



INSTITUTO SUPERIOR
UNIVERSITARIO

SUCE

**GUÍA GENERAL DE ESTUDIO
DE AUTOMATIZACIÓN
INDUSTRIAL**



Guía General de Estudio de Automatización Industrial

Christian Andrés Ortega Hidalgo

Verónica Johana Andrade García

Verónica del Rocío Cuasquer Gualoto

Gladys Rocío Herrera Panchi

Fabrizio Manuel Tipantocta Pillajo

2026

Esta publicación ha sido sometida a revisión por pares académicos específicos por:

Oscar Omar Gonzales Zurita
Universidad de las Américas

Corrección de estilo:

- Fabrizio Manuel Tipantocta Pillajo - Docente - Sucre

Diseño y diagramación:

- Freddy Javier Centeno Martínez - Docente - Sucre

Editorial RIMANA

Primera Edición
Quito – Ecuador

INSTITUTO SUPERIOR UNIVERSITARIO SUCRE

ISBN: 978-9942-590-17-6

Esta publicación está bajo una licencia de Creative Commons Reconocimiento-No Comercial-Compartir Igual 4.0 Internacional.



MISIÓN

Ser una Institución Superior Universitaria con estándares de calidad académica e innovación, reconocida a nivel nacional con proyección internacional.

VISIÓN

Formamos profesionales competentes con espíritu emprendedor, capaces de contribuir al desarrollo integral del país.

Los contenidos de este trabajo están sujetos a una licencia internacional Creative Commons Reconocimiento-No Comercial-Compartir Igual 4.0 (CC BY-NC-SA 4.0). Usted es libre de Compartir — copiar y redistribuir el material en cualquier medio o formato. Adaptar — remezclar, transformar y construir a partir del material citando la fuente, bajo los siguientes términos: Reconocimiento- debe dar crédito de manera adecuada, brindar un enlace a la licencia, e indicar si se han realizado cambios. Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que usted o su uso tienen el apoyo de la licenciante. No Comercial-no puede hacer uso del material con propósitos comerciales. Compartir igual-Si remezcla, transforma o crea a partir del material, debe distribuir su contribución bajo la misma licencia del original. No puede aplicar términos legales ni medidas tecnológicas que restrinjan legalmente a otras a hacer cualquier uso permitido por la licencia.

<https://creativecommons.org/licenses/by-nc-sa/4.0/>



**Reconocimiento-NoComercial-CompartirIgual
4.0 Internacional (CC BY-NC-SA 4.0)**

Usted acepta y acuerda estar obligado por los términos y condiciones de esta Licencia, por lo que, si existe el incumplimiento de algunas de estas condiciones, no se autoriza el uso de ningún contenido.

INDICE

UNIDAD 1: AUTOMATIZACION INDUSTRIAL.....	7
UNIDAD 2: CONTROLADOR LÓGICO PROGRAMABLE (PLC) - INTRODUCCIÓN Y LÓGICA DE PROGRAMACIÓN EN LADDER.....	16
UNIDAD 3: PROGRAMACIÓN SECUENCIAL CON GRAFCET.....	27
UNIDAD 4: INSTRUMENTACIÓN VIRTUAL Y SISTEMAS SCADA.....	37
UNIDAD 5: COMUNICACIONES INDUSTRIALES	50
REFERENCIAS BIBLIOGRÁFICAS	61

Presentación de la asignatura

La asignatura de Automatización Industrial introduce a los estudiantes de la Carrera de Electrónica en los principios fundamentales de la automatización de procesos, proporcionando las bases necesarias para la comprensión, análisis y diseño de sistemas de control secuencial y continuo en entornos industriales. A través del estudio de los elementos de automatización, como sensores, actuadores, sistemas de mando y control, se desarrollan competencias para interpretar y estructurar procesos automatizados de manera lógica y eficiente.

En este contexto, se aborda la programación secuencial utilizando GRAFCET como método principal de enseñanza, permitiendo al estudiante modelar, analizar y documentar el comportamiento de sistemas automatizados desde sus etapas iniciales hasta su operación final. Posteriormente, estos conceptos se integran con la programación de controladores lógicos programables (PLC) y la introducción a los sistemas SCADA, sentando las bases para la supervisión, adquisición de datos y control de procesos industriales.

Esta asignatura es fundamental para los estudiantes de la Carrera de Electrónica, ya que proporciona los conocimientos necesarios para comprender y aplicar la automatización en sistemas productivos, de control e instrumentación industrial. De esta manera, se fortalece la formación técnica y analítica del estudiante para enfrentar desafíos reales del sector industrial, donde la automatización es un pilar clave para la eficiencia, seguridad y optimización de los procesos.

Resultados del aprendizaje

Conoce los conceptos de la automatización industrial, arquitectura, normativa y tipos de sistemas de control.

Maneja y selecciona la instrumentación adecuada para el sistema automatizar.

Cumple con la normativa vigente en el diseño de sistemas de automatización industrial.

Conoce la arquitectura, partes y aplicación de los controladores lógicos programables.

Realiza la programación del PLC en lenguajes ladder, GRAFCET y de bloques.

Cumpla la normativa y simbología en el diseño de la automatización.

Identifica los protocolos industriales usados en la industria y conoce sus características principales.

UNIDAD 1: AUTOMATIZACION INDUSTRIAL

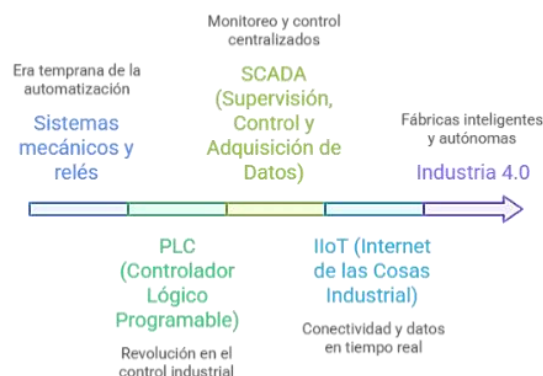
Fundamentos de automatización y Arquitectura

Definición de automatización industrial, sistema automatizado y su evolución tecnológica.

La automatización industrial se define como el conjunto de tecnologías (mecánicas, electrónicas, computacionales) aplicadas para operar y controlar procesos industriales con una intervención humana mínima o nula, destinadas a aumentar la productividad, la calidad, la seguridad y la flexibilidad de la manufactura. Un sistema automatizado es, por tanto, una integración sinérgica de dispositivos de potencia (actuadores), sensores, unidades de control y elementos de comunicación que trabajan coordinadamente para ejecutar una secuencia de acciones sobre un proceso físico (Brunete et al., 2025). Su evolución tecnológica, inició con la mecanización durante la Revolución Industrial, donde máquinas de vapor reemplazaron el esfuerzo humano. A principios del siglo XX, la electrificación introdujo el control eléctrico y los motores, permitiendo cadenas de montaje más eficientes. La tercera fase, la automatización, surgió a mediados del siglo XX con el uso de controladores lógico-programables (PLC) y computadores, permitiendo secuencias complejas y flexibles. Actualmente, nos encontramos en la era de la digitalización e integración representada en la Figura 1, caracterizada por sistemas ciberfísicos, el Internet Industrial de las Cosas (IIoT), la inteligencia artificial y la integración total de la información a través de modelos como la Industria 4.0 (Brunete et al., 2025)

Figura 1

Evolución de la automatización industrial



Ventajas, desventajas y criterios de aplicación de la automatización en procesos industriales.

La Tabla 1 resume las principales ventajas y desventajas asociadas a la automatización en una industria, detallando los criterios de aplicación técnica.

Tabla 1

Cuadro Comparativo de ventajas y desventajas

VENTAJAS	DESVENTAJAS	CRITERIOS DE APLICACIÓN
1. Incremento de productividad y capacidad de producción – Permite operación continua y ritmos sostenidos sin fatiga.	1. Alta inversión inicial – Los costos de adquisición, instalación y puesta en marcha son significativos.	1. Volumen de producción alto y estable – Se justifica la inversión en grandes series con demanda predecible.
2. Mejora de la calidad y uniformidad – Reduce la variabilidad humana, garantizando estándares consistentes.	2. Necesidad de personal especializado – Requiere capacitación en diseño, programación y mantenimiento, generando costos adicionales.	2. Operaciones peligrosas – Entornos con riesgos físicos, químicos o biológicos donde la seguridad humana es prioritaria.
3. Optimización de recursos – Disminuye el desperdicio de materias primas y energía, aumentando la eficiencia.	3. Resistencia al cambio y posible desplazamiento de mano de obra – Impacto social y cultural en la organización.	3. Especificaciones de calidad extremadamente estrictas – Procesos que exigen precisión y repetibilidad superior a la humana.
4. Mayor seguridad laboral – Aleja a los operarios de tareas riesgosas o ambientes hostiles.	4. Reducción de flexibilidad – Sistemas fijos pueden dificultar los cambios en el diseño del producto o proceso.	4. Tareas monótonas y repetitivas – Actividades que generan fatiga o errores por falta de concentración.
5. Reducción de costos laborales directos – Minimiza la intervención humana en operaciones repetitivas.	5. Dependencia tecnológica y vulnerabilidad ante fallos – Una avería puede detener toda la línea de producción.	5. Análisis costo-beneficio favorable – Cuando los ahorros operativos y las ganancias superan claramente la inversión inicial a largo plazo.

VENTAJAS	DESVENTAJAS	CRITERIOS DE APLICACIÓN
6. Disponibilidad 24/7 – Posibilidad de funcionamiento en turnos extendidos sin interrupciones.	6. Complejidad de integración – Dificultad para adaptar la automatización a infraestructuras o procesos existentes.	6. Procesos con alto impacto en eficiencia global – Donde la automatización permite mejoras significativas en tiempo, costo o calidad.

Clasificación tecnológica y grados de automatización industrial

Según Ragno (2024), una clasificación tecnológica común la divide en:

Automatización Fija o Dedicada: Es muy eficiente en producción masiva (ej: líneas de embotellado clásicas) pero rígida y costosa de modificar.

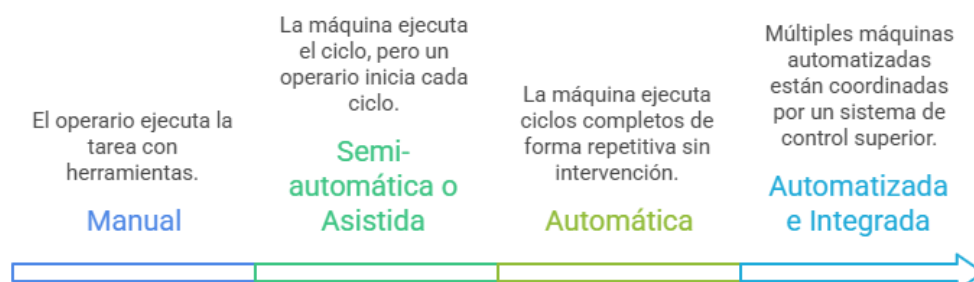
Automatización Programable: Utiliza controladores (como PLCs) cuyo software puede ser reconfigurado para adaptarse a diferentes secuencias o productos. Es ideal para lotes de producción medianos (ej: máquinas CNC, robótica industrial).

Automatización Flexible: Un subconjunto avanzado de la programable, se adapta a las cambiantes necesidades de producción con un mínimo tiempo de parada, mediante instrucciones enviadas por un sistema supervisor

Respecto a los grados de automatización, van desde procesos manuales asistidos por instrumentos hasta automatización total e integrada, según lo detallado en la Figura 2:

Figura 2

Grados de automatización industrial



Arquitectura de los sistemas de automatización

El modelo conceptual más aceptado es la Pirámide de la Automatización que organiza los niveles de información y control según se observa en la Figura 3:

Nivel 0 - Campo: Contiene los dispositivos físicos: sensores y actuadores.

Nivel 1 - Control de Procesos: Donde los controladores (PLCs, controladores PID) ejecutan la lógica en tiempo real.

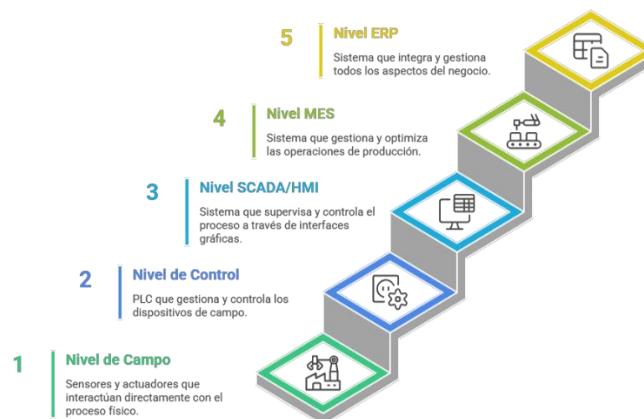
Nivel 2 - Supervisión: Sistemas SCADA (Supervisory Control and Data Acquisition) y HMIs (Interfaces Hombre-Máquina) que monitorean, visualizan y permiten la interacción operativa.

Nivel 3 - Planificación de la Producción: Sistemas MES (Manufacturing Execution Systems) que gestionan órdenes de trabajo, eficiencia y trazabilidad.

Nivel 4 - Planificación Empresarial: Sistemas ERP (Enterprise Resource Planning) que gestionan los recursos de toda la empresa (finanzas, logística, ventas) (Centro de Ciberseguridad Industrial, 2017).

Figura 3

Niveles de la automatización industrial



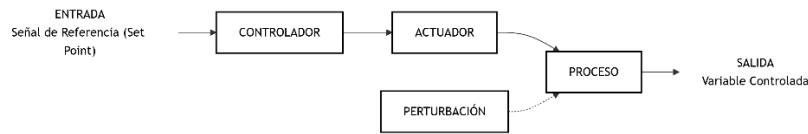
Sistemas de control: lazo abierto y lazo cerrado

Principios de funcionamiento del control en lazo abierto y lazo cerrado

Villamarín Reinoso (2024) resume que el sistema de control en lazo abierto de la Figura 4 ejecuta una acción de control predeterminada sin considerar el resultado real obtenido. Su operación es secuencial y no mide la variable controlada para corregir posibles desviaciones. Es simple y económico, pero su precisión depende de la calibración y es vulnerable a perturbaciones externas.

Figura 4

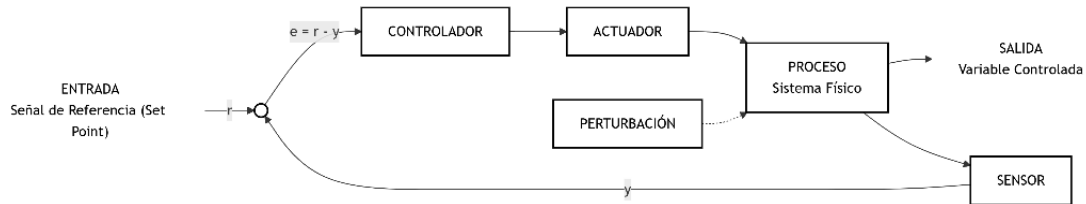
Diagrama de bloques de un sistema de control en lazo abierto



Por el contrario, el sistema de control en lazo cerrado (o de realimentación) de la Figura 5 sí mide continuamente la variable controlada, la compara con un valor deseado o set-point, y utiliza la señal de error (diferencia entre ambos) para calcular y aplicar una acción correctiva sobre el proceso. Este ciclo de medición-comparación-acción se repite constantemente, haciendo al sistema más preciso, estable y capaz de rechazar perturbaciones.

Figura 5

Diagrama de bloques de un sistema de control en lazo cerrado



Instrumentación industrial y normativas

Según Cuásquer Gualoto et al. (2025), la instrumentación industrial es el conjunto de dispositivos que permiten medir, transmitir, indicar, registrar y controlar variables de proceso (presión, temperatura, nivel, flujo, etc.). Los componentes clave son: el transductor/sensor (convierte la variable física en una señal), el transmisor (acondiciona y envía la señal a un formato estándar, como 4-20 mA o digital Fieldbus), los controladores (PLC, DCS, controladores dedicados), y los elementos finales de control (válvulas, motores, cilindros, etc.).

Las normativas más relevantes se encuentran detalladas en la Tabla 2:

Tabla 2

Normativas Clave en Automatización Industrial

NORMA	ÁMBITO DE APLICACIÓN	IMPORTANCIA PRINCIPAL
ISA-88 (S88)	Control de procesos por lotes	Establece un modelo estructural y terminología estandarizada para la automatización de procesos por lotes, permitiendo la reutilización de software y facilitando la integración entre equipos de diferentes fabricantes (International Society of Automation, n.d.-a).
ISA-95 (S95) / IEC 62264	Integración de sistemas empresariales y de control	Define un modelo de datos e interfaces común entre los sistemas de planificación empresarial (nivel 4, ERP) y los sistemas de control de producción (nivel 3, MES y nivel 2, SCADA), optimizando el flujo de información (International Society of Automation, n.d.-b).
ISA-5.1	Simbología y diagramas de instrumentación	Establece una notación gráfica uniforme para representar instrumentos, lazos de control y sistemas en diagramas de ingeniería (P&ID), siendo fundamental para la comunicación clara en proyectos de automatización (International Society of Automation, n.d.-c).
IEC 61131-3	Programación de Controladores Lógicos Programables (PLC)	Estandariza los lenguajes de programación para PLCs (Diagrama Escalera, Texto Estructurado, Diagrama de Bloques, etc.), aumentando la portabilidad del código y reduciendo la curva de aprendizaje para ingenieros (International Electrotechnical Commission, 2013).
IEC 61508 / IEC 61511	Seguridad funcional de sistemas eléctricos/electrónica	Proporciona el marco para implementar Sistemas Instrumentados de Seguridad (SIS). Define el Nivel de Integridad de Seguridad (SIL) para especificar y validar el rendimiento de seguridad requerido, siendo crítica en procesos de alto riesgo (International Electrotechnical Commission, 2010).

NORMA	ÁMBITO DE APLICACIÓN	IMPORTANCIA PRINCIPAL
	cos/ programables (E/E/PE)	
IEC 62443	Seguridad de la automatización y sistemas de control industrial (IACS)	Es una serie de normas que aborda la ciberseguridad integral (organizacional, de proceso y técnica) para proteger las redes y sistemas industriales de amenazas cibernéticas, esencial en la era de la Industria 4.0 (International Society of Automation, n.d.-d).

Fuente: International Electrotechnical Commission (2013), International Society of Automation (2009)

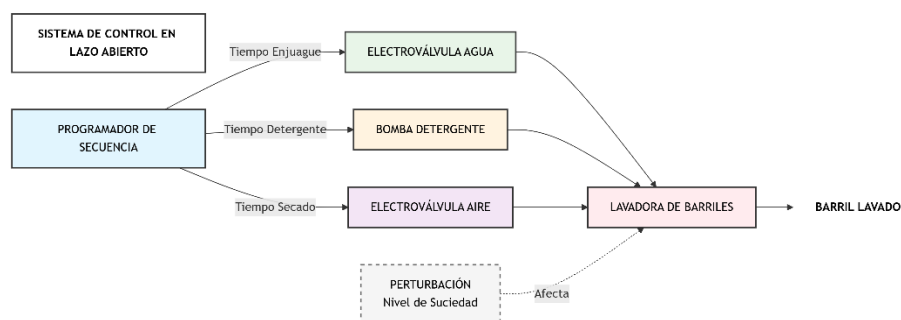
Aplicaciones industriales de sistemas de control en lazos abiertos y cerrados

- Proceso de lavado de barriles

El sistema de lavado industrial ilustrado en la Figura 6 representa una aplicación típica de control en lazo abierto. Se asume que la suciedad de los barriles es relativamente uniforme y que los tiempos preprogramados son suficientes para la mayoría de los casos. La perturbación principal (nivel de suciedad inicial) no se compensa automáticamente, lo que puede resultar en barriles insuficientemente lavados o en desperdicio de recursos (si los barriles están más limpios de lo normal).

Figura 6

Sistema de control de lavado de barriles en lazo abierto

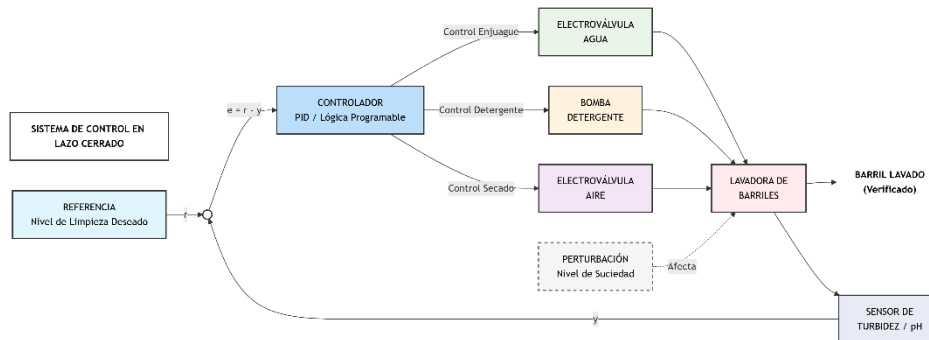


Por su parte, el sistema en lazo cerrado de la Figura 7 implementa un ciclo de medición-corrección continuo. El sensor de turbidez/pH proporciona información en tiempo real sobre la efectividad del lavado, permitiendo al controlador extender o acortar cada fase según sea necesario. Esto resulta en: (1)

menor consumo de agua y productos químicos, (2) calidad consistente del lavado, y (3) capacidad de compensar perturbaciones como variaciones en presión de agua o concentración de detergente.

Figura 7

Sistema de control de lavado de barriles en lazo cerrado

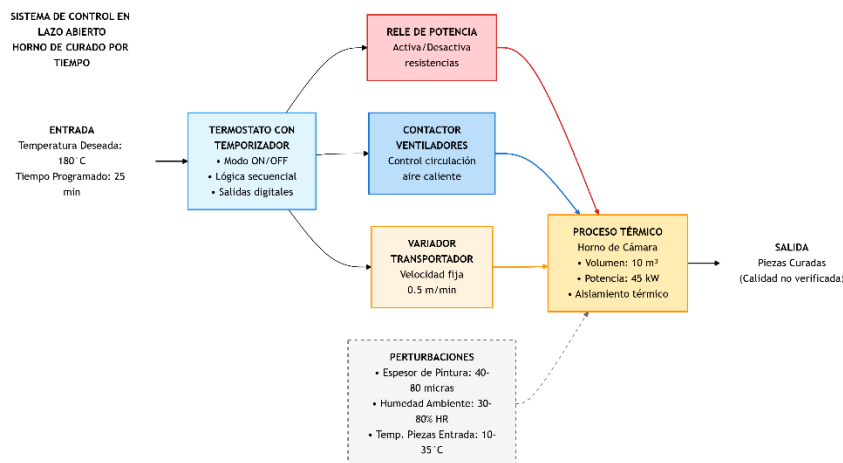


- Sistema de Secado por Tiempo en Horno de Curado

En la industria de pintura y recubrimientos, un horno de curado por convección opera frecuentemente en lazo abierto para el secado de piezas metálicas. El sistema calienta piezas pintadas a una temperatura constante durante un tiempo preestablecido, sin medir la humedad residual o el grado de curado real. La Figura 8 ilustra un ejemplo con datos reales.

Figura 8

Diagrama de bloques del sistema de secado en horno de curado operando en lazo abierto.



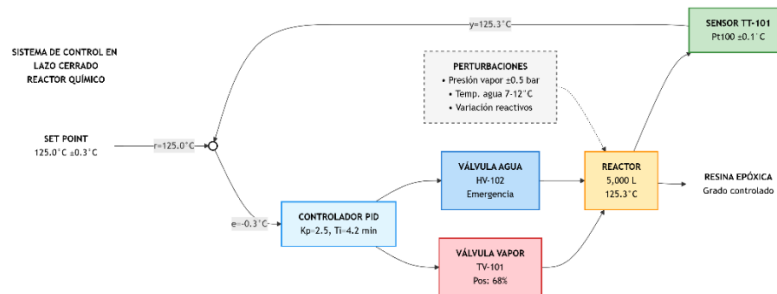
- Reactor químico para producción de resinas

El reactor químico de resinas epóxicas representa un caso paradigmático donde el control en lazo cerrado (Figura 9) no es una opción, sino una necesidad técnica y económica. La naturaleza exotérmica

de la reacción, combinada con especificaciones estrictas de calidad del producto, requiere un control preciso que solo puede lograrse mediante realimentación continua.

Figura 9

Control de temperatura en reactor químico para producción de resinas en lazo cerrado



EJERCICIOS DE APLICACIÓN

1. En un sistema automatizado de estacionamiento que guía vehículos a lugares vacíos:
 - a) Identifique tres componentes del sistema y clasifíquelos como sensores, actuadores o controladores.
 - b) Describa cómo interactúan estos componentes en una secuencia básica.
2. Una panadería artesanal produce 500 panes/día manualmente. Considera automatizar con una máquina que cuesta \$20,000 y reduciría el personal de 4 a 1 persona.
 - a) Calcule el tiempo de recuperación de la inversión si cada trabajador gana \$15/día.
 - b) Mencione dos criterios de la sección Ventajas, desventajas y criterios de aplicación de la automatización en procesos industriales que justificarían esta automatización.
3. En una planta embotelladora, el gerente revisa reportes de eficiencia mensual.
 - a) Identifique en qué nivel de la pirámide de automatización se encuentra esta actividad.
 - b) Describa dos funciones que realizaría un sistema SCADA en esta planta.
4. Compare un ventilador con interruptor manual (ON/OFF) con un ventilador con control de velocidad automático.
 - a) Dibuje el diagrama de bloques de cada sistema.
 - b) Señale la diferencia fundamental entre ambos esquemas de control.
5. Un sistema de dosificación por volumen en una envasadora de aceite opera en lazo abierto.
 - a) Identifique dos posibles perturbaciones que afectarían la precisión del volumen envasado.
 - b) Proponga una modificación para convertir el sistema a lazo cerrado.

UNIDAD 2: CONTROLADOR LÓGICO PROGRAMABLE (PLC) - INTRODUCCIÓN Y LÓGICA DE PROGRAMACIÓN EN LADDER

Evolución y fundamentos del PLC

El Controlador Lógico Programable (PLC) es básicamente una computadora de tipo industrial diseñada para el control de maquinaria y procesos productivos, surgió a finales de la década de 1960 como una solución para sustituir los complejos sistemas de control basados en lógica de relés, los cuales eran costosos, ocupaban mucho espacio y carecían de flexibilidad para modificaciones. El primer autómata comercial, el MODICON 084, fue introducido en 1969 por Bedford Associates, marcando el inicio de la automatización de procesos secuenciales mediante tecnología programada. A diferencia de las computadoras convencionales, los PLC están diseñados con hardware robusto para soportar condiciones industriales adversas como vibraciones, ruido eléctrico y temperaturas extremas.

Para sintetizar este desarrollo histórico y sus bases operativas, la siguiente tabla resume los hitos y ventajas que definieron su adopción en la industria:

Tabla 3

Hito Histórico de la automatización

Fundamento	Impacto en la Industria
Origen (1968-1969)	Concebidos para eliminar los costos de los sistemas de relés; el primer PLC comercial fue el MODICON 084.
Confiabilidad	Diseño robusto capaz de operar en ambientes hostiles sin personal especializado constante
Flexibilidad	Capacidad de reprogramación y reutilización inmediata para distintos procesos
Tiempo Real	Ejecución de ciclos de control breves que permiten una reacción rápida ante eventos del proceso secuenciales en ambientes industriales.
Ventajas de Diseño	Alta confiabilidad, flexibilidad ante cambios de programa, funciones de diagnóstico y resistencia a condiciones adversas (vibración, temperatura, ruido).

Además del control secuencial, los PLC modernos gestionan lazos regulados (PID) y comunicaciones en red (Ethernet/PROFINET).

Como se observa, estos fundamentos permitieron que el PLC se consolidara como el cerebro de la automatización industrial, ofreciendo una plataforma capaz de evolucionar desde tareas binarias simples hasta el control de lazos regulados complejos.

Arquitectura interna de un PLC

La arquitectura de un PLC sigue un modelo tipo Harvard, donde la memoria de programa y la de datos están físicamente separadas, permitiendo que el procesador gestione las instrucciones y las variables de forma independiente. El corazón del sistema es el microprocesador (CPU), que se encarga de ejecutar la lógica de usuario, realizar autodiagnósticos (Autotest) y vigilar el tiempo de ejecución mediante el Watchdog. Este procesamiento se apoya en memorias ROM para el sistema operativo y memorias RAM (con respaldo de batería) para el programa del usuario y el almacenamiento de datos críticos.

La interacción entre estos componentes internos se organiza en la tabla 4 según la siguiente estructura funcional:

Tabla 4

Componentes de un autómata

Componente	Función y Detalles Técnicos
CPU (Procesador)	Es el cerebro que ejecuta el programa de usuario, vigila el tiempo de ejecución, crea la imagen de entradas y actualiza salidas.
Memoria	Se divide en ROM (firmware y sistema operativo) y RAM (programa del usuario, variables internas y memoria buffer para E/S).
Fuente Alimentación	Proporciona las tensiones necesarias para el funcionamiento de la CPU y los módulos (típicamente 24 VCD).
Bus Datos/Direcciones	Transporta la información y los datos entre la CPU, la memoria y las interfaces de entrada/salida.

Interfaces de E/S Circuitos que aíslan eléctricamente la CPU de las señales de campo mediante optoacopladores.

Este diseño modular interno garantiza que la información fluya a través de los buses de datos y direcciones de manera eficiente, permitiendo que las interfaces de entrada y salida se comuniquen con la CPU bajo altos estándares de seguridad eléctrico.

Tipos de PLC (Compactos, modulares, distribuidos)

La constitución física de los autómatas varía considerablemente para adaptarse a las necesidades de escala y flexibilidad de cada planta. Los PLC compactos integran en un solo bloque la fuente, la CPU y una cantidad fija de entradas y salidas, siendo la opción más económica para aplicaciones sencillas. Por el contrario, los PLC modulares ofrecen una estructura basada en un bastidor o rack, donde se pueden añadir módulos de comunicación, control PID o entradas especiales según los requerimientos del sistema. Finalmente, en sistemas de gran envergadura, se utilizan PLC distribuidos, donde la inteligencia se descentraliza mediante periféricos remotos conectados por buses de campo.

A continuación, en la tabla 5 se comparan estas configuraciones según su aplicación y capacidad de expansión:

Tabla 5

Tipo de estructuras de un autómata

Tipo de PLC	Características Principales	Aplicación Típica
Compacto	Bloque único, costo reducido, potencia de Maquinaria pequeña y cuadros de mando proceso limitada.	simples.
Modular	Alta flexibilidad, permite sustituir o ampliar módulos de E/S y comunicación.	Procesos de gama alta con gran número de señales.
Distribuido	Basado en una red de procesadores independientes comunicados entre sí.	Instalaciones geográficamente dispersas (oleoductos, refinerías).

La elección entre estas estructuras depende directamente de la variabilidad del proceso y la necesidad de futuras ampliaciones, permitiendo a las empresas optimizar costos desde la puesta en marcha.

Entradas y salidas digitales/analógicas

El PLC interactúa con el mundo físico mediante módulos de interfaz que captan y emiten dos tipos de señales: discretas y continuas. Las entradas digitales reciben señales "todo o nada" de dispositivos como pulsadores o sensores de proximidad (NPN/PNP), mientras que las salidas digitales activan actuadores por ejemplo como electroválvulas o motores. Por otro lado, las señales analógicas representan variables físicas continuas por ejemplo (temperatura, presión) mediante rangos estándar de tensión (0-10V) o corriente (4-20mA). Para procesar estos datos, el PLC utiliza convertidores ADC para digitalizar las lecturas y DAC para generar salidas de control proporcional.

La siguiente tabla detalla la naturaleza de estas señales y su manejo dentro del controlador:

Tabla 6

Tipos de señales

Tipo de Señal	Naturaleza y Componentes	Dispositivos de Ejemplo
Digital (discreta)	Señales todo-nada que solo adoptan dos estados (0V o +24V).	Pulsadores, interruptores de límite, sensores fotoeléctricos, válvulas solenoide.
Analógica	Señales continuas que pueden adoptar cualquier valor dentro de un rango (0-10V, 4-20mA).	Sensores de temperatura (RTD), caudalímetros y potenciómetros.
Conversión (ADC/DAC)	El PLC utiliza convertidores ADC para digitalizar entradas y DAC para generar salidas continuas proporcionales.	Escalamiento de valores (ej. 0-27648) para a unidades de ingeniería.

El ciclo de escaneo (Scan cycle)

El funcionamiento operativo del PLC se basa en una ejecución cíclica e indefinida denominada ciclo de scan, la cual garantiza el control en tiempo real. Este proceso comienza con la lectura de las entradas físicas para crear una imagen de los sensores en la memoria interna; posteriormente, la CPU ejecuta el programa de usuario instrucción por instrucción y, finalmente, realiza el refresco de las salidas hacia los actuadores. Un factor crítico es que los cambios en los sensores ocurridos durante la ejecución del programa no tendrán efecto hasta el siguiente ciclo de scan.

Para comprender la secuencia lógica de este proceso, se presenta la tabla 7 con el siguiente desglose de fases

Tabla 7

Conceptos Críticos de Operación del PLC

Concepto	Función Técnica	Puntos Críticos
Ciclo de Scan	Conjunto de tareas ejecutadas de forma cíclica e indefinida; incluye la lectura de entradas físicas, ejecución de instrucciones de usuario y refresco de salidas.	El tiempo de respuesta total depende de la longitud del programa; los cambios en los sensores no afectan al programa hasta el siguiente bloque de refresco.
Mapeo y Direccionamiento de Memoria	Organización y asignación de direcciones únicas en la memoria (áreas IR, SR, DM, etc.) para vincular el hardware de campo con la lógica del programa.	Es indispensable utilizar la sintaxis de direccionamiento correcta para cada área (bits o canales de 16 bits) para que la CPU interactúe con los datos previstos.
Watchdog (Perro guardián)	Temporizador de seguridad crítica diseñado para detectar bloqueos del procesador o bucles infinitos en el algoritmo de control.	El temporizador se reinicia al inicio de cada scan; si no se borra antes de que expire su tiempo (típicamente 100 ms), el sistema se detiene por seguridad.

La programación de los Controladores Lógicos Programables (PLC) representa la fase donde se define la inteligencia de un sistema automatizado, sustituyendo la rigidez de la lógica cableada por algoritmos flexibles almacenados en la memoria del procesador. Este proceso consiste en diseñar instrucciones lógicas y de conmutación para controlar, en tiempo real, maquinaria y procesos industriales basándose en el estado de las señales de entrada provenientes de sensores. Se emplean lenguajes estandarizados bajo la norma IEC 61131-3, que es un estándar internacional cuyo objetivo es establecer una sintaxis de programación común que permita desarrollar automatismos sin depender de la sintaxis específica de un fabricante. Esta normativa estandariza diversos lenguajes gráficos y textuales para controladores lógicos programables (PLC), como se detalla en la tabla 8, de esta manera se garantiza que los programas sean más fáciles de mantener y potencialmente portables entre diferentes marcas de equipos industriales.

Tabla 8

Lenguajes de Programación (Norma IEC 61131-3)

Tipo	Lenguaje	Características y Uso
	Gráfico Ladder (LD)	Basado en esquemas eléctricos; intuitivo para técnicos que conocen lógica de relés.
	Gráfico SFC (GRAFCET)	Representación de alto nivel que utiliza etapas y transiciones para procesos secuenciales.
	Gráfico FBD	Utiliza bloques de funciones lógicas interconectados para el flujo de señales.
Textual	ST Estructurado)	(Texto Lenguaje de alto nivel similar a C o Pascal, ideal para cálculos complejos.
Textual	IL (Lista de Instrucciones)	Lenguaje de bajo nivel similar al ensamblador; potente pero menos autoexplicativo.

Lenguaje Ladder (Diagrama de Escalera) según IEC 61131-3

El lenguaje Ladder es un lenguaje de programación gráfico estandarizado, cuya popularidad radica en su similitud con los esquemas eléctricos de control clásicos basados en relés. La esencia del Ladder es la continuidad lógica. Gráficamente, se compone de dos líneas verticales (rieles) que representan la tensión y la masa; las instrucciones lógicas se ubican en líneas horizontales denominadas escalones o peldaños. La energía fluye desde el riel izquierdo hacia el derecho solo si los contactos intermedios permiten una trayectoria cerrada hacia la bobina o instrucción de salida. El PLC procesa diferentes tipos de información dependiendo de la naturaleza del sensor o actuador conectado.

Contactos, bobinas y lógicas combinacionales

Las funciones de los contactos, las bobinas y los principios de las lógicas combinacionales en el entorno de la automatización industrial y la programación en Ladder, se presenta la tabla 9 comparativa de los elementos representativos de este lenguaje gráfico, el cual se basa en esquemas eléctricos de control clásicos donde la energía fluye de izquierda a derecha.

Tabla 9

Elementos representativos de este lenguaje gráfico

Nombre del Elemento	Descripción Representativa	Ejemplo de Aplicación
Contacto Normalmente Abierto (NA)	Representa un interruptor que se cierra únicamente cuando la variable asociada tiene un valor lógico de "1".	Pulsador de inicio (Start) de una banda transportadora.
Contacto Normalmente Cerrado (NC)	Representa un interruptor que permanece cerrado en reposo y se abre cuando la variable asociada es "1".	Botón de parada de emergencia o sensores de seguridad.
Bobina de Salida (Simple)	Activa un bit de salida o una variable interna cuando existe continuidad lógica en el escalón.	Activación de una electroválvula para expandir un cilindro.

Bobinas Set (S) y Reset (R)	Elementos de enclavamiento que retienen su estado lógico, aunque la condición de entrada desaparezca.	Mantenimiento del estado de marcha de un motor tras un pulso momentáneo.
--------------------------------	---	--

• **Lógicas Combinacionales:** Estos sistemas carecen de memoria y de retroalimentación; si las entradas cambian, la salida se actualiza inmediatamente sin considerar estados anteriores. Para implementarlas en PLC, se utiliza la lógica de contactos donde la conexión en serie implementa la función AND y la conexión en paralelo la función OR

• **Continuidad Lógica:** En el diagrama Ladder, se considera que la "energía" se desplaza de izquierda a derecha; para activar una bobina, debe haber un camino cerrado de contactos desde el riel de tensión.

• **Regla de Oro de la Bobina:** En una red o escalón (rung), no puede haber contactos a la derecha de la bobina; ella debe ser siempre el último elemento.

• **Diferencia Crítica:** Mientras que en la lógica combinacional la salida se "apaga" si dejas de presionar el botón, en los sistemas secuenciales (que usan bobinas Set/Reset o realimentación) el sistema "recuerda" y mantiene la acción.

• **Direccionamiento:** En equipos como el PLC Mitsubishi o Siemens, cada contacto y bobina debe tener una dirección física (ej. X400 para entrada, Y430 para salida) para saber qué cable real está controlando.

Timers, counters y comparadores.

Los temporizadores (timers) y contadores (counters) permiten gestionar eventos dependientes del tiempo y la cantidad de ocurrencias, siendo esenciales para la lógica secuencial y el control de procesos repetitivos. Por su parte, los comparadores posibilitan la toma de decisiones lógicas mediante la evaluación de valores numéricos, mientras que el tratamiento de señales analógicas permite al PLC trabajar con variables físicas reales como temperatura, presión, nivel o caudal. Para ello, se emplean procesos de normalización y escalado, que transforman los valores digitales internos del PLC en unidades de ingeniería comprensibles para el operador y el sistema de control.

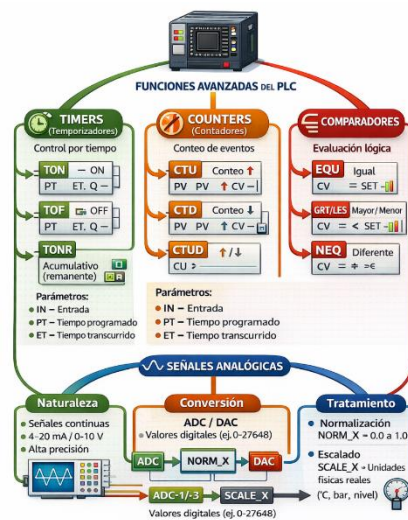
Tabla 10

Características avanzadas del PLC

Nombre del Elemento	Descripción Representativa	Ejemplo de Aplicación
Temporizador al Trabajo (TON)	Bloque de función que retarda la activación de su salida una vez transcurrido un tiempo predefinido.	Retraso de 5 segundos antes de encender una sirena tras detectar un fallo.
Contador Ascendente (CTU)	Incrementa un valor numérico cada vez que detecta un flanco de subida en su entrada de conteo.	Conteo de 24 botellas antes de proceder al embalaje en cajas.
Contactos de Flanco (P / N)	Detectan la transición de 0 a 1 (subida) o de 1 a 0 (bajada) durante un solo ciclo de scan.	Detección exacta del paso de un objeto frente a un sensor inductivo.

Figura 10

Principio de funcionamiento de un contador, temporizador y comparadores



EJERCICIOS DE APLICACIÓN

Para que este sistema funcione en un entorno real (usando lenguaje Ladder o FBD), los bloques operarían de la siguiente manera:

- **Contador (CTU - Count Up):**
- **Entrada (CU):** Sensor fotoeléctrico (pulso cada vez que pasa una botella).
- **Valor de Preselección (PV):** 24.
- **Salida (Q):** Se activa al llegar a 24, disparando la lógica de "Lote Completo".

Temporizador (TON - Timer On Delay):

- **Entrada (IN):** Se activa cuando el contador llega a 24.
- **Tiempo Programado (PT):** 5 segundos.
- **Función:** Mantiene la banda detenida y permite que el actuador neumático trabaje sin movimiento en la línea.

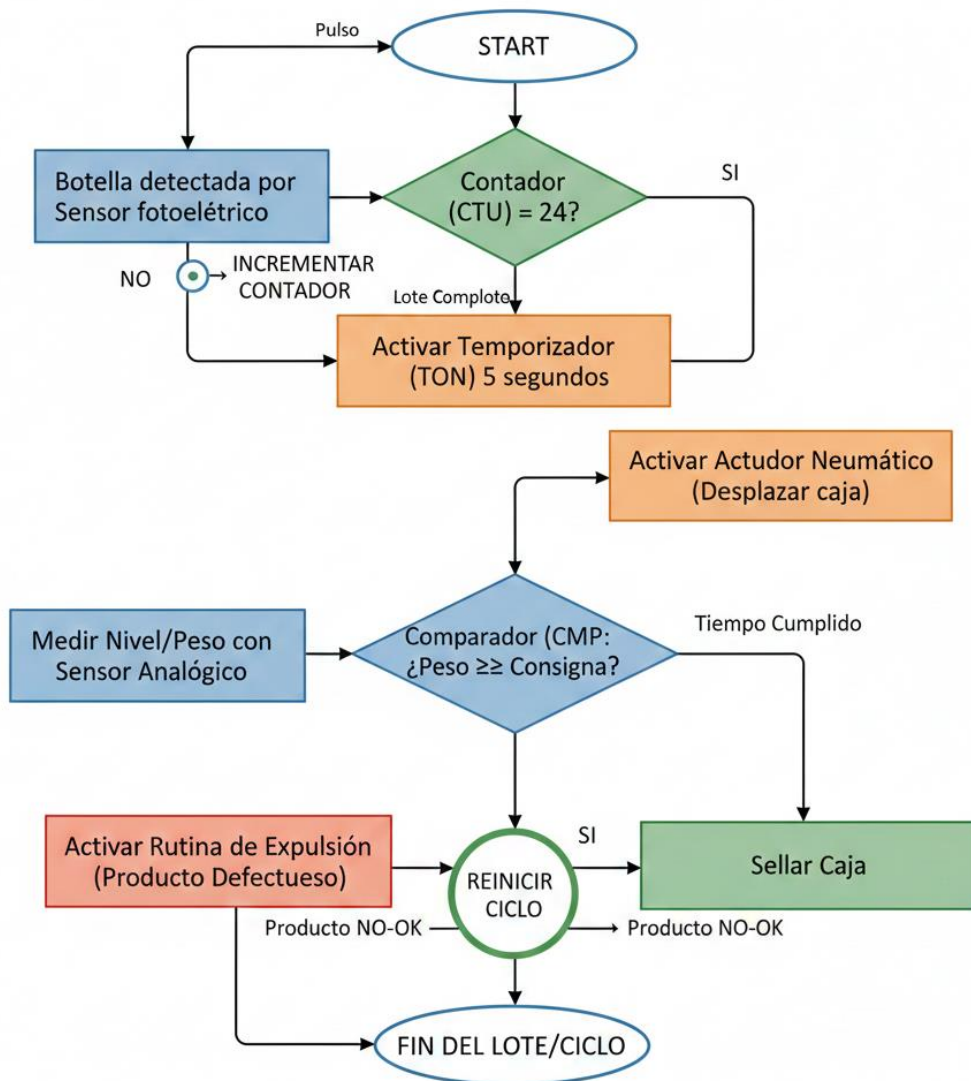
Comparador (CMP - GEQ / Greater Than or Equal):

- **Entrada A:** Señal analógica del sensor de peso (ej. 0-10V convertidos a gramos).
- **Entrada B:** Valor de consigna (ej. 500g).
- **Resultado:** Si $A < B$, se activa el pistón de rechazo.

<u>Componente</u>	<u>Acción Industrial</u>	<u>Condición de Activación</u>
<u>Contador</u>	<u>Conteo de unidades</u>	<u>Sensor de presencia = Flanco de subida</u>
<u>Comparador</u>	<u>Control de Calidad</u>	<u>Peso medido < Peso objetivo</u>
<u>Temporizador</u>	<u>Pausa de seguridad/traslado</u>	<u>Conteo = 24 unidades</u>

Figura 11

Diagrama de flujo del comportamiento del sistema en mención.



UNIDAD 3: PROGRAMACIÓN SECUENCIAL CON GRAFCET

Conceptos Fundamentales de GRAFCET

GRAFCET (Gráfico Funcional de Control de Etapas/Transiciones) es un lenguaje gráfico normalizado por la norma IEC 60848, utilizado para describir de manera estructurada el funcionamiento secuencial de sistemas automatizados. Su principal ventaja radica en que permite representar el comportamiento funcional del sistema de forma independiente al lenguaje de programación del PLC. (IEC, 2013)

Un sistema secuencial se caracteriza porque su comportamiento no depende únicamente de las entradas actuales, sino también del estado previo del sistema. El GRAFCET aborda esta característica mediante una representación basada en etapas y transiciones, permitiendo modelar procesos donde las operaciones se ejecutan en un orden determinado y bajo condiciones específicas.

Desde el punto de vista pedagógico, el GRAFCET es especialmente valioso en la formación en Automatización Industrial, ya que permite al estudiante desarrollar habilidades de análisis lógico, secuenciación de procesos y diseño estructurado de sistemas de control. Además, actúa como un puente conceptual entre el análisis del proceso y lenguajes de programación industrial como LADDER, FBD o Structured Text. (Bolton, 2015)

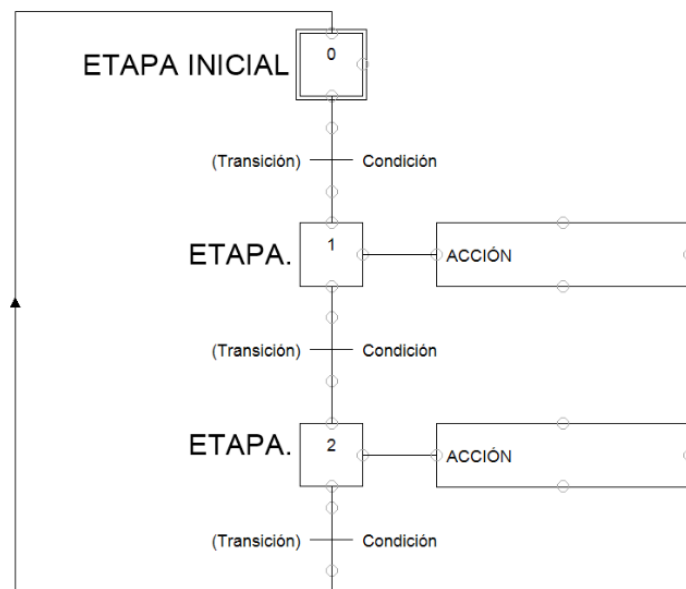
Elementos básicos de un GRAFCET

El GRAFCET (*Grphe Fonctionnel de Commande Étape/Transition*) es un lenguaje gráfico formal utilizado para la especificación funcional de sistemas secuenciales automatizados. Su estructura se basa en tres elementos fundamentales: etapas, acciones, transiciones y receptividades (condiciones), los cuales permiten describir de forma clara y ordenada la evolución de un proceso industrial en función del tiempo y de los eventos del sistema (IEC, 2013).

En la figura 12 se puede apreciar con detenimiento estas definiciones sobre GRAFCET de manera general.

Figura 12.

Estructura básica de un GRAFCET considerando las partes fundamentales



- Etapas

Las etapas representan los estados discretos del sistema automatizado. Cada etapa define una situación estable del proceso durante la cual se ejecutan una o varias acciones asociadas. Desde el punto de vista funcional, una etapa indica qué operación o conjunto de operaciones está realizando el sistema en un momento determinado (Petruzella, 2017)

Gráficamente, las etapas se representan mediante rectángulos, generalmente numerados para facilitar su identificación. Una etapa puede encontrarse en dos estados posibles: activa o inactiva. Cuando una etapa está activa, las acciones asociadas a ella se ejecutan; cuando está inactiva, dichas acciones permanecen deshabilitadas (IEC, 2013)

Dentro de un GRAFCET siempre existe al menos una etapa inicial, la cual representa el estado del sistema al momento de su puesta en marcha. Esta etapa se identifica mediante un doble contorno, indicando el punto de inicio del proceso secuencial. La correcta definición de la etapa inicial es fundamental para garantizar un arranque controlado y seguro del sistema automatizado (Bolton, 2015).

- Acciones

Las acciones se representan gráficamente mediante bloques rectangulares conectados a las etapas correspondiente como se aprecia en la figura 12, lo que permite identificar de manera clara qué operaciones se ejecutan en cada fase del proceso. Esta representación facilita la lectura del diagrama y

contribuye a una documentación estructurada del comportamiento funcional del sistema. (Bolton, 2015).

Desde el punto de vista lógico, las acciones pueden ser de tipo continuo, ejecutándose durante todo el tiempo en que la etapa está activa, o de tipo condicionado, dependiendo de variables adicionales o receptividades internas. Asimismo, el GRAFCET contempla diferentes calificadores de acción, tales como:

- Acción normal (N): se ejecuta mientras la etapa esté activa.
- Acción temporizada (D): se ejecuta tras un retardo determinado.
- Acción impulsional (P): se ejecuta durante un breve instante al activarse la etapa.
- Acción mantenida o memorizada (S/R): permite activar o desactivar salidas mediante órdenes de set y reset.

Estos calificadores amplían la capacidad expresiva del GRAFCET y permiten modelar comportamientos complejos sin comprometer la claridad del diagrama

- Transiciones

Las transiciones representan el cambio de una etapa a otra dentro del GRAFCET y permiten la evolución del sistema a lo largo del tiempo. Gráficamente, se representan mediante líneas horizontales situadas entre dos etapas consecutivas. (IEC, 2013)

Una transición no provoca el cambio de estado por sí sola. Para que una transición se dispare, deben cumplirse simultáneamente dos condiciones fundamentales:

- Todas las etapas inmediatamente precedentes a la transición deben estar activas.
- La condición lógica asociada a la transición debe ser verdadera. (Bolton, 2015)

Cuando una transición se dispara, las etapas precedentes se desactivan y las etapas siguientes se activan de manera simultánea. Este principio garantiza una evolución ordenada, coherente y determinista del sistema secuencial, evitando cambios de estado inesperados o no. (Bolton, 2015)

- Condiciones o Receptividades

Las condiciones, también denominadas receptividades, son expresiones lógicas asociadas a las transiciones y determinan el momento en que estas pueden dispararse. Una condición representa un evento o conjunto de eventos que deben cumplirse para permitir el paso de una etapa a la siguiente. (IEC, 2013)

Desde el punto de vista lógico, las condiciones se expresan mediante variables booleanas que pueden combinarse utilizando operadores lógicos como AND (Y), OR (O) y NOT (NO). Estas variables suelen corresponder a señales provenientes de sensores, pulsadores, finales de carrera, temporizadores, contadores o estados internos del sistema. (Petruzella, 2017)

Es importante destacar que una condición solo tiene efecto si las etapas precedentes están activas. Esto significa que, aunque una condición sea verdadera, la transición no se disparará si el sistema no se encuentra en el estado adecuado. Este principio refuerza el carácter secuencial del GRAFCET y evita comportamientos erráticos. (Bolton, 2015).

Reglas de evolución de GRAFCET

Las reglas de evolución de una GRAFCET dentro de un sistema secuencial se describen a continuación:

- o Regla de activación de las etapas iniciales

Al iniciar el sistema (puesta en marcha o reinicialización), todas las etapas iniciales del GRAFCET se activan simultáneamente. Una etapa activa se representa gráficamente mediante un doble recuadro. Las acciones asociadas a estas etapas se ejecutan inmediatamente.

- o Regla de validada de una transición

Una transición está validada cuando todas las etapas inmediatamente anteriores a ella están activas. Si al menos una de las etapas precedentes no está activa, la transición no puede franquearse, aunque su condición lógica sea verdadera.

- o Regla de franqueamiento (franqueo) de una transición

Una transición se dispara cuando:

- Está sensibilizada
- Su receptividad (condición lógica asociada) es verdadera

El disparo provoca:

- La desactivación simultánea de todas las etapas precedentes
- La activación simultánea de todas las etapas siguientes
- Este cambio es instantáneo y atómico.
 - o Regla de evolución simultánea

Todas las transiciones disparables en un mismo instante se disparan simultáneamente.

El GRAFCET no admite prioridades implícitas; la evolución es paralela y determinista.

Esto garantiza coherencia lógica y evita conflictos de secuencia.

Conversión de GRAFCET a LADDER

Para realizar un proceso de conversión de GRAFCET a LADDER, se presenta un ejercicio que permita clarificar esta metodología y tener como una plantilla de programación.

Para ello, se considerará que, se desea programar el arranque Estrella-Triángulo de un motor trifásico de inducción, donde, se establecerá 2 segundos como intervalo de tiempo para la conmutación de la configuración.

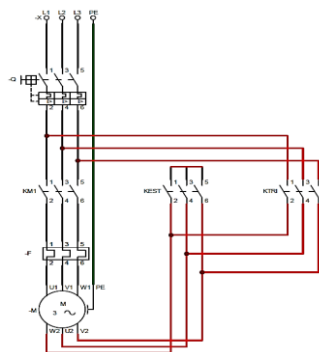
Todo este procedimiento se realizará siempre y cuando se presione el pulsador P1 (Marcha) que es un contacto normalmente abierto y el motor se detendrá al presionar el pulsador P2 (Paro) como un contacto normalmente cerrado.

Diagrama de potencia.

Considerando el ejercicio planteado, el diagrama de potencia que no se debe perder de vista el mostrado en la figura 13.

Figura 13.

Diagrama de potencia de un arranque estrella-triángulo de un motor trifásico



Nota: Elaborado en CADeSIMU

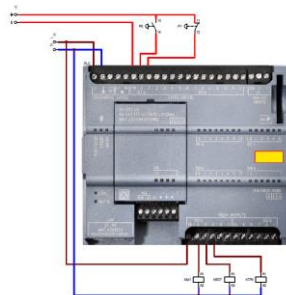
Bajo esta consideración, es importante mencionar que el sistema de potencia del proceso a automatizar no cambia en absolutamente nada si se desea aplicar como metodología de programación a la conversión de GRAFCET A LADDER

Diagrama de IO del PLC

Para tener el contexto completo del ejercicio planteado, se presenta el diagrama de IO del PLC que servirá como elementos del proceso de conversión de GRAFCET a LADDER y se muestra en la figura 14.

Figura 14.

Diagrama de I/O del PLC para el arranque estrella-triángulo desde el PLC



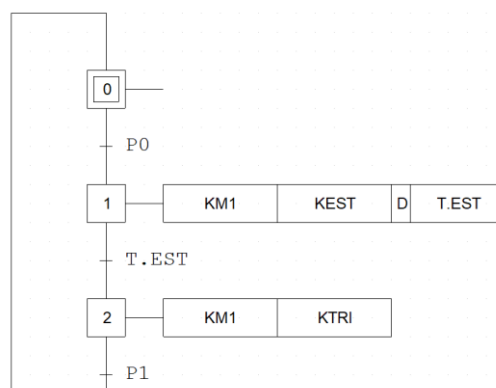
Nota: Elaborado en CADeSIMU

GRAFCET del sistema

El modelado del proceso dentro de la metodología denominada GRAFCET puede mostrar un ejemplo como el que se puede apreciar en la figura 15.

Figura 15.

GRAFCET de un sistema de arranque Estrella-Triángulo



Nota: Elaborado en CADeSIMU

Partiendo del criterio establecido entre, el diagrama de potencia, el diagrama de I/O del PLC y el GRAFCET, se procede a establecer la metodología para la conversión a el lenguaje de programación LADDER.

Metodología para convertir GRAFCET a LADDER

Es necesario considerar los espacios de memoria del PLC que serán definidos de acorde al GRAFCET de la figura 15, donde, por ejemplo, se debe definir:

- Etapas del GRAFCET

Para cumplir con esta definición de variables, se considera que las etapas serán parte de los espacios de memoria RAM donde se albergan las variables aleatorias (Marcas del Sistema)

- o Etapa 0 (X0) – Marca 0 (M0.0)
- o Etapa 1 (X1) – Marca 1 (M0.1)
- o Etapa 2 (X2) – Marca 2 (M0.2)
- Entradas del PLC

Estas variables se necesitan definir en el espacio de memoria de imagen de las entradas del PLC (PII) con la respectiva dirección relativa según la distribución interna de bits del Autómata, por ejemplo.

- o P0 – Pulsador de Marcha (I0.0)
- o P1 – Pulsador de Paro (I0.1)
- Salidas del PLC

Al igual que las entradas, se define estas señales, en la memoria de salida del autómata considerando la distribución de la memoria del equipo, por ejemplo:

- o KM1 – Contactor directo del arranque (Q0.0)
- o KEST – Contactor de la configuración en estrella de los terminales del motor. (Q0.1)
- o KTRI – Contactor de la configuración en triángulo de los terminales del motor. (Q0.2)

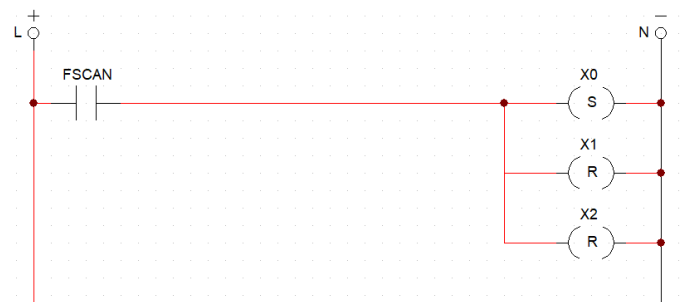
Una vez conocidas las variables del sistema, es necesario que, para el procedimiento conversión, se van a considerar 3 pasos para programar en el PLC, los cuales, de manera teórica y práctica se describen a continuación:

a. Programación de la Inicialización del PLC

En esta sección del código, se debe considerar cumplir con la primera regla de evolución del GRAFCET donde se requiere que; al inicio del proyecto siempre, se debe encontrar activa la etapa 0 y el resto debe estar desactivadas. Esta parte del código se muestra en la figura 16.

Figura 16.

Programación de la inicialización del GRAFCET en el PLC



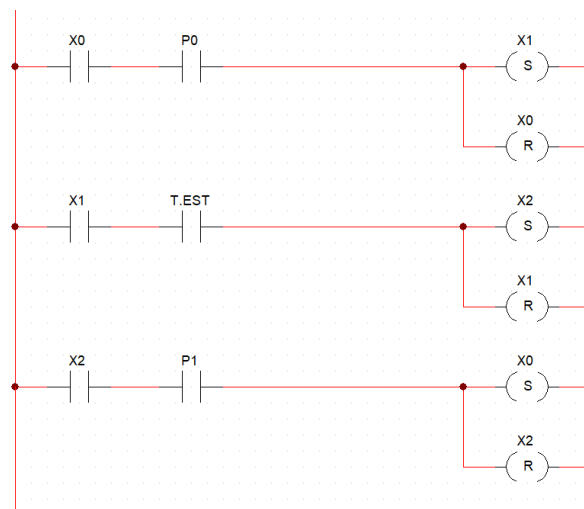
Nota: Elaborado en CADeSIMU – El bit de activación de esta parte del código es el First-Scan en el modo RUN.

b. Programación de las transiciones del PLC

Para esta zona de programación, es necesario considerar las reglas de validación y transición del GRAFCET, para esto, se aprecia en la figura 17 el código que se aplicaría en el arranque del motor estrella-triángulo.

Figura 17.

Programación de las transiciones del GRAFCET en el PLC



Nota: Elaborado en CADeSIMU

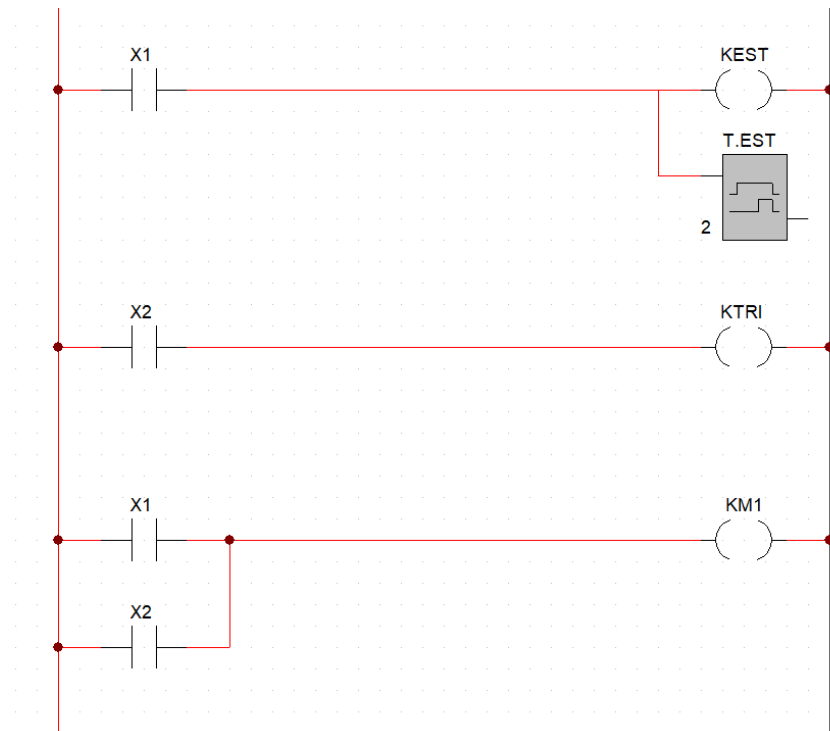
Con esta sección de código, se ha conseguido programar el flujo de evolución del GRAFCET ante las distintas transiciones que se van ejecutando en el proyecto. Es importante notar como en cada transición se evidencia la programación de la *validación* y del *franqueamiento* de la misma.

c. Programación de las acciones del GRAFCET

Para ejecutar las distintas acciones del GRAFCET se programar la dependencia existente en cada bit de memoria que representa una etapa con la acción establecida en cada una. En este fragmento de código intervienen los pre-actuadores del sistema.

Figura 18.

Programación de las acciones del GRAFCET en el PLC



Nota: Elaborado en CADeSIMU

Con este fragmento de código, el proyecto ha sido coficado completamente y manteniendo los 3 pasos que forman parte de una receta de programación a fin de disponer de un método sistemático en la codificación.

EJERCICIOS DE APLICACIÓN

EJERCICIO 1: *Control de una cinta transportadora con estación de taladrado*

Se dispone de una cinta transportadora que lleva piezas hasta una estación de taladrado automático. El sistema funciona de la siguiente manera:

1. El operario pulsa el botón de marcha, y la cinta transportadora comienza a moverse.
2. Cuando una pieza llega a la estación de taladrado, un sensor de presencia detecta la pieza y la cinta se detiene.
3. Un cilindro neumático vertical desciende para realizar el taladrado.
4. Al llegar al final de carrera inferior, el cilindro permanece abajo durante 3 segundos.
5. Finalizado el tiempo, el cilindro asciende hasta el final de carrera superior.
6. Una vez el cilindro está arriba, la cinta transportadora vuelve a ponerse en marcha y el ciclo se repite.
7. Al pulsar el botón de paro, el sistema se detiene de forma segura y vuelve al estado inicial.

EJERCICIO 2: Sistema de llenado de un tanque con dos niveles

Se desea automatizar el llenado y vaciado de un tanque mediante un PLC. El sistema cuenta con una válvula de llenado, una bomba de vaciado, y dos sensores de nivel (mínimo y máximo).

La secuencia de funcionamiento es la siguiente:

1. Al pulsar el botón de inicio, el sistema comienza el llenado del tanque abriendo la válvula de entrada.
2. Cuando el líquido alcanza el nivel máximo, la válvula de llenado se cierra automáticamente.
3. El sistema permanece en reposo durante 5 segundos.
4. Transcurrido ese tiempo, se activa la bomba de vaciado.
5. Cuando el nivel desciende hasta el sensor de nivel mínimo, la bomba se detiene.
6. El sistema queda listo para iniciar un nuevo ciclo.
7. Un botón de paro de emergencia puede detener el proceso en cualquier momento.

UNIDAD 4: INSTRUMENTACIÓN VIRTUAL Y SISTEMAS SCADA

El constante avance de la tecnología en los campos de la electrónica, la informática y las comunicaciones ha transformado de manera significativa los sistemas de automatización industrial, haciéndolos más eficientes y accesibles. Esta evolución ha facilitado una interacción más intuitiva entre los operadores y los sistemas de control, así como una gestión más efectiva de los procesos industriales. En este contexto, la instrumentación virtual aparece como una solución clave para el desarrollo de interfaces Hombre-Máquina (HMI) que permiten supervisar, controlar y monitorear los procesos de forma flexible y amigable. De igual manera, los sistemas SCADA desempeñan un papel fundamental al posibilitar la adquisición y transmisión de datos en tiempo real, contribuyendo a una mejor organización de la información y a una toma de decisiones más oportuna. Por estas razones, en esta unidad se abordarán los fundamentos y aplicaciones de la instrumentación virtual y los sistemas SCADA dentro del ámbito de la automatización industrial.

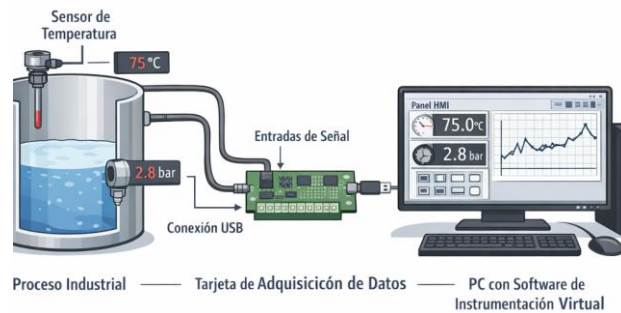
Sistemas de instrumentación virtual

Un instrumento virtual es básicamente un módulo de software diseñado para reproducir el funcionamiento de un instrumento físico real. Para lograrlo, aprovecha todos los dispositivos y sensores que el ordenador puede reconocer y utilizar. Cuando el usuario ejecuta un instrumento virtual, en la pantalla aparece un panel que imita al del instrumento original, permitiendo visualizar las mediciones y controlar su funcionamiento de una manera intuitiva y cercana a la experiencia real. (Manuel et al., 2021).

En la Figura 19 se muestra un ejemplo del funcionamiento de un instrumento virtual. El proceso representado corresponde a un tanque en el que se realiza la medición de temperatura y presión. Para ello, se emplean sensores y transductores que captan las señales físicas y las envían a una tarjeta de adquisición de datos. Estas señales también pueden dirigirse a un microcontrolador o a un PLC. Posteriormente, cualquiera de estos dispositivos transmite la información hacia una pantalla de visualización generalmente una computadora donde se encuentra instalado el software de instrumentación virtual, en el cual se ha diseñado la interfaz gráfica correspondiente.

Figura 19

Ejemplo de instrumento Virtual



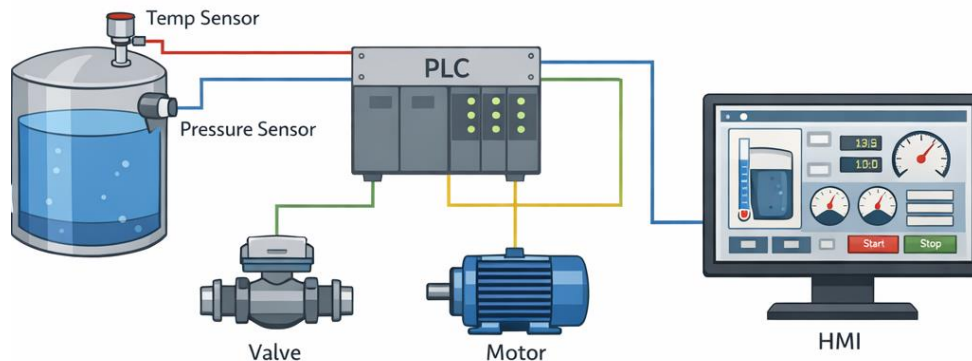
Interfaz Humano Maquina (HMI)

Una vez comprendido el funcionamiento de los instrumentos virtuales y su utilidad para mostrar y analizar datos, surge la necesidad de ir más allá de la simple observación. En escenarios reales, las personas deben interactuar con los sistemas, tomar decisiones y ejecutar acciones en tiempo real. En este contexto aparecen los HMI, que convierten la información del proceso en una interfaz clara y accesible, permitiendo al usuario supervisar y controlar las operaciones de manera directa y sencilla.

En la Figura 20 se observa que, además del instrumento virtual para la medición de temperatura y presión, la interfaz gráfica incluye botones para el encendido y apagado del proceso, así como opciones para configurar las variables de control de velocidad. Esto indica que el HMI no solo permite monitorear el sistema, sino que también proporciona un cierto nivel de control sobre él.

Figura 20

Sistema HMI para medición de temperatura y control de presión



Principios prácticos del diseño de HMI

Una Interfaz Humano-Máquina (HMI) es un sistema que permite visualizar y gestionar de manera gráfica el comportamiento de un proceso industrial automatizado, facilitando la comunicación del usuario con el proceso completo de forma sencilla e interactiva. Según Rodríguez (2021), existen tres principios fundamentales para el desarrollo de una HMI orientados a mitigar errores y optimizar las herramientas y elementos disponibles. Estos principios se presentan en la Tabla 11.

Tabla 11:

Principios para el desarrollo de un HMI

CLARIDAD	COHERENCIA	RETROALIMENTACIÓN
<ul style="list-style-type: none"> - Los gráficos deben ser entendibles y los mensajes deben facilitar la lectura. - Se debe ver claramente el estado y las condiciones del proceso. - Los elementos utilizados para la manipulación del proceso deben ser claramente distinguible. - Las HMIs debe contener solo información importante, que 	<ul style="list-style-type: none"> - Las funciones gráficas deben ser estandarizadas, sencillas y que utilicen funciones de ayuda para la manipulación de las HMI. - La HMI debe establecer un manejo lógico acorde al modelo mental de los operadores del proceso y un desarrollo orientado a las buenas prácticas. 	<ul style="list-style-type: none"> - El diseño de las HMIs debe minimizar la fatiga del usuario. - Las acciones importantes del proceso deben tener mecanismos de confirmación para evitar activaciones inadvertidas. - Los elementos gráficos y objetos de control deben ser desarrollados de acuerdo con el funcionamiento del proceso y a los estándares y normas existentes.

permita determinar el estado del proceso.

- Las alarmas e indicadores de situaciones anormales deben ser claros, resaltados y distinguibles.
-

Normativa ISA 101.

La norma ISA-101 busca que las interfaces HMI sean más claras, intuitivas y amigables para las personas que operan los procesos industriales. Más que centrarse solo en la parte visual, esta norma ayuda a que el operador pueda entender rápidamente lo que está ocurriendo en planta, tomar decisiones con seguridad y evitar errores que podrían afectar la producción. Su enfoque combina buenas prácticas de diseño, organización de la información y consistencia en la navegación, de manera que cada pantalla aporte valor real y no solo datos aislados. En esencia, la ISA-101 promueve HMIs que apoyen al operador, que reduzcan la carga mental y que faciliten la interacción diaria con el sistema, creando entornos más eficientes y confiables. (International Society of Automation, 2015).

Los HMI deben ser diseñados de tal forma que cumplan con estos tres principios básicos_

- Los HMI deben ser una herramienta eficaz para el control seguro y eficiente del proceso.
- Debe ayudar a la detección temprana, el diagnóstico y la respuesta adecuada antes situaciones anormales.
- Debe estar estructurada para ayudar a los operadores a priorizar la respuesta ante fallas mayores o múltiples fallas del sistema.
- La falla de una pantalla o de los elementos en la pantalla debe ser evidente para el operador.

Consideraciones de color

La norma ISA-101 propone una manera más consciente y cuidadosa de usar el color en las interfaces HMI. Su intención es sencilla: ayudar al operador a ver lo verdaderamente importante sin que la pantalla

lo abruma. Por eso sugiere emplear colores solo cuando realmente aporten información valiosa, como indicar una alarma o una condición fuera de lo normal. Los fondos en tonos grises y elementos visuales suaves son una forma de reducir la fatiga visual y permitir que aquello que necesita atención resalte de inmediato. De esta manera, el color deja de ser un adorno y se convierte en una herramienta clara y coherente que guía al operador, mejora su comprensión del proceso y facilita decisiones más rápidas y seguras. Aunque no impone un código cromático rígido, recomienda seguir patrones ampliamente aceptados:

- Rojo_ Indica condición peligrosa o alarma
- Amarillo_ Indica Advertencia
- Verde_ Indica funcionamiento seguro
- Azul_ información auxiliar

Densidad de la información mostrada

En una pantalla HMI es fundamental que el operador encuentre solo la información que realmente necesita y que pueda interpretarla sin esfuerzo. Cuando una pantalla está saturada o desordenada, entender lo que ocurre se vuelve más difícil, por lo que a veces es mejor rediseñarla para mostrar solo lo esencial. Además, el texto y los números deben mantenerse estables: nada debería moverse o parpadear sin una razón de peso, ya que esto distrae y dificulta la lectura. El parpadeo debe reservarse exclusivamente para situaciones que requieren atención inmediata, como una alarma, y siempre debe existir una opción para detener ese parpadeo si el operador lo necesita. En esencia, la HMI debe ser clara, tranquila y enfocada en guiar al usuario sin agobiarlo. (International Society of Automation, 2015).

Animaciones

Para gestionar las animaciones de una manera eficiente se debe considerar los siguientes aspectos:

- El parpadeo se debe usar en símbolos o bordes para la identificación de alarmas no reconocidas.
- No se recomienda usar parpadeo en texto o números debido a que esto dificultaría la lectura.
- No es recomendable que una parte de la pantalla parpadee o destelle a menos que se requiera una acción del operador.
- Colocar un mecanismo para que el operador pueda detener el parpadeo o destello.

Señales auditivas

Para mejor gestión del HMI se recomienda colocar señales auditivas para alertar al usuario sobre situaciones anormales como una alarma, una entrada incorrecta o un fallo del HMI al procesa una entrada, estas señales deben ser diseñadas de tal manera que llamen la atención del operador, pero no que no aumentes los niveles de ruido o impidan la comunicación entre operadores. El significado de cada señal auditiva debe ser claro e inequívoco.

Visualización

En la Tabla 12 se puede observar un ejemplo de las diferentes formar de visualización de una pantalla HMI.

Tabla 12

Estilos de visualización recomendada

TIPO	DESCRIPCION	EJEMPLOS
Lista	Las listas de datos se pueden mezclar con símbolo de equipos de proceso.	Tablas de sujeción de tanques, tabla de límite de operación seguro.
Proceso	Representación gráfica del equipo de procesos, tuberías e instrumentación	Representación de diagrama P&ID
Resumen esquemático	Los tipos de controles e indicadores necesarios dependerán de los requisitos funcionales.	Área de proceso o visiones generales del transporte.
Resumen funcional del tablero	Representación de la relación funcional de los datos	Tipos de datos según se requiera (por ejemplo, indicadores clave de rendimiento)

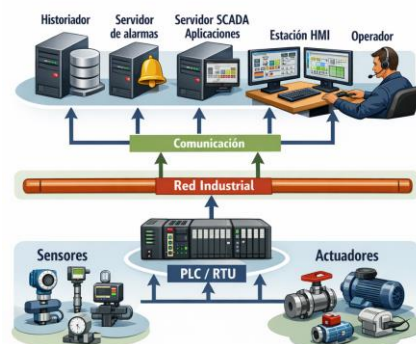
Topología	Representación del diseño lógico del sistema	Sistema de red Sistema eléctrico Diagrama unifilar Estado de detección de incendios
Gráficos	Gráficos de datos en tiempo real o históricos	Tendencia en tiempo real o históricos Gráficos de control de calidad estadístico

Sistemas de Supervisión, Control y Adquisición de Datos SCADA

Un sistema SCADA es una combinación de software y hardware que supervisa, controla y registra en tiempo real los procesos industriales para garantizar que una planta opere de forma segura y eficiente. (Rodríguez, 2013). Este sistema muestra la información en pantallas HMI y puede enviar órdenes automáticas a distintos equipos. En cambio, un HMI es solo la interfaz visual donde el operador ve lo que ocurre y controla una máquina puntual, mientras que el SCADA actúa como el “cerebro” que coordina múltiples máquinas, analiza datos, genera alarmas y permite gestionar toda la instalación desde un centro de control.

Figura 21

Sistema SCADA



Criterios de selección y diseño

➤ **Funcionalidad del sistema.**

Antes de elegir un SCADA, hay que asegurarse de que realmente haga lo que la planta necesita: mostrar datos, guardar registros, lanzar alarmas y permitir controlar equipos. En pocas palabras, debe ser capaz de acompañar y mejorar el trabajo diario, no complicarlo.

➤ **Robustez**

El sistema SCADA debe ser capaz de seguir operando incluso ante fallos, accidentes o intentos de intrusión, permitiendo que sus componentes mantengan funciones básicas de manera autónoma. Si la estación maestra (MTU) deja de funcionar, las estaciones remotas deben activar un protocolo automático de desconexión y continuar operando de forma segura hasta que el sistema se restablezca.

➤ **Seguridad**

Para mantener el SCADA a salvo de usos indebidos, conviene pensar en la seguridad desde el diseño: dar a cada persona solo los permisos que necesita, exigir contraseñas fuertes, cifrar lo que se envía y almacena, filtrar qué información se comparte y usar herramientas que vigilen el sistema y avisen cuando algo raro ocurra.

➤ **Prestaciones**

El sistema debe estar diseñado para que, cuando se genere una alerta, el equipo tenga tiempo suficiente para comprender toda la información recibida. Aunque los datos no puedan procesarse o mostrarse al instante, es esencial que no se pierda ningún detalle, garantizando que todo quede registrado y disponible para su análisis posterior.

➤ **Mantenibilidad**

El sistema debe contar con herramientas de diagnóstico que faciliten el mantenimiento preventivo y permitan hacer ajustes, pruebas o modificaciones mientras el proceso sigue en marcha. De esta manera, se pueden evitar paradas largas e innecesarias y mantener la operación continua sin afectar la producción.

➤ **Escalabilidad**

El sistema debe diseñarse pensando en que pueda crecer con el tiempo, permitiendo incorporar nuevas funciones o herramientas sin dificultades. Para lograrlo, es importante considerar desde el inicio el

espacio disponible, la capacidad del hardware y el rendimiento del sistema informático. Así, la aplicación podrá evolucionar sin limitaciones; en otras palabras, el SCADA debe ser fácil de ampliar y actualizar cuando surjan nuevas necesidades. (Rodríguez, 2013)

Ejercicio de demostración

Implementar un HMI conforme a la normativa ISA-101 para el control y monitoreo de nivel, considerando una jerarquía de usuarios que diferencie las funciones y permisos del operador y del administrador. El HMI deberá presentar la información de forma clara, priorizada y segura, de acuerdo con las buenas prácticas de diseño establecidas por la norma.

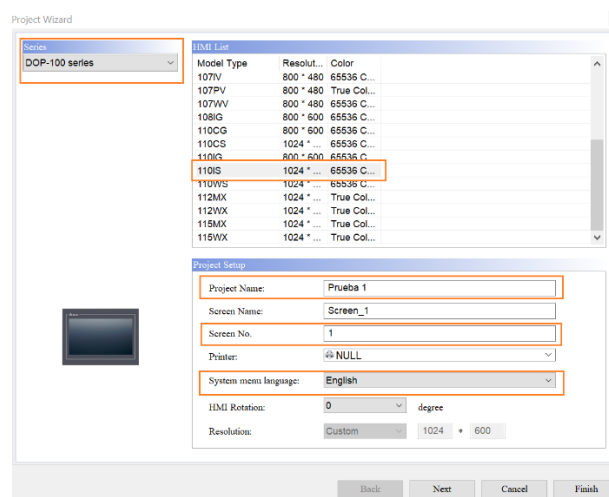
Procedimiento

El desarrollo se realiza utilizando tecnología Delta para aprender de manera práctica cómo diseñar una interfaz HMI industrial, comenzando con la creación de pantallas, botones y variables en el software DOPSoft, lo que servirá como base para comprender la comunicación entre el HMI y el sistema de control, así como la aplicación de buenas prácticas en automatización.

1. Abrimos el software DOPSoft, aquí vamos a seleccionar el tipo de pantalla que vamos a configurar, asignaremos nombre al proyecto, el número de pantalla, y el idioma.

Figura 22

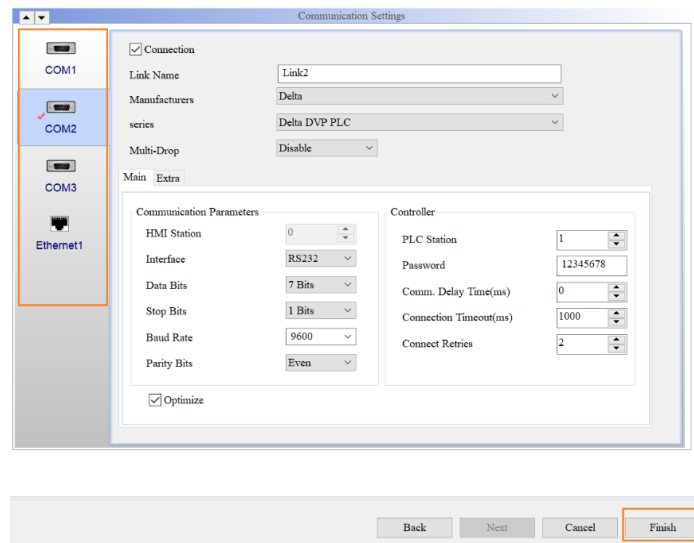
Configuración de nuevo proyecto



2. Una vez configurada la pantalla anterior se dará clic en siguiente, lo que abrirá una nueva pantalla para la configuración de las comunicaciones de la HMI, aquí se debe seleccionar el puerto de comunicación que se desea usar de la pantalla y se dará clic en finalizar.

Figura 23

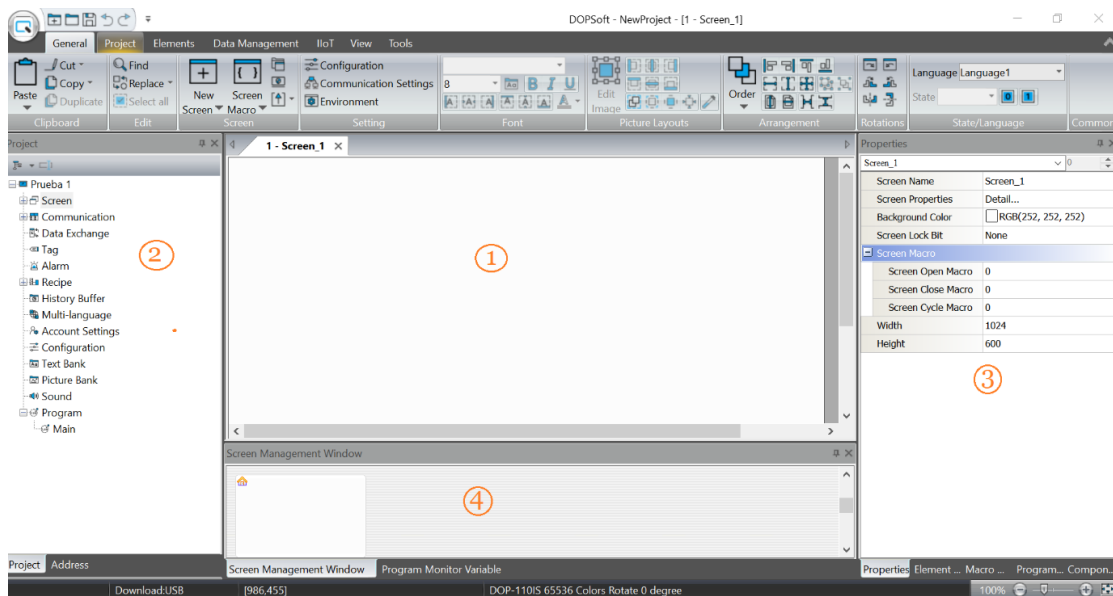
Selección de puerto de comunicación



3. Al finalizar la configuración inicial se abrirá la pantalla de diseño en donde se puede distinguir cuatro sub-pantallas principales:
 1. Pantalla de trabajo
 2. Pantalla de configuraciones
 3. Pantalla de descripción de elementos seleccionados
 4. Distribución de pantallas del HMI

Figura 24

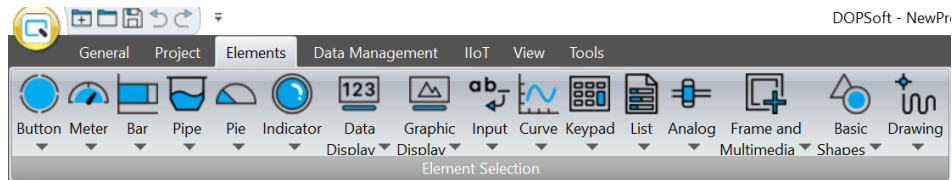
Interfaz del software



- Procedemos con el diseño, para ello en la barra principal, damos clic en elementos y procedemos arrastrar todos los que sean necesarios para la pantalla.

Figura 25

Barra de elementos



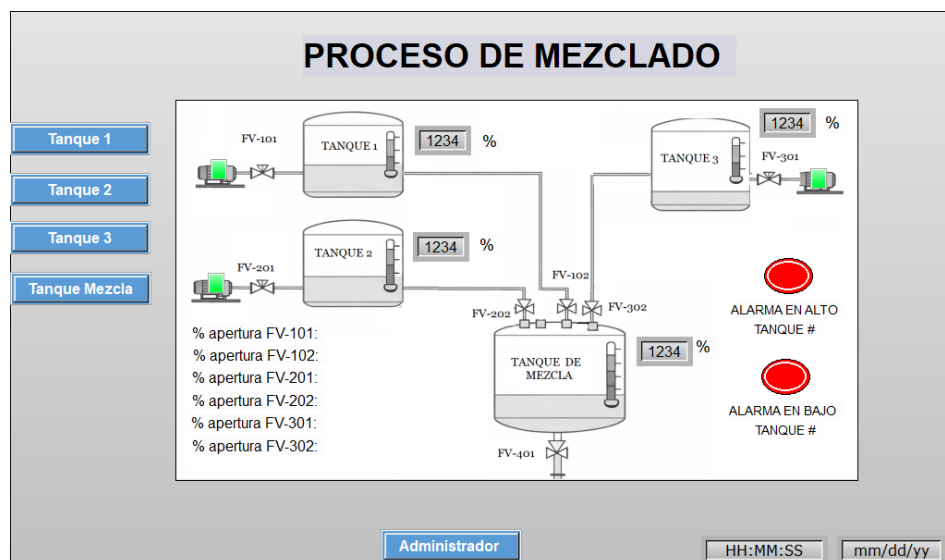
- Se analiza el proceso y en función de las necesidades del usuario se procede a diseñar la pantalla del HMI, para el diseño se respeta la normativa ISA 101, en donde se establecerá 3 niveles de visualización:

Nivel 1: Vista general del proceso

La pantalla muestra una vista general del proceso mediante un diagrama tipo P&ID según la normativa ISA, con botones para acceder al control individual de cada tanque y la fecha y hora para apoyar la operación. Incluye un texto de alarma asociado a luces intermitentes para alertar sobre condiciones anómalas, así como un acceso para administradores que permite, tras validar credenciales, modificar los set points y consultar la base de datos de alarmas.

Figura 26

Pantalla principal



Nivel 2: Vista de unidad o área

En esta pantalla se muestra el proceso de control individual de cada tanque, ofreciendo la opción de arrancar o detener cada operación de forma remota. Se incluye una luz indicadora del estado de la bomba, así como los porcentajes de apertura de las válvulas, y se incorporan alarmas visuales que señalan condiciones de nivel alto o bajo para alertar oportunamente al operador.

Figura 27

Pantalla de área



Nivel 3: Pantalla de detalle tendencia y alarmas específicas.

En la pantalla de tendencias se muestra el historial de alarmas, permitiendo al operador analizar el comportamiento del proceso, así como un gráfico de tendencias que facilita la verificación de la variación de la variable de nivel en cada uno de los tanques.

Figura 28

Área de reportes de una HMI



EJERCICIO DE APLICACIÓN

Diseñar en un HMI para el control de temperatura de un horno eléctrico. La interfaz debe mostrar la temperatura actual, permitir ajustar el set point (SP), visualizar el estado del elemento calefactor (encendido/apagado) y generar alarmas visuales cuando la temperatura exceda los límites de seguridad. Además, deberá integrar una pantalla de tendencias donde se observe la variación de la temperatura en el tiempo.

Nota: solo el Administrador podrá realizar acciones como cambio de setpoint o encendido y apagado de proceso de forma remota.

UNIDAD 5: COMUNICACIONES INDUSTRIALES

Fundamentos y Protocolos de Comunicaciones Industriales

En la automatización industrial, los protocolos de comunicación actúan como el vínculo esencial que interconecta máquinas, sistemas y aplicaciones, facilitando su operación coordinada y eficaz. Estos estándares definen la forma en que los dispositivos comparten datos, un factor clave para asegurar la interoperabilidad, así como el control y la supervisión de los procesos. Dada la extensa gama de protocolos creados para cubrir diversas exigencias, la adecuada elección y aplicación de los mismos resulta fundamental para maximizar el rendimiento de cualquier entorno industrial.

Topologías de una red – medios físicos

En el contexto de las redes, la topología describe la estructura física o lógica de las conexiones entre los dispositivos. Estos sistemas, denominados redes, tienen como función principal permitir la comunicación y el intercambio de datos entre dos o más computadores o microprocesadores. Mientras que el término nodo identifica cualquier punto de la red donde convergen las líneas de comunicación o se conecta un equipo a ellas.

Modelos y topologías de red industrial: bus, estrella, anillo y línea

Las topologías de red más comunes son:

- **Bus de datos (Multidrop)**

Esta topología se caracteriza por un canal de comunicación principal (bus lineal), al que se conectan todas las estaciones de la red. Es una configuración habitual para conectar múltiples terminales en un esquema multipunto y suele ser la opción más adecuada cuando la separación entre los nodos supera los 100 metros. Figura 29.a

- **Estrella**

Esta arquitectura asigna un canal de comunicación directo entre cada estación y un conmutador central en forma de anillo (o hub), que actúa como un nodo obligatorio para todo el tráfico de la red. Es la topología típica de las centrales telefónicas privadas (PBX) empresariales, donde todas las líneas convergen en una central. También es común para integrar terminales locales y remotas con un ordenador central (host). Su principal punto débil es la dependencia del conmutador central: una falla en este anillo provoca el colapso de toda la red. Figura 29.b

- **Jerárquica o de árbol**

En esta topología, múltiples ramas se conectan de manera indirecta hasta converger en un punto central situado en la raíz o cima del árbol. Entre dos estaciones cualesquiera, existe un único camino de comunicación posible. Esta configuración se logra interconectando varios sistemas de bus lineal. Al igual que el bus simple, su uso es recomendable cuando las distancias entre nodos superan los 100 metros. Figura 29.c

- **Anillo**

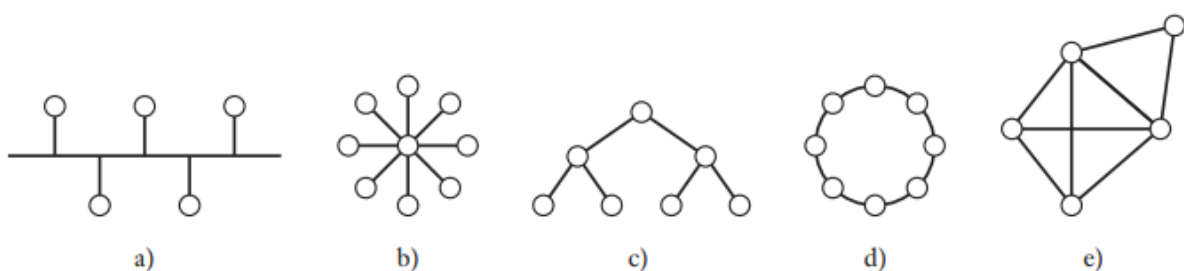
Esta es una topología muy extendida en las redes de área local (LAN), donde cada estación se conecta directamente a un anillo. Las distancias entre nodos en esta configuración suelen ser inferiores a 100 metros. Los datos introducidos en el anillo circulan por él de forma secuencial hasta que la estación destino los retira. Esta característica permite que todas las estaciones del anillo tengan acceso potencial a la información transmitida. Figura 29.d

- **Malla**

Este método no presenta una topología fija o predefinida para las conexiones entre estaciones, permitiendo la existencia de múltiples rutas de datos entre cualquier par de ellas. El concepto de Red de Área Local (LAN) describe precisamente este tipo de redes, las cuales están confinadas a una zona geográfica específica, como puede ser un edificio o un campus. Figura 29.e

Figura 29.

Topologías de red industrial



Nota: Tomado de Acciomate Ingeniería & Proyectos. (2025). Protocolos de Comunicación Industrial: Tipos y Aplicaciones. Acciomate

Por lo general, la estructura de estas redes adopta las formas básicas de bus, estrella o anillo. En contraste, una Red de Área Ampla (WAN) se encarga de interconectar equipos, terminales y redes

locales a una escala nacional o incluso global. El enfoque principal de este capítulo recae en el estudio de las redes de área local (LAN).

Modelo de interconexión de sistemas abiertos

Es imprescindible contar con protocolos de comunicación que operen en diferentes niveles. Para ello, la Organización Internacional de Normalización (ISO) estableció un estándar de protocolos estructurado en siete niveles, conocido como Modelo de Interconexión de Sistemas Abiertos (OSI). Este modelo sirve como marco de referencia fundamental para el diseño de un sistema de normas coherente y coordinado. Las capas que lo componen son las siguientes:

Es fundamental implementar protocolos de comunicación en distintos estratos. La Organización Internacional de Normalización (ISO) desarrolló un estándar de protocolos de siete niveles denominado Modelo de Interconexión de Sistemas Abiertos (OSI), que sirve como marco de referencia para crear un sistema de normas unificado y armonioso. Estas capas son:

- **Capa Física**

Define los medios y características técnicas para transmitir el flujo de bits brutos a través del hardware de la red. Se ocupa de especificaciones como el tipo de cables, conectores, sincronización y niveles de voltaje de las señales. Estándares como Ethernet y Token Ring se definen primordialmente en este nivel.

- **Capa de Enlace de Datos**

Establece los protocolos para el envío y recepción de mensajes entre dos nodos directamente conectados. Sus funciones incluyen el encapsulamiento de datos en tramas, la detección y corrección de errores, y el control del flujo para asegurar la secuencia correcta. Ethernet y Token Ring también se especifican en esta capa.

- **Capa de Red**

Gestiona el direccionamiento lógico, el enrutamiento y el control del tráfico de paquetes a través de la red, asegurando que la información encuentre el camino más eficiente hacia su destino final. Protocolos comunes en esta capa son el Protocolo de Internet (IP) y el IPX de Novell.

- **Capa de Transporte**

Brinda un servicio de comunicación confiable y completo de extremo a extremo entre los sistemas finales. Es responsable de establecer, mantener y finalizar la conexión, además de garantizar la entrega ordenada y libre de errores. Los protocolos TCP (de Internet) y SPX (de Novell) operan en este nivel.

- **Capa de Sesión**

Administra y sincroniza el "diálogo" o sesión entre las aplicaciones que se comunican. Controla el establecimiento, la coordinación (mediante puntos de control) y la terminación de los intercambios de comunicación.

- **Capa de Presentación**

Actúa como traductor de los datos, encargándose de su sintaxis y semántica. Transforma la información del formato de la aplicación a un formato común para la transmisión por la red (y viceversa), manejando aspectos como la compresión, el cifrado y la codificación.

- **Capa de Aplicación**

Es la interfaz directa con el usuario o el proceso de aplicación, proporcionando los servicios de red específicos que estos requieren, como el acceso a archivos remotos, el correo electrónico o la navegación web. Protocolos como HTTP, FTP y SMTP funcionan en esta capa.

Características Eléctricas de las señales – Modos de Funcionamiento

La interfaz en serie puede ser con protocolo síncrono o asíncrono. Las interfaces asíncronas comúnmente utilizadas son la RS-232 y versiones posteriores, el lazo de corriente 20 mA, I2C, CAN y USB.

- **Interfaz RS-232**

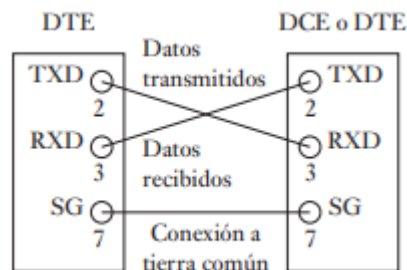
La interfaz serial más utilizada es la RS-232, cuya primera especificación fue publicada en 1962 por la Asociación de Industrias Electrónicas de Estados Unidos. Este estándar distingue dos tipos de equipos: el Equipo Terminal de Datos (DTE), como una computadora o microcontrolador que genera o consume datos, y el Equipo de Terminación del Circuito de Datos (DCE), como un módem, que actúa como intermediario en la comunicación.

Para establecer un enlace bidireccional básico (full-duplex simple), solo son necesarias tres señales: el pin 2 para transmitir datos (TxD), el pin 3 para recibir datos (RxD) y el pin 7 como tierra de señal común (GND). Por lo tanto, la conexión mínima requiere un cable de tres conductores.

Los estándares RS-232C definen niveles de voltaje no compatibles con la lógica digital común: un "0" lógico (espacio) se representa con un voltaje positivo (+3 V a +15 V, típicamente +12 V), y un "1" lógico (marca) con un voltaje negativo (-3 V a -15 V, típicamente -12 V). Esta diferencia hace imprescindible el uso de circuitos convertidores de nivel (como el chip MAX232) para adaptar las señales entre un dispositivo con lógica TTL/CMOS (por ejemplo, 0V y 5V) y el puerto RS-232.

Figura 20

Puerto de comunicación serial



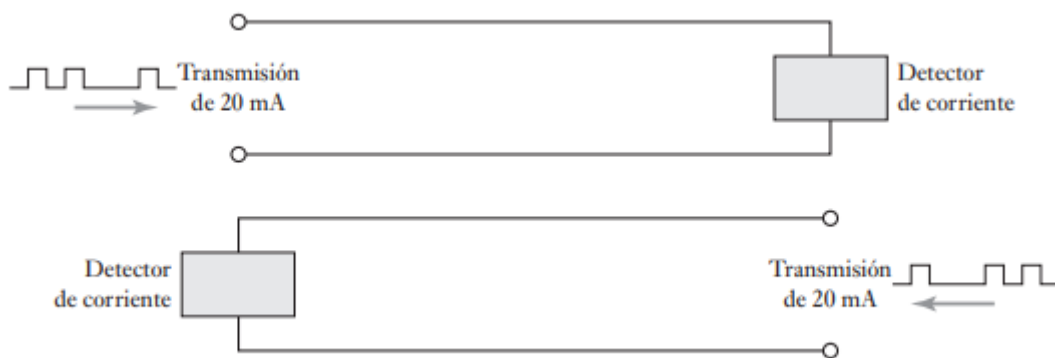
Nota: Tomado de *Acciomate Ingeniería & Proyectos. (2025). Protocolos de Comunicación Industrial: Tipos y Aplicaciones. Acciomate*

- **Lazo de corriente 20 mA**

Otra técnica que se basa en la RS-232 pero que no es parte de la norma, es el lazo de corriente 20 mA ver Figura 3. Éste emplea una señal de corriente en lugar de una señal de voltaje. Para la transmisión se utilizan un par de cables separados y los lazos receptores con un nivel de corriente de 20 mA que se usa para indicar un 1 lógico y 0 mA un 0 lógico. Los datos en serie están codificados con un bit de inicio, 8 bits de datos y dos bits de paro. Estas señales de corriente permiten una distancia mucho mayor, unos kilómetros, entre el transmisor y el receptor que con las conexiones de voltaje de la norma RS-232.

Figura 31.

Transmisor de corriente



Nota: Tomado de Acciomate Ingeniería & Proyectos. (2025). Protocolos de Comunicación Industrial:

Tipos y Aplicaciones. Acciomate

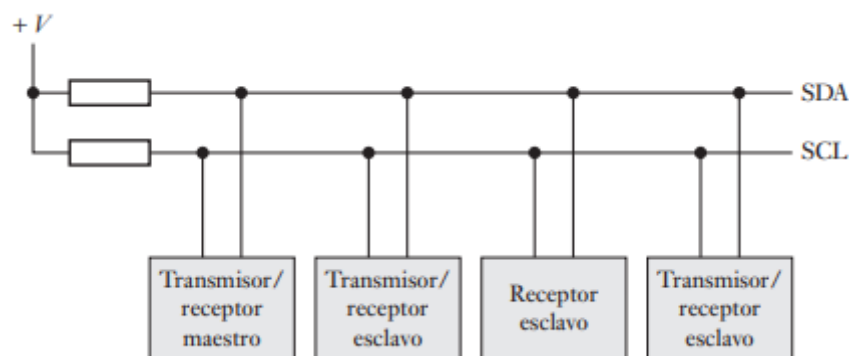
Buses de campo

- **Bus I²C**

El bus de Intercomunicación con Circuitos Integrados, más conocido como bus I²C, es el bus de datos diseñado por Philips para la comunicación entre circuitos integrados o módulos. A través de este bus los dispositivos intercambian datos e instrucciones, con sólo dos cables, lo que simplifica de manera considerable los circuitos.

Figura 32

Bus I²C



Nota: Tomado de Acciomate Ingeniería & Proyectos. (2025). Protocolos de Comunicación Industrial:

Tipos y Aplicaciones. Acciomate

Ambas líneas son líneas de datos bidireccionales (SDA) y una línea de temporización (SCL). Las dos líneas se conectan a la fuente de alimentación positiva, a través de resistores.

- **Bus CAN**

Un automóvil moderno puede tener hasta setenta unidades de control electrónico (ECU) para varios subsistemas, como los sistemas de administración del motor, frenos antibloqueo, control de tracción, suspensión activa, bolsas de aire, control de cruce, de ventanas, etcétera. Todo esto implicaría un cableado excesivo. Sin embargo, una alternativa es el uso de un bus de datos común, el cual transmite datos a través de sí mismo y los hace llegar a todas las partes del automóvil. Bosch desarrolló un protocolo conocido como CAN o Controller Area Network o red de área de control. Actualmente el bus CAN también se utiliza como un bus de campo en otros sistemas de automatización.

- **USB**

El Bus Serial Universal (USB) está diseñado para que monitores, impresoras, módems y otros dispositivos de entrada se conecten fácilmente a las PC; se utiliza el término plug-and-play. El USB utiliza una topología de estrella; por tanto, sólo necesita un dispositivo para conectarse en una PC con otros dispositivos que de esta manera se puedan conectar al anillo resultante en una topología de estrella.

- **Firewire**

El firewire es un bus en serie desarrollado por Apple Computers, las especificaciones las da IEEE 1394. El firewire ofrece capacidad de conexión y reproducción y se utiliza en aplicaciones como unidades de disco duro, impresoras y cámaras.

Buses de Nivel Sensores/Actuadores:

- AS-Interface (AS-i):
- Topología: Libre (árbol, estrella, línea)
- Velocidad: 167 kbps
- Característica: Alimentación y datos por el mismo cable
- Aplicación: Sensores y actuadores binarios simples

DeviceNet:

- Basado en CAN (Controller Area Network)

- Topología: Tronco-ramal
- Velocidad: 125, 250, 500 kbps
- Aplicación: E/S distribuidas

Buses de Nivel de Campo:

- PROFIBUS DP: Para automatización de procesos
- PROFIBUS PA: Para áreas con riesgo de explosión (intrínsecamente seguro)
- Foundation Fieldbus H1: Para instrumentación de proceso

Buses de Nivel de Control:

- PROFINET: Ethernet industrial
- Ethernet/IP: Ethernet con protocolo CIP
- Modbus TCP: Modbus sobre Ethernet
- Comparativa según Industriales Pérez (2015):

Protocolos de comunicaciones industriales

Principios de funcionamiento, arquitectura maestra–esclavo y variantes RTU/TCP.

Arquitectura Maestra-Esclavo:

- Maestro: Inicia todas las transacciones, controla la comunicación
- Esclavos: Responden solo cuando son interrogados
- Ventajas: Predictibilidad, control determinístico
- Desventajas: Ineficiente si hay muchos esclavos

Variantes RTU/TCP:

- RTU (Remote Terminal Unit): Implementación sobre RS-485/232
- TCP/IP: Implementación sobre Ethernet

Ciclo Típico de Comunicación:

- Maestro envía comando a dirección específica
- Todos esclavos reciben el mensaje
- Solo el esclavo dirigido responde
- Maestro procesa respuesta y continúa ciclo

Modbus RTU-ASCII

Modbus RTU (Remote Terminal Unit):

- Transporte: RS-485/RS-232
- Codificación: Binaria (más eficiente)
- Formato de trama:
[Dirección] [Función] [Datos] [CRC]
- 1 byte 1 byte N bytes 2 bytes
- Velocidad: 300-115200 baudios
- Características: Mayor densidad de datos, chequeo por CRC

Modbus ASCII:

- Codificación: Caracteres ASCII imprimibles
- Formato de trama:
[:] [Dirección(2 chars)] [Función(2 chars)] [Datos] [LRC] [CR][LF]
- Ventajas: Fácil depuración (mensajes legibles)
- Desventajas: Menor eficiencia (el doble de bytes que RTU)

Funciones Modbus Comunes:

- 01 (0x01): Leer coils (salidas digitales)
- 02 (0x02): Leer inputs discretos (entradas digitales)
- 03 (0x03): Leer holding registers
- 04 (0x04): Leer input registers
- 05 (0x05): Escribir single coil
- 06 (0x06): Escribir single register
- 16 (0x10): Escribir múltiples registers

Modbus TCP/IP

Arquitectura Modbus TCP:

- Capa Aplicación: Modbus Protocol
- Capa Transporte: TCP (puerto 502)
- Capa Red: IP
- Capa Enlace: Ethernet

Formato de Mensaje Modbus TCP:

- [MBAP Header] [Modbus PDU]
7 bytes N bytes
- MBAP Header:
- [Transaction ID] [Protocol ID] [Length] [Unit ID]
2 bytes 2 bytes 2 bytes 1 byte

Ventajas sobre Modbus RTU:

- Distancias ilimitadas (vía routers)
- Mayor velocidad (100 Mbps vs. 115 kbps)
- Conexiones múltiples simultáneas
- Integración con redes empresariales

Profibus – Profinet

PROFIBUS DP (Decentralized Periphery):

- Para automatización de fabricación
- Velocidad: 9.6 kbps a 12 Mbps
- Topología: Bus lineal con terminaciones

PROFIBUS PA (Process Automation):

- Para áreas explosivas (intrínsecamente seguro)
- Velocidad: 31.25 kbps
- Alimentación por bus

Arquitectura, comunicación en tiempo real

Tipo 1: Tiempo Real Suave (Soft Real-Time)

- Ejemplo: PROFINET Clase A/B
- Basado en prioridades (IEEE 802.1Q)
- Jitter aceptable: 10-100ms

Tipo 2: Tiempo Real Duro (Hard Real-Time)

- Ejemplo: PROFINET IRT, EtherCAT
- Planificación determinística
- Jitter: $<1\mu\text{s}$
- Requiere switches especiales

Técnicas para Tiempo Real:

- Time-Slicing: División del tiempo en slots
- Prioridades (VLAN tagging): IEEE 802.1Q/p
- IRT (Isochronous Real Time): Reserva de ancho de banda

5.1.1 Aplicaciones en sistemas PLC–HMI.

Directa (Punto a Punto):

PLC ————— HMI

(RS-485/232 o Ethernet)

- Protocolos: Modbus RTU, PPI (Siemens), DF1 (Allen-Bradley)
- Aplicación: Máquinas individuales

REFERENCIAS BIBLIOGRÁFICAS

- Brunete, A., San Segundo, P., & Herrero, R. (2025). *Introducción a la automatización industrial*.
Obtenido de: https://bookdown.org/alberto_brunete/intro_automatica/intro_automatica.pdf
- Centro de Ciberseguridad Industrial (2017). Guía de bolsillo: Ciberseguridad en la pirámide de automatización industrial. <https://www.cci-es.org/wp-content/uploads/Guia-Bolsillo-Piramide-de-Automatizacion-Industrial-1.pdf>
- Cuásquer Gualoto, V. del R., Durán Pardo, V. H., Lema Parco, R., Cajas Oña, E. J., Andrade García, V. J., & Sarzosa Villarroel, M. A. (2025). *Guía general de estudio de Instrumentación* (ISBN 978-9942-686-86-2). Rimana Editorial. <https://doi.org/10.59602/re.201>
- International Electrotechnical Commission. (2013). Programmable controllers – Part 3: Programming languages (IEC 61131-3). IEC.
- International Society of Automation. (2009). *Instrumentation symbols and identification* (ANSI/ISA-5.1-2009). ISA.
- Ragno, M. (2024, abril 2). Introduction to types of automation systems. RT Engineering. Obtenido de: <https://www.rteng.com/blog/types-of-industrial-automation-systems>
- Villamarín Reinoso, J. V. (2024). *Automatización y control industrial*. Instituto Superior Universitario Cotopaxi. Obtenido de: <https://libros.isuc.edu.ec/index.php/omp/catalog/book/45>
- Acciomate Ingeniería & Proyectos. (2025, 12 de enero). Protocolos de Comunicación Industrial: Tipos y Aplicaciones. Acciomate. <https://www.acciomate.com/post/protocolos-de-comunicaci%C3%B3n-industrial-tipos-y-aplicaciones>
- International Society of Automation. (2015). *ISA-101.01: Human-Machine Interfaces for Process Automation Systems*. ISA.
- Manuel, A., Biel, D., Olivé, J., Prat, J., & Sánchez, F. (2021). *Instrumentación Virtual Aquisición, procesamiento y análisis de señales*. Barcelona: Edicions de la Univesitat Politecnica de Catalunya.
- Rodríguez, A. (2013). *Sistemas SCADA*. Alfaomega – Marcombo.

- Rodriguez, J. (2012). *Buenas prácticas para diseño de HMI del alto rendimiento*. Cartagena: Universidad Tecnologica de Bolivar.
- Bolton, W. (2015). *Programmable logic controllers (6th ed.)*. Elsevier.
- IEC. (2013). *IEC 60848: Specification language GRAFCET for sequential function charts*. IEC.
- Petruzella, F. D. (2017). *Programmable logic controllers (5th ed.)*. McGraw-Hill Education.

SUCRE



ISBN: 978-9942-590-17-6



 SUCREInstitutooficial  @SUCREInstituto  @SUCREInstituto