

AUTOMATAS PROGRAMABLES

Objetivo General:

Aplicar los autómatas programables a la gestión de proyectos.

Horas totales: 60

CODIGO : ELEE018PO

CONTENIDOS:

1. ESTRUCTURA, MANEJO E INSTALACIÓN DE LOS AUTÓMATAS PROGRAMABLES:

- 1.1. Arquitectura interna de un autómata:
 - 1.1.1. Introducción.
 - 1.1.2. Bloques esenciales de un autómata.
 - 1.1.3. Unidad central de proceso, CPU.
 - 1.1.4. Memoria del autómata.
 - 1.1.5. Interfases de entrada y salida.
 - 1.1.6. Fuente de alimentación.
- 1.2. Ciclo de funcionamiento de un autómata y control en tiempo real:
 - 1.2.1. Introducción.
 - 1.2.2. Modos de operación.
 - 1.2.3. Ciclo de funcionamiento.
 - 1.2.4. Chequeos del sistema.
 - 1.2.5. Tiempo de ejecución y control en tiempo real.
 - 1.2.6. Elementos de proceso rápido.
 - 1.2.7. Procesado rápido de programas.
 - 1.2.8. Contador de alta velocidad.
 - 1.2.9. Entradas detectoras de flanco.
- 1.3. Instalación y mantenimiento de autómatas programables:
 - 1.3.1. Introducción.
 - 1.3.2. Fase de proyecto con autómatas programables.
 - 1.3.3. Selección del autómata.
 - 1.3.4. Fase de instalación.
 - 1.3.5. Fijaciones y condiciones mecánicas.
 - 1.3.6. Espacios de ventilación.
 - 1.3.7. Distancias de seguridad eléctrica.
 - 1.3.8. Condiciones ambientales.
 - 1.3.9. Compatibilidad electromagnética.
 - 1.3.10. Alimentación y protecciones.
 - 1.3.11. Distribución y cableado interno del armario de control.
 - 1.3.12. Cableado externo.
 - 1.3.13. Diseño e instalación del software.
 - 1.3.14. Fiabilidad de las instalaciones con autómatas.
 - 1.3.15. Mantenimiento de instalaciones con autómatas

2. LENGUAJES Y PROGRAMAS EN LOS AUTÓMATAS PROGRAMABLES:

- 2.1. Diseño de automatismos lógicos:
 - 2.1.1. Introducción.
 - 2.1.2. Modelos y funciones de transferencia.
 - 2.1.3. Automatismos combinacionales y secuenciales.

- 2.1.4. Diseño de automatismos combinacionales.
- 2.1.5. Diseño de automatismos secuenciales.
- 2.1.6. GRAFCET: Resumen histórico.
- 2.1.7. Diseño basado en GRAFCET.
- 2.1.8. GRAFCET: Elementos de base y reglas de evolución.
- 2.1.9. GRAFCET: Ejemplo de diseño.
- 2.1.10. Macroetapas y representación en detalle.
- 2.1.11. Estructuras básicas del GRAFCET.
- 2.1.12. Diagramas de flujo y diagramas GRAFCET.
- 2.1.13. Etapas iniciales, preposicionamiento y alarmas.
- 2.1.14. Puestas en marcha y paradas: GEMMA.
- 2.1.15. Método general de diseño basado en GEMMA.
- 2.1.16. Paros de emergencia.
- 2.1.17. Ejemplo de diseño.
- 2.2. Programación del autómatas:
 - 2.2.1. Introducción.
 - 2.2.2. Representación de sistemas de control.
 - 2.2.3. Descripciones literales.
 - 2.2.4. Identificación de variables y asignación de direcciones.
 - 2.2.5. Lenguajes de programación.
 - 2.2.6. Lenguajes booleanos y lista de instrucciones.
 - 2.2.7. Diagramas de contactos.
 - 2.2.8. Plano de funciones.
 - 2.2.9. Lenguajes de alto nivel.
- 2.3. Programación de bloques funcionales:
 - 2.3.1. Introducción.
 - 2.3.2. Bloques secuenciales básicos.
 - 2.3.3. Bloques funcionales de expansión.
 - 2.3.4. Instrucciones especiales.
- 2.4. Estructuras de programación:
 - 2.4.1. Introducción.
 - 2.4.2. Programación lineal.
 - 2.4.3. Programación estructurada.
 - 2.4.4. Programación multitarea.
 - 2.4.5. Tareas rápidas e interrupciones.
 - 2.4.6. Parametrización de módulos funcionales.
 - 2.4.7. Programación de procesadores periféricos inteligentes.
- 2.5. Programación en TIA PORTAL:
 - 2.5.1. Introducción a TIA PORTAL.
 - 2.5.2. Estructura interna de un TIA PORTAL.
 - 2.5.3. Direccionamiento.
 - 2.5.4. Estructura de un programa y módulos de TIA PORTAL
 - 2.5.5. Operaciones combinacionales.
 - 2.5.6. Operaciones de memoria.
 - 2.5.7. Operaciones de tiempo.
 - 2.5.8. Operaciones con contadores.
 - 2.5.9. Operaciones de comparación.

2.5.10. Operaciones aritméticas.

2.5.11. Operaciones entre bloques.

2.5.12. Relaciones entre bloques.

3. EJEMPLOS BÁSICOS DE PROGRAMACIÓN:

3.1. Introducción.

3.2. Identificación de entradas y salidas.

3.3. Programas con operaciones combinatorias.

3.4. Programas con contadores.

3.5. Programas con operadores de comparación.

3.6. Programas con operadores de memoria.

3.7. Programas con temporizadores.

3.8. Ejemplo: Máquina empaquetadora.

3.9. Ejemplo: Máquina mezcladora.