de palabra.

Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	es reservado.
Overflow (V)	se establece si un overflow se detecta; en caso contrario, se restablece. Durante un overflow, el bit de error menor S:5/0 también se establece y el valor $-32,768$ ó $32,767$ se coloca en el destino. La presencia de un overflow se verifica antes y después de la aplicación del valor de offset. ¹
Cero (Z)	se establece cuando el valor de destino es cero.
Signo (S)	se establece si el valor de destino es negativo; en caso contrario, se restablece.

ADVERTENCIA:

Si el resultado de la fuente multiplicado por el régimen, dividido entre 10,000 es mayor que 32,767, la instrucción SCL provoca un overflow y causa un error 0020 (bit de error menor) y coloca 32,767 en el destino. Esto ocurre independientemente del offset actual.

Absoluto (ABS)



Instrucciones de salida

Usar la instrucción ABS para calcular el valor absoluto de la fuente y colocar el resultado en el destino. Esta instrucción tiene capacidad para los valores de entero y punto (coma) flotante. Usar la instrucción con procesadores SLC 5/03 OS302 y SLC 5/04 OS401.

Cómo introducir los parámetros

Introducir los parámetros siguientes al programar esta instrucción:

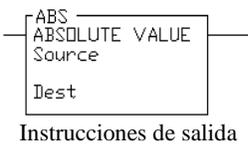
- **La fuente** puede ser una dirección de palabra, una constante de entero, elemento de datos de punto (coma) flotante o una constante de punto (coma) flotante.

- **El destino** sólo puede ser una dirección de palabra o un elemento de datos de punto (coma) flotante.

Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	siempre se restablece.
Overflow (V)	siempre se restablece con un valor de punto (coma) flotante; se establece si la entrada es $-32,768$ (valor de entero).
Cero (Z)	se establece cuando el valor de destino es cero; en caso contrario, se restablece.
Signo (S)	siempre se restablece.

Calcular (CPT)



La instrucción CPT efectúa operaciones de copiar, aritméticas, lógicas y conversión. Se define la operación en la expresión y el resultado se escribe en el destino. El CPT usa funciones para operar en uno o más valores en la expresión para efectuar operaciones tales como:

- convertir de un formato de número a otro
- manejar los números
- efectuar funciones trigonométricas

Usar esta instrucción con los procesadores SLC 5/03 OS302 y SLC 5/04 OS401.

Las instrucciones que se pueden usar en la expresión incluyen:

+, -, *, | (DIV), SQR, - (NEG), NOT, XOR, OR, AND, TOD, FRD, LN, TAN, ABS, DEG, RAD, SIN, COS, ATN, ASN, ACS, LOG y ** (XPY).

Nota

El tiempo de ejecución de una instrucción CPT es mayor que el de una sola operación aritmética y usa más palabras de instrucción.

Cómo introducir parámetros

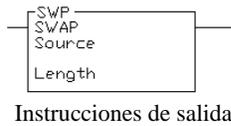
Introducir los parámetros siguientes al programar esta instrucción:

- **El destino** puede ser una dirección de palabra o la dirección de un elemento de datos de punto (coma) flotante.
- **La expresión** es cero o más líneas, con hasta 28 caracteres por línea, hasta 255 caracteres.

Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	se establece según el resultado de la última instrucción en la expresión.
Overflow (V)	se establece cuando un overflow ocurre durante la evaluación de la expresión.
Cero (Z)	se establece según el resultado de la última instrucción en la expresión.
Signo (S)	se establece según el resultado de la última instrucción en la expresión. Los bits anteriores son puestos a cero al inicio de la instrucción CPT.

Intercambio (SWP)



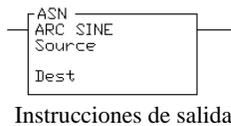
Usar esta instrucción para intercambiar los bytes bajos y altos de un número de palabras especificado en un archivo de bit, entero, ASCII o cadena. Usar la instrucción con los procesadores SLC 5/03 OS302 y SLC 5/04 OS401.

Cómo introducir parámetros

Introducir los parámetros siguientes al programar esta instrucción:

- **La fuente** sólo puede ser una dirección de palabra indexada.
- **La longitud** es una referencia al número de palabras que van a intercambiarse, pese al tipo de archivo. La dirección se limita a constantes de entero. Para los archivos de tipo bit, entero y ASCII, el rango de longitud es de 1 a 128. Para el archivo de tipo cadena, el rango de longitud es de 1 a 41. Notar que esta instrucción se restringe a un solo elemento de cadena y no puede cruzar un límite de elemento de cadena.

Arco seno (ASN)

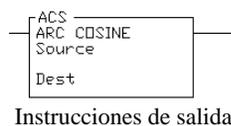


Usar la instrucción ASN para tomar el arco seno de un número (fuente en radianes) y almacenar el resultado (en radianes) en el destino. La fuente debe ser mayor o igual que -1 y menor o igual que 1 . El valor resultante en el destino siempre es mayor o igual que $-\pi/2$ y menor o igual que $\pi/2$ donde $\pi = 3.141592$. Usar la instrucción con los procesadores SLC 5/03 OS302 y SLC 5/04 OS401.

Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	siempre se restablece.
Overflow (V)	se establece si un overflow es generado o una entrada sin capacidad se detecta; si no, se restablece.
Cero (Z)	se establece si el resultado es cero; en caso contrario, se restablece.
Signo (S)	se establece si el resultado es negativo; en caso contrario, se restablece.

Arco coseno (ACS)



Usar la instrucción ACS para tomar el arco seno de un número (fuente en radianes) y almacenar el resultado (en radianes) en el destino. La fuente debe ser mayor o igual que -1 y menor o igual que 1 . El valor resultante en el destino siempre es mayor o igual que 0 y menor o igual que π , donde $\pi = 3.141592$. Usar la instrucción con los procesadores SLC 5/03 OS302 y SLC 5/04 OS401.

Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	siempre se restablece.
Overflow (V)	se establece si un overflow es generado o una entrada sin capacidad se detecta; en caso contrario, se restablece.
Cero (Z)	se establece si el resultado es cero; en caso contrario, se restablece.
Signo (S)	siempre se restablece.

Arco tangente (ATN)

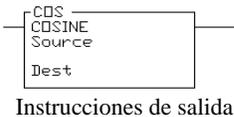


Se usa para tomar el arco tangente de un número (fuente) y almacenar el resultado (en radianes) en el destino. El valor resultante en el destino siempre es mayor o igual que $-\pi/2$ y menor o igual que $\pi/2$, donde $\pi = 3.141592$. Usar la instrucción con los procesadores SLC 5/03 OS302 y SLC 5/04 OS401.

Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	siempre se restablece.
Overflow (V)	se establece si un overflow es generado o una entrada sin capacidad se detecta; en caso contrario, se restablece.
Cero (Z)	establece si el resultado es cero; en caso contrario, se restablece.
Signo (S)	se establece si el resultado es negativo; en caso contrario, se restablece.

Coseno (COS)



Usar la instrucción COS para tomar el coseno de un número (fuente en radianes) y almacenar el resultado (en radianes) en el destino. La fuente debe ser mayor o igual que $-205,887.4$ y menor o igual que $205,887.4$. La óptima exactitud se obtiene cuando la fuentes es mayor que -2π y menor que 2π , donde $\pi = 3.141592$. El valor resultante en el destino siempre es mayor o igual que -1 y menor o igual que 1 . Usar la instrucción con los procesadores SLC 5/03 OS302 y SLC 5/04 OS401.

Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	siempre se restablece.
Overflow (V)	se establece si un overflow es generado o una entrada sin capacidad se detecta; en caso contrario, se restablece.
Cero (Z)	establece si el resultado es cero; en caso contrario, se restablece.
Signo (S)	siempre se restablece.

Logaritmo natural (LN)

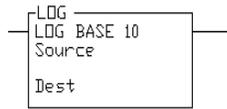


Usar la instrucción LN para tomar el logaritmo natural del valor en la fuente y almacenar el resultado en el destino. La fuente debe ser mayor que cero. El valor resultante en el destino siempre es mayor que o igual que -87.33654 y menor o igual que 88.72284 . Usar la instrucción con los procesadores SLC 5/03 OS302 y SLC 5/04 OS401.

Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	siempre se restablece.
Overflow (V)	se establece si un overflow es generado o una entrada sin capacidad se detecta; en caso contrario, se restablece.
Cero (Z)	se establece si el resultado es cero; en caso contrario, se restablece.
Signo (S)	siempre se restablece.

Logaritmo a la base 10 (LOG)



Instrucciones de salida

Usar la instrucción LOG para tomar el logaritmo de base 10 del valor en la fuente y almacenar el resultado en el destino. La fuente debe ser mayor que cero. El valor resultante en el destino siempre es mayor o igual que -37.92978 y menor o igual que 38.53184 . Usar la instrucción con los procesadores SLC 5/03 OS302 y SLC 5/04 OS401.

Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	siempre se restablece.
Overflow (V)	se establece si un overflow es generado o una entrada sin capacidad se detecta; en caso contrario, se restablece.
Cero (Z)	se establece si el resultado es cero; en caso contrario, se restablece.
Signo (S)	siempre se restablece.

Seno (SIN)



Instrucciones de salida

Usar la instrucción SIN para tomar el seno de un número (fuente en radianes) y almacenar el resultado en el destino. La fuente debe ser mayor o igual que $-205,887.4$ y menor o igual que $205,887.4$. La óptima exactitud se obtiene cuando la fuente es mayor que -2π y menor que 2π , donde $\pi = 3.141592$. El valor resultante en el destino siempre es mayor o igual que -1 y menor o igual que 1 . Usar la instrucción con los procesadores SLC 5/03 OS302 y SLC 5/04 OS401.

Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	siempre se restablece.
Overflow (V)	se establece si un overflow es generado o una entrada sin capacidad se detecta; en caso contrario, se restablece.
Cero (Z)	se establece si el resultado es cero; en caso contrario, se restablece.
Signo (S)	siempre se restablece.

Tangente (TAN)



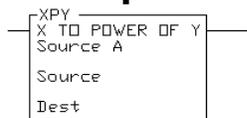
Instrucciones de salida

Usar la instrucción TAN para tomar la tangente de un número (fuente en radianes) y almacenar el resultado en el destino. El valor de la fuente debe ser mayor o igual que -102943.7 y menor o igual que 102943.7 . La óptima exactitud se obtiene cuando la fuente es mayor que -2π y menor que 2π , donde $\pi = 3.141592$. El valor resultante en el destino es un número real o infinito. Usar la instrucción con los procesadores SLC 5/03 OS302 y SLC 5/04 OS401.

Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	siempre se restablece.
Overflow (V)	se establece si un overflow es generado o una entrada sin capacidad se detecta; en caso contrario, se restablece.
Cero (Z)	se establece si el resultado es cero; en caso contrario, se restablece.
Signo (S)	siempre se restablece.

X a la potencia de Y (XPY)



Instrucciones de salida

Usar la instrucción XPY para elevar un valor (fuente A) a una potencia (fuente B) y almacenar el resultado en el destino. Si el valor en la fuente A es negativo, la exponente (fuente B) debe ser un número entero. Si no es un número entero, el bit de overflow se establece y el valor absoluto de la base se usa en el cálculo. Usar la instrucción con los procesadores SLC 5/03 OS302 y SLC 5/04 OS401.

La instrucción XPY usa el algoritmo siguiente:

$$XPY = 2^{**} (Y * \log_2 (X))$$

Si cualquiera de las operaciones intermedias en este algoritmo provoca un overflow, se establece el bit de estado de overflow aritmético (S:01/).

Actualizaciones de los bits de estado aritmético

Con este bit:	El procesador:
Acarreo (C)	siempre se restablece.
Overflow (V)	se establece si un overflow es generado o una entrada sin capacidad se detecta; en caso contrario, se restablece.
Cero (Z)	se establece si el resultado es cero; en caso contrario, se restablece.
Signo (S)	siempre se restablece.

Instrucciones de comparación

En esta sección estudiaremos el uso y funcionamiento de las instrucciones de comparación. Además cada una de las instrucciones de comparación incluye información acerca de:

- cómo debe aparecer el símbolo de instrucción
- cómo usar la instrucción

Instrucción		Propósito	Página
Mnemónico	Nombre		
EQU	Igual	Probar si dos valores son iguales	59
NEQ	No igual	Probar si un valor no es igual que un segundo valor	60
LES	Menor que	Probar si un valor es menor que un segundo valor.	60
LEQ	Menor que o igual que	Probar si un valor es menor o igual que un segundo valor.	60
GRT	Mayor que	Probar si un valor es mayor que otro.	60
GEQ	Mayor o igual que	Probar si un valor es mayor o igual que un segundo valor.	61
MEQ	Comparación igualdad con máscara	Probar porciones de dos valores para saber si son iguales. Compara datos de 16 bits de una dirección de fuente contra datos de 16 bit en una dirección de referencia mediante una máscara.	61
LIM	Prueba de límite	Probar si un valor se encuentra dentro del rango de límite de otros dos valores.	61

Acerca de las instrucciones de comparación

Las instrucciones de comparación se usan para probar parejas de valores para establecer condiciones de la continuidad lógica de un renglón.

3.3.2. Descripción general de las instrucciones de comparación.

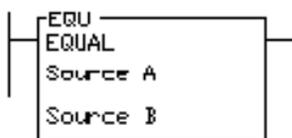
Uso de direcciones de palabra indexadas

Al usar las instrucciones de comparación, tiene la opción de usar direcciones de palabra indexadas para parámetros de instrucción especificando direcciones de palabra.

Uso de direcciones de palabra indirectas

Tiene la opción de usar direcciones indirectas a nivel de palabra y a nivel de bit para instrucciones especificando direcciones de palabra cuando usa los procesadores SLC 5/03 OS302 y SLC 5/04 OS401.

Igual (EQU)



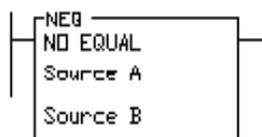
Instrucciones de salida



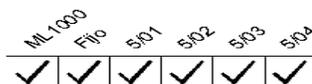
Usar la instrucción EQU para probar si dos valores son iguales. Si la fuente A y la fuente B son iguales, la instrucción es lógicamente verdadera. Si estos valores no son iguales, la instrucción es lógicamente falsa.

La fuente A debe ser una dirección. La fuente B puede ser una constante de programa o una dirección. Los enteros negativos se almacenan de forma complementaria de dos.

No igual (NEQ)



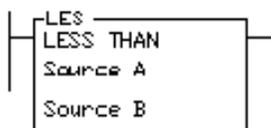
Instrucciones de salida



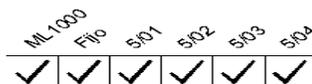
Usar la instrucción NEQ para probar si dos valores no son iguales. Si la fuente A y la fuente B no son iguales, la instrucción es lógicamente verdadera. Si los dos valores son iguales, la instrucción es lógicamente falsa.

La fuente A debe ser una dirección. La fuente B puede ser un constante de programa o una dirección. Los enteros negativos se almacenan de forma complementaria de dos.

Menor que (LES)

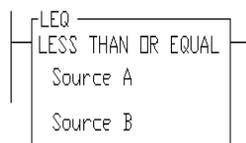


Instrucciones de salida

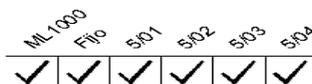


Usar la instrucción LES para probar si un valor (fuente A) es menor que otro (fuente B). Si la fuente A es menor que el valor en la fuente B, la instrucción es lógicamente verdadera. Si el valor en la fuente A es mayor o igual que el valor en la fuente B, la instrucción es lógicamente falsa. La fuente A debe ser una dirección. La fuente B puede ser una constante de programa o una dirección. Los enteros negativos se almacenan de forma complementaria de dos.

Menor o igual que (LEQ)

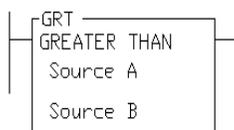


Instrucciones de salida

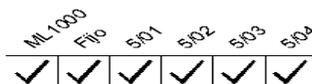


Usar la instrucción LEQ para probar si un valor (fuente A) es menor o igual que otro (fuente B). Si la fuente A es menor o igual que el valor en la fuente B, la instrucción es lógicamente verdadera. Si el valor en la fuente A es mayor que el valor en la fuente B, la instrucción es lógicamente falsa. La fuente A debe ser una dirección. La fuente B puede ser una constante de programa o una dirección. Los enteros negativos se almacenan de forma complementaria de dos.

Mayor que (GRT)



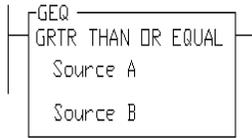
Instrucciones de salida



Usar la instrucción GRT para probar si un valor (fuente A) es mayor que otro (fuente B). Si la fuente A es mayor que el valor en la fuente B, la instrucción es lógicamente verdadera. Si el valor en la fuente A es menor o igual que el valor en la fuente B, la instrucción es lógicamente falsa. La fuente A debe ser una dirección. La fuente B puede ser un constante de programa o una dirección. Los enteros negativos se almacenan de forma complementaria de dos.

Mayor o igual que (GEQ)

ML1000	Fifo	5/01	5/02	5/03	5/04
✓	✓	✓	✓	✓	✓

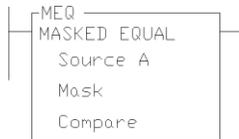


Instrucciones de salida

Usar la instrucción GEQ para probar si un valor (fuente A) es mayor o igual que otro (fuente B). Si la fuente A es mayor o igual que el valor en la fuente B, la instrucción es lógicamente verdadera. Si el valor en la fuente A es menor que el valor en la fuente B, la instrucción es lógicamente falsa. La fuente A debe ser una dirección. La fuente B puede ser un constante de programa o una dirección. Los enteros negativos se almacenan de forma complementaria de dos.

Comparación con máscara para igual (MEQ)

ML1000	Fifo	5/01	5/02	5/03	5/04
✓	✓	✓	✓	✓	✓



Instrucciones de salida

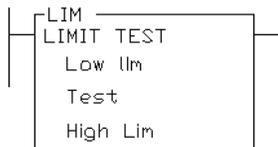
Usar la instrucción MEQ para comparar datos en una dirección de fuente contra datos en una dirección de comparación. El uso de esta instrucción permite que una palabra separada enmascare porciones de datos.

Cómo introducir parámetros

- **Fuente** es la dirección del valor que desea comparar.
- **Máscara** es la dirección de la máscara mediante la cual la instrucción mueve datos. La máscara puede ser un valor hexadecimal.
- **Comparación** es un valor de entero o la dirección de la referencia. Si los 16 bits de datos en la dirección de fuente son iguales a los 16 bits de datos en la dirección de comparación (menos los bits con máscara), la instrucción es verdadera. La instrucción se hace falsa en el momento en que detecta una desigualdad. Los bits en la palabra de máscara enmascaran los datos al restablecerse; transmiten datos al establecerse.

Prueba de límite (LIM)

ML1000	Fifo	5/01	5/02	5/03	5/04
✓	✓	✓	✓	✓	✓



Instrucciones de salida

Usar la instrucción LIM para probar los valores dentro o fuera de un rango especificado, según cómo se haya establecido los límites.

Cómo introducir parámetros

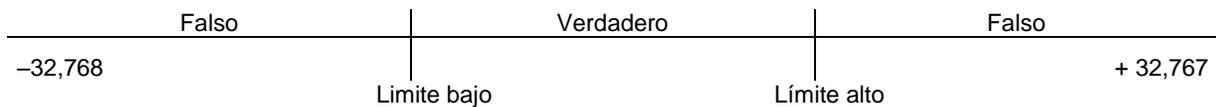
Los valores de límite bajo, prueba y límite alto pueden ser direcciones de palabra o constantes restringidas a las combinaciones siguientes:

Si el parámetro de prueba es una constante de programa, los parámetros de límite bajo y límite alto deben ser direcciones de palabra.

Si el parámetro de prueba es una dirección de palabra, los parámetros de límite bajo y límite alto pueden ser una constante de programa o una dirección de palabra.

Estado verdadero/falso de la instrucción

Si el límite bajo tiene un valor igual o menor que el límite alto, la instrucción es verdadera cuando el valor de prueba se encuentra entre los límites o cuando es igual a cualquiera de los límites. Si el valor de prueba se encuentra fuera de los límites, la instrucción es falsa, según se indica a continuación.



Ejemplo. – límite bajo menor que el límite alto:

Límite bajo	Límite alto	La instrucción es verdadera cuando el valor de prueba es	La instrucción es falsa cuando el valor de prueba es
5	8	5 a 8	-32,768 a 4 y 9 a 32,767

Si el límite bajo tiene un valor mayor que el límite alto, la instrucción es falsa cuando el valor de prueba se encuentra entre los límites. Si el valor de prueba es igual a cualquiera de los límites o se encuentra fuera de los límites, la instrucción es verdadera, según se indica a continuación.

Ejemplo – límite bajo mayor que el límite alto:



Ejemplo – límite bajo menor que el límite alto:

Límite bajo	Límite alto	La instrucción es verdadera cuando el valor de prueba es	La instrucción es falsa cuando el valor de prueba es
5	8	-32,768 a 5 y 8 a 32,767	6 y 7

3.3.3. Descripción general de las instrucciones de movimiento.

Cómo introducir parámetros

- **La fuente** es la dirección del valor en que la operación de mover o lógica se debe efectuar. La fuente puede ser una dirección de palabra o una constante de programa, a menos que se describa lo contrario. Si la instrucción tiene dos operandos de fuente, no acepta constantes de programa en ambos operandos.

Cuando usa un procesador SLC 5/03 OS301, OS302 ó SLC 5/04 OS400, OS401, soportan los valores de punto (coma) flotante y cadena.

- **El destino** es la dirección de resultado de una operación de mover o lógica. Debe ser una dirección de palabra.

Uso de direcciones de palabra indexadas

Se tiene la opción de usar direcciones de palabra indexadas como parámetros de instrucción especificando las direcciones de palabra.

Actualizaciones de los bits de estado aritmético

Si los bits de estado aritmético se encuentran en la palabra 0, bits 0–3 en el archivo de estado del controlador. Después de la ejecución de una instrucción, se actualizan los bits de estado aritmético en el archivo de estado.

Uso de direcciones de palabra indirectas

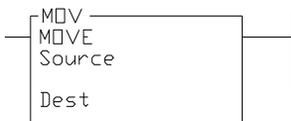
Se tiene la opción de usar direcciones indirectas a nivel de palabra y a nivel de bit para instrucciones especificando las direcciones de palabra cuando usa un procesador SLC 5/03 OS302 y SLC 5/04 OS401.

Cambios del registro matemático, S:13 y S:14

Las instrucciones de mover y lógicas no afectan el registro matemático.

Mover (MOV)

ML1000	Flt	5/01	5/02	5/03	5/04
✓	✓	✓	✓	✓	✓



Instrucciones de salida

La instrucción de salida mueve el valor de fuente al lugar de destino. Siempre que el renglón permanezca verdadero, la instrucción mueve los datos durante cada escán.

Cómo introducir parámetros

Introducir los parámetros siguientes al programar esta instrucción:

- **La fuente** es la dirección o constante de los datos que desea mover.
- **El destino** es la dirección a la cual la instrucción mueve los datos.

Nota de aplicación:

Si se desea mover una palabra de datos sin afectar los indicadores matemáticos, usar una instrucción de copiar (COP) con una longitud de 1 palabra en vez de la instrucción MOV.

Actualizaciones de los bits de estado aritmético

Con este bit:	El controlador:
S:0/0 Acarreo (C)	siempre se restablece.
S:0/1 Overflow (V)	siempre se restablece.
S:0/2 Cero (Z)	se establece si el resultado es cero; en caso contrario, se restablece.
S:0/3 Signo (S)	se establece si el resultado es negativo (el bit más significativo establecido); en caso contrario, se restablece.

3.4 Instrucciones de flujo de programa

En esta sección se tiene información general acerca de las instrucciones de flujo de programa y explica cómo funcionan en su programa de aplicación. Cada una de las instrucciones incluye información acerca de:

- cómo aparece el símbolo de instrucción
- cómo usar la instrucción

Instrucciones de flujo de programa

Instrucción		Propósito	Página
Mnemónico	Nombre		
JMP y LBL	Saltar a etiqueta y Etiqueta.	Saltar hacia adelante o hacia atrás a la instrucción de etiqueta especificada.	64
JSR, SBR, y RET	Saltar a subrutina, subrutina, y retornar de la subrutina.	Saltar a una subrutina designada y retornar.	65
MCR	Restablecimiento del control maestro.	Desactivar todas las salidas no retentivas en una sección de un programa de escalera.	68
TND	Fin temporal	Marcar un fin temporal que detiene la ejecución del programa.	69
SUS	Suspender.	Identifica condiciones específicas para la depuración del programa, la localización y corrección de fallos del sistema.	69

Acerca de las instrucciones de control de flujo de programa

Usar estas instrucciones para controlar la secuencia en que se ejecuta el programa. Las instrucciones de control permiten cambiar el orden en que el procesador realiza un escán de un programa de escalera. Estas instrucciones típicamente se usan para minimizar el tiempo de escán, crear un programa más eficiente para localizar y corregir fallos de un programa de escalera.

3.4.1. Salto (JMP) y etiqueta (LBL)

Usar estas instrucciones conjuntamente para saltar porciones del programa de escalera.

—(JMP)—	Si el renglón que contiene la instrucción de salto es:	El programa:
—[LBL]—	Verdadero	Salta del renglón que contiene la instrucción JMP al renglón que contiene la instrucción LBL designada y sigue ejecutando. Puede saltar hacia adelante o hacia atrás.
Instrucciones de salida	Falso	No ejecuta la instrucción JMP.

Nota

Al saltar hacia adelante a una etiqueta ahorra el tiempo de escán del programa eliminando un segmento de programa hasta que sea necesario. Al saltar hacia atrás le permite al controlador ejecutar segmentos de programa repetidamente.

Tener cuidado de no saltar hacia atrás excesivamente. El temporizador de control (watchdog) podría sobrepasar el límite de tiempo y causar un fallo del controlador.

Usar un contador, temporizador o el registro de “escán de programa” (registro de estado de sistema, palabra S:3, bits 0–7) para limitar el tiempo que se pasa realizando lazos dentro de las instrucciones JMP/LBL.

Cómo introducir parámetros

Introducir un número de etiqueta decimal de 0 a 999. Se puede colocar hasta:

- 256 etiquetas en cada archivo de subrutina para los procesadores SLC

Uso de JMP

La instrucción JMP causa que el controlador salte renglones. Se puede saltar a la misma etiqueta desde una o más instrucciones JMP.

Uso de LBL

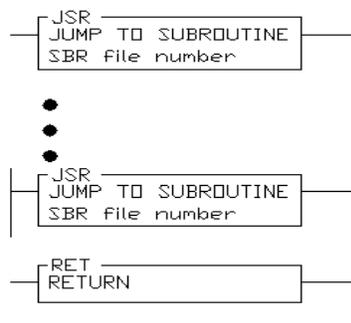
La instrucción de entrada es el blanco de las instrucciones JMP que tienen el mismo número de etiqueta. Se debe programar esta instrucción como la primera instrucción de un renglón. Esta instrucción no tiene bits de control.

Se puede programar saltos múltiples a la misma etiqueta asignando el mismo número de etiqueta a instrucciones JMP múltiples. Sin embargo, los números de etiqueta deben ser únicos.

Nota:

No salte (JMP) en una zona MCR. Las instrucciones programadas dentro de la zona MCR a partir de la instrucción LBL hasta la instrucción 'END MCR' siempre serán evaluadas como si la zona MCR fuera verdadera, sin importar el estado verdadero de la instrucción “Start MCR”.

3.4.2. Saltar a subrutina (JSR), subrutina (SBR), y retornar (RET)



Instrucciones de salida

Las instrucciones JSR, SBR y RET se usan para indicar al controlador que ejecute un archivo de subrutina separado dentro del programa de escalera y retornar a la instrucción siguiente a la instrucción JSR.

Nota:

Si se usa la instrucción SBR, ésta debe ser la primera instrucción en el primer renglón en el archivo de programa que contiene la subrutina.

Usar una subrutina para almacenar secciones repetidas de lógica de programa que se debe ejecutar desde varios puntos dentro de su programa de aplicación. Una subrutina ahorra memoria porque se programa sólo una vez.

Actualizar E/S críticas dentro de subrutinas usando las instrucciones de entrada y/o salida inmediata (IIM, IOM), especialmente si la aplicación requiere subrutinas anidadas o largas. En caso contrario, el controlador no actualizará la E/S hasta que llegue al final del programa principal (después de ejecutar todas las subrutinas).



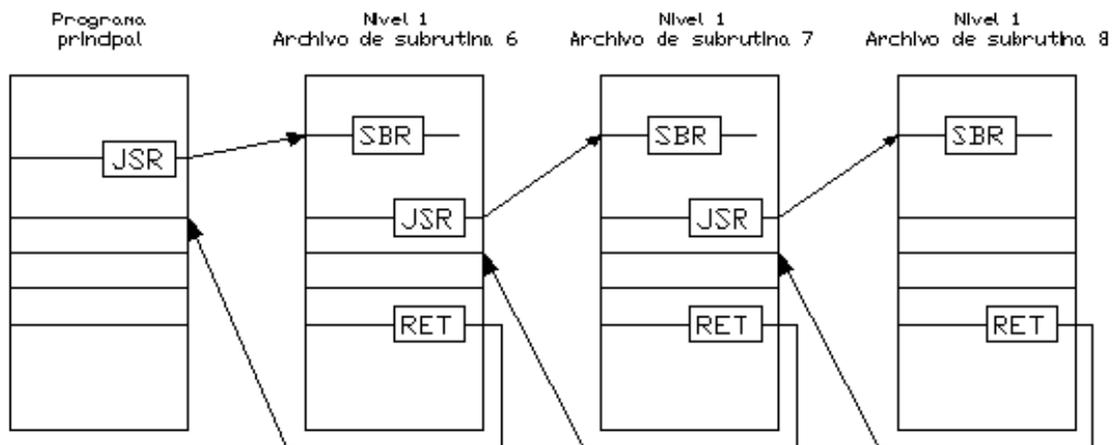
Las salidas controladas dentro de una subrutina permanecen en su último estado hasta que la subrutina se vuelva a ejecutar.

Cómo anidar archivos de subrutina

El anidar subrutinas le permite dirigir el flujo de programa desde el programa principal hasta una subrutina y luego a otra subrutina. Las reglas siguientes se aplican al anidar subrutinas: Se puede anidar hasta ocho niveles de subrutinas. Si se usa una subrutina STI, una subrutina de interrupción HSC o una rutina de fallo del usuario, puede anidar subrutinas hasta tres niveles desde cada subrutina.

- Con los procesadores fijos y SLC 5/01, se puede anidar subrutinas hasta cuatro niveles.
- Con los procesadores SLC 5/02, SLC 5/03, SLC 5/04, se puede anidar subrutinas hasta ocho niveles. Si usa una subrutina STI, subrutina de interrupción provocada por evento de E/S, una rutina de fallo del usuario o una subrutina de interrupción HSC, se puede anidar subrutinas hasta tres niveles desde cada subrutina. La ilustración siguiente muestra cómo se pueden anidar las subrutinas.

Ejemplo de cómo anidar subrutinas hasta el nivel 3



Ejemplo de cómo anidar subrutinas hasta el nivel 3

Ocurrirá un error si se llaman más niveles de subrutinas que los permitidos (overflow de pila de subrutina) o si se ejecutan más retornos que niveles de llamada existentes (underflow de pila de subrutina).

Uso de JSR

Cuando la instrucción JSR se ejecuta, el controlador salta a la instrucción de subrutina (SBR) al inicio del archivo de subrutina destino y reanuda la ejecución desde aquel punto. No se puede saltar en una parte de una subrutina con excepción de la primera instrucción en ese archivo.

Se debe programar cada subrutina en su propio archivo de programa asignando un número de archivo único:

- 3–255 para los procesadores SLC

Fijo y específico del SLC 5/01 – La instrucción JSR no se debe programar en bifurcaciones de salida anidadas. Un error de compilador ocurrirá si se encuentra un renglón que contenga salidas múltiples con lógica condicional y una instrucción JSR.

Uso de SBR

La subrutina de destino se identifica por el número de archivo que usted introdujo en la instrucción JSR. Esta instrucción sirve como etiqueta o identificador de un archivo de programa designado como un archivo de subrutina normal.

La instrucción no tiene bits de control. Siempre se evalúa como verdadera. La instrucción se debe programar como la primera instrucción en el primer renglón de una subrutina. El uso de esta instrucción es opcional; sin embargo, se recomienda su uso para obtener mayor claridad.

Uso de RET

Esta instrucción de salida indica el fin de ejecución de subrutina o el fin del archivo de subrutina. Causa que el controlador reanude la ejecución en la instrucción siguiente a la instrucción JSR. Si se involucra una secuencia de subrutinas anidadas, la instrucción causa que el procesador retorne la ejecución de programa a la subrutina anterior.

El renglón que contiene la instrucción RET puede ser condicional si este renglón precede el final de la subrutina. De esta manera el controlador elimina el resto de una subrutina sólo si su condición de renglón es verdadera.

Sin instrucción RET, la instrucción END (siempre presente en la subrutina) retorna automáticamente la ejecución de programa a la instrucción siguiente a la instrucción JSR en el archivo de escalera que llama.

Nota:

La instrucción RET termina la ejecución de la subrutina DII (procesadores SLC 5/03 y SLC 5/04), la subrutina STI, la subrutina de interrupción provocada por evento de E/S y el indicador de error del usuario cuando se usa un procesador SLC 5/02, SLC 5/03 ó SLC 5/04.

3.4.3. Restablecimiento de control maestro (MCR)

—(MCR)—

Instrucciones de salida

Usar las instrucciones MCR conjuntamente para crear zonas de programa que desactiven todas las salidas no retentivas en la zona. Los renglones dentro de la zona MCR todavía son escaneados, pero el tiempo de escán se reduce debido al estado falso de las salidas no retentivas.

Si el renglón MCR que inicia la zona es:	El controlador:
Verdadero	Ejecuta los renglones en la zona MCR según la condición de entrada de cada renglón (como si la zona no existiera)
Falso	Restablece todas las instrucciones de salida no retentiva en la zona MCR pese a las condiciones de entrada de cada renglón.

Las zonas MCR le permiten habilitar o inhabilitar segmentos del programa.

Cuando se programa las instrucciones MCR, observe lo siguiente:

- Se debe terminar la zona con una instrucción MCR no condicional.
- No puede anidar una zona MCR dentro de otra.
- No saltar una zona MCR. Si la zona es falsa, el saltar a ella activa la zona.
- Siempre coloque la instrucción MCR como la última instrucción en un renglón.

Nota:

La instrucción MCR no substituye un relé cableado de control maestro que proporciona la capacidad de detención de emergencia. Todavía debe instalar un relé cableado de control maestro para proporcionar la interrupción de alimentación eléctrica de E/S en casos de emergencia.



Si inicia instrucciones tales como temporizadores o contadores en una zona MCR, la operación de instrucción se detiene cuando la zona se inhabilita. Vuelva a programar operaciones críticas fuera de la zona si fuese necesario.

Operación del procesador SLC

No saltar (JMP) a una zona MCR. Las instrucciones programadas dentro de la zona MCR, que comienzan con una instrucción LBL y terminan con la instrucción 'END MCR', siempre son evaluadas como si la zona MCR fuera verdadera, sin importar el estado verdadero de la instrucción "Start MCR". Si la zona es falsa, el saltar a ella activa la zona desde la LBL hasta el final de la zona.



Si inicia instrucciones tales como temporizadores o contadores en una zona MCR, la operación de instrucción se detiene cuando la zona se inhabilita. Volver a programar operaciones críticas fuera de la zona si fuese necesario. El temporizador TOF se activará (cuando se coloque)

dentro de una zona MCR falsa.

La instrucción MCR no sustituye el relé cableado de control maestro. Se recomienda que el sistema de controlador programable incluya un relé de cableado de control maestro e interruptores de parada de emergencia para proporcionar la interrupción de alimentación eléctrica de E/S. Los interruptores de parada de emergencia se pueden monitorizar pero no deben ser controlados por el programa de escalera. Cablear estos dispositivos según lo descrito en el manual de instalación.



Específico de SLC 5/03 y SLC 5/04 – Cuando está en línea y existe en su programa una instrucción MCR desigual, la instrucción END sirve como la segunda instrucción MCR no condicional y todos los renglones siguientes a la primera instrucción MCR se ejecutan por medio del estado de instrucción MCR actual.

Se puede guardar el programa cuando está en línea si existen instrucciones MCR desatendidas. Sin embargo, si está fuera de línea y existen instrucciones MCR desatendidas, ocurrirá un error.

3.4.4. Fin temporal (TND)

—————<TND>—————

Instrucciones
de salida

Cuando el renglón de esta instrucción es verdadero, previene que el procesador realice un escán del resto del archivo de programa, actualiza la E/S y reanuda el escaneado a partir del renglón 0 del programa principal (archivo 2). Si la instrucción de este renglón es falsa, el procesador sigue realizando el escán hasta la próxima instrucción TND o el comando END. Usar la instrucción para depurar progresivamente un programa o eliminar condicionalmente el resto de su archivo de programa o subrutinas existentes.

Nota:

Si se usa la instrucción dentro de una subrutina anidada, se termina la ejecución de todas las subrutinas anidadas.

3.4.5. Suspend (SUS)



Instrucciones de salida

Cuando esta instrucción se ejecuta, causa que el procesador entre en el modo de Suspend/Idle y almacena la identificación de suspender en palabra 7 (S:7) del archivo de estado. Todas las salidas se desactivan. Suspend (SUS).

Usar esta instrucción para capturar e identificar condiciones específicas para la depuración de programas y la localización y corrección de fallos de sistemas.

Cómo introducir parámetros

Introducir un número de identificación de suspender de $-32,768$ a $+32,767$ al programar la instrucción.

Cuando la instrucción SUS se ejecuta, la identificación programada así como la identificación del archivo de programa desde el cual se ejecuta la instrucción SUS se colocan en el archivo de estado del sistema.

Capítulo

4

AUTOMATIZACIÓN DE UNA LINEA DE LLENADO DE PASTILLAS

En este capítulo describiremos la secuencia de operación por etapas de la Llenadora de Pastillas. Aquí haremos mención de los dispositivos eléctricos y de control que intervienen en la automatización, para lo cual nos auxiliaremos de los diagramas de conexión punto a punto, los diagramas eléctricos y por supuesto se presentara el programa en escalera.

4.1 Automatización de una Línea de Llenado de pastillas.

Hay infinidad de temas que se pueden tratar con la aplicación de los Controladores Lógicos Programables, ¿el porque la automatización de un sistema de Llenado de pastillas?, podría ser porque fue uno de los primeros equipos que en lo personal pude trabajar y automatizar. El sistema consiste en el mejoramiento del llenado de pastillas. En el comienzo del proceso se requiere del suministro de la metería prima, en este caso son pastillas almacenadas en tambores metálicos que después serán vaciados mediante una grúa eléctrica-manual para la acumulación de producto en el fondo de la tolva para dejar caer las pastillas a una charola con doble fondo y que tiene pendiente para el desplazamiento de la pastilla. La charola esta acoplada a un vibrador eléctrico que hace que se desplace la pastilla mediante la vibración, la charola de arriba sirve como un colador ya que tiene orificios circulares con la medida suficiente para no dejar pasar las pastillas que estén enteras, pero las rotas y se encuentren pedazos deben caer en la placa del fondo. Conforme se desplaza la pastilla va acumulando pastillas a la Llenadora para el dosificado de las cajillas.



1ra Etapa.

Toma de estuche (cajilla) y formación. Se debe alimentar de cajas al contenedor que es lineal-vertical que mediante la acción de un brazo mecánico que en su interior es hueco y al extremo del brazo tiene un chupón que con la ayuda de un sistema de vacio realiza la toma del estuche para ser depositado sobre el transportador.



La toma de estuche se deposita sobre la banda para la formación de la caja.

2da Etapa.

Consiste en formar la bolsa la cual será introducida dentro de cajilla para formar una envoltente y no se contamine el producto. En esta parte se coloca un sensor para la detección de la bolsa el cual esta sincronizado con una leva mecánica para determina ausencia o presencia de bolsa para que al final del transportador sea expulsado.



3ra Etapa.

En esta tercera etapa que es llenado de las cajillas las cuales en cada paso o ciclo de la maquina se dosifica dos pastillas para esto se cuenta con una regleta que contiene veinte huecos para que en un lapso de diez ciclos la cajilla llegue a un total de veinte pastillas.

Aquí en la llenadora se colocó un soporte con veinte sensores con el mismo espaciado de la regleta de pastillas para detectar una a una de las pastillas. Para detectar un hueco o pastilla se basa en el principio siguiente, existe un seguidor el cual debe mantener una posición vertical y que mediante la sincronización de una leva mecánica y un sensor determinan la posición de la pastilla. El seguidor está construido de material laminado grado alimenticio y en la parte superior tiene incrustado un poste metálico y en la parte inferior tiene una rueda del mismo material que el seguidor. El seguidor mediante un movimiento de péndulo determina ausencia o presencia de pastilla, este movimiento lo realiza con una rueda que va detectando pastillas en la carrillera para así mantener su posición vertical, cuando hay faltante de pastilla el seguidor detecta un hueco y pierde su posición vertical al extremo superior tiene el incruste metálico y cuando el sensor del tipo inductivo pierde la posición vertical, el cual está en sincronía con la leva mecánica mandan una señal al PLC de ausencia de pastilla.



Ya se detectó ausencia de pastilla, ¿qué pasa si falta una pastilla en la regleta? La cajilla va a tener menos pastillas ¡sí! en un principio lo que hacía es que se regalaba una pastilla en cada cajilla y con esto garantizaban que llevase mínimo diecinueve pastillas, ya que la confiabilidad del sistema soportaba de una a dos huecos en la regleta de dosificación en la llenadora.

4ta Etapa.

Cuando se detecta un hueco en la regleta se tiene que dosificar una pastilla. Por lo cual se implementa un dosificador con dos carrileras para adicionar una o dos como máximo.



Para la dosificación de una o dos pastillas, se tiene un pequeño contenedor, con un vibrador, dos res baladilla y un accionamiento neumático por resbaladilla para la dosificación de pastillay en caso de ser necesario en el siguiente ciclo dosifica la segunda pastilla simple y cuando haya detectado para la misma caja dos huecos.

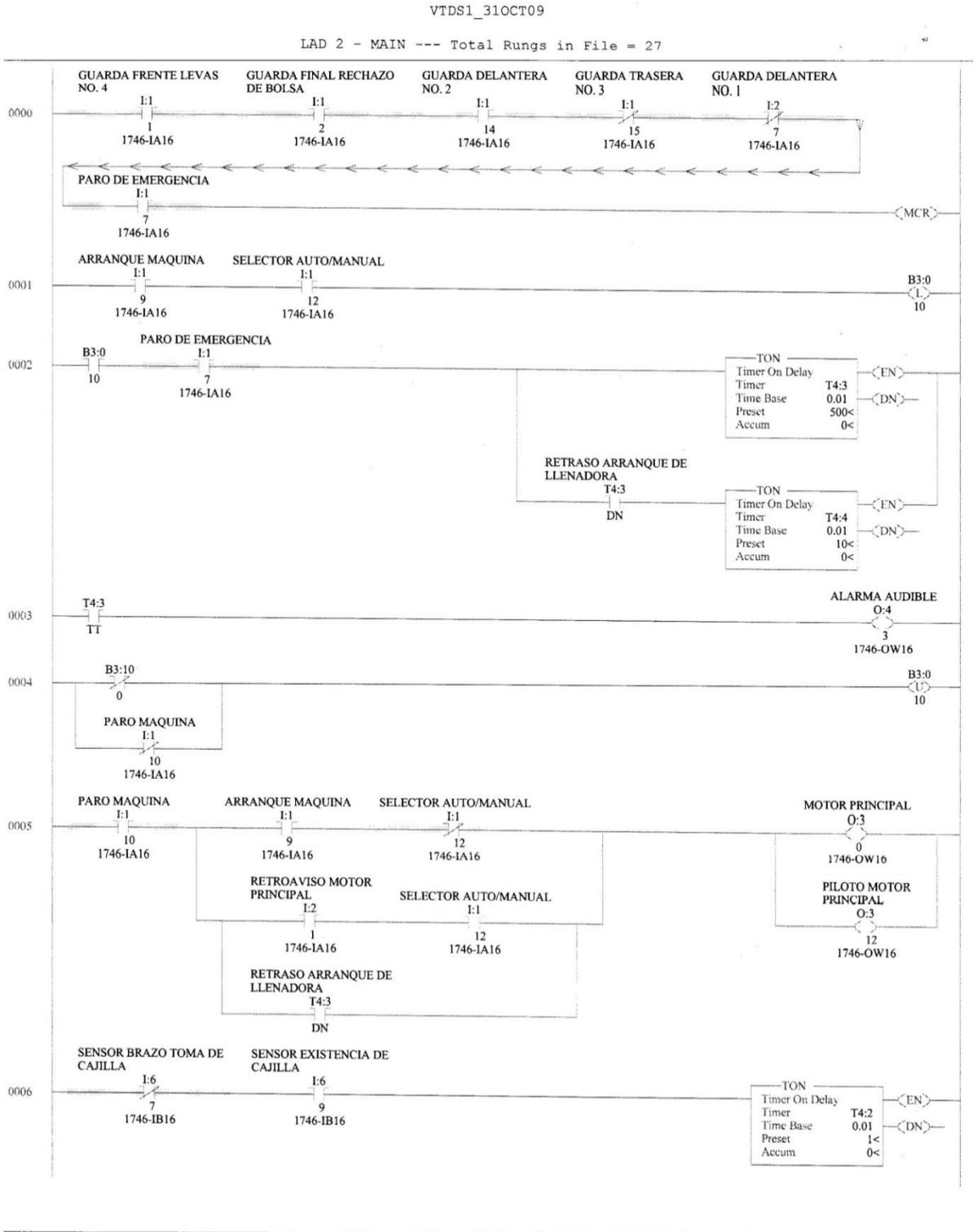
5ta Etapa.

En esta última etapa de rechazo. Se tienen dos condiciones, una cuando no detecta bolsa dentro de la cajilla y la otra si en la dosificación de una cajilla hay menos de diecinueve pastillas será expulsada. Para lo cual se implemento en sincronía un brazo mecánico, un pistón neumático y una leva mecánica. El pistón actúa como freno al brazo mecánico, es decir, cuando algunas de las condiciones de operación descritas anteriormente se da. La leva mecánica le manda la señal al PLC para que actúe el pistón, libere el brazo mecánico y rechace la cajilla.

Actualmente tiene otro dispositivo por peso para determinar el número aproximado de pastillas de acuerdo a una tolerancia en su peso la cual rechaza el producto en cajas que sirven para reciclar el producto. **Hasta antes de la medición de peso de la cajilla, se lleva a cabo el sistema de llenado.**

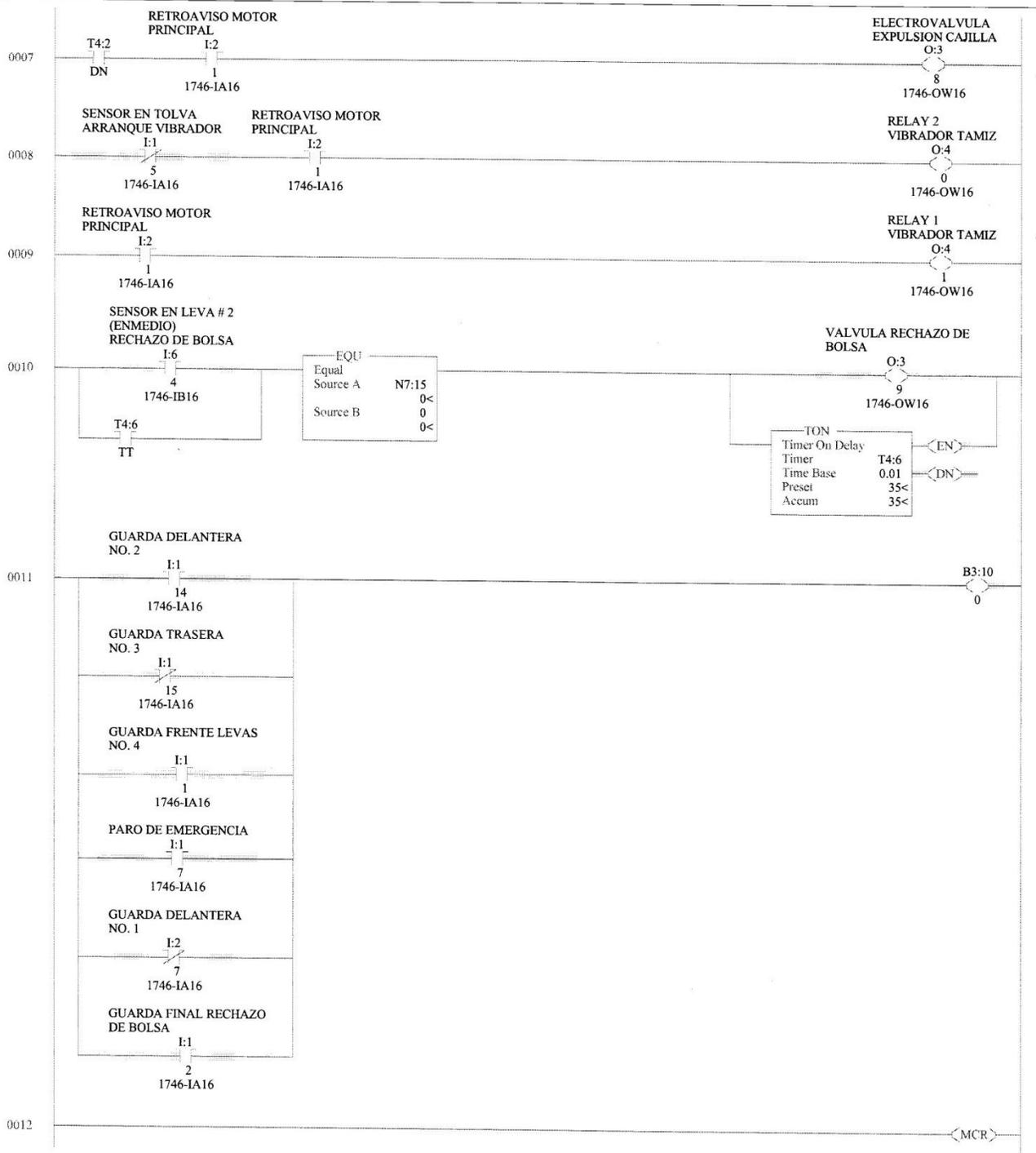
Existen otros sistemas que son colocar la envolvente a cada cajilla, pasa a otra etapa que es formar paquetes de cinco o diez cajas de pastillas, para luego pasar a otra envolvedora de paquetes y finalmente estos paquetes sean colocados en corrugados (cajas de cartón) para ser transportador como producto terminado. Casi todo el sistema es automático, sin embargo, existen etapas en la cual interviene la mano de obra de la gente que labora ahí, como es el traslape de los paquetes de cajillas a la envolvedora y el reciclaje de producto a la salida de la Llenadora cuando pasa por el checador de peso.

4.2 Programación para el control y automatización de la línea de empaque.



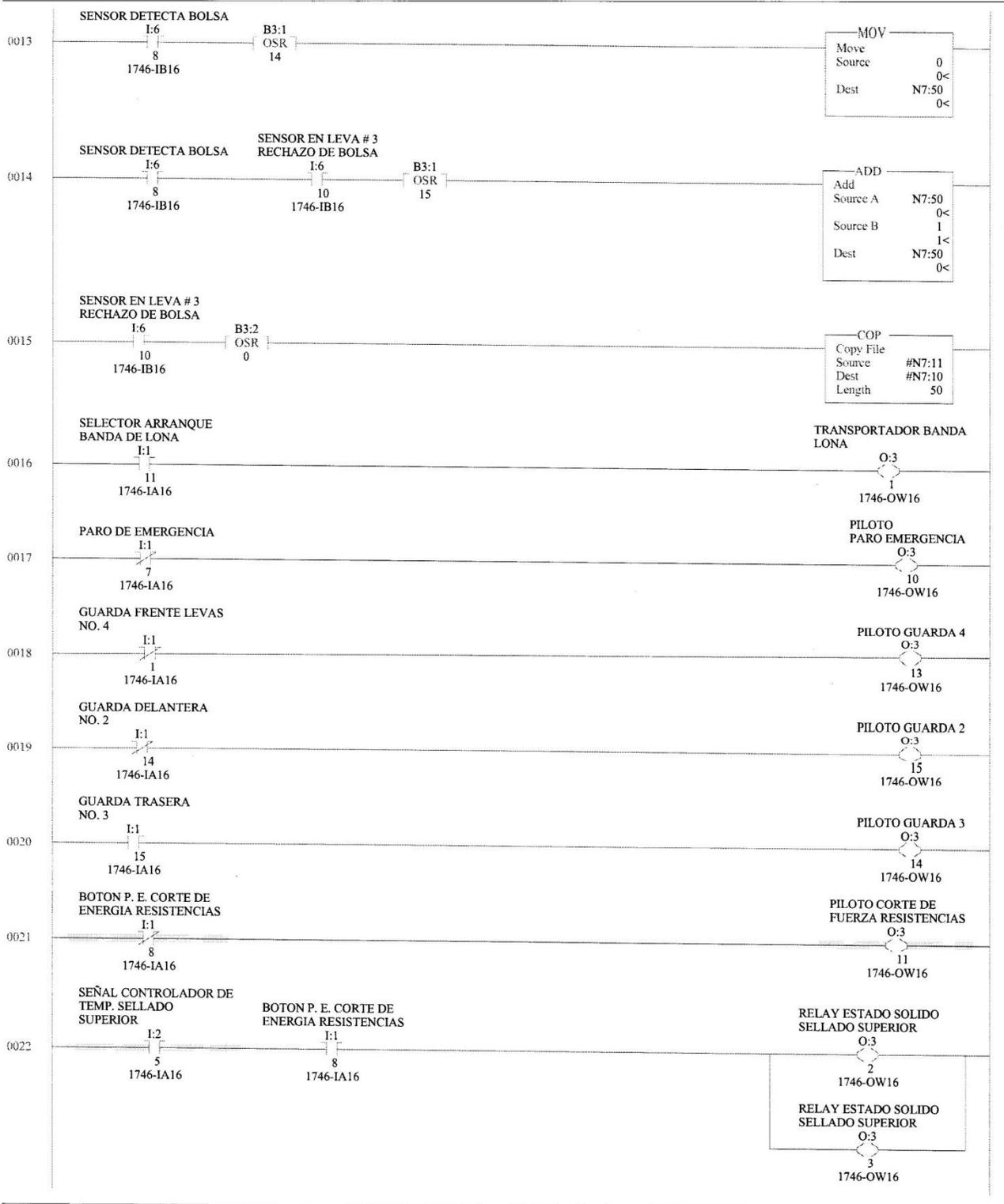
VTDS1_31OCT09

LAD 2 - MAIN --- Total Rungs in File = 27



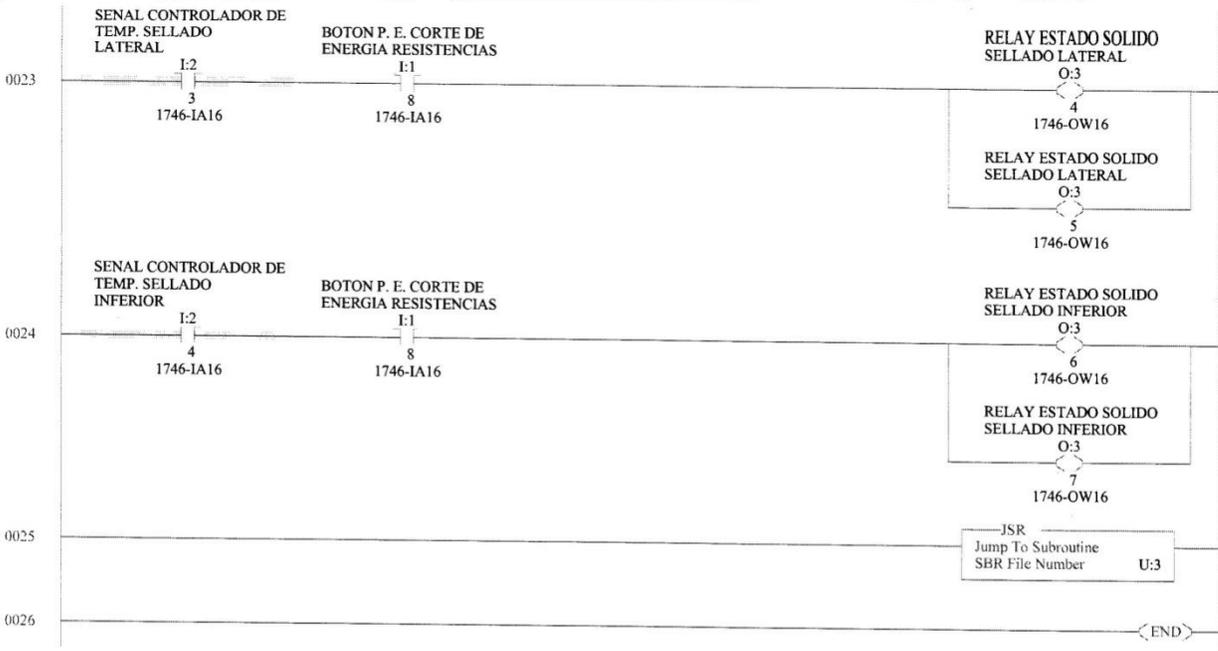
VTDS1_31OCT09

LAD 2 - MAIN --- Total Rungs in File = 27



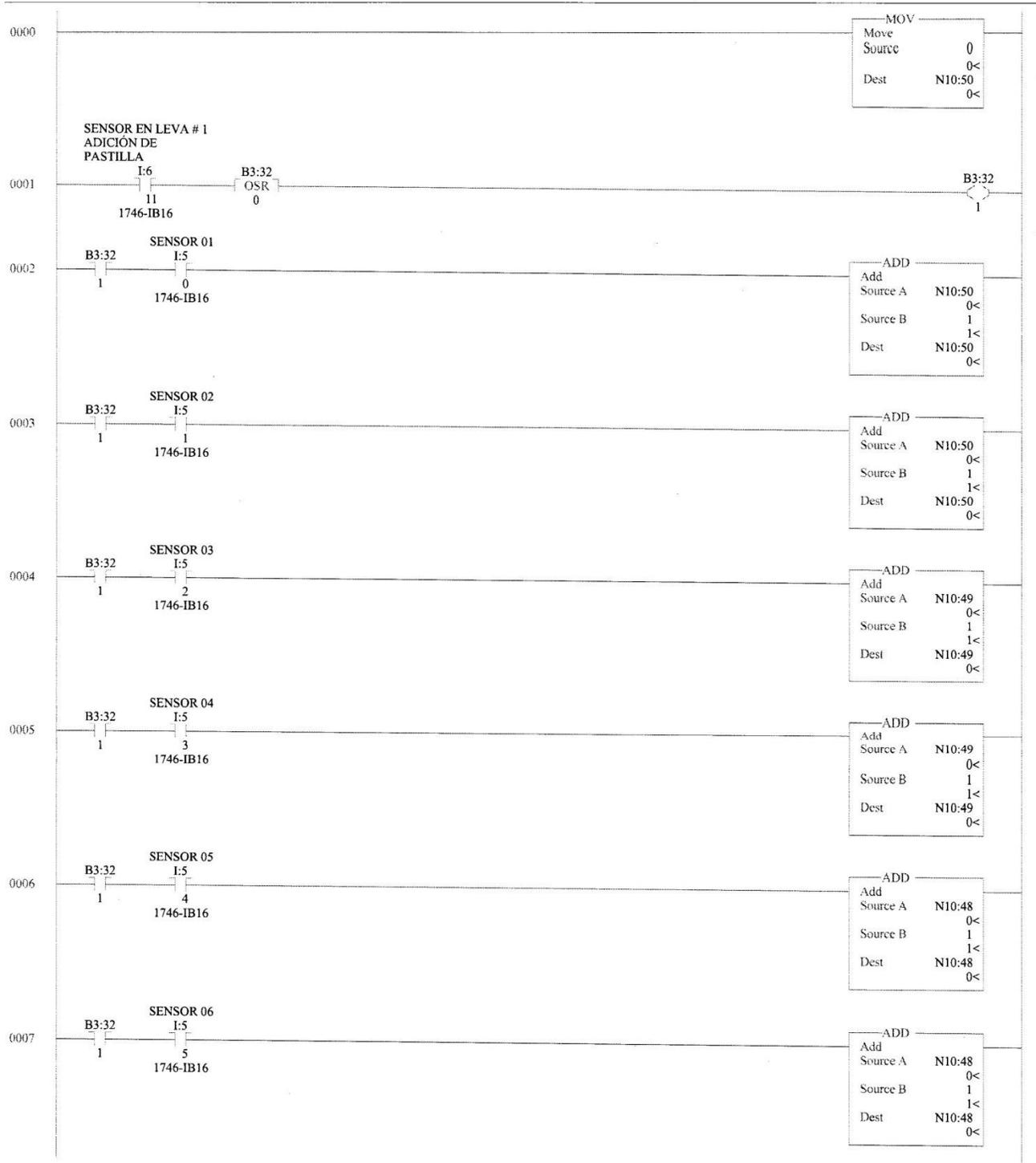
VTDS1_31OCT09

LAD 2 - MAIN --- Total Rungs in File = 27



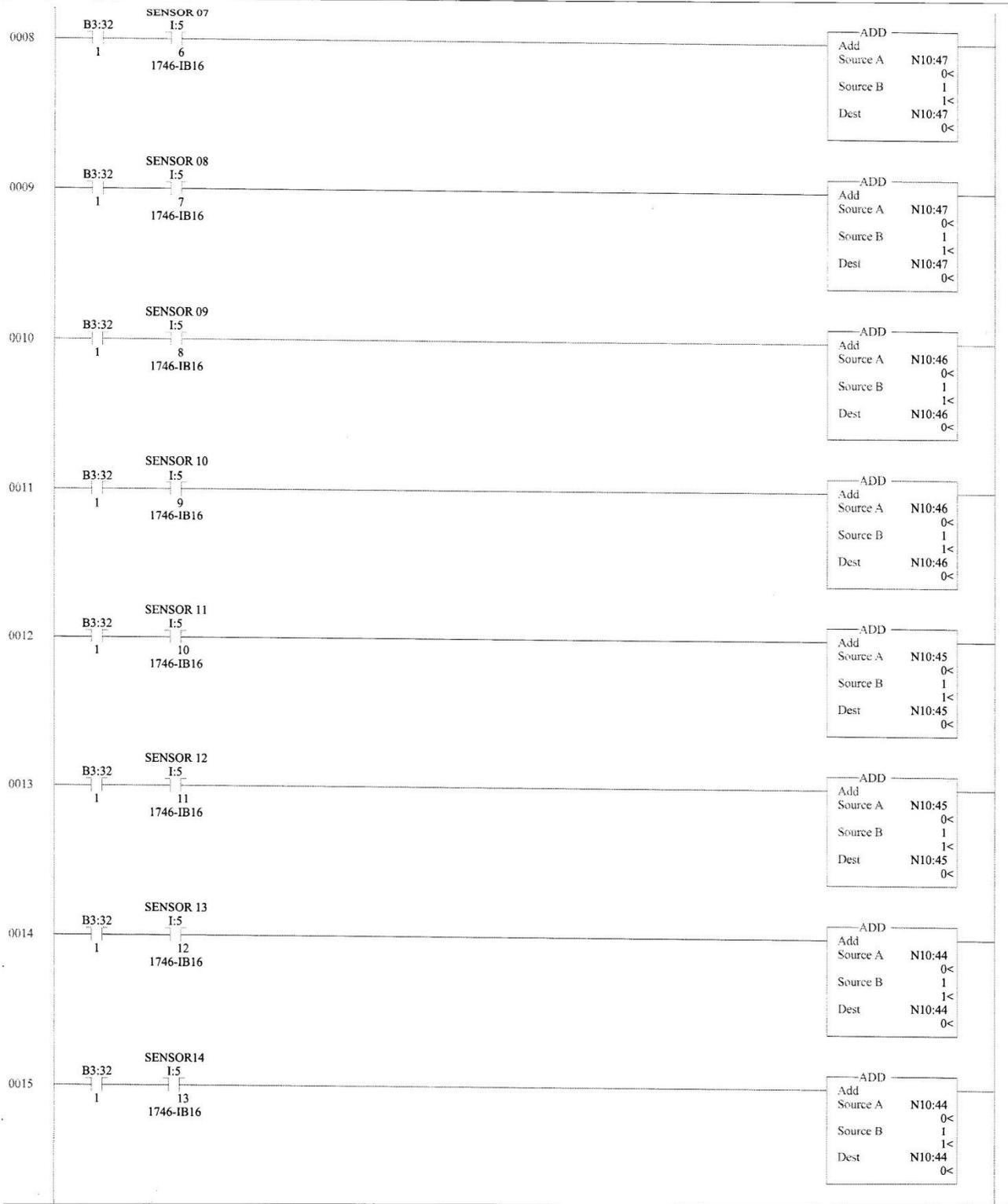
VTDS1_31OCT09

LAD 3 - PASTILLAS --- Total Rungs in File = 26



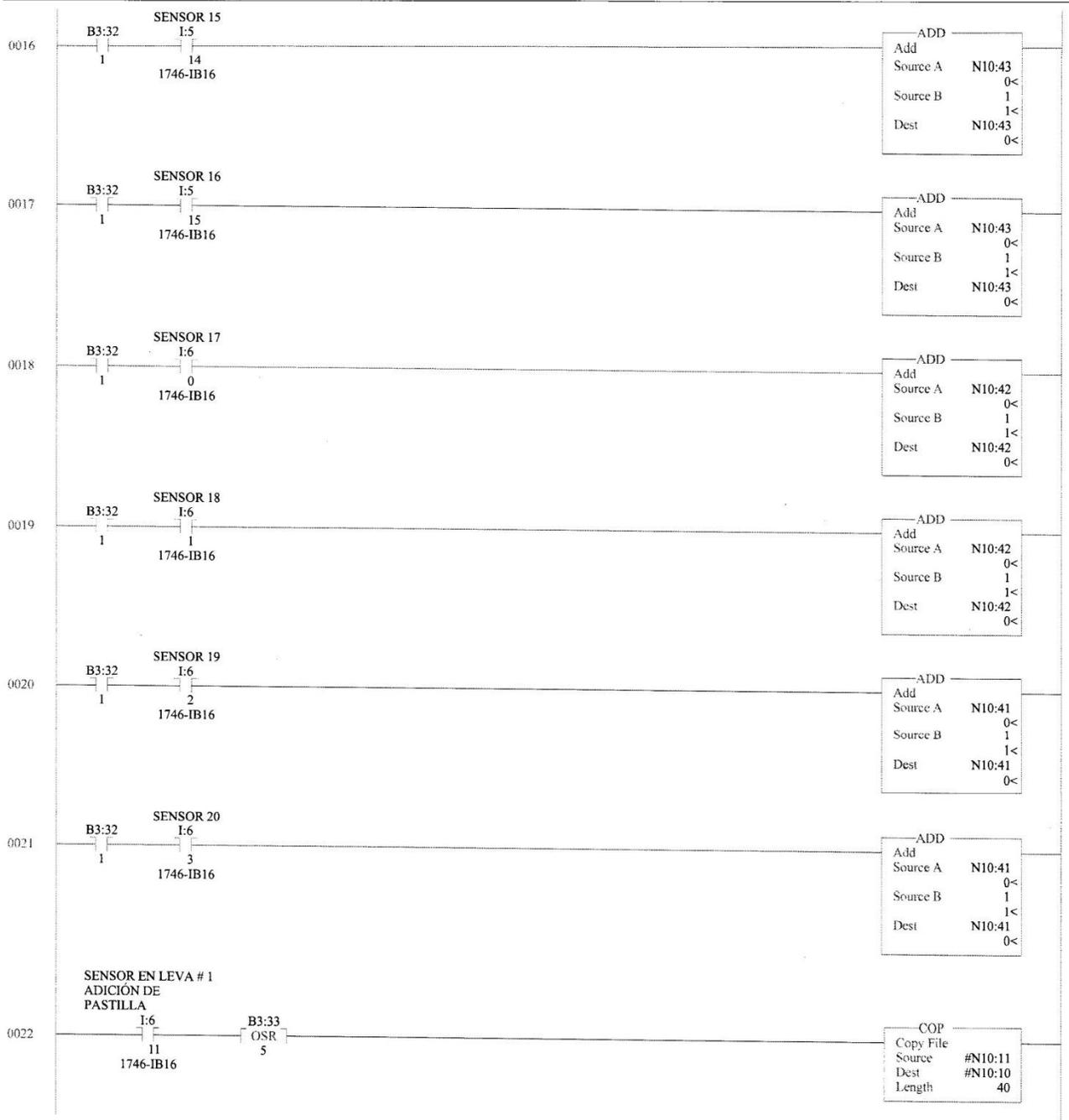
VTDS1_31OCT09

LAD 3 - PASTILLAS --- Total Rungs in File = 26



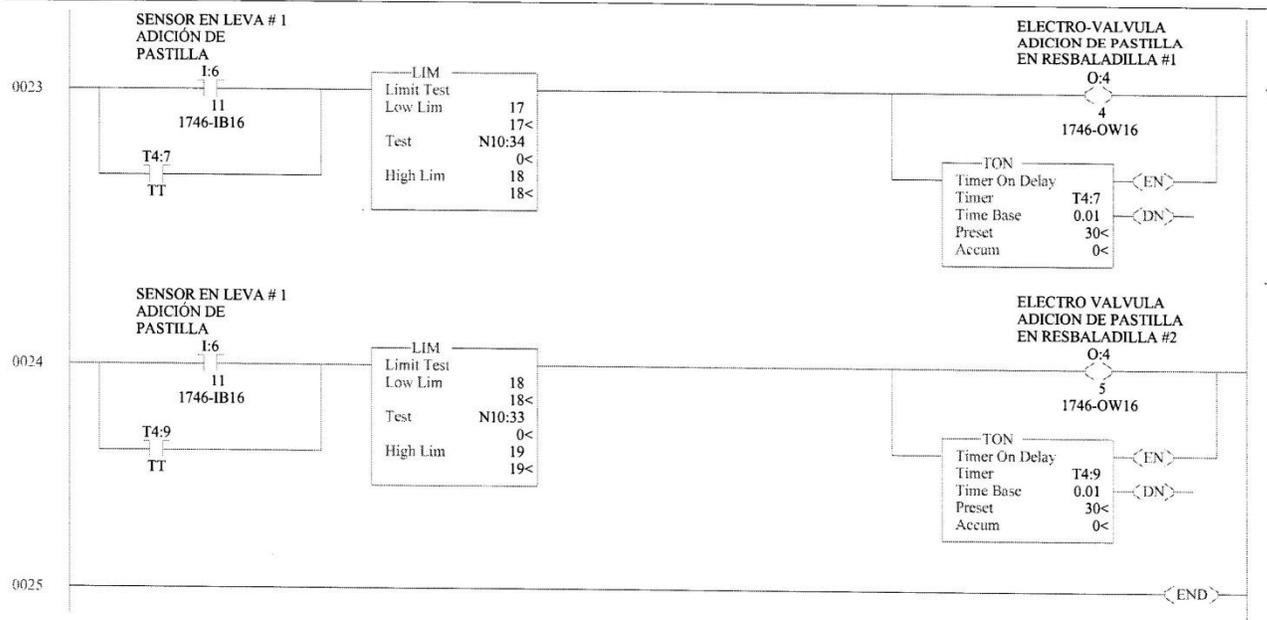
VTDS1_31OCT09

LAD 3 - PASTILLAS --- Total Rungs in File = 26



VTDS1_31OCT09

LAD 3 - PASTILLAS --- Total Rungs in File = 26



VTDS1_31OCT09

Address/Symbol Database

Address	Symbol	Scope	Description	Sym Group	Dev. Code	ABV
I:1/1			GUARDA FRENTE LEVAS NO. 4			
I:1/2			GUARDA FINAL RECHAZO DE BOLSA			
I:1/5			SENSOR EN TOLVA ARRANQUE VIBRADOR			
I:1/7			PARO DE EMERGENCIA			
I:1/8			BOTON P. E. CORTE DE ENERGIA RESISTENCIAS			
I:1/9			ARRANQUE MAQUINA			
I:1/10			PARO MAQUINA			
I:1/11			SELECTOR ARRANQUE BANDA DE LONA			
I:1/12			SELECTOR AUTO/MANUAL			
I:1/14			GUARDA DELANTERA NO. 2			
I:1/15			GUARDA TRASERA NO. 3			
I:2/1			RETROAVISO MOTOR PRINCIPAL			
I:2/3			SEÑAL CONTROLADOR DE TEMP. SELLADO LATERAL			
I:2/4			SEÑAL CONTROLADOR DE TEMP. SELLADO INFERIOR			
I:2/5			SEÑAL CONTROLADOR DE TEMP. SELLADO SUPERIOR			
I:2/6			???????			
I:2/7			GUARDA DELANTERA NO. 1			
I:5/0			SENSOR 01			
I:5/1			SENSOR 02			
I:5/2			SENSOR 03			
I:5/3			SENSOR 04			
I:5/4			SENSOR 05			
I:5/5			SENSOR 06			
I:5/6			SENSOR 07			
I:5/7			SENSOR 08			
I:5/8			SENSOR 09			
I:5/9			SENSOR 10			
I:5/10			SENSOR 11			
I:5/11			SENSOR 12			
I:5/12			SENSOR 13			
I:5/13			SENSOR14			
I:5/14			SENSOR 15			
I:5/15			SENSOR 16			
I:6/0			SENSOR 17			
I:6/1			SENSOR 18			
I:6/2			SENSOR 19			
I:6/3			SENSOR 20			
I:6/4			SENSOR EN LEVA # 2 (ENMEDIO) RECHAZO DE BOLSA			
I:6/7			SENSOR BRAZO TOMA DE CAJILLA			
I:6/8			SENSOR DETECTA BOLSA			
I:6/9			SENSOR EXISTENCIA DE CAJILLA			
I:6/10			SENSOR EN LEVA # 3 RECHAZO DE BOLSA			
I:6/11			SENSOR EN LEVA # 1 ADICIÓN DE PASTILLA			
O:3/0			MOTOR PRINCIPAL			
O:3/1			TRANSPORTADOR BANDA LONA			
O:3/2			RELAY ESTADO SOLIDO SELLADO SUPERIOR			
O:3/3			RELAY ESTADO SOLIDO SELLADO SUPERIOR			
O:3/4			RELAY ESTADO SOLIDO SELLADO LATERAL			
O:3/5			RELAY ESTADO SOLIDO SELLADO LATERAL			
O:3/6			RELAY ESTADO SOLIDO SELLADO INFERIOR			
O:3/7			RELAY ESTADO SOLIDO SELLADO INFERIOR			
O:3/8			ELECTROVALVULA EXPULSION CAJILLA			
O:3/9			VALVULA RECHAZO DE BOLSA			
O:3/10			PILOTO PARO EMERGENCIA			
O:3/11			PILOTO CORTE DE FUERZA RESISTENCIAS			
O:3/12			PILOTO MOTOR PRINCIPAL			
O:3/13			PILOTO GUARDA 4			
O:3/14			PILOTO GUARDA 3			
O:3/15			PILOTO GUARDA 2			
O:4/0			RELAY 2 VIBRADOR TAMIZ			
O:4/1			RELAY 1 VIBRADOR TAMIZ			
O:4/2						
O:4/3			ALARMA AUDIBLE			
O:4/4			ELECTRO-VALVULA ADICION DE PASTILLA EN RESBALADILLA #1			
O:4/5			ELECTRO VALVULA ADICION DE PASTILLA EN RESBALADILLA #2			
O:4/6			???????			
S:0			Arithmetic Flags			
S:0/0			Processor Arithmetic Carry Flag			
S:0/1			Processor Arithmetic Underflow/ Overflow Flag			
S:0/2			Processor Arithmetic Zero Flag			
S:0/3			Processor Arithmetic Sign Flag			
S:1			Processor Mode Status/ Control			
S:1/0			Processor Mode Bit 0			
S:1/1			Processor Mode Bit 1			
S:1/2			Processor Mode Bit 2			
S:1/3			Processor Mode Bit 3			
S:1/4			Processor Mode Bit 4			
S:1/5			Forces Enabled			
S:1/6			Forces Present			
S:1/7			Comms Active			
S:1/8			Fault Override at Powerup			
S:1/9			Startup Protection Fault			
S:1/10			Load Memory Module on Memory Error			
S:1/11			Load Memory Module Always			
S:1/12			Load Memory Module and RUN			

VTDS1_31OCT09

Address/Symbol Database

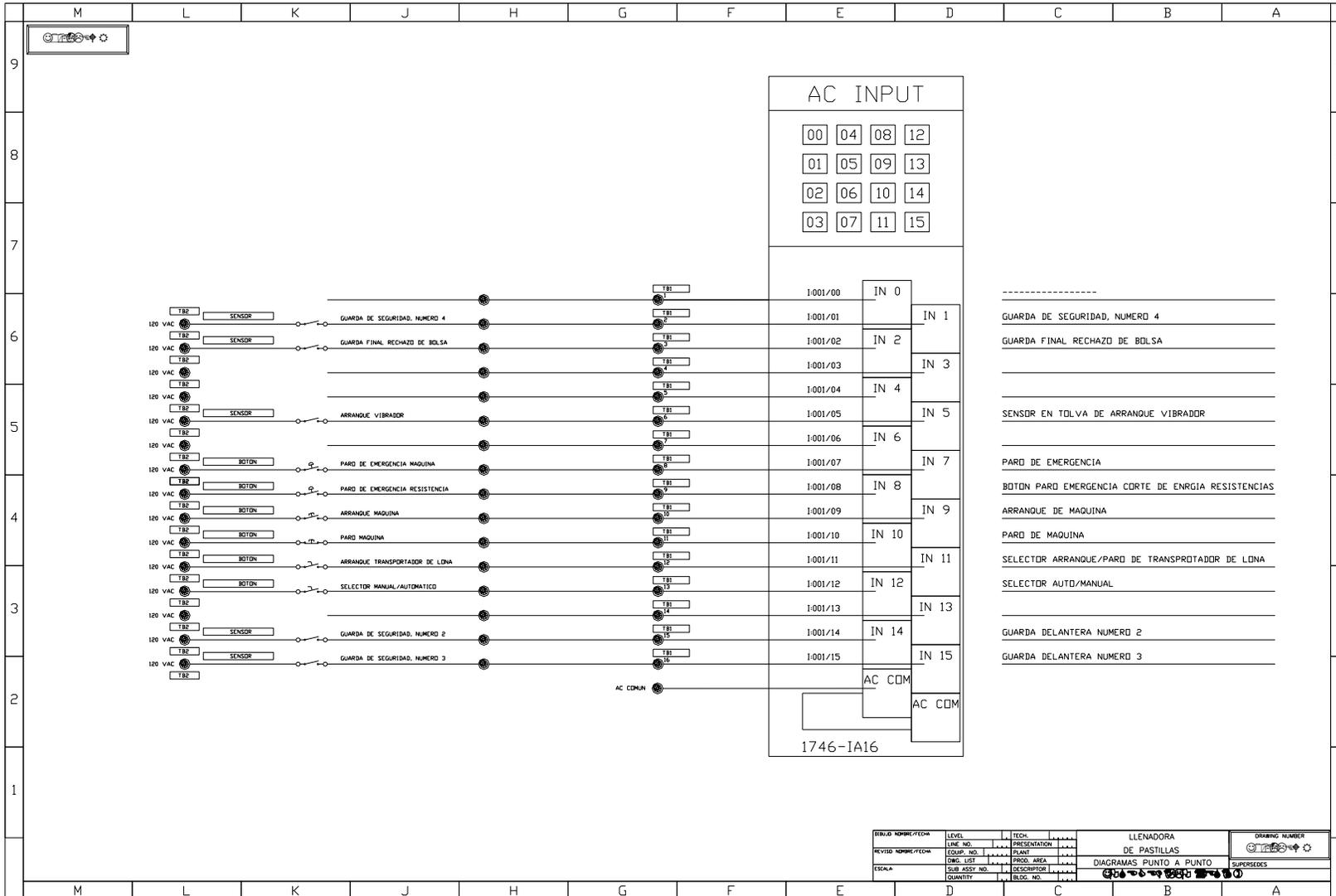
Address	Symbol	Scope	Description	Sym Group	Dev. Code	ABV
S:1/13			Major Error Halted			
S:1/14			Access Denied			
S:1/15			First Pass			
S:2/0			STI Pending			
S:2/1			STI Enabled			
S:2/2			STI Executing			
S:2/3			Index Addressing File Range			
S:2/4			Saved with Debug Single Step			
S:2/5			DH-485 Incoming Command Pending			
S:2/6			DH-485 Message Reply Pending			
S:2/7			DH-485 Outgoing Message Command Pending			
S:2/15			Comms Servicing Selection			
S:3			Current Scan Time/ Watchdog Scan Time			
S:4			Time Base			
S:5/0			Overflow Trap			
S:5/2			Control Register Error			
S:5/3			Major Err Detected Executing UserFault Routine			
S:5/4			M0-M1 Referenced on Disabled Slot			
S:5/8			Memory Module Boot			
S:5/9			Memory Module Password Mismatch			
S:5/10			STI Overflow			
S:5/11			Battery Low			
S:6			Major Error Fault Code			
S:7			Suspend Code			
S:8			Suspend File			
S:9			Active Nodes			
S:10			Active Nodes			
S:11			I/O Slot Enables			
S:12			I/O Slot Enables			
S:13			Math Register			
S:14			Math Register			
S:15			Node Address/ Baud Rate			
S:16			Debug Single Step Rung			
S:17			Debug Single Step File			
S:18			Debug Single Step Breakpoint Rung			
S:19			Debug Single Step Breakpoint File			
S:20			Debug Fault/ Powerdown Rung			
S:21			Debug Fault/ Powerdown File			
S:22			Maximum Observed Scan Time			
S:23			Average Scan Time			
S:24			Index Register			
S:25			I/O Interrupt Pending			
S:26			I/O Interrupt Pending			
S:27			I/O Interrupt Enabled			
S:28			I/O Interrupt Enabled			
S:29			User Fault Routine File Number			
S:30			STI Setpoint			
S:31			STI File Number			
S:32			I/O Interrupt Executing			
S:33			Extended Proc Status Control Word			
S:33/0			Incoming Command Pending			
S:33/1			Message Reply Pending			
S:33/2			Outgoing Message Command Pending			
S:33/3			Selection Status User/DF1			
S:33/4			Communicat Active			
S:33/5			Communicat Servicing Selection			
S:33/6			Message Servicing Selection Channel 0			
S:33/7			Message Servicing Selection Channel 1			
S:33/8			Interrupt Latency Control Flag			
S:33/9			Scan Toggle Flag			
S:33/10			Discrete Input Interrupt Reconfigur Flag			
S:33/11			Online Edit Status			
S:33/12			Online Edit Status			
S:33/13			Scan Time Timebase Selection			
S:33/14			DTR Control Bit			
S:33/15			DTR Force Bit			
S:34			Pass-thru Disabled			
S:34/0			Pass-Thru Disabled Flag			
S:34/1			DH+ Active Node Table Enable Flag			
S:34/2			Floating Point Math Flag			
S:35			Last 1 ms Scan Time			
S:36			Extended Minor Error Bits			
S:36/8			D11 Lost			
S:36/9			STI Lost			
S:36/10			Memory Module Data File Overwrite Protection			
S:37			Clock Calendar Year			
S:38			Clock Calendar Month			
S:39			Clock Calendar Day			
S:40			Clock Calendar Hours			
S:41			Clock Calendar Minutes			
S:42			Clock Calendar Seconds			
S:43			STI Interrupt Time			
S:44			I/O Event Interrupt Time			
S:45			D11 Interrupt Time			
S:46			Discrete Input Interrupt- File Number			

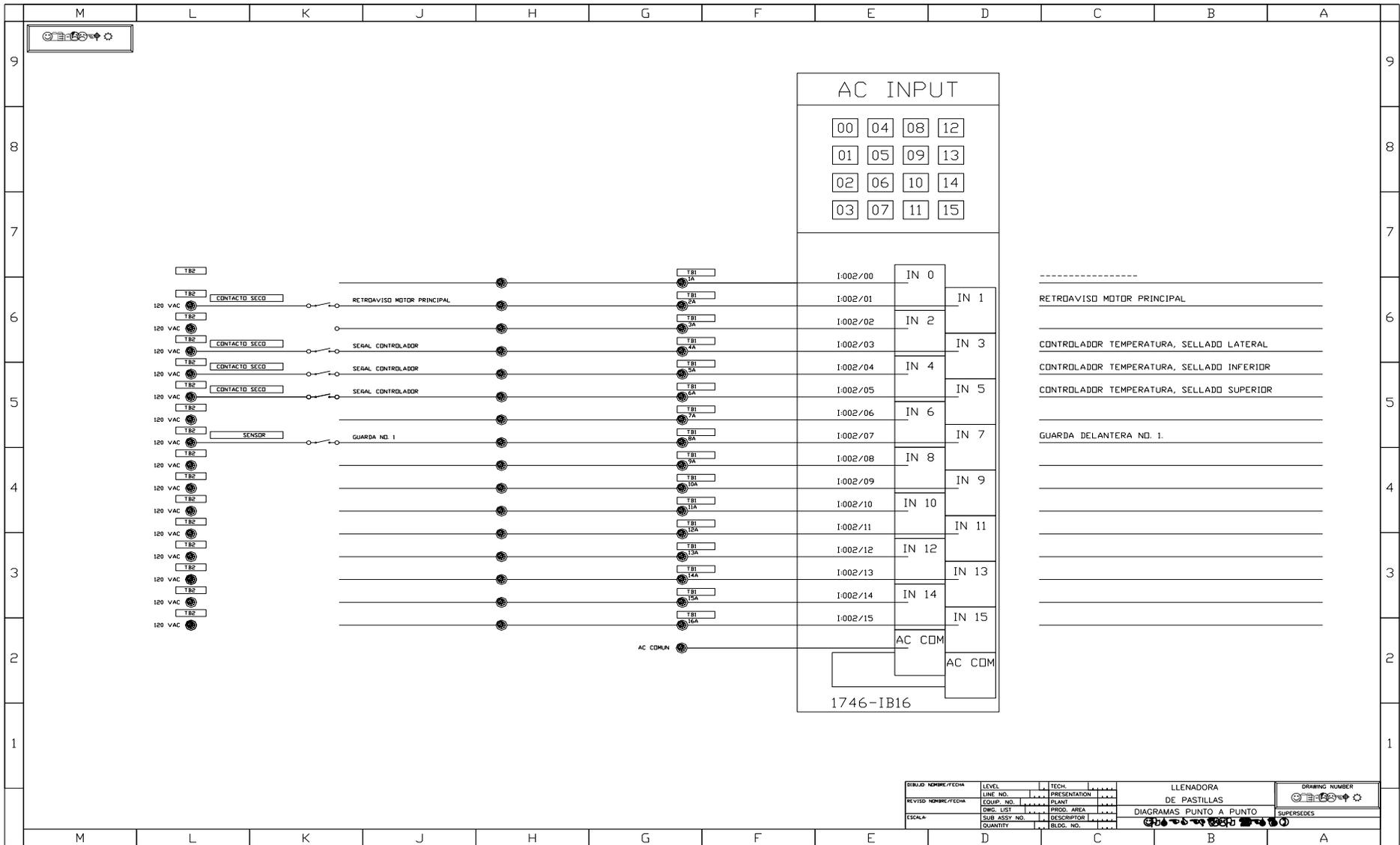
VTDS1_31OCT09

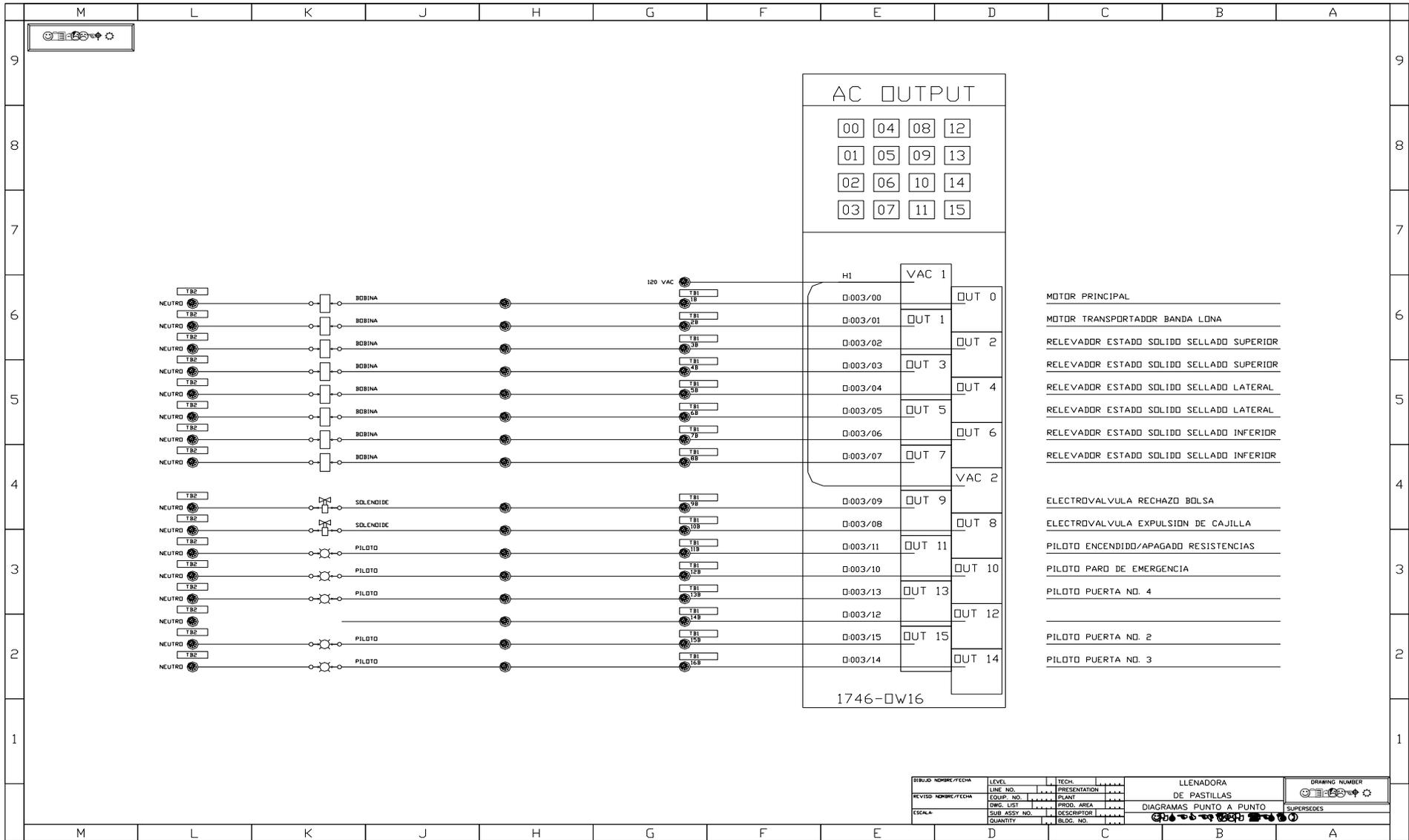
Address/Symbol Database

Address	Symbol	Scope	Description	Sym Group	Dev. Code	ABV
S:47			Discrete Input Interrupt- Slot Number			
S:48			Discrete Input Interrupt- Bit Mask			
S:49			Discrete Input Interrupt- Compare Value			
S:50			Processor Catalog Interrupt- Preset			
S:51			Discrete Input Interrupt- Return Number			
S:52			Discrete Input Interrupt- Accumulat			
S:53			Discrete Input Interrupt- Timer			
S:54			Discrete Input Interrupt- Timer			
S:55			Last Dll Scan Time			
S:56			Maximum Observed Dll Scan Time			
S:57			Operating System Catalog Number			
S:58			Operating System Series			
S:59			Operating System FRN			
S:61			Processor Series			
S:62			Processor Revision			
S:63			User Program Type			
S:64			User Program Functional Index			
S:65			User RAM Size			
S:66			Flash EEPROM Size			
S:67			Channel 0 Active Nodes			
S:68			Channel 0 Active Nodes			
S:69			Channel 0 Active Nodes			
S:70			Channel 0 Active Nodes			
S:71			Channel 0 Active Nodes			
S:72			Channel 0 Active Nodes			
S:73			Channel 0 Active Nodes			
S:74			Channel 0 Active Nodes			
S:75			Channel 0 Active Nodes			
S:76			Channel 0 Active Nodes			
S:77			Channel 0 Active Nodes			
S:78			Channel 0 Active Nodes			
S:79			Channel 0 Active Nodes			
S:80			Channel 0 Active Nodes			
S:81			Channel 0 Active Nodes			
S:82			Channel 0 Active Nodes			
S:83			DH+ Active Nodes			
S:84			DH+ Active Nodes			
S:85			DH+ Active Nodes			
S:86			DH+ Active Nodes			
T4:3/DN			RETRASO ARRANQUE DE LLENADORA			

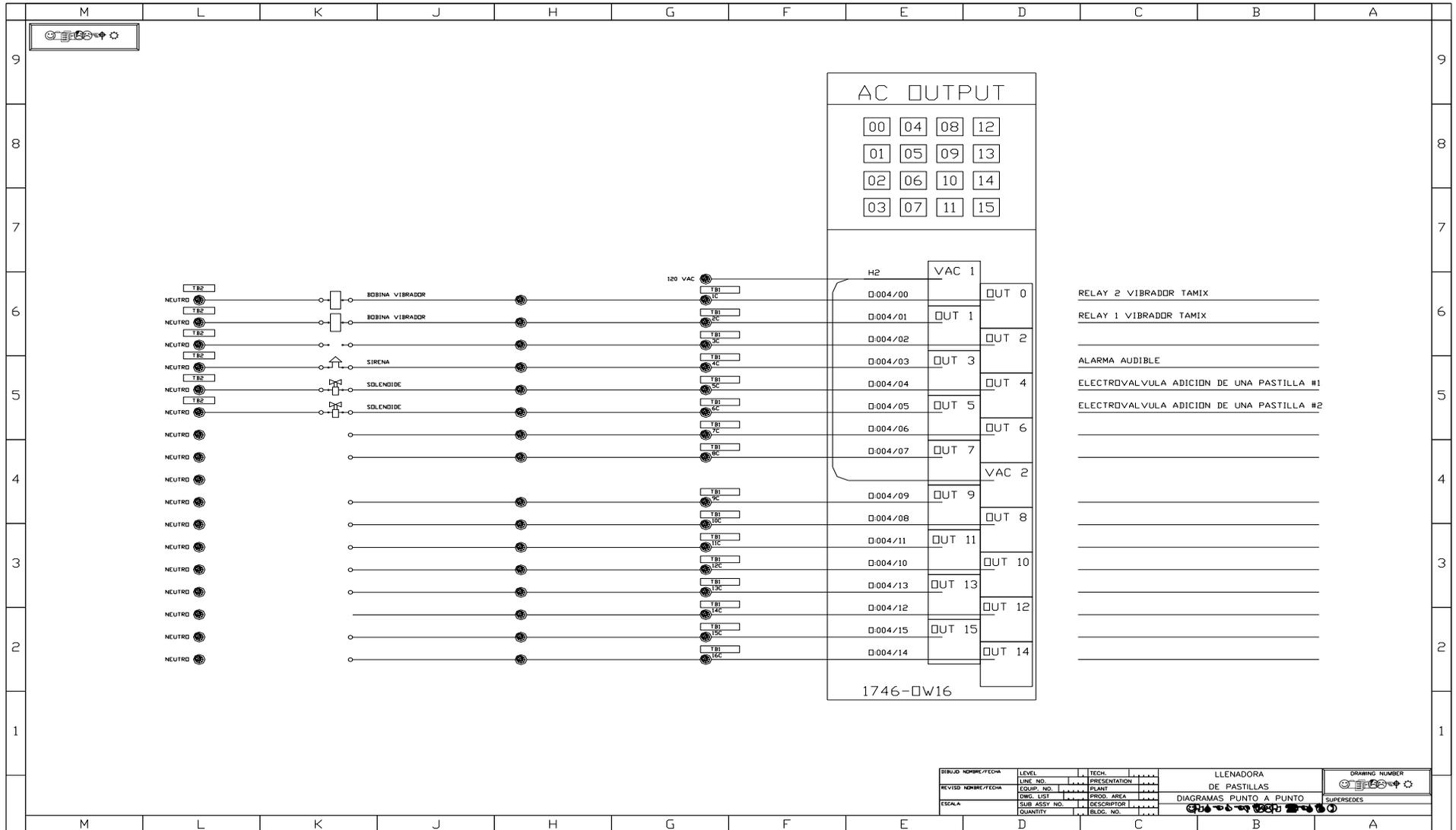
4.3. Diagramas de conexión punto a punto

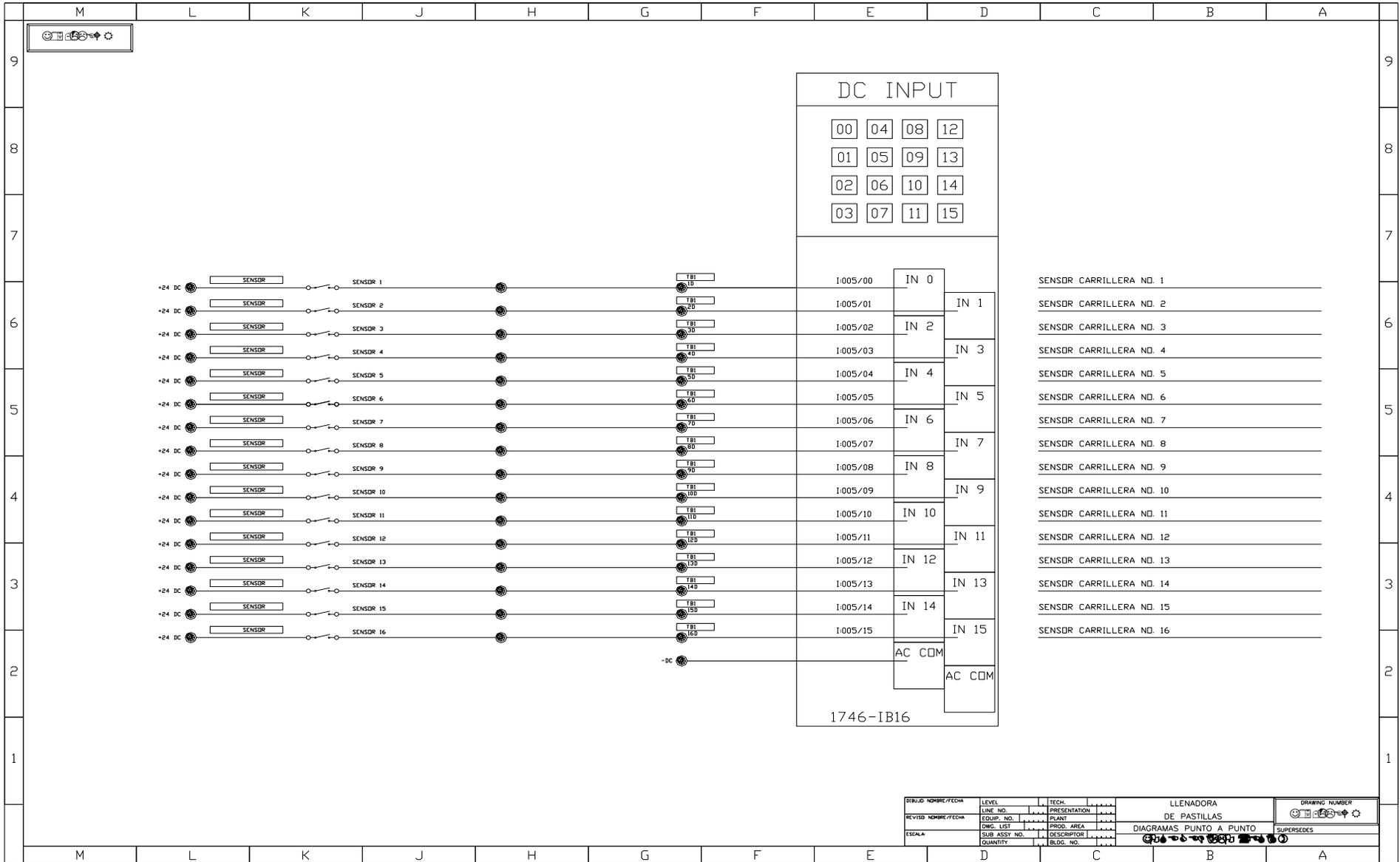




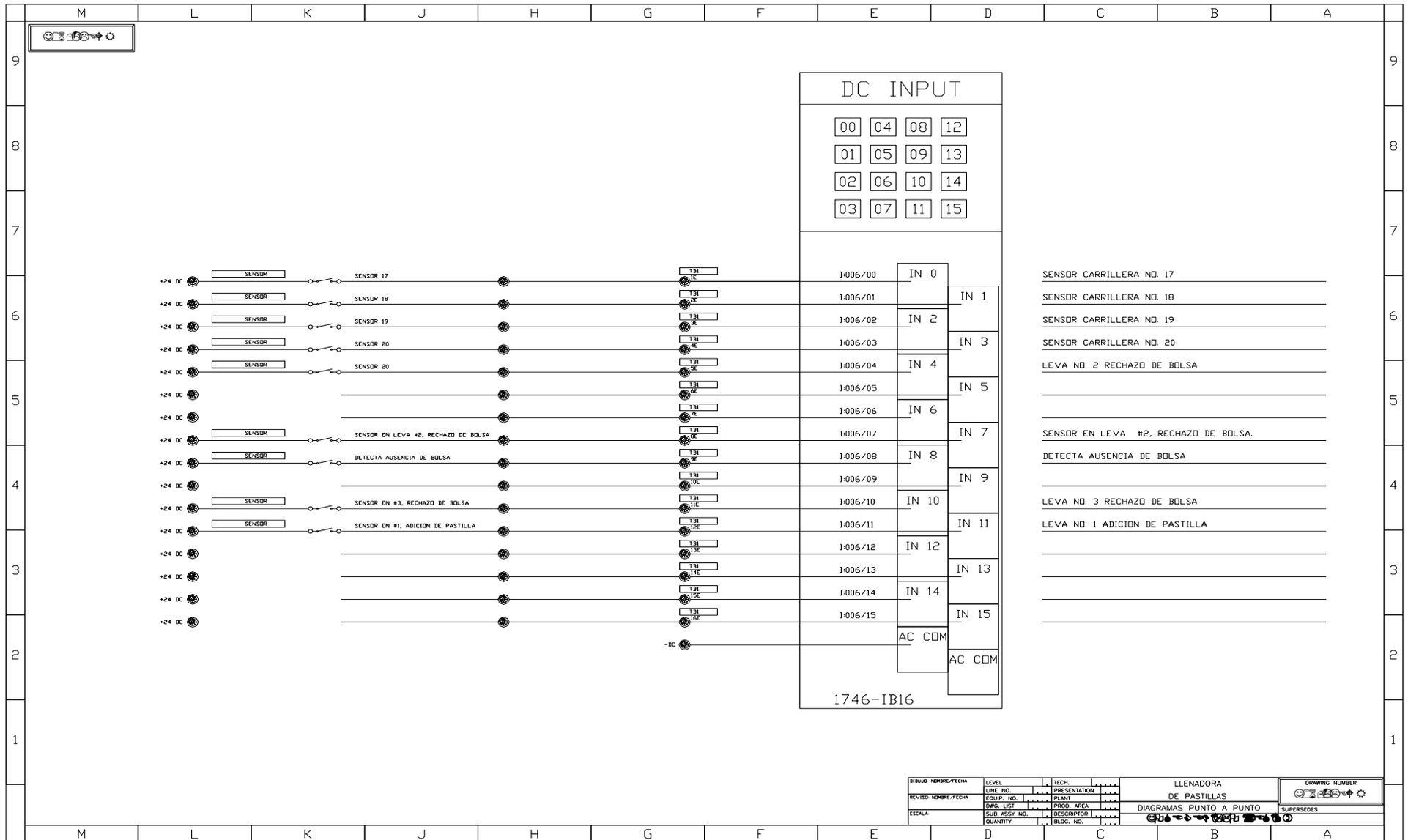


REVISOR	FECHA	TECH.	LLENADORA	DRAWING NUMBER
REVISOR	FECHA	TECH.	DE PASTILLAS	
ESCALA	QUANTITY	BLOG. NO.	DIAGRAMAS PUNTO A PUNTO	SUPERSEDES

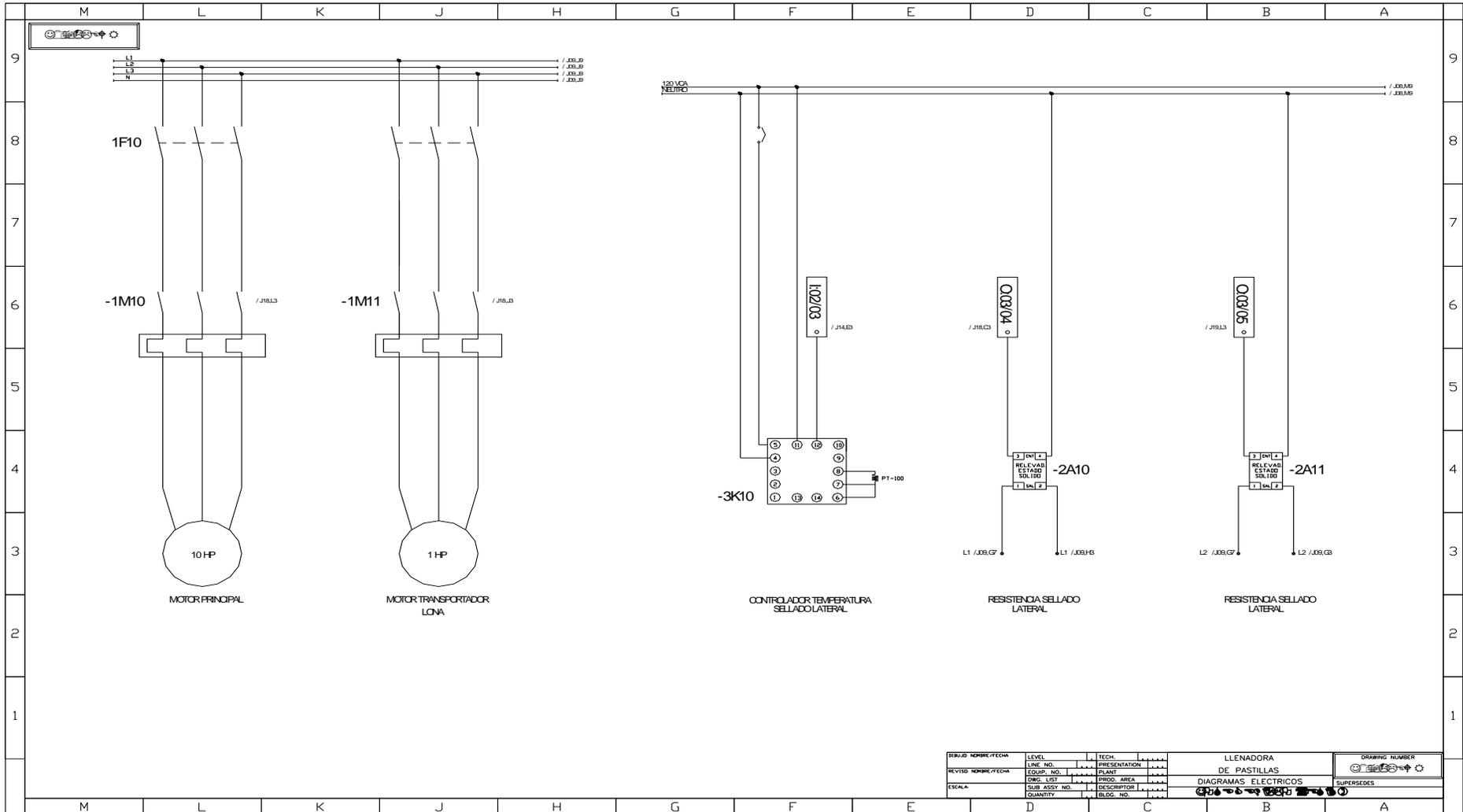


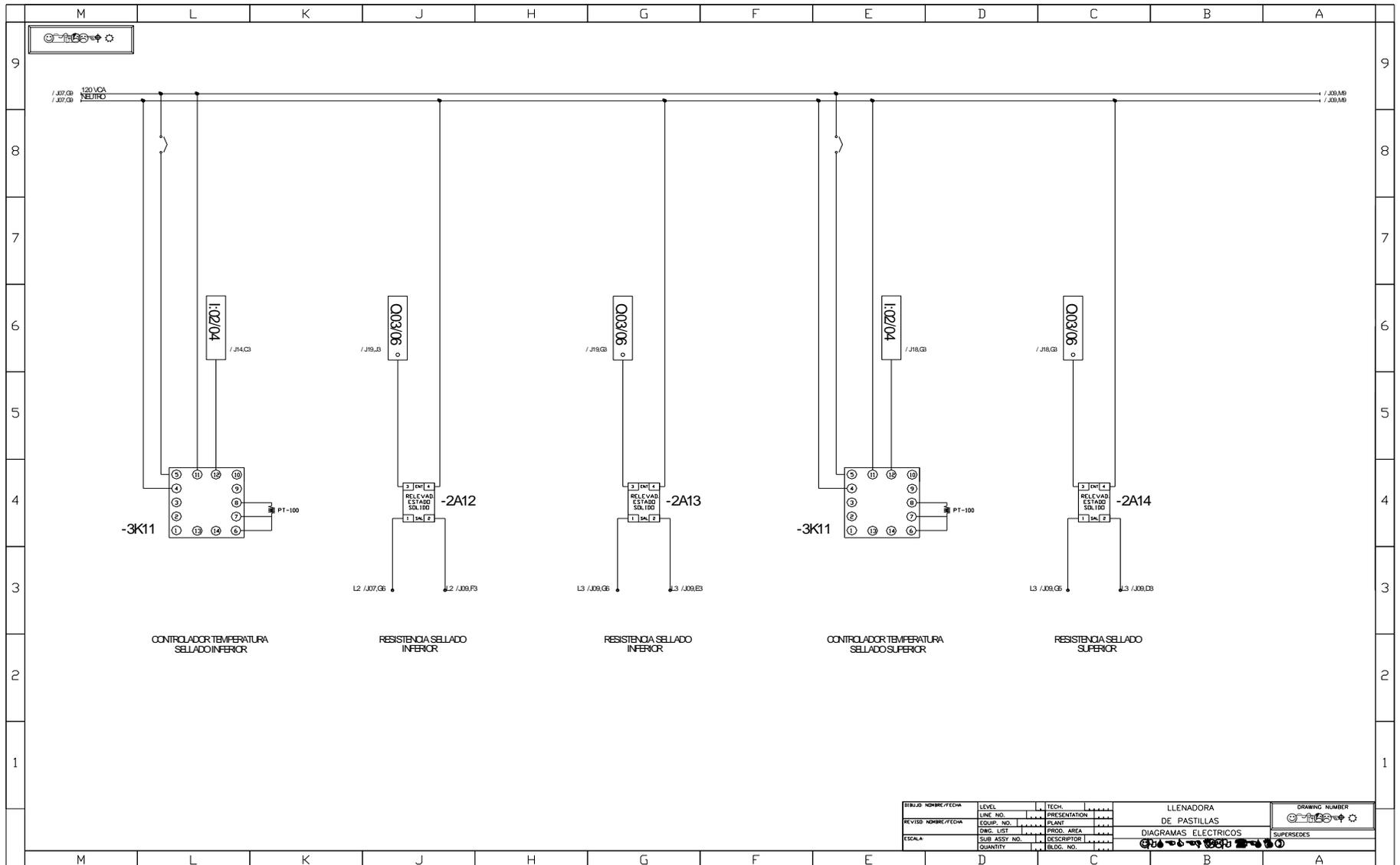


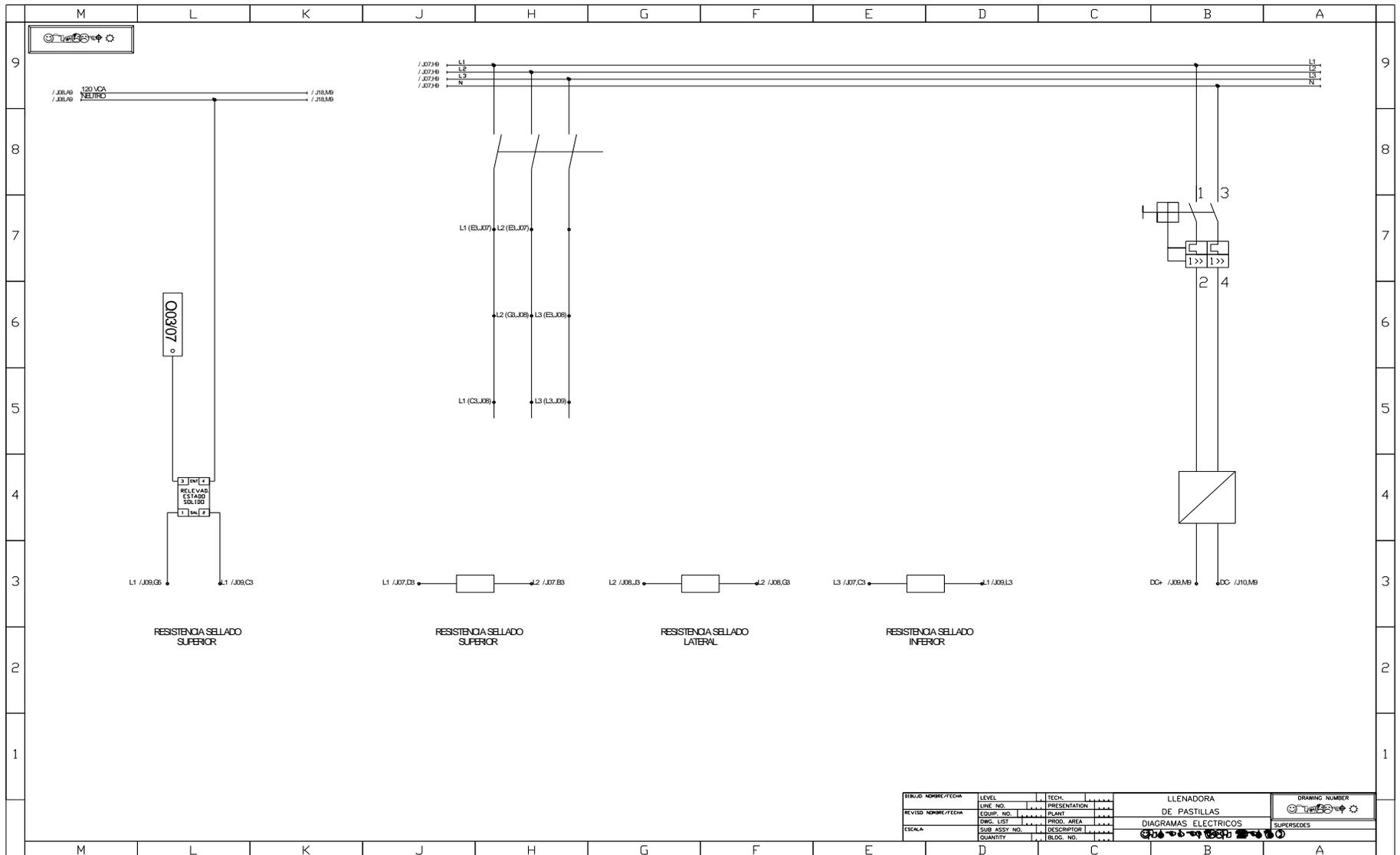
SERVIDOR/FECHA	LEVEL	TECH.	LLENADORA	DRAWING NUMBER
REVISO	LINE NO.	PRESENTATION	DE PASTILLAS	DIAGRAMAS PUNTO A PUNTO
Escala	EQUIP. NO.	PLANT	PROD. AREA	SUPERSEDES
	SUB ASSY NO.	DESCRIPTOR		
	QUANTITY	BLOC. NO.		



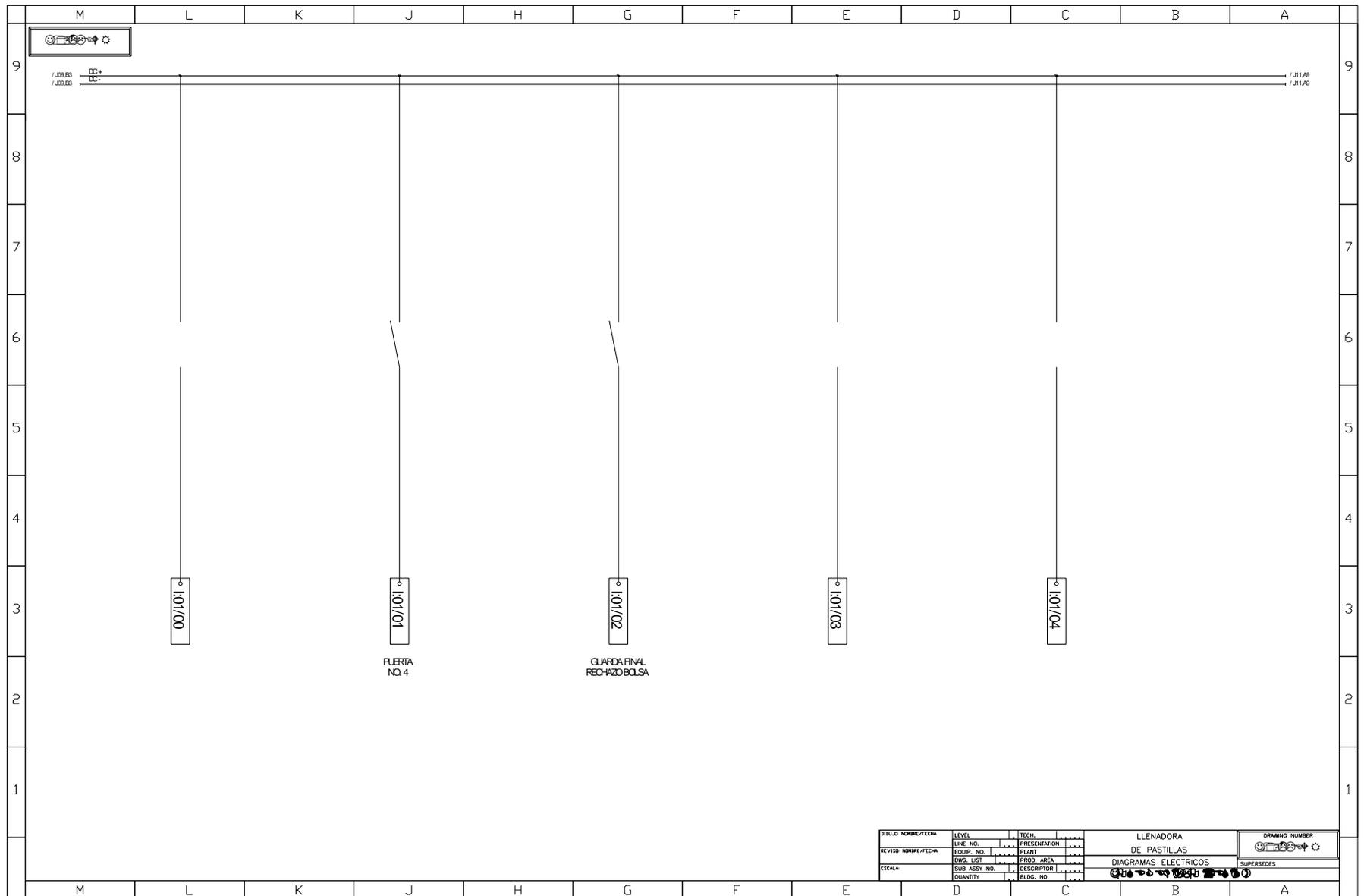
4.4. Diagramas de conexión punto a punto

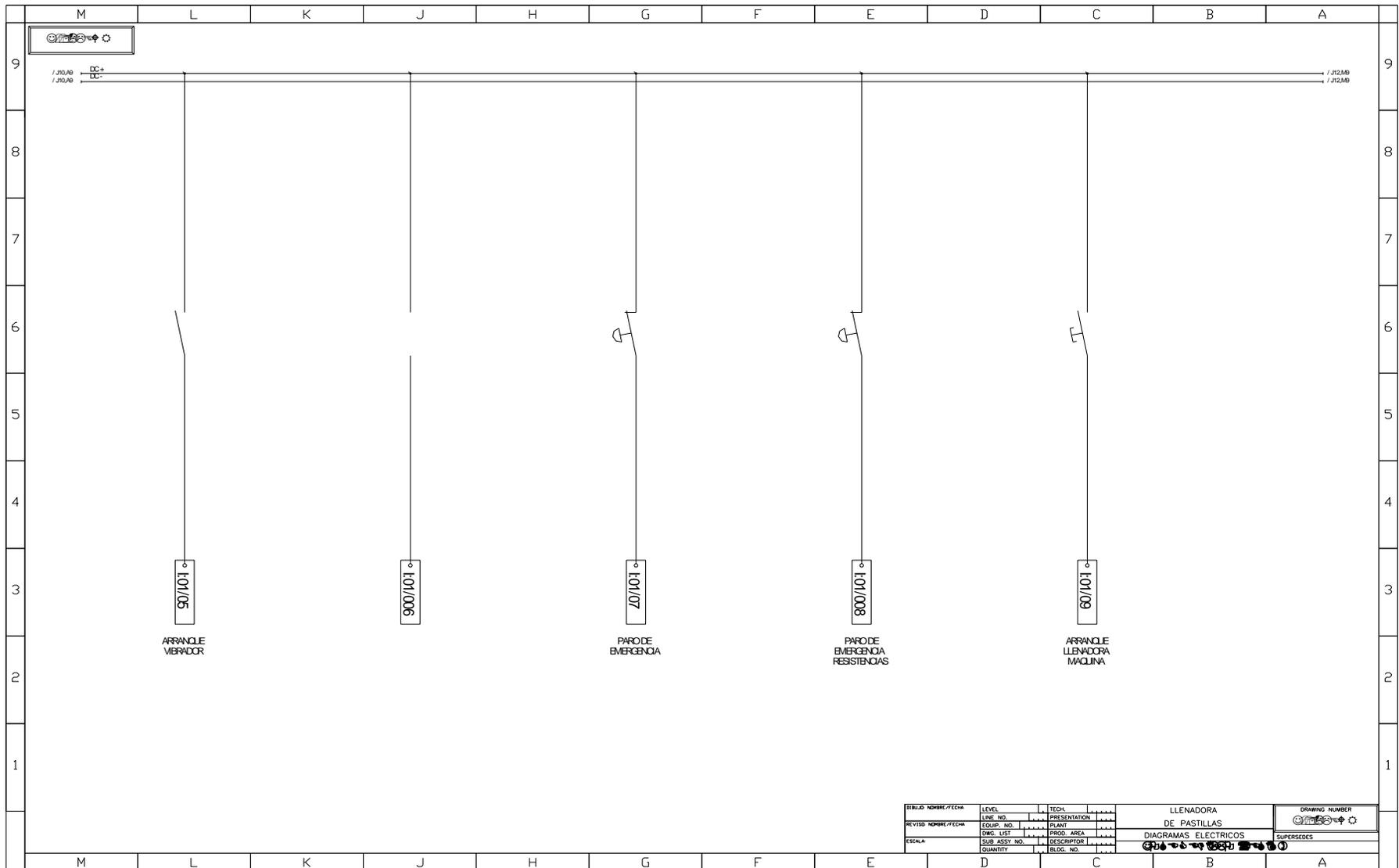


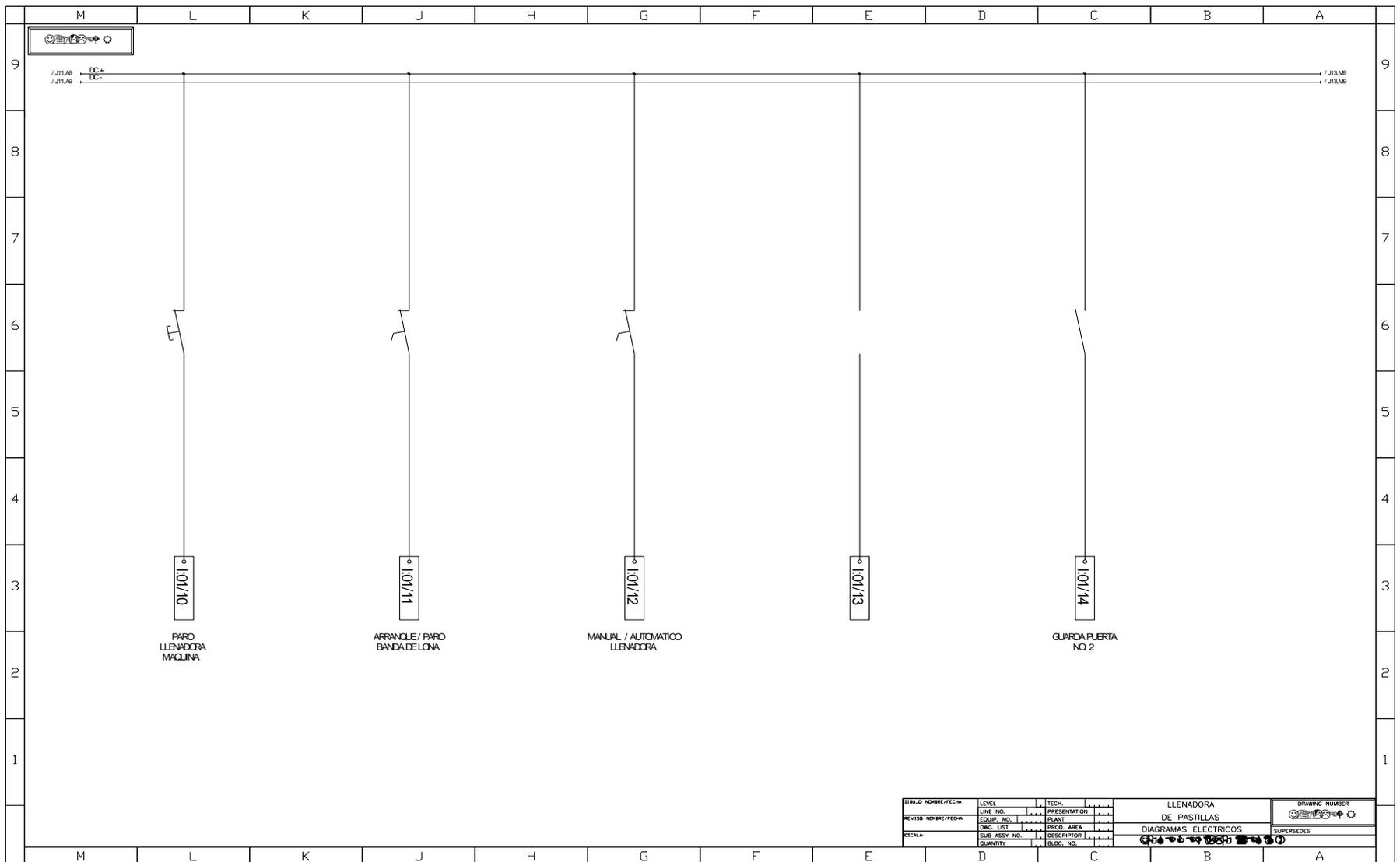


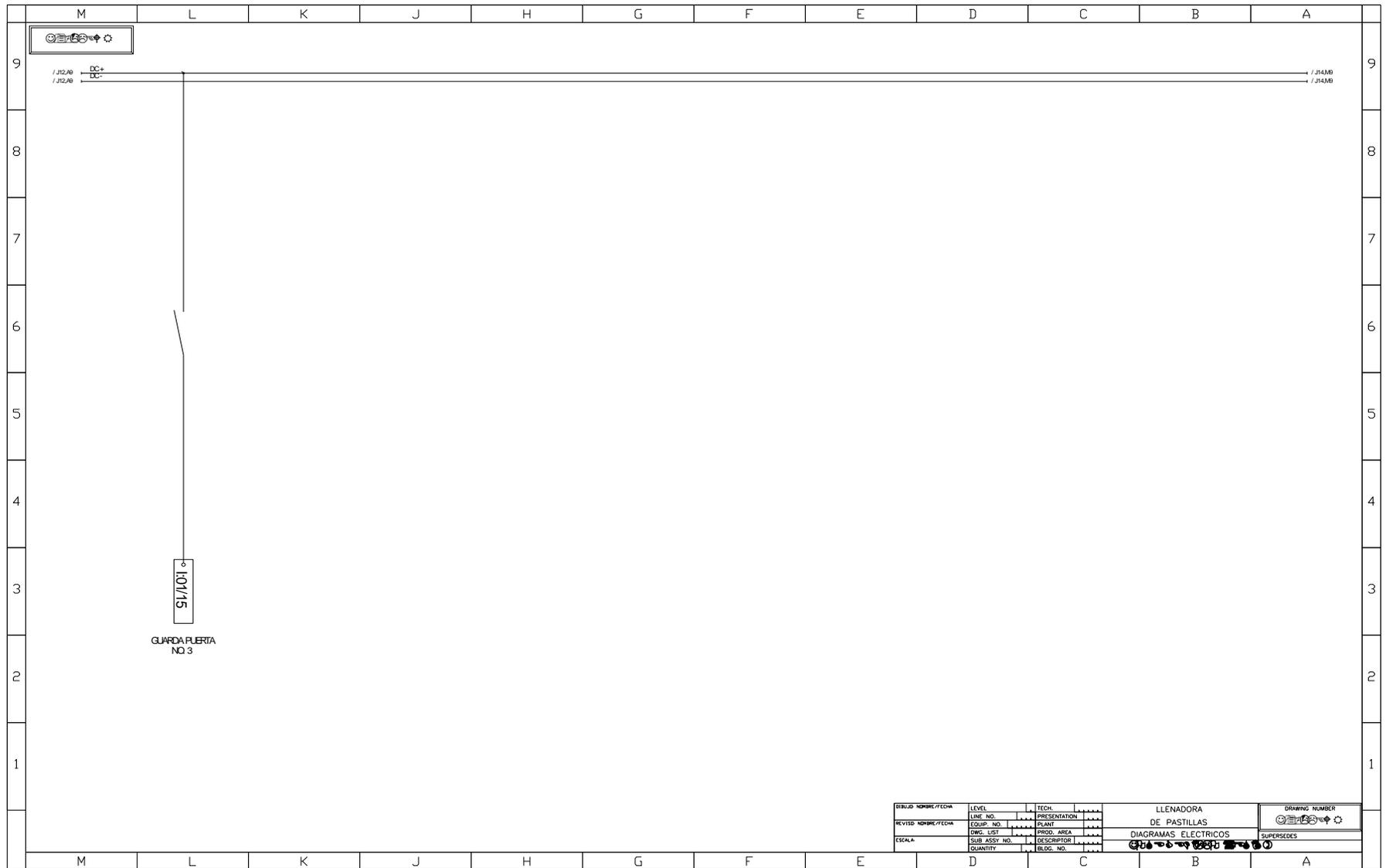


DIBUJO	NOMBRE/FECHA	LEVEL	TECH.	LLENADORA	DRAWING NUMBER
REVISO	NOMBRE/FECHA	LINE NO.	PRESENTATION	DE PASTILLAS	
ESCALA		EQUIP. NO.	PLANT	DIAGRAMAS ELECTRICOS	
		DWG. LIST	PROD. AREA		
		SUB. ASSY NO.	DESCRIPTOR		
		QUANTITY	BLOC. NO.		

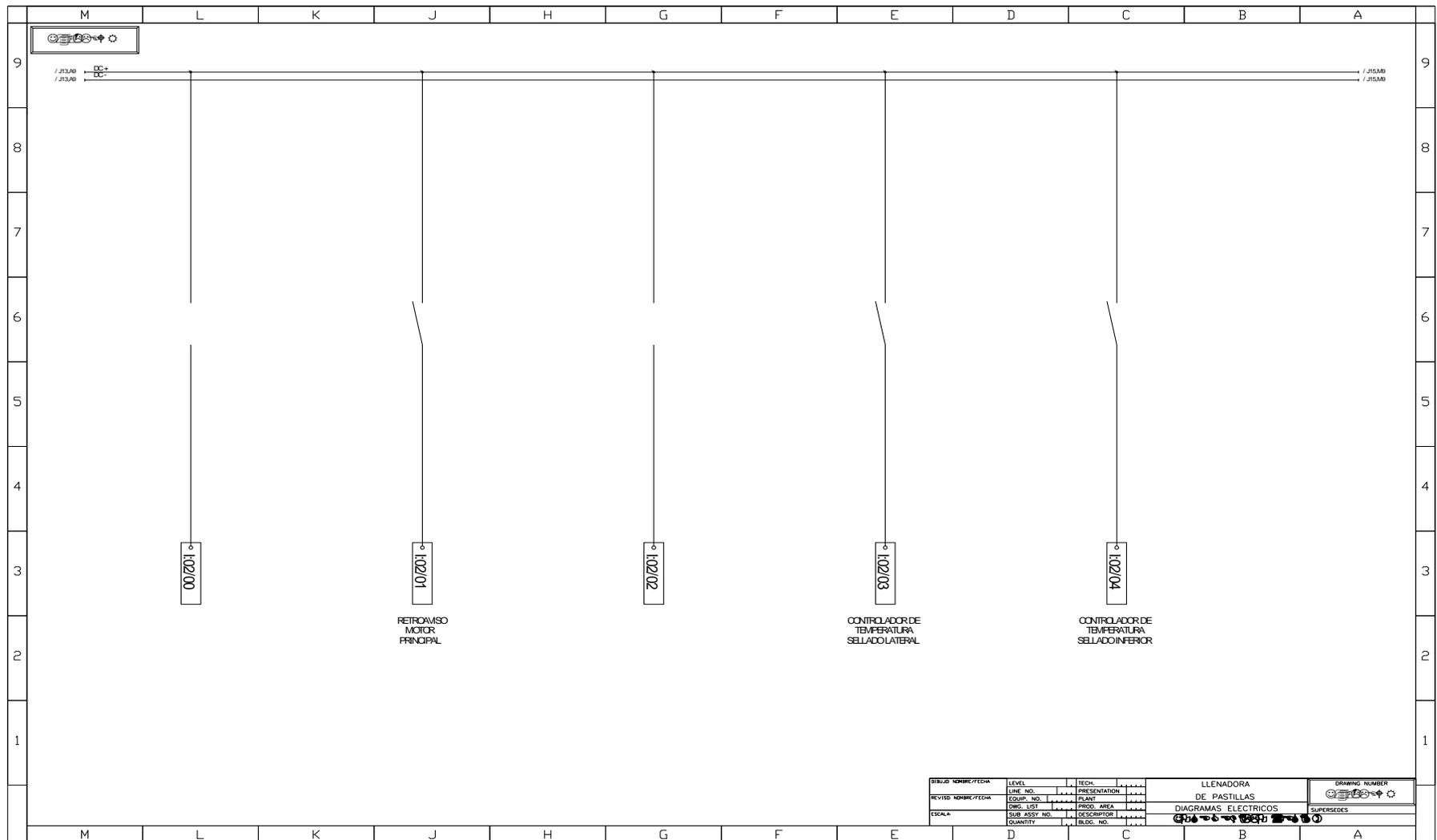


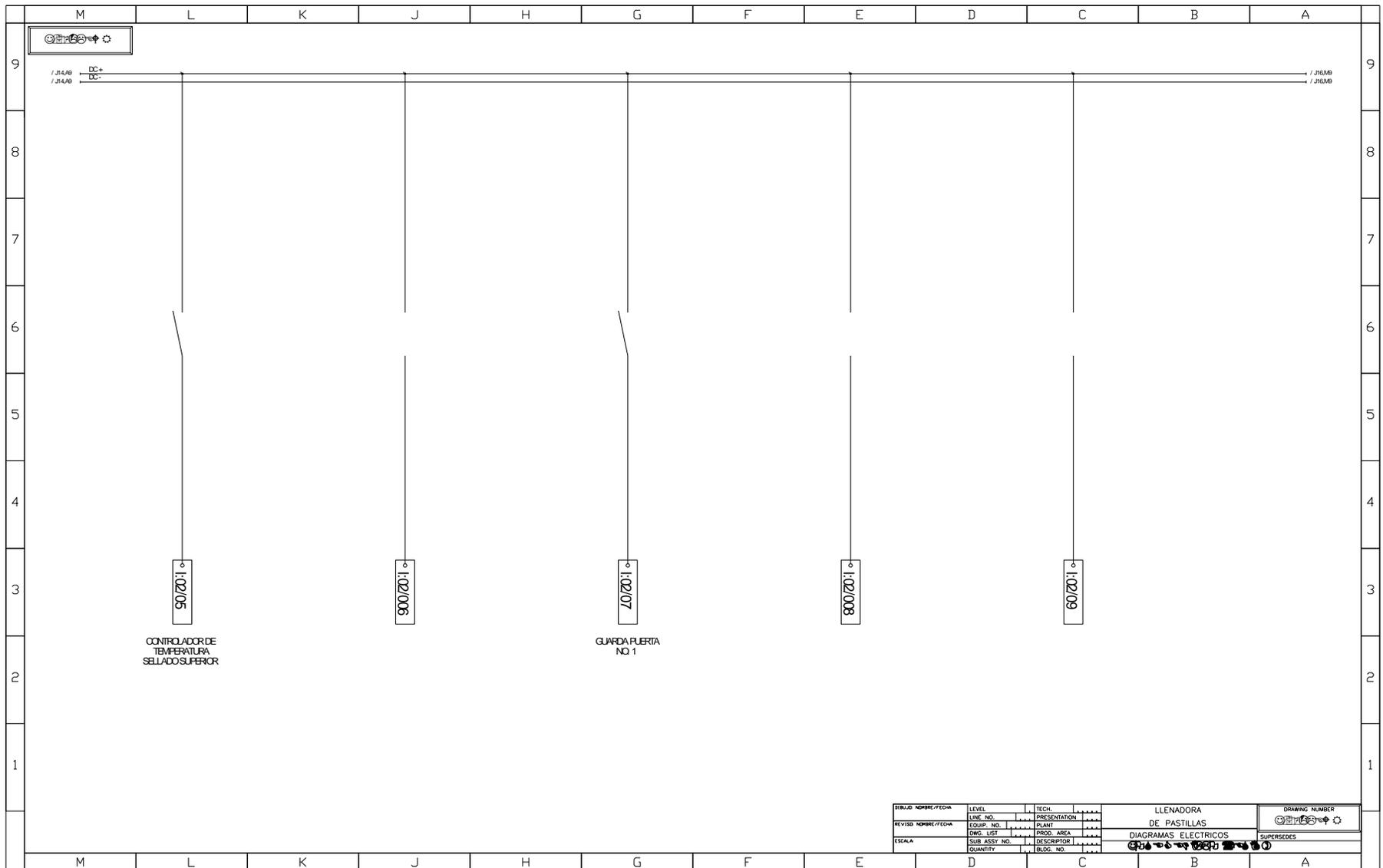


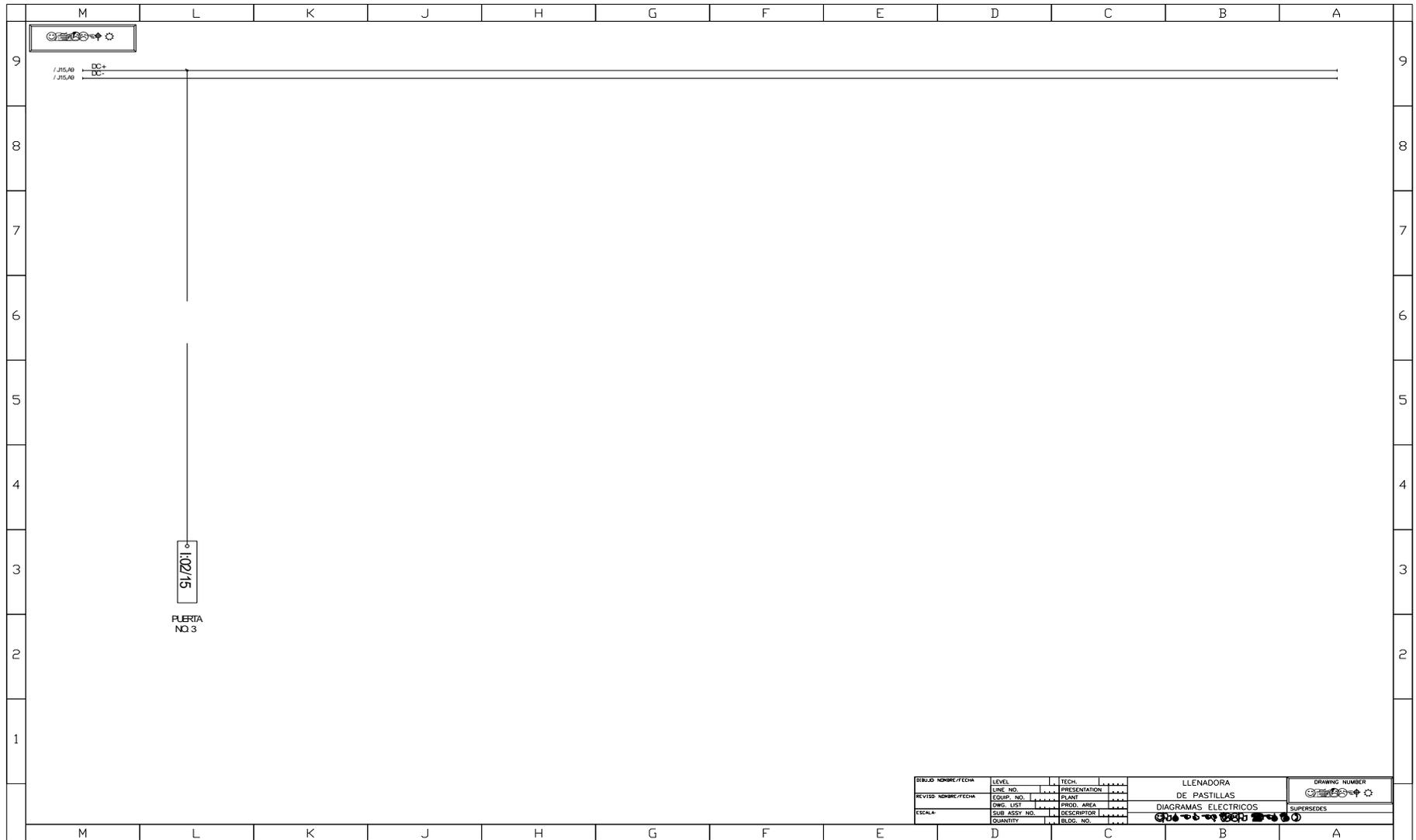


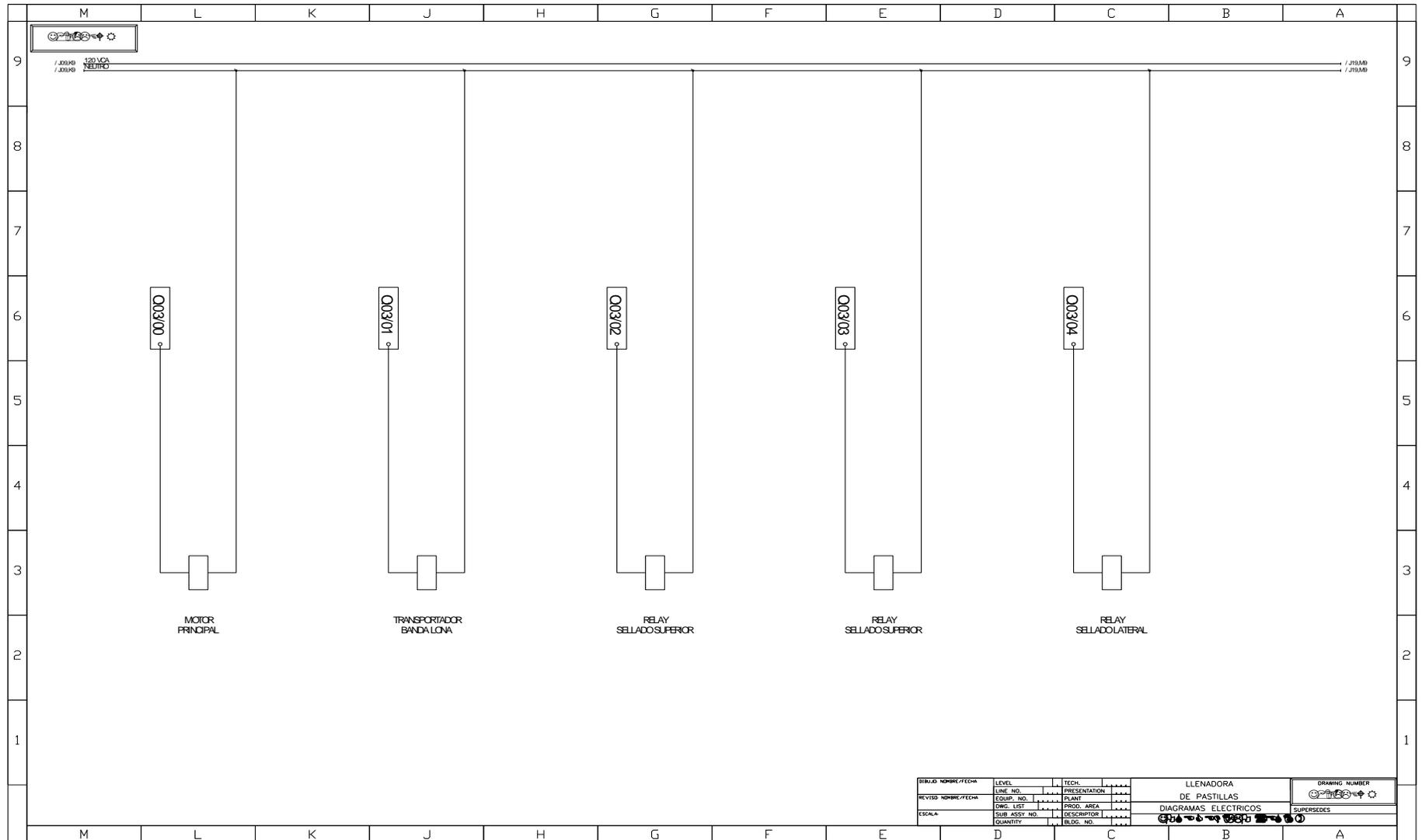


DIBUJO: NOMBRE / FECHA		LEVEL	TECH.	LLENADORA		DRAWING NUMBER
REVISO: NOMBRE / FECHA		EQUIP. NO.	PRESENTATION	DE PASTILLAS		
ESCALA		DWG. LIST	PROD. AREA	DIAGRAMAS ELECTRICOS		SUPERSEDES
		EQUIP. ASSY. NO.	DESCRIPTOR			
		QUANTITY	BLOG. NO.			









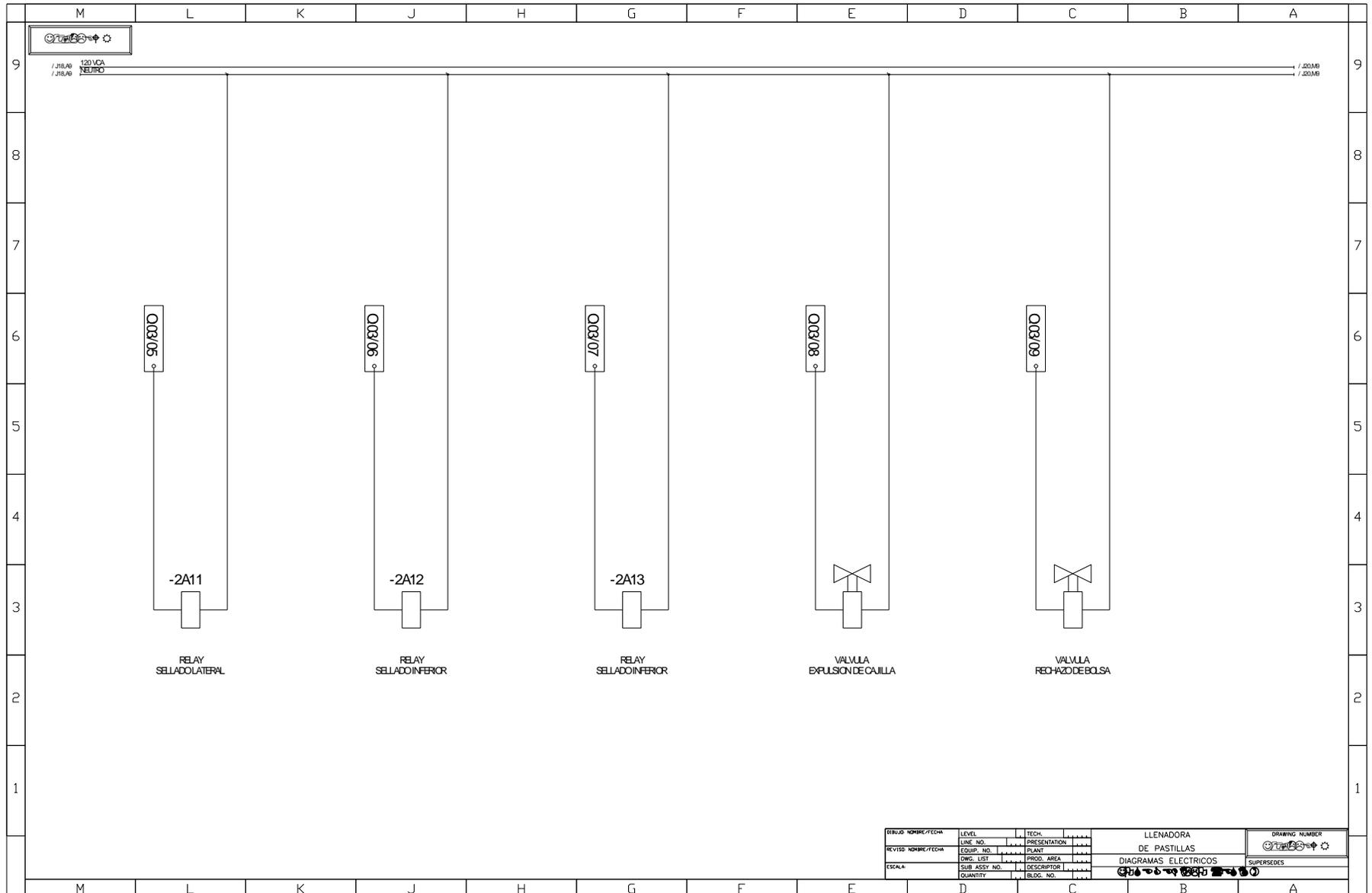
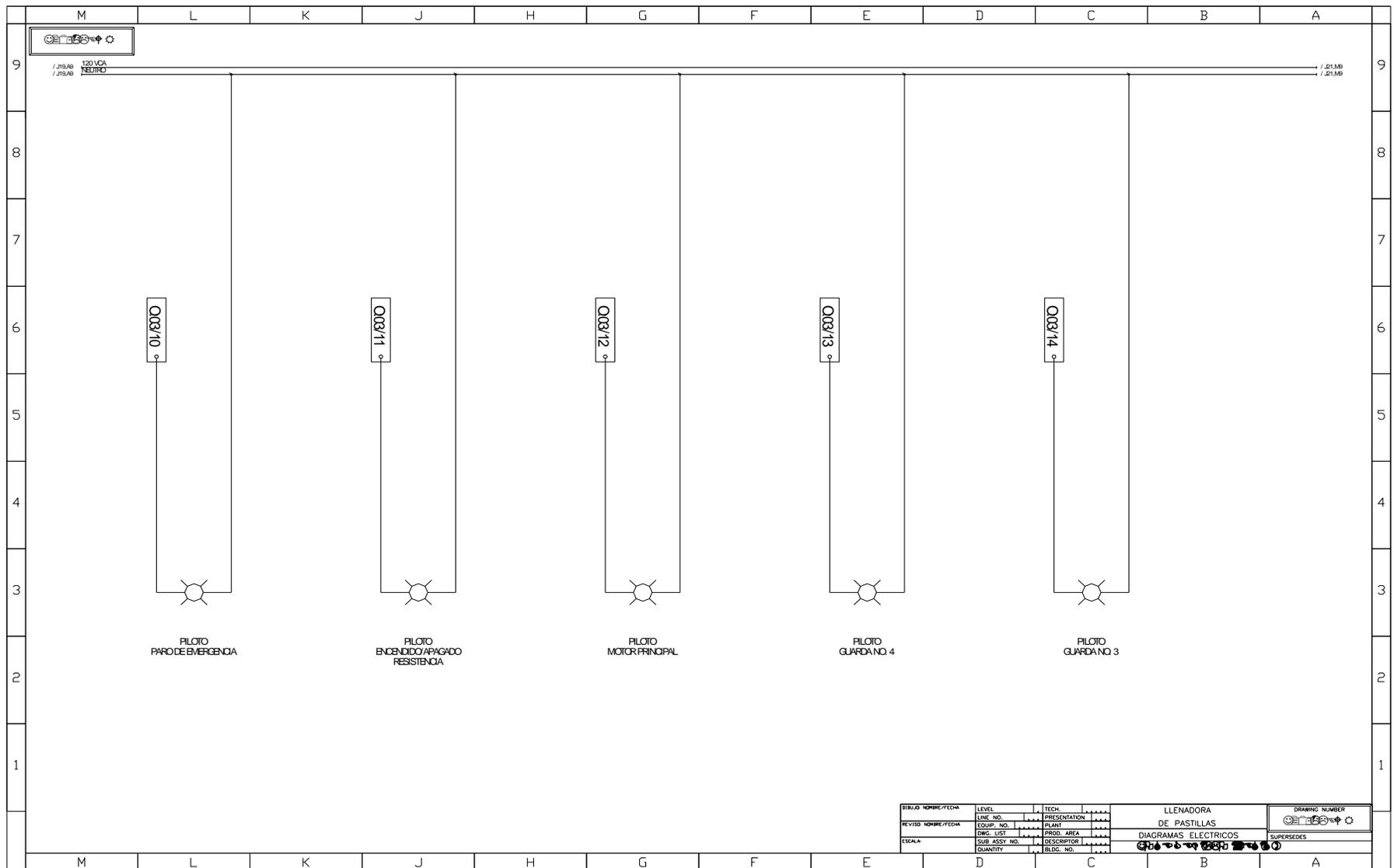
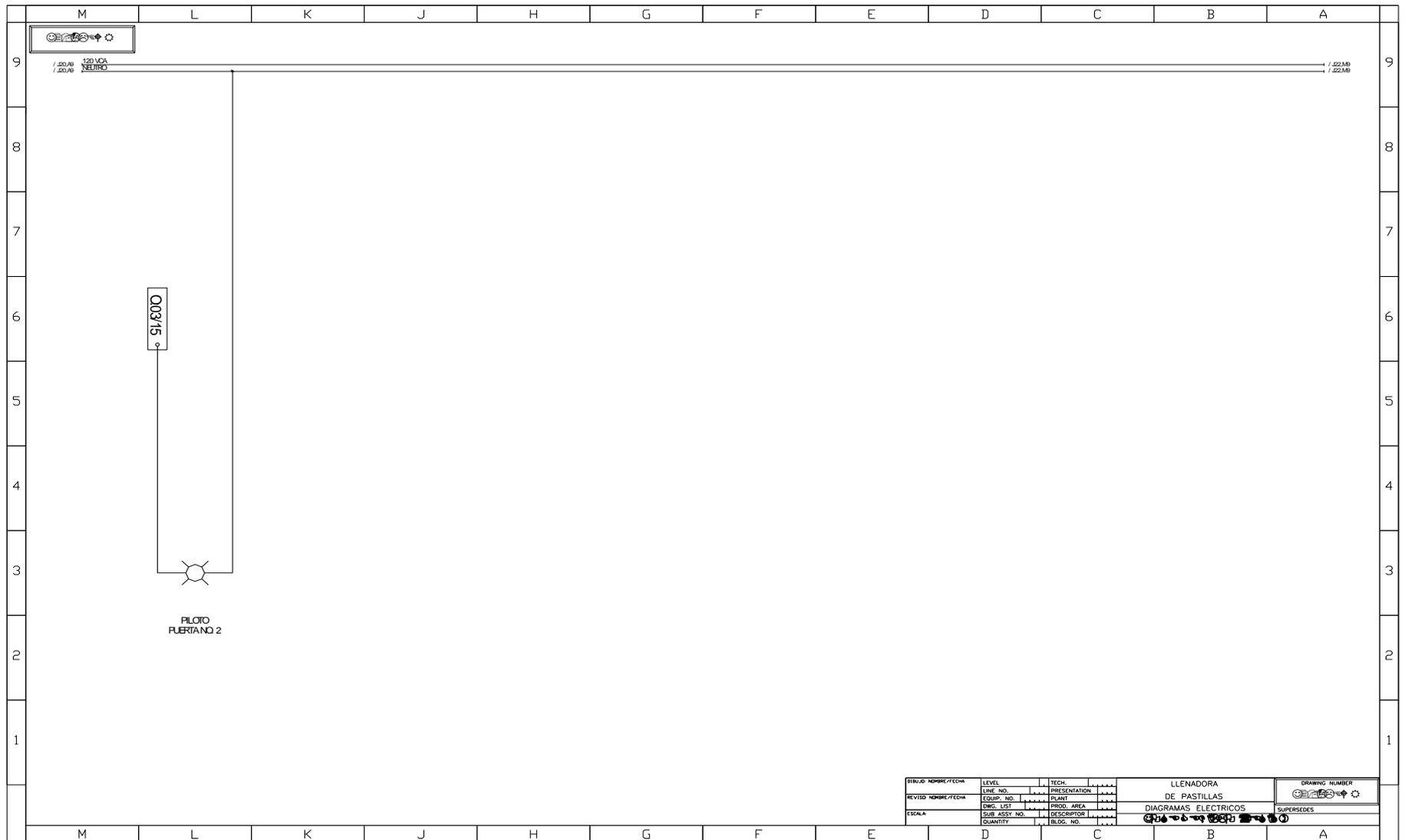
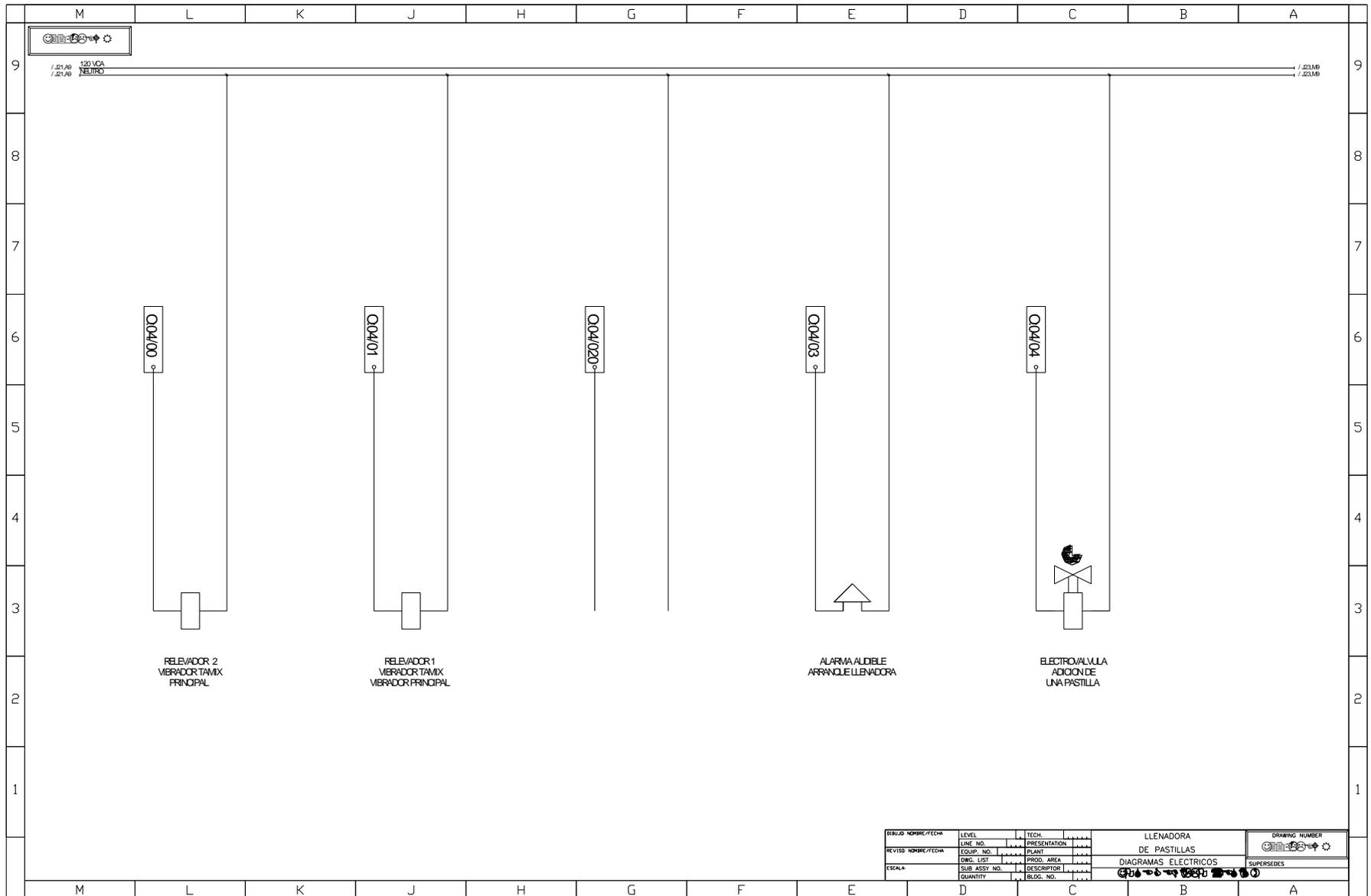


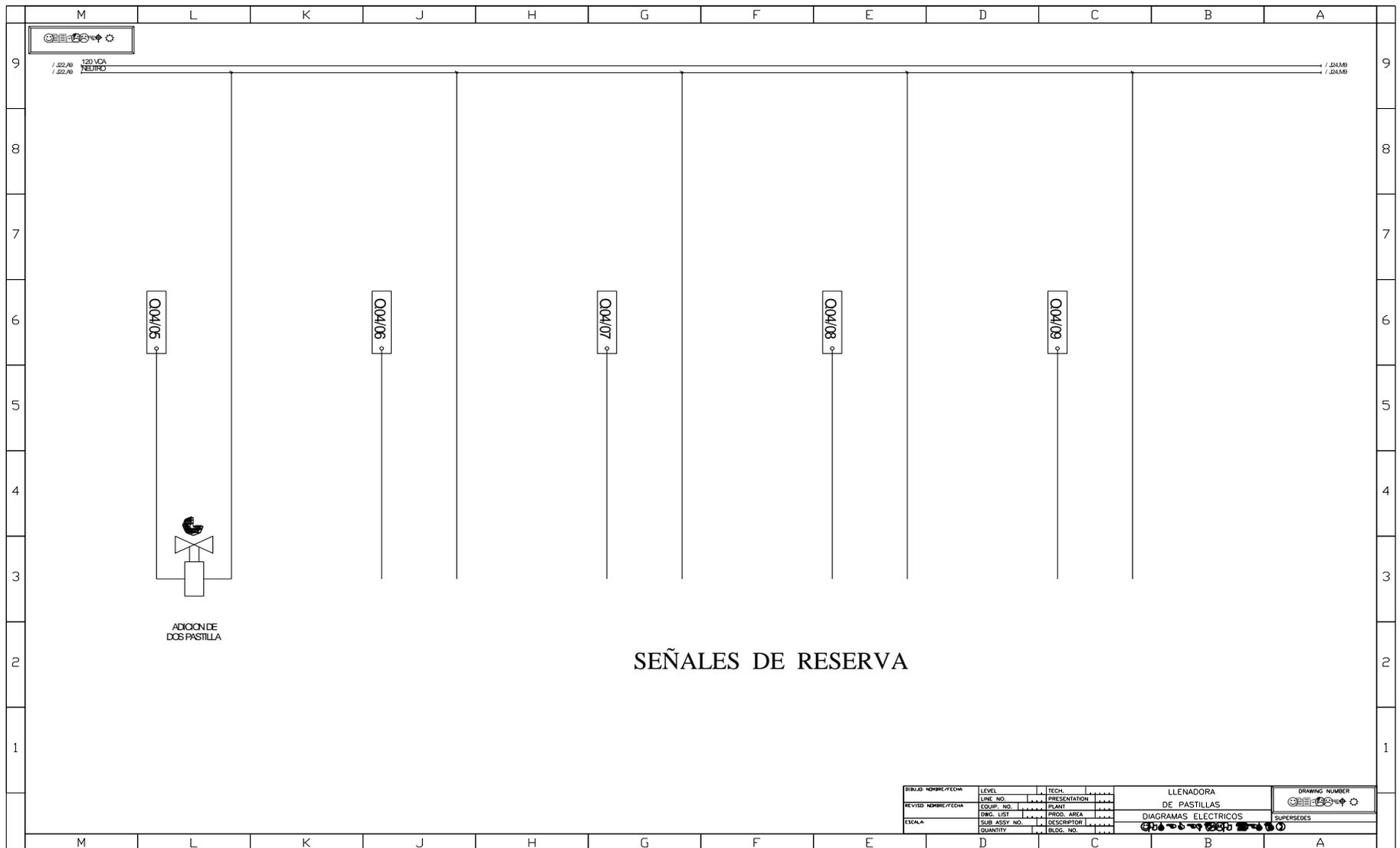
DIAGRAMA	REVISOR	FECHA	LÍNEA	TECH.	LLENADORA DE PASTILLAS	DRAWING NUMBER
REVISOR	FECHA	PRESENCIA	PLANT	DIAGRAMAS ELECTRICOS	SUPERSEDES	
ESCALA	SUB ASSY NO.	DESCRIPCIÓN	BLOC. NO.			





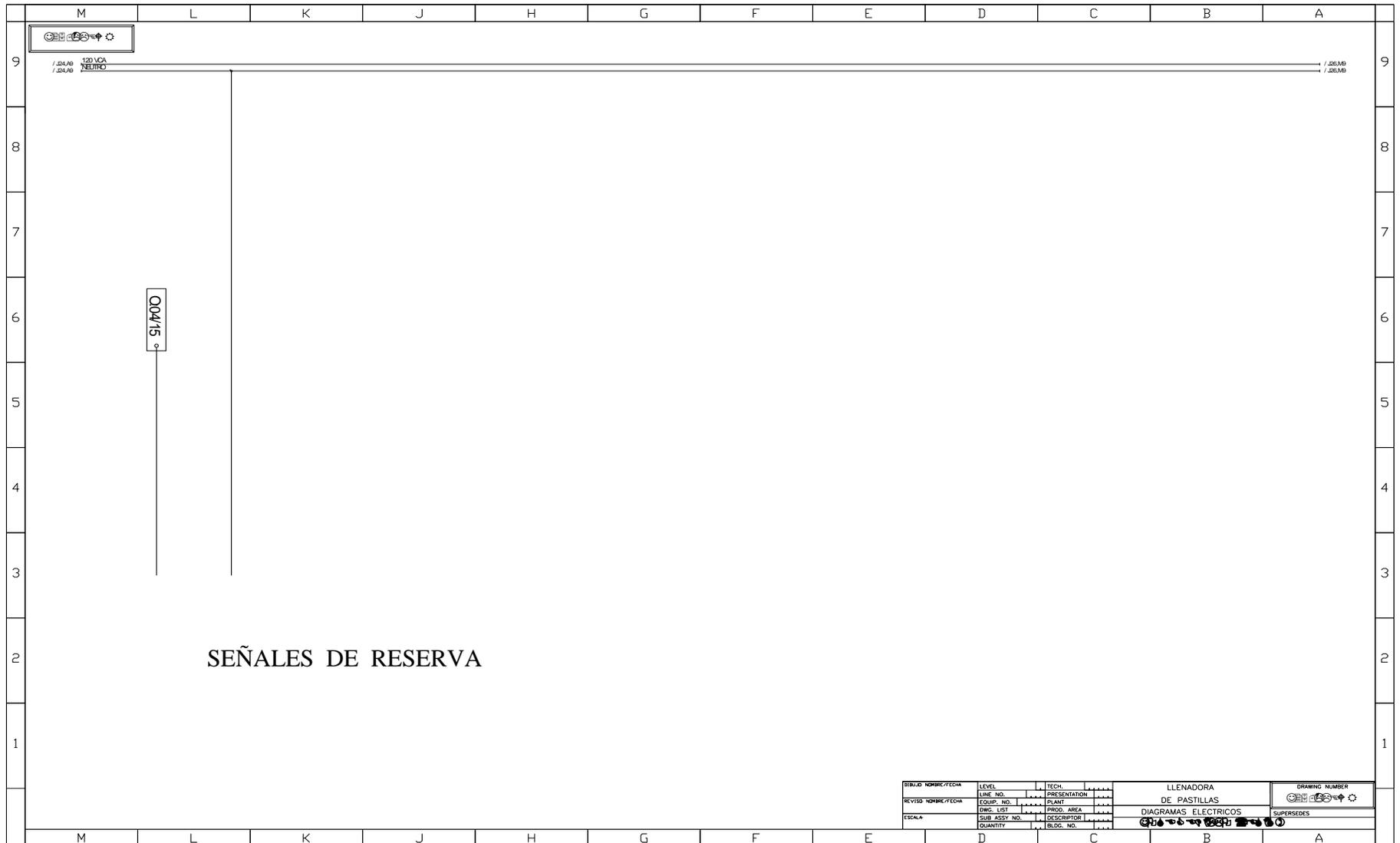


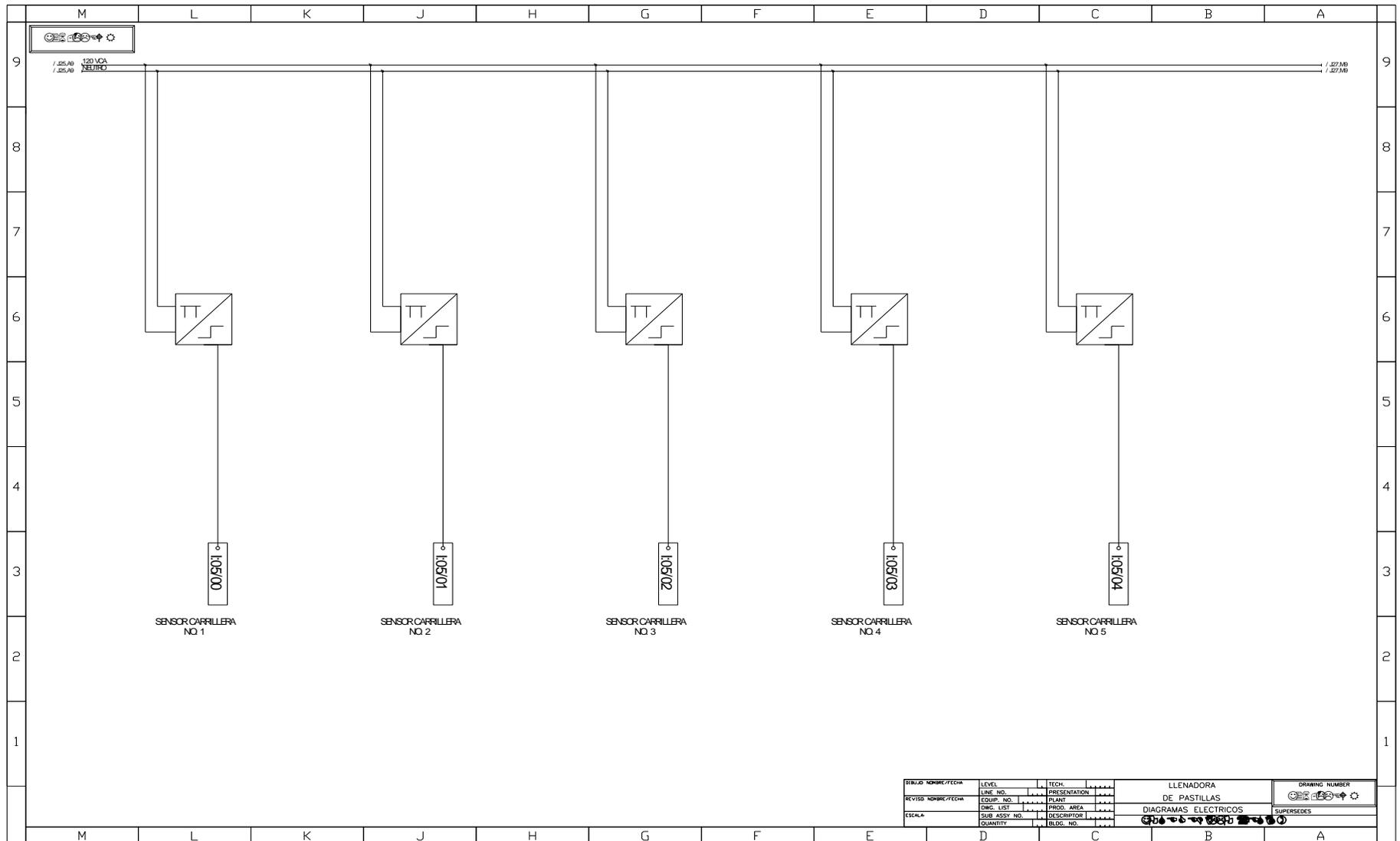
DIBUJO NOMBRE/TECNO REVISO NOMBRE/TECNO ESCALA	LEVEL LINE NO. EQUIP. NO. ENG. LIST SUB ASSY NO. QUANTITY	TECH. PRESENTATION PLANT PRD. AREA DESCRIPTOR BLDG. NO.	LLENADORA DE PASTILLAS DIAGRAMAS ELECTRICOS SUPLENEDORES	DRAWING NUMBER
--	--	--	---	--------------------



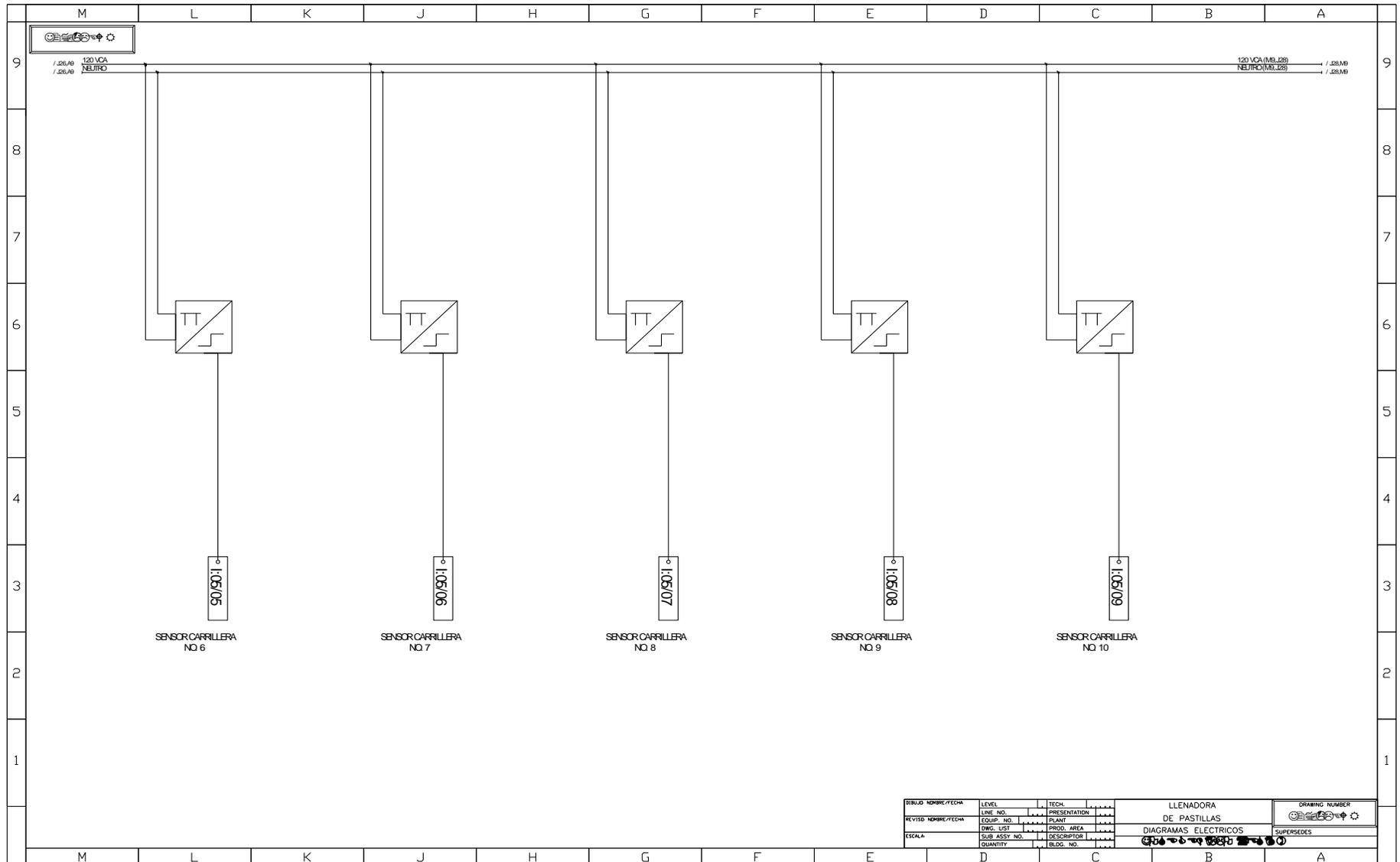
REVISOR: NOMBRE/FECHA	LEVEL NO.	TECH.	LLENADORA DE PASTILLAS	DRAWING NUMBER
REVISOR: NOMBRE/FECHA	EQUIP. NO.	PRESENTATION	DIAGRAMAS ELECTRICOS	004/05-09
ESCALA:	PROG. LIST	PROG. AREA	SUPERSEDES	
	SUB ASSY NO.	DESCRIPTOR		
	QUANTITY	BLOQ. NO.		

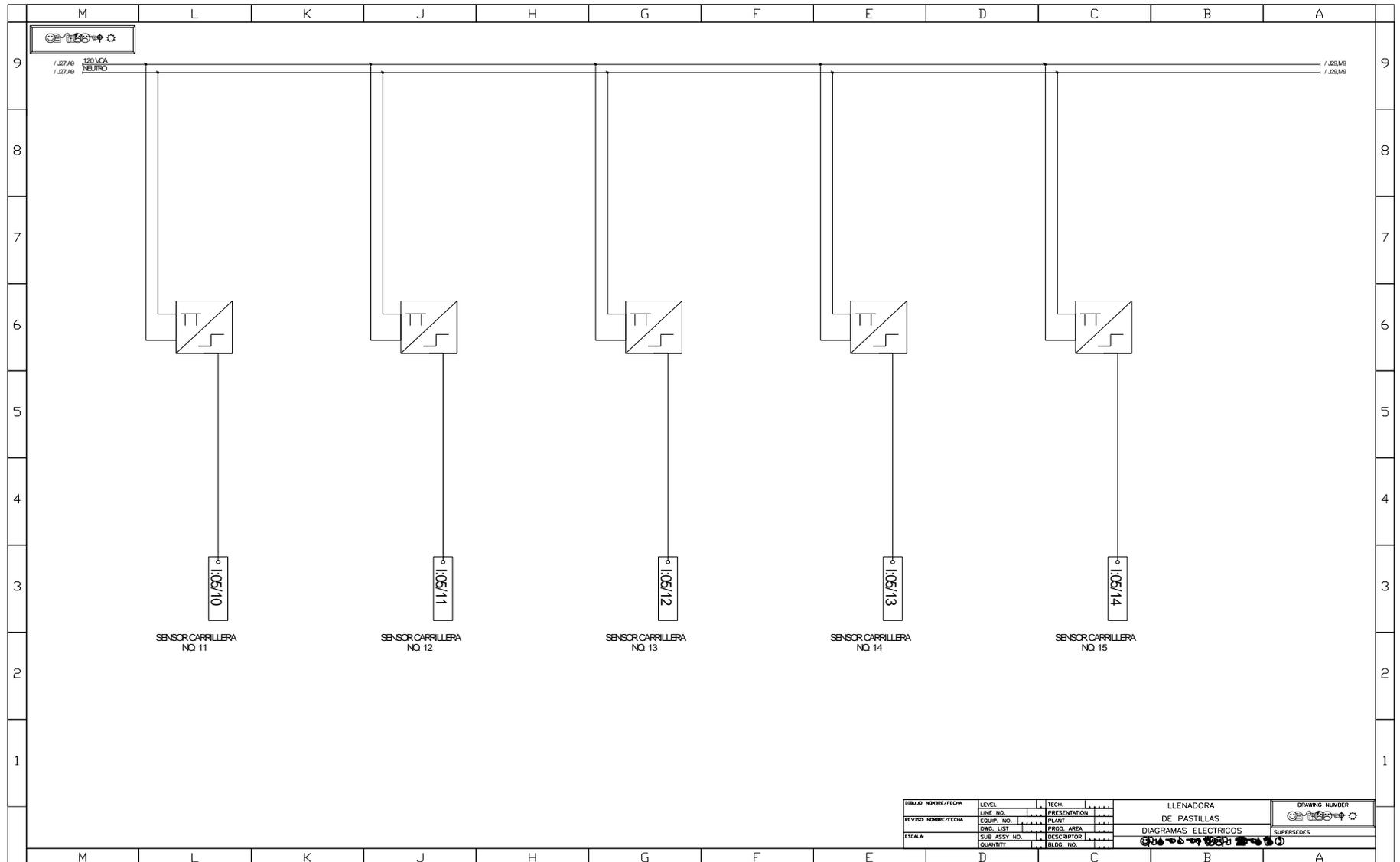


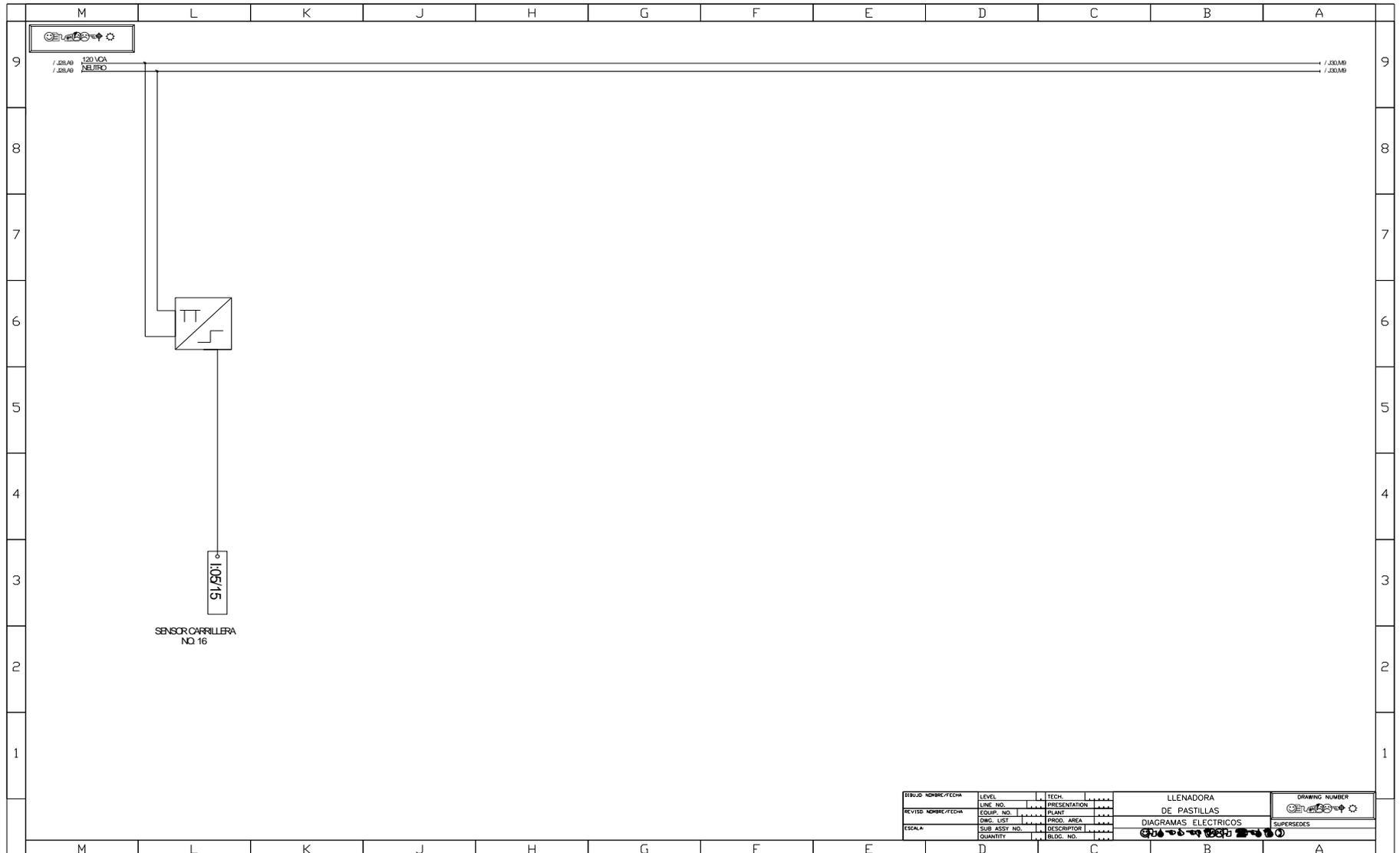




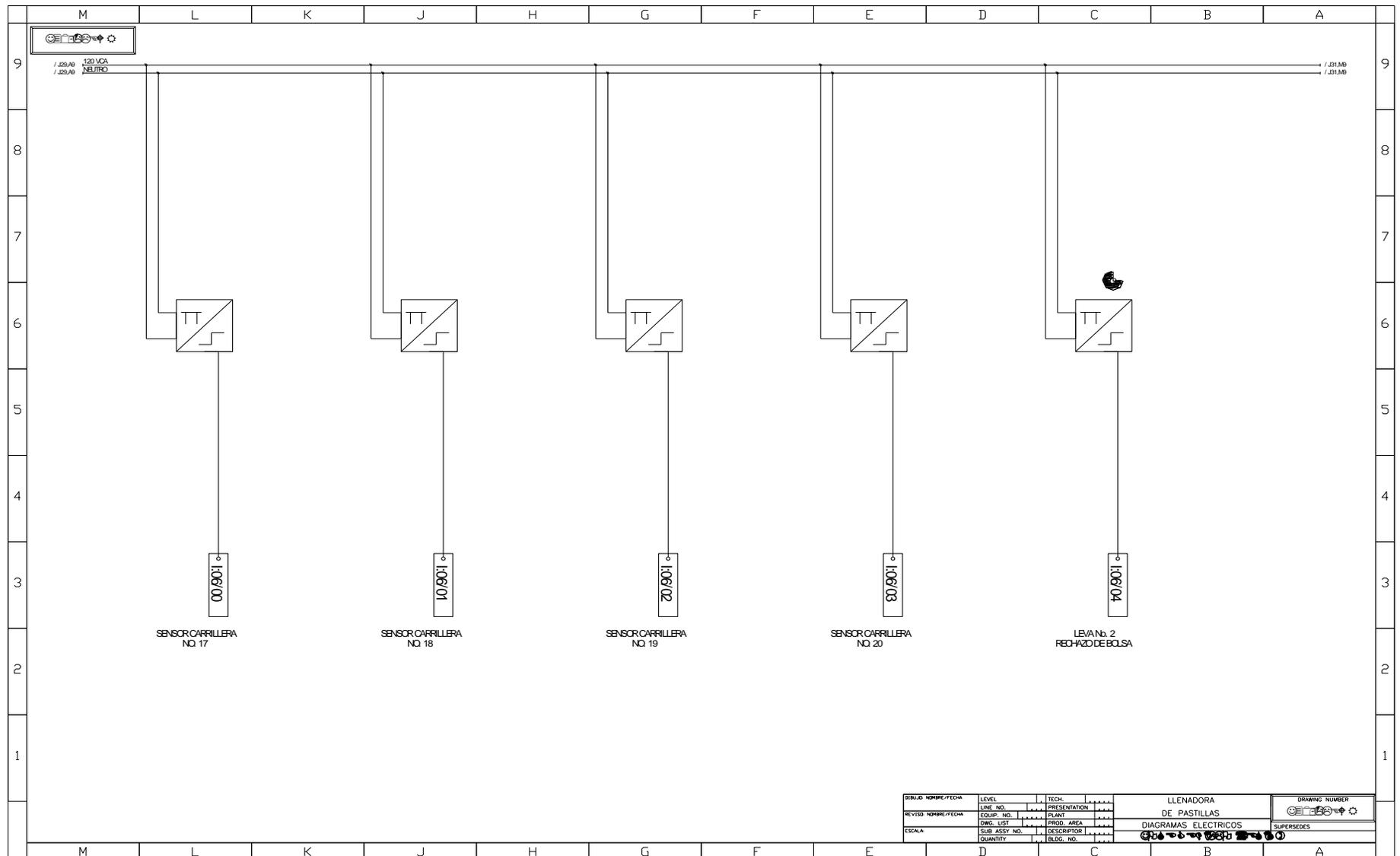
ESTILO NOMBRE/FECHA	LEVEL	TECH.	LLENADORA	DRAWING NUMBER
REVISO NOMBRE/FECHA	LINE NO.	PRESENTATION	DE PASTILLAS	
	EQUIP. NO.	PLANT		
	DWG. LIST	PRD. AREA	DIAGRAMAS ELECTRICOS	SUPERSEDES
ESCALA	SUB. ASST. NO.	DESCRIPTOR		
	QUANTITY	BLDG. NO.		

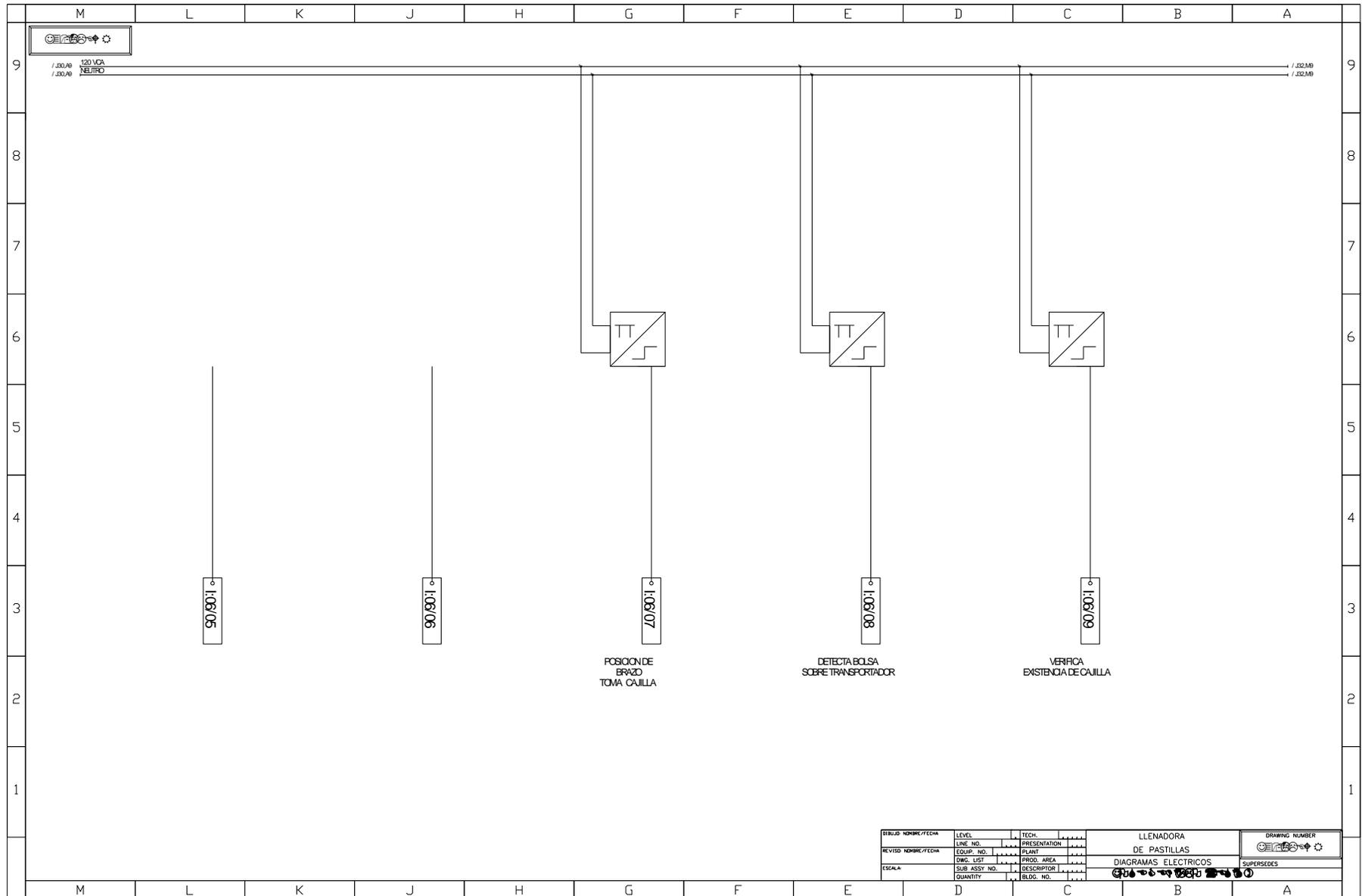


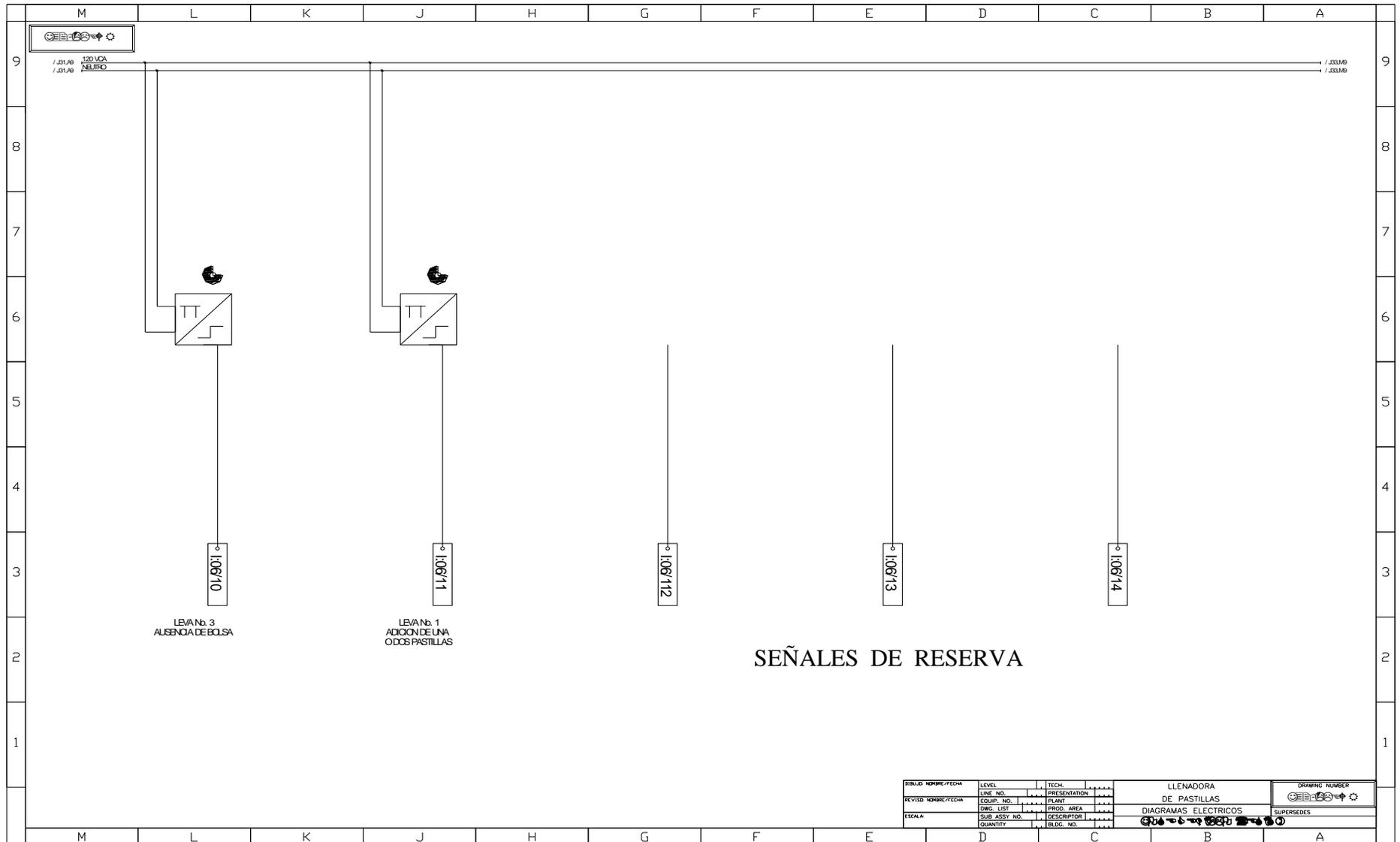


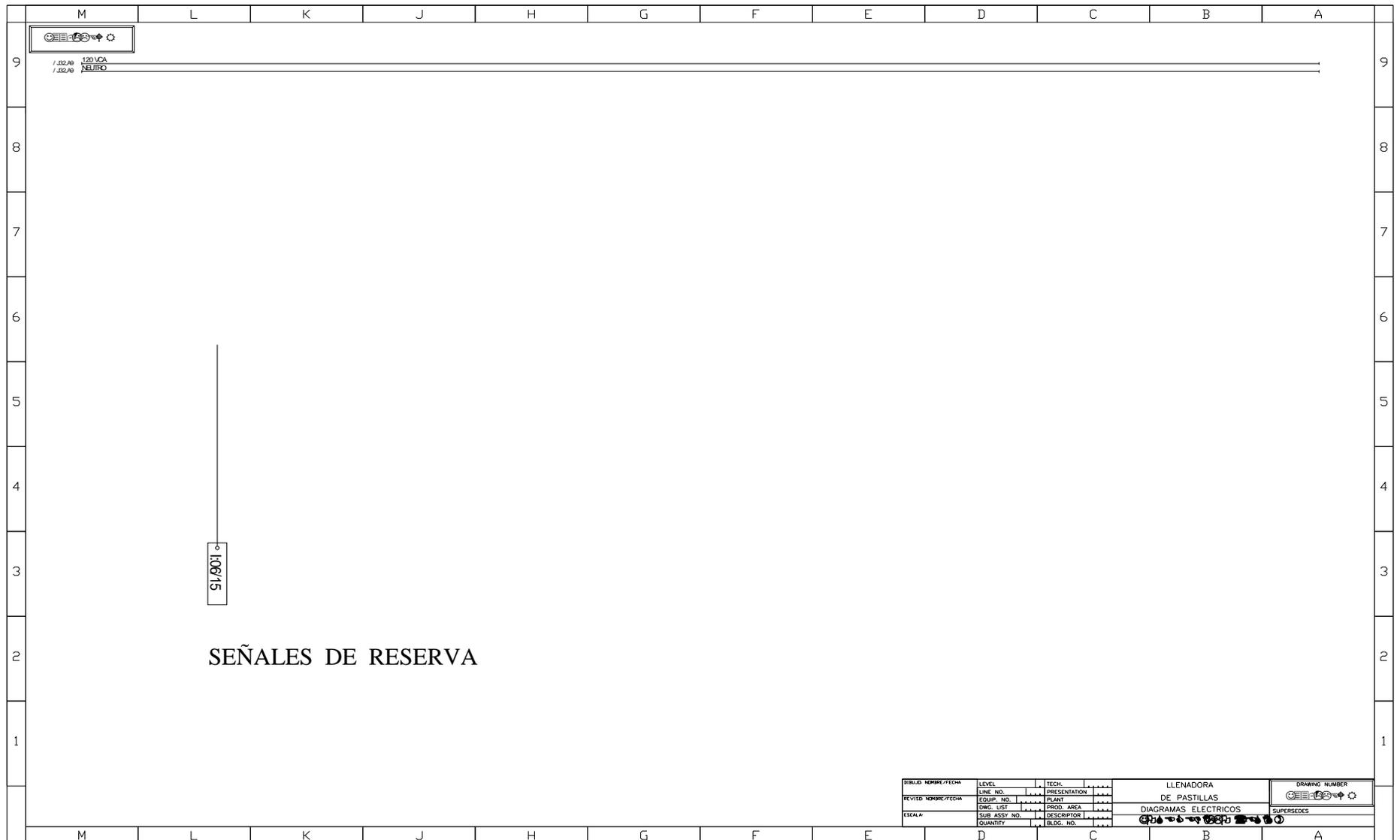


BUILD NO/REV/FECHA	LEVEL	TECH.	LLENADORA	DRAWING NUMBER
REVISD NO/REV/FECHA	EQUIP. NO.	PRESENTATION	DE PASTILLAS	
ETCAL*	QTY. LIST	PLANT	DIAGRAMAS ELECTRICOS	
	SUB ASSY NO.	PROD. AREA	SUPERSEDES	
	QUANTITY	DESCRIPOR		
		BLDG. NO.		









CONCLUSIONES

Concluir siempre ha sido difícil en cualquier tarea, trabajo pero es la culminación de un proyecto. Cuando se escribe acerca de un trabajo una experiencia siempre es difícil determinar el final pero es satisfactorio llegar a la meta propuesta.

La automatización de una línea de producción llenado de pastillas ha sido un proyecto que me llena de experiencias y satisfacciones. Cuando se plantea los requerimientos de automatización de la línea, inicialmente se quería la automatización de la llenadora para que se asegurara que el dosificado de veinte pastillas para ello se plantea la dosificación de una o dos pastillas al final de la llenadora. Después se plantea la opción del rechazo de la cajilla siempre y cuando no tenga bolsa y/o no se tenga las veinte pastillas. Se van presentando diferentes opciones de solución y conforme se resuelven unas van surgiendo nuevas hasta llegar a un fin común por ambas partes.

Me enseña que se debe conocer de diferentes disciplinas que aunque uno no sea un experto en cada de ellos se deben de afrontar e irlos sorteando de la mejor manera para darles una solución

Puedo comentarles que es necesario conocer de las siguientes disciplinas:

- Neumática.
- Eléctrica
- Mecánica
- Control
- Programación.
- Electrónica.

Uno aplica lo que se nos enseña en la escuela pero hay algo que no se aprende y que se va adquiriendo conforme pasa el tiempo y es: experiencia y confianza.

Se cumple con el objetivo se logra, se termina y funciona.

Bibliografía.

Instrumentación industrial, Antonio Creus.

Automatización reingeniería en los procesos de negocios a través de la simulación, Gregory A. Hansen

Autómatas Programables, Albert Mayol Ibadia

Automatización Neumática y Electro neumática. Salvador Millan.

Programación de Autómatas Programables Omron, Vincet Lladonosa / Ferrán Ibáñez

Detección (catalogo octubre 1994), Grupo Schneider (Telemecanique)

Installation & Operation Manual, Bulletin 1746 & 1747 SLC 500 fixed hardware style Programmable controllers, Allen Bradley.

Programa de Fabricación, Festo Pneumatic

Intouch, Wonderware Training.

Sysmac C-series and CVM1 PCS, Sysmac Support Software, Operation Manual C-Serie